



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

La Universidad Católica de Loja

ÁREA TÉCNICA

TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y
COMPUTACIÓN

**Implementación de una aplicación para gestionar los procesos de
elaboración, valoración postulación y asignación de trabajos de titulación
par la carrera de Ingeniería en Sistemas Informáticos y Computación e
Ingeniería en Informática**

TRABAJO DE TITULACIÓN

AUTOR: Quichimbo Suárez, José Luis

DIRECTOR: Abad Espinoza, Marco Patricio

LOJA – ECUADOR

2017



Esta versión digital, ha sido acreditada bajo la licencia Creative Commons 4.0, CC BY-NY-SA: Reconocimiento-No comercial-Compartir igual; la cual permite copiar, distribuir y comunicar públicamente la obra, mientras se reconozca la autoría original, no se utilice con fines comerciales y se permiten obras derivadas, siempre que mantenga la misma licencia al ser divulgada. <http://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>

Septiembre, 2017

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN

Ingeniero.

Marco Patricio Abad Espinoza

DOCENTE DE LA TITULACIÓN

De mi consideración:

El presente trabajo de titulación: “Implementación de una aplicación para gestionar los procesos de elaboración, valoración postulación y asignación de trabajos de titulación par la carrera de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática” realizado por José Luis Quichimbo Suárez, ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo.

Loja, Marzo de 2017

f

Ing. Marco Patricio Abad Espinoza

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

Yo José Luis Quichimbo Suárez declaro ser autor del presente trabajo de titulación: “Implementación de una aplicación para gestionar los procesos de elaboración, valoración postulación y asignación de trabajos de titulación par la carrera de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática”, de la Titulación de Sistemas Informáticos Computación, siendo Marco Patricio Abad Espinoza director del presente trabajo; y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales. Además certifico que las ideas, conceptos, procedimientos, desarrollo y resultados vertidos en el presente trabajo de investigación, son de mi exclusiva responsabilidad.

Adicionalmente declaro conocer y aceptar la disposición del Art. 88 del Estatuto Orgánico de la Universidad Técnica Particular de Loja que en su parte pertinente textualmente dice: “Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través, o con el apoyo financiero, académico o institucional (operativo) de la Universidad”.

f.....

Autor: Quichimbo Suárez José Luis

Cédula: 1104134232

DEDICATORIA

Este trabajo lo dedico a mis padres, por su esfuerzo en cada etapa de mi vida para que pueda alcanzar mis objetivos, de quienes su afecto y confianza han constituido la mayor fuerza para avanzar en cada etapa personal y académica.

A mis hermanos y demás familiares por la confianza depositada y por la motivación en cada circunstancia difícil que he tenido que superar.

A mis amigos desde que iniciamos la carrera universitaria, y a los que permanecen a lo largo de los años y que a pesar de la distancia demostrando que ese vínculo constituye una fuerza personal que motiva a avanzar e ir por más logros.

AGRADECIMIENTO

A Dios y a San Gregorio, gracias a ellos el poder estar aquí, y poder haber culminado este trabajo.

A mis docentes a lo largo de la carrera, que además de impartirme sus conocimientos, han constituido una guía por los consejos personales y la motivación que han sabido compartir a su momento.

Un agradecimiento especial al Ingeniero. Patricio Abad, quien dirige este trabajo de titulación por su tiempo y confianza depositada en mí como desarrollador, quien me ha permitido formar parte de una solución para mi titulación.

Al Ingeniero Richar Guaya, por brindarme su ayuda para la integración con el Sistema del Vicerrectorado Académico, por su buena voluntad, su apoyo y sus consejos y sugerencias para poder culminar con éxito este trabajo.

A la Lic. Lady Sanmartín secretaria de mi titulación, por su entrega en su oficio y por su voluntad y vocación para ayudarnos a los estudiantes desde que ingresé a esta carrera.

A mis compañeros de la carrera agradezco infinitamente, quienes me brindaron su apoyo en todo sentido, cuando tuve el problema de salud grave y estuvieron conmigo para motivarme a culminar mis estudios a pesar de mi estado.

A mis padres, familiares y amigos, por la confianza depositada y su apoyo incondicional para culminar con éxito mi carrera universitaria.

ÍNDICE DE CONTENIDO

Contenido

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN.....	ii
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE DE CONTENIDO	vi
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE TABLAS	xiii
RESUMEN.....	1
ABSTRACT	2
INTRODUCCIÓN.....	3
OBJETIVOS	4
Objetivo General.	4
Objetivos específicos.	4
ALCANCE	5
ACTIVIDADES	5
RESULTADOS ESPERADOS.....	5
CAPÍTULO 1: MARCO TEÓRICO.....	6
1.1. Arquitecturas de software	7
1.1.1. Cloud Computing.....	7
1.1.2. Cliente servidor.....	10
1.1.3. Modelo 4+1.....	11
1.2. Metodologías.....	12
1.2.1. Metodologías tradicionales	13
1.2.2. Metodologías ágiles.....	14
1.3. Lenguajes y herramientas de programación	24
1.3.1. PHP.....	25
1.3.2. Ruby.....	26
1.3.3. Python	28
1.3.4. Node.js.....	30
1.3.5. Java.....	31
1.3. Bases de datos.....	33
1.4.1. Base de datos NoSQL.....	33
1.4.2. MongoDB	35
1.5. Servidores Cloud.....	36

1.6.	Requerimientos	39
1.6.1.	Herramientas y técnicas	40
1.7.	Servicios Web.....	47
1.7.1.	SOAP	47
1.7.2.	REST.....	48
1.8.	Análisis	49
1.8.1.	Arquitectura	49
1.8.2.	Lenguaje de programación	52
1.8.3.	Servidor	55
1.8.4.	Base de datos.....	58
1.8.5.	Resultados del análisis.....	61
CAPÍTULO 2: PROCESO DE NEGOCIO.....		63
2.1.	Proceso de trabajos de titulación UTPL.....	64
2.1.2.	Proceso	64
2.1.3.	Matriz de responsabilidades	70
2.2.	Categorización de interesados (Stakeholders).....	73
2.3.	Matriz de perfiles de interesados	75
2.4.	El problema	85
2.4.1.	Falta de información de los Trabajos de Titulación (T.T.).....	86
2.4.2.	Estudiantes reprueban o desertan de los T.T.	90
2.4.3.	Retraso en postulación de propuestas para T.T.	93
2.4.4.	Retraso en revisiones de propuestas para T.T.	97
2.4.5.	Dificultad para notificaciones.	100
2.5.	Propuesta de Solución.....	103
CAPÍTULO 3: DESARROLLO DE LA SOLUCIÓN		108
3.1.	Introducción	109
3.2.	Requerimientos	110
3.2.1.	Problema.....	110
3.2.2.	Objetivos.	110
3.2.3.	Requerimientos Funcionales.	111
3.2.4.	Product BackLog.	112
3.2.5.	Identificación de actores y responsabilidades.....	112
3.2.6.	Identificación de necesidades.....	114
3.2.7.	Identificación de casos de uso.....	118
3.2.8.	Requerimientos No Funcionales.....	118
3.2.9.	Diagramas de secuencia para cada caso de uso.....	120
3.2.10.	Vista de escenarios 4+1.....	120

3.3.	Análisis	120
3.3.1.	Identificación de objetos entidad, límites, y objetos de control	120
3.3.1.	Modelado de interacción entre objetos.	120
3.3.6.	Vista Lógica 4+1	121
3.4.	Diseño	121
3.4.1.	Identificación de subsistemas	121
3.4.2.	Modelado de componentes en Django.....	122
3.4.3.	Vista de Procesos 4+1	122
3.5.	Implementación	122
3.5.1.	Diagrama de Datos.....	123
3.5.2.	Despliegue en Heroku	123
3.5.3.	Sprint BackLog	124
3.5.4.	Integración con servicio REST.....	131
CAPÍTULO 4: PLAN DE PRUEBAS Y VALIDACIÓN.		134
4.1.	Introducción	135
4.2.	Ejecución de Pruebas.....	135
4.2.1.	Creación de convocatoria.	136
4.2.2.	Creación de esquema de evaluación para propuesta.	138
4.2.3.	Subida de propuesta de trabajo de titulación.	139
4.2.4.	Asignación de revisores.....	140
4.2.5.	Calificación de propuesta de trabajo de titulación.	141
4.2.6.	Validación de revisiones.	142
4.2.7.	Autorización y envío de propuestas a S.V.A.	143
4.2.8.	Consumo de estudiante para propuesta de S.V.A.	144
CAPÍTULO 5: RESULTADOS		146
5.1.	Introducción	147
5.2.	Resultados de la encuesta.....	147
5.2.1.	Oferta de convocatorias.....	149
5.2.2.	Creación de esquema de evaluación (rúbrica).....	153
5.2.3.	Subida de propuestas.....	157
5.2.4.	Calificación de propuesta.....	162
5.2.5.	Validación de revisiones	166
5.2.6.	Autorización de propuestas	171
5.3.	Cómo la herramienta resuelve los problemas identificados	175
5.3.1.	Falta de información de propuestas para trabajos de titulación.....	176
5.3.2.	Estudiantes reprueban o desertan de los trabajos de titulación.	177
5.3.3.	Retraso en propuesta y postulación de trabajos de titulación.	178

5.3.4.	Retraso en revisiones de propuestas para trabajos de titulación.	179
5.3.5.	Dificultad para notificación a involucrados.	180
5.3.6.	Análisis.	181
CONCLUSIONES		182
RECOMENDACIONES		184
BIBLIOGRAFÍA.....		185
ANEXOS.....		189
ANEXO 1.	Diagrama de flujo de proceso de postulación.....	190
ANEXO 2.	Preguntas para entrevista	191
ANEXO 3.	Reporte de entrevistas	192
ANEXO 4.	Caso de uso global	208
ANEXO 5.	Especificación de casos de uso.	209
ANEXO 6.	Diagramas de secuencia.....	241
ANEXO 7.	Diagramas de Datos	248
ANEXO 8.	Product BackLog.....	253
ANEXO 9.	Diagrama de componentes - Subsistemas.....	262
ANEXO 10.	Modelado de Componentes Django.....	265
ANEXO 11.	Diagrama de estados.....	266
ANEXO 12.	Prototipos exploratorios	267
ANEXO 13.	Lineamientos para la presentación, aprobación y seguimiento de los Trabajos de Titulación.	270
ANEXO 14.	Instalación y configuración de Heroku	302
ANEXO 15.	Formularios para pruebas y validación.....	305

ÍNDICE DE FIGURAS

Figura 1. Proceso Global Scrum	18
Figura 2. Ciclo de vida DevOps	21
Figura 3. Esquema relación de Django	29
Figura 4. Pasos para elicitación de requerimientos.....	39
Figura 5. Proceso General de Postulación de Trabajos de Titulación.	66
Figura 6. Diagrama causa efecto – Falta de información de T.T.	86
Figura 7. Histograma análisis– Falta de información de T.T.....	89
Figura 8. Diagrama causa efecto – Estudiantes reprueban o desertan de T.T.....	90
Figura 9. Histograma análisis– Reprobación de estudiantes.....	92
Figura 10. Diagrama causa efecto – Retraso en postulación de T.T.....	93
Figura 11. Histograma análisis– Retraso en postulación.....	96
Figura 12. Diagrama causa efecto – Retraso en revisiones de T.T.....	97
Figura 13. Histograma análisis– Retraso en revisión.	99
Figura 14. Diagrama causa efecto – Dificultad para notificaciones	100
Figura 15. Histograma análisis– Dificultad para notificaciones.....	102
Figura 16. Diagrama Propuesta de solución.	105
Figura 17. Tablero Herramienta Kanban AppSpot.	125
Figura 18. Salida servicio REST formato JSON.....	132
Figura 19. Formulario creación de convocatoria.	137
Figura 20. Formulario creación de esquema de evaluación.	138
Figura 21. Formulario subida de propuesta.....	140
Figura 22. Formulario asignación de revisores.....	141
Figura 23. Formulario calificación de propuesta.....	141
Figura 24. Interfaz visualización de calificaciones.....	142
Figura 25. Interfaz visualización de calificaciones.....	143
Figura 26. Confirmación de autorización propuestas.	144
Figura 27. Interfaz visualización de propuestas autorizadas y postuladas.	144
Figura 28. Porcentaje personas encuestadas.....	148
Figura 29. Gráfico circular – Soporte al proceso de negocio “Oferta de convocatoria”.	149
Figura 30. Gráfico circular – Complejidad “Oferta de convocatoria”.	150
Figura 31. Gráfico circular – Velocidad “Oferta de convocatoria”.	151
Figura 32. Gráfico circular – Confiabilidad “Oferta de convocatoria”.	152
Figura 33. Gráfico circular – Soporte al proceso de negocio “Esquemas de evaluación”. ..	153
Figura 34. Gráfico circular – Complejidad “Esquemas de evaluación”.	154

Figura 35. Gráfico circular – Velocidad “Esquemas de evaluación”.....	155
Figura 36. Gráfico circular – Confiabilidad “Esquemas de evaluación”.....	156
Figura 37. Gráfico circular – Soporte al proceso de negocio “Subida de propuestas”.	158
Figura 38. Gráfico circular – Complejidad “Subida de propuestas”.....	159
Figura 39. Gráfico circular – Velocidad “Subida de propuestas”.....	160
Figura 40. Gráfico circular – Confiabilidad “Subida de propuestas”.....	161
Figura 41. Gráfico circular – Soporte al proceso de negocio “Calificación de propuesta”. ..	162
Figura 42. Gráfico circular – Complejidad “Calificación de propuesta”.	163
Figura 43. Gráfico circular – Velocidad “Calificación de propuesta”.	164
Figura 44. Gráfico circular – Confiabilidad “Calificación de propuesta”.	165
Figura 45. Gráfico circular – Soporte al proceso de negocio “Validación de revisiones”.....	167
Figura 46. Gráfico circular – Complejidad “Validación de revisiones”.....	168
Figura 47. Gráfico circular – Velocidad “Validación de revisiones”.....	169
Figura 48. Gráfico circular – Confiabilidad “Validación de revisiones”.....	170
Figura 49. Gráfico circular – Soporte al proceso de negocio “Autorización y envío de propuestas”.....	171
Figura 50. Gráfico circular – Complejidad “Autorización y envío de propuestas”.....	172
Figura 51. Gráfico circular – Velocidad “Autorización y envío de propuestas”.....	173
Figura 52. Gráfico circular – Confiabilidad “Autorización y envío de propuestas”.....	174
Figura 53. Gráfico circular – Resolución del problema "Falta de información de T.T."	176
Figura 54. Gráfico circular – Resolución del problema "Reprobación y abandono de estudiantes a T.T.".....	177
Figura 55. Gráfico circular – Resolución del problema "Retraso en propuesta y postulación de T.T.".....	178
Figura 56. Gráfico circular – Resolución del problema "Retraso en revisiones de T.T."	179
Figura 57. Gráfico circular – Resolución del problema "Dificultad para notificaciones.".....	180
Figura 58. Diagrama de Flujo proceso postulacion de trabajos de titulación	190
Figura 59. Reporte entrevista coordinador titulación	195
Figura 60. Reporte entrevista responsable sección	198
Figura 61. Reporte entrevista tutor de Gestión Productiva	202
Figura 62. Reporte entrevista secretaria titulación Sistemas informáticos y computación ..	204
Figura 63. Reporte entrevista secretaria titulación Ingeniería Informática	207
Figura 64. Caso de uso.....	208
Figura 65. Diagrama de secuencia – Ofertar Convocatoria.....	241
Figura 66. Diagrama de secuencia – Crear Rúbrica (Esquema Evaluación)	242
Figura 67. Diagrama de secuencia – Asignar Revisión	243
Figura 68. Diagrama de secuencia – Subir Propuesta	244

Figura 69. Diagrama de secuencia – Eliminar Propuesta.....	245
Figura 70. Diagrama de secuencia – Calificar Propuesta.....	246
Figura 71. Diagrama de secuencia – Aprobar Revisión	247
Figura 72. Modelo de datos Usuario Titulación	249
Figura 73. Modelo de datos: Propuesta - Convocatoria	250
Figura 74. Modelo de datos: Rúbrica - Revisión.....	251
Figura 75. Modelo de datos: Rúbrica - Revisión.....	252
Figura 76. Descomposición de sistema - Package.....	262
Figura 77. Descomposición subsistemas.....	263
Figura 78. Modelo de componentes Django.....	265
Figura 79. Diagrama de estados “Propuesta T.T”	266
Figura 80. Prototipo Subir Propuesta para T.T.....	267
Figura 81. Prototipo Calificación de propuesta para T.T	268
Figura 82. Configuración Heroku	304
Figura 83. Configuración Heroku	304
Figura 84. Validación escenario “Lanzar convocatoria”.....	305
Figura 85. Validación escenario “Crear esquema evaluación”	306
Figura 86. Validación escenario “Subida de Propuesta”.....	307
Figura 87. Validación escenario “Calificación de Propuesta”.....	308
Figura 88. Validación escenario “Validación de revisión”	309
Figura 89. Validación escenario “Autorización de propuestas”.....	310
Figura 90. Validación escenario “Consumo de estudiantes para Propuesta”	311

ÍNDICE DE TABLAS

Tabla 1. Herramientas para elicitación	40
Tabla 2. Esquema de valoración de arquitectura	49
Tabla 3. Evaluación de arquitecturas	51
Tabla 4. Esquema de valoración lenguajes de programación.	52
Tabla 5. Evaluación de lenguajes de programación.....	54
Tabla 6. Esquema de valoración servicios cloud.....	55
Tabla 7. Evaluación servicios cloud.	57
Tabla 8. Esquema de valoración motores de bases de datos.	58
Tabla 9. Evaluación motores de bases de datos.....	60
Tabla 10. Resultados del análisis.....	61
Tabla 11. Matriz de Responsabilidades del Proceso	71
Tabla 12. Categorización de Interesados.....	73
Tabla 13. Matriz de perfiles de interesados.....	78
Tabla 14. Análisis: “Falta de información de T.T.”	87
Tabla 15. Análisis: “Estudiantes reprueban o desertan T.T.”	91
Tabla 16. Análisis: “Retraso en postulaciones de propuestas T.T.”	94
Tabla 17. Análisis: “Retraso en revisión de propuestas T.T.”	98
Tabla 18. Análisis: “Dificultad para notificaciones.”	101
Tabla 19. Identificación de actores.....	112
Tabla 20. Matriz de identificación de necesidades.	114
Tabla 21. Requerimientos no funcionales.	118
Tabla 22. Plan de Iteraciones.	126
Tabla 23. Valores encuesta - Soporte al proceso de negocio “Oferta de convocatoria”.	149
Tabla 24. Valores encuesta – Complejidad “Oferta de convocatoria”.....	150
Tabla 25. Valores encuesta – Velocidad “Oferta de convocatoria”.....	151
Tabla 26. Valores encuesta – Confiabilidad “Oferta de convocatoria”	152
Tabla 27. Análisis encuesta - Escenario “Oferta de convocatoria”.	153
Tabla 28. Valores encuesta - Soporte al proceso de negocio “Esquemas de evaluación” ..	154
Tabla 29. Valores encuesta – Complejidad “Esquemas de evaluación”.	155
Tabla 30. Valores encuesta – Velocidad “Esquemas de evaluación”.	156
Tabla 31. Valores encuesta – Confiabilidad “Esquemas de evaluación”.	157
Tabla 32. Análisis Encuesta - Escenario “Esquemas de evaluación”.	157
Tabla 33. Valores encuesta - Soporte al proceso de negocio “Subida de propuestas”	158
Tabla 34. Valores encuesta – Complejidad “Subida de propuestas”.	159

Tabla 35. Valores encuesta – Velocidad “Subida de propuestas” .	160
Tabla 36. Valores encuesta – Confiabilidad “Subida de propuestas” .	161
Tabla 37. Análisis Encuesta - Escenario “Subida de propuestas” .	162
Tabla 38. Valores encuesta - Soporte al proceso de negocio “Calificación de propuesta” .	163
Tabla 39. Valores encuesta – Complejidad “Calificación de propuesta” .	164
Tabla 40. Valores encuesta – Velocidad “Calificación de propuesta” .	165
Tabla 41. Valores encuesta – Confiabilidad “Calificación de propuesta” .	166
Tabla 42. Análisis Encuesta - Escenario “Calificación de propuesta” .	166
Tabla 43. Valores encuesta - Soporte al proceso de negocio “Validación de revisiones” .	167
Tabla 44. Valores encuesta – Complejidad “Validación de revisiones” .	168
Tabla 45. Valores encuesta – Velocidad “Validación de revisiones” .	169
Tabla 46. Valores encuesta – Confiabilidad “Validación de revisiones” .	170
Tabla 47. Análisis Encuesta - Escenario “Validación de revisiones” .	171
Tabla 48. Valores encuesta - Soporte al proceso de negocio “Autorización y envío de propuestas” .	172
Tabla 49. Valores encuesta – Complejidad “Autorización y envío de propuestas” .	173
Tabla 50. Valores encuesta – Velocidad “Autorización y envío de propuestas” .	174
Tabla 51. Valores encuesta – Confiabilidad “Autorización y envío de propuestas” .	175
Tabla 52. Análisis Encuesta - Escenario “Autorización y envío de propuestas” .	175
Tabla 53. Valores encuesta – Resolución del problema "Falta de información de T.T." .	176
Tabla 54. Valores encuesta – Resolución del problema "Reprobación y abandono de estudiantes a T.T." .	177
Tabla 55. Valores encuesta – Resolución del problema "Retraso en propuesta y postulación de T.T." .	178
Tabla 56. Valores encuesta – Resolución del problema "Retraso en revisiones de T.T." .	179
Tabla 57. Valores encuesta – Resolución del problema "Dificultad para notificaciones" .	180
Tabla 58. Análisis Encuesta - Escenario “Autorización y envío de propuestas” .	181
Tabla 59. Product BackLog .	253

RESUMEN

El presente proyecto describe el diseño y la implementación de una aplicación basada en la web, que dé soporte y solución al proceso de elaboración, valoración postulación y asignación de Trabajos de Titulación de las titulaciones: Sistemas Informáticos y Computación e Ingeniería Informática. Ésta aplicación debe comunicarse con los datos disponibles del personal de la Universidad Técnica Particular de Loja y de los estudiantes de la misma.

La implementación de la solución propuesta en esta investigación se encuentra en el desarrollo de una aplicación web que esté disponible en la nube y su desarrollo comprenda despliegues funcionales frecuentes con el uso de una metodología que soporte éstas entregas continuas.

Palabras Clave: Metodologías Ágiles, Arquitectura de Software, Lenguajes de Programación, Servicios Web, Heroku, Integración, JSON, 4+1, Krutchen, Ishikawa.

ABSTRACT

The present project describes the design and implementation of a web-based application, which supports and solves the process of preparation, postulation evaluation and assignment of Titling Jobs of the degrees: “Sistemas Informáticos y Computación” and “Ingeniería Informática”. This application must communicate with the available data of the personnel of the Universidad Técnica Particular de Loja and its students.

The implementation of the solution proposed in this research is in the development of a web application that is available in the cloud and its development includes frequent functional deployments using a methodology that supports these continuous deliveries.

Keywords: Agile Methodologies, Software Architecture, Programming Languages, Web Services, Heroku, Integration, JSON, 4+1, Krutchen, Ishikawa.

INTRODUCCIÓN

El reglamento de régimen académico (CES, 2015) al entrar en vigencia en la Universidad Técnica Particular de Loja, ha establecido la necesidad de adaptar las opciones de titulación y los procesos, para las diferentes titulaciones, disponiendo que las opciones de titulación se circunscriben a una de las opciones propuestas en el mismo.

El problema que actualmente reside en las titulaciones de la Universidad Técnica Particular de Loja (UTPL), específicamente en las titulaciones de “Sistemas Informáticos y Computación” e “Ingeniería Informática”, es que no existe un sistema de software que dé soporte al proceso, puesto que cada titulación de la UTPL maneja un propio esquema de evaluación para validar una propuesta de trabajo de titulación antes que el estudiante pueda postular a la misma.

Este proceso heterogéneo en estas y todas las titulaciones de la UTPL, ocasiona dificultad en el seguimiento del estado de cada propuesta de trabajo de titulación.

Lo que se plantea es desarrollar una herramienta de software, que de soporte a este proceso, utilizando una metodología de desarrollo que pueda abarcar los cambios periódicos en el proceso, que además utilice un lenguaje de programación ágil que permita el desarrollo eficiente del software, y que la herramienta pueda integrarse con los esquemas de datos de la UTPL.

El presente proyecto se enfoca a la opción “*Proyecto Técnico*”, el cuál dispone que se combinen elementos de desarrollo técnico en mayor medida, elementos de investigación y elementos de innovación, los cuáles deben ser medidos y controlados.

OBJETIVOS

Objetivo General.

Desarrollar una solución de software, que dé soporte para optimizar el proceso de gestión para la presentación, validación, postulación, aprobación y asignación, de trabajos de titulación de las titulaciones de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática, y homogeneice los procesos para el área técnica.

Objetivos específicos.

- Comprender el proceso establecido por la titulación y la problemática asociada al mismo.
- Establecer una metodología que de soporte al desarrollo de la solución.
- Elaborar un modelo de requerimientos de describa con claridad las necesidades de solución del sistema.
- Diseñar un modelo arquitectónico eficiente que de soporte a la solución.
- Implementar una aplicación web, necesaria para dar soporte al proceso.
- Validar la aplicación de modo que se garantice su utilidad y cumplimiento de los criterios de calidad establecidos por la titulación.

ALCANCE

El alcance del proyecto se centra en el proceso que corresponde a la presentación, evaluación, aprobación y postulación de trabajos de titulación. La aplicación validará la pertinencia del trabajo de titulación, antes de que el mismo sea publicado para la postulación de los estudiantes.

La aplicación deberá comunicarse con los servicios de datos disponibles de la Universidad Técnica Particular de Loja, utilizando el método de comunicación más adecuado según el análisis que se describa durante el desarrollo del presente proyecto.

ACTIVIDADES

- Análisis del proceso y lineamientos para la presentación, aprobación y seguimiento de los trabajos de titulación del área técnica de las titulaciones: *Ingeniería en sistemas informáticos* y *Computación e Ingeniería en informática*.
- Revisar los documentos que sustentan el problema, que son:
 - Reglamento de Régimen académico,
 - Documentación del CES sobre las opciones de titulación.
- Análisis y selección de la metodología de desarrollo de software a implementar.
- Estructuración y modelado de los requerimientos tomando en cuenta la notación y metodología, y la arquitectura e infraestructura seleccionadas como óptimas.
- Definición de la arquitectura de software a implementar, considerando que la misma sea centrada en la nube, se pueda vincular con aplicaciones móviles y soporte los procesos de gestión y comunicación de la aplicación.
- Desarrollo de la aplicación web, con la documentación elaborada, y las consideraciones con respecto a la arquitectura seleccionada.
- Integración de los esquemas de datos de la Universidad Técnica particular de Loja.

RESULTADOS ESPERADOS

- Documento de requerimientos.
- Especificación técnica de modelo de desarrollo e infraestructura tecnológica.
- Arquitectura y modelado del sistema.
- Aplicación web.

CAPÍTULO 1: MARCO TEÓRICO

A continuación se realizará un estudio de los componentes necesarios para la implementación de un sistema de software, estos componentes van desde la metodología, la arquitectura de software, el lenguaje de programación, la base de datos, el servidor y el framework a utilizar.

Existe un sinnúmero de permutaciones entre éstos componentes, y como ya se dijo antes, no existe una metodología, y tampoco una arquitectura predeterminada para un proyecto. Por tanto, se va a analizar los conceptos más importantes que van acorde con nuestro proyecto, se describirá los lenguajes más importantes conjunto con sus frameworks, con mayor influencia actualmente dentro del desarrollo ágil.

1.1. Arquitecturas de software

Además de la metodología de desarrollo, es necesario definir la arquitectura con la que se va a desarrollar el proyecto.

Puede observarse que al hablar de arquitectura de software, se hace alusión a la especificación de la estructura del sistema, entendida como la organización de componentes y relaciones entre ellos; los requerimientos que debe satisfacer el sistema y las restricciones a las que está sujeto, así como las propiedades no funcionales del sistema y su impacto sobre la calidad del mismo; las reglas y decisiones de diseño que gobiernan esta estructura y los argumentos que justifican las decisiones tomadas (Camacho, Cardeso, & Nuñez, 2004).

1.1.1. Cloud Computing

Uno de los conceptos a considerar en nuestro proyecto es el Cloud Computing. Una arquitectura cloud involucra recursos virtuales de almacenamiento, procesamiento, aplicaciones, hardware y plataformas de desarrollo, los cuales son accesibles al usuario mediante una interfaz web.

Según Rodríguez (2011) “Estos recursos son proporcionados como servicios y pueden ser dinámicamente reconfigurados para adaptarse a una carga de trabajo variable (escalabilidad), permitiendo una óptima utilización de los mismos y evitando sobre-provisión e infra-provisión (elasticidad)”. (Rodríguez, Pettoruti, Chichizola, & De Giusti, 2011)

La computación en nube es un término general para la entrega de servicios alojados a través de Internet. La computación en nube permite a las organizaciones utilizar los recursos informáticos como una utilidad como la electricidad, en lugar de tener que construir y mantener la infraestructura informática en la empresa. Por lo tanto, las organizaciones

pueden externalizar sus grandes datos en la nube. Reduce los costes de gestión y aumenta la facilidad de acceso.

Hay varios beneficios del almacenamiento en la nube: el uso más significativo del almacenamiento en la nube es almacenar gran cantidad de datos en la nube y utilizar la potencia de procesamiento del sistema de nube para recuperar eficientemente el resultado deseado basado en el requerimiento del usuario y reducir la sobrecarga de procesamiento de datos del propietario u organización (S. Ahmad & Kumar, 2016).

Cloud Computing comprende tres componentes principales: la Infraestructura como Servicio (IaaS), la plataforma como servicio (PaaS) y el software como servicio (SaaS), que se describirán a continuación.

1.1.1.1. SaaS (Software as a Service)

Dado que el alcance del proyecto considera que la implementación de nuestra solución a la gestión de la postulación de trabajos de titulación como un servicio para la titulación de Sistemas Informáticos y Computación e Ingeniería en Informática, que será utilizada por docentes, estudiantes y secretarías, se ha considerado la implementación como un servicio SaaS, Software como servicio.

Según Torbacki en su artículo “*SaaS – direction of technology development in ERP/MRP systems*”, define la tecnología SaaS como “la prestación de servicios de acceso remoto al software (software de contratación), que actualmente experimenta un desarrollo dinámico y empieza a ser apoyado por mayores productores de sistemas ERP / MRP en el mundo”.

Algunas de las ventajas de trabajar con modelos SaaS son:

- Bajos costes iniciales
- Gran velocidad de iniciación
- Bajos costos de suscripción
- No hay necesidad de instalar ningún software en una estación de trabajo que pertenece a un usuario del sistema
- El acceso en línea al sistema desde cualquier ubicación por ordenador con una conexión a Internet
- Eliminación de la licencia y actualización de costos,
- Costos TCO bajas (Total Cost of Ownership), debido a la falta de necesidad de grandes inversiones en servidores, bases de datos y mantenimiento,
- Acceso continuo a las versiones de software más recientes compatibles con las normas de derecho reales,

- El acceso a las tecnologías más nuevas de negocio,
- Previsibilidad de costos en los sistemas informáticos.

Parece que en los trabajos de investigación y desarrollo futuros cercanos realizados sobre ERP / sistemas MRP se concentrarán en:

- Amplia aplicación de la tecnología SaaS.
- Posibilidades de aplicación de la definición fácil de usuario y la aplicación de los informes multi-criterio generan en una base de información almacenada en esos sistemas.
- Aplicación de métodos modernos de gestión estratégica.(Torbacki, 2008).

Susana Chavez, Martín, Rodríguez, Murazzo, y Valenzuela, (2012), en su artículo en conferencia sobre “*Metodología ágil para el desarrollo de SaaS*” expone: “El Software como Servicio (SaaS) entrega software y datos como un servicio sobre internet usualmente por medio de un browser que corre del lado del cliente sin tener que instalarlo en este dispositivo”.

Con esto SaaS asegura que una sola copia del software se ejecute en un entorno homogéneo con respecto al sistema operativo y hardware, elementos que son controlados por los desarrolladores. La ventaja es que la interface (API) y la aplicación del lado del cliente no se modifican y los desarrolladores pueden mejorar el software y el hardware subyacente.

SaaS permite actualizar sólo una copia del software, y esta característica se alinea completamente con el ciclo de vida del desarrollo ágil. Las empresas proveedores SaaS se mantienen en innovación constante, agregando características que mantengan la fidelidad de sus clientes.

Por lo tanto, si el software, como en nuestro caso, está modelado como servicio, los requerimientos pueden ser mucho más simples.

1.1.1.2. PaaS (*Platform as a service*)

Los sistemas en la nube pueden ofrecer un nivel de abstracción adicional: en lugar de proporcionar una infraestructura virtualizada, pueden proporcionar la plataforma de software donde se ejecutan los sistemas. El dimensionamiento de los recursos de hardware exigidos por la ejecución de los servicios se realiza de manera transparente. Esto se denomina plataforma como un servicio (PaaS). (Vaquero, Rodero-Merino, Caceres, & Lindner, 2008).

Un ejemplo de Plataforma como servicio es ofertado por Google con Google App Engine. En su página oficial define: “Google App Engine es una plataforma como servicio (PaaS), que permite crear y ejecutar aplicaciones en la infraestructura de Google. Aplicaciones App Engine son fáciles de construir, fácil de mantener y fácil de escalar a medida que cambian las necesidades de tráfico y almacenamiento de datos. Con App Engine, no hay servidores que mantener. Sólo tiene que cargar su aplicación y que está listo para probar.”

1.1.1.3. IaaS (Infraestructure as a Service)

Los proveedores de infraestructura gestionan un gran conjunto de recursos informáticos, como la capacidad de almacenamiento y procesamiento. A través de la virtualización, son capaces de dividir, asignar y redimensionar dinámicamente estos recursos para crear sistemas ad-hoc según lo exigido por los clientes, los proveedores de servicio. Despliegan las pilas de software que ejecutan sus servicios. Este es el escenario Infraestructura como un servicio (IaaS). (Vaquero et al., 2008)

Como ejemplo, Google también ofrece un producto IaaS, que es Compute Engine, en su página define: “Ejecuta cargas de trabajo a gran escala en las máquinas virtuales alojadas en la infraestructura de Google. Elija una máquina virtual que se ajuste a sus necesidades y obtenga el rendimiento de la red de fibra en todo el mundo de Google.”

Un proyecto SaaS requiere de la infraestructura de la IT para lograr comunicación, que les permita a los clientes interactuar con los servicios; escalabilidad, en que el servicio pueda agregar nuevos usuarios rápidamente; y disponibilidad, en que la comunicación y el servicio estén continuamente disponibles. Internet provee la comunicación para SaaS, y Cloud Computing provee el hardware para la escalabilidad y almacenamiento para SaaS (Chavez et al., 2012)

1.1.2. Cliente servidor

Según el libro de (Sommerville, 2005), en una arquitectura cliente servidor, una aplicación se modela como un conjunto de servicios prestados por servidores y un conjunto de clientes que utilizan dichos servicios.

Los clientes conocen qué servicios pueden utilizar, pero no es necesario que conozcan la existencia de otros clientes.

Los clientes acceden a los servidores a través de una red que puede ser una intranet. No existe una correspondencia 1 a uno entre los procesos que puede ejecutar un servidor, puesto que un servidor puede correr varios procesos servidor invocados por varios clientes.

1.1.3. Modelo 4+1

Según el artículo (Kruchten, Philippe, s. f.) "*Planos Arquitectónicos, El modelo 4+1 Vistas de la Arquitectura del Software*", el modelo 4+1 de Kruchten, es un modelo de vistas diseñado por el profesor Philippe Kruchten que se utiliza para describir la arquitectura de un sistema software intensivo basado en el uso de múltiples puntos de vista.

Lo que propone Kruchten es que: "Un sistema software se ha de documentar y mostrar con 4 vistas bien diferenciadas y estas 4 vistas se han de relacionar entre sí con una vista más, que es la denominada vista +1".

Las cuatro vistas son:

- Vista lógica.
- Vista de procesos.
- Vista de despliegue
- Vista física
- Vista +1

Esta última vista relaciona las cuatro vistas anteriores y se llama "Vista de Escenario".

Cada una de estas vistas muestra toda la arquitectura del sistema software que se esté documentando, pero cada una de ellas ha de documentarse de forma diferente y ha de mostrar aspectos diferentes del sistema software.

Vista Lógica: En esta vista se representa la funcionalidad que el sistema proporcionara a los usuarios finales. Es decir, se ha de representar lo que el sistema debe hacer, y las funciones y servicios que ofrece. Para completar la documentación de esta vista se pueden incluir los diagramas de clases, de comunicación o de secuencia de UML.

Vista de Despliegue: En esta vista se muestra el sistema desde la perspectiva de un programador y se ocupa de la gestión del software; o en otras palabras, se va a mostrar cómo está dividido el sistema software en componentes y las dependencias que hay entre esos componentes. Para completar la documentación de esta vista se pueden incluir los diagramas de componentes y de paquetes de UML.

Vista de Procesos: En esta vista se muestran los procesos que hay en el sistema y la forma en la que se comunican estos procesos; es decir, se representa desde la perspectiva de un integrador de sistemas, el flujo de trabajo paso a paso de negocio y operacionales de los componentes que conforman el sistema. Para completar la documentación de esta vista se puede incluir el diagrama de actividad de UML.

Vista Física: En esta vista se muestra desde la perspectiva de un ingeniero de sistemas todos los componentes físicos del sistema así como las conexiones físicas entre esos componentes que conforman la solución (incluyendo los servicios). Para completar la documentación de esta vista se puede incluir el diagrama de despliegue de UML.

+1 Vista de Escenarios: Esta vista va a ser representada por los casos de uso de software y va a tener la función de unir y relacionar las otras 4 vistas, esto quiere decir que desde un caso de uso podemos ver cómo se van ligando las otras 4 vistas, con lo que tendremos una trazabilidad de componentes, clases, equipos, paquetes, etc., para realizar cada caso de uso. Para completar la documentación de esta vista se pueden incluir el diagrama de casos de uso de UML.

1.2. Metodologías

Ya que el desarrollo de software no es una tarea fácil, según afirma José H. Canós, es por eso que existen numerosas propuestas metodológicas para el proceso de desarrollo. Existe el desarrollo tradicional que se enfoca en el control del proceso, estableciendo inflexiblemente las actividades implicadas, los artefactos que se deben generar, y las herramientas y notaciones que se usarán. Sin embargo, el resultado final es un proceso de desarrollo más complejo que puede incluso restringir la propia habilidad del equipo para llevar a cabo el proyecto. Existe el otro enfoque que se centra en factor humano y el producto de software, es la filosofía denominada metodologías ágiles. Éstas metodologías dan mayor valor a la persona, a la cooperación con el cliente y al desarrollo de software de manera incremental con iteraciones y entregas muy cortas. Este enfoque está mostrando su efectividad en proyectos donde los requerimientos son muy cambiantes y cuando los tiempos de desarrollo son demasiado cortos, pero manteniendo una alta calidad (Canós, Letelier, & Penadés, 2013).

Cada proyecto constituye un mundo diferente, por tanto no existe una metodología determinada para un proyecto, por tanto, es necesario antes, comprender y definir la naturaleza del sistema a implementar. Las metodologías tradicionales se enfocan en el uso exhaustivo de documentación, mientras que las metodologías ágiles priorizan la constante respuesta a cambios y a la relación del equipo con el usuario final. Ambas tienen el mismo objetivo, que es el de desarrollar un sistema de calidad, que satisfaga las necesidades del cliente y genere competitividad con respecto a otros sistemas, en ambas es necesaria la documentación, puesto que los diagramas, documentos, explicaciones, descripciones, constituyen la fuente principal para la comprensión del entorno, de las necesidades y del problema.

Debido al entorno cambiante de nuestro proyecto, y a los continuos cambios en el proceso de postulación, se considera como opción más viable la implementación de metodologías ágiles, para gestionar de manera más eficiente los continuos cambios al proceso. Aún así se considera necesario mantener algunos aspectos sobre la documentación pertinente para el correcto desarrollo del sistema que contemplan las metodologías tradicionales especificadas a continuación.

1.2.1. Metodologías tradicionales

Según Figueroa, Solís y Cabrera sobre las metodologías tradicionales afirman que “Surgieron a los inicios del desarrollo de software, pues el desarrollo era netamente artesanal, y por la necesidad de mejorar el proceso para llegar a la meta deseada, tuvieron que importarse las metodologías de otras áreas y ajustarlas al desarrollo de software”.

Estas metodologías se centran en un proceso desarrollado por etapas, en dónde es estrictamente necesario culminar la fase precedente para proseguir con el proceso.

Los modelos principales son el modelo en cascada y el modelo en espiral. El modelo cascada es usado cuando se conocen a profundidad los requisitos, y no están completamente sujetos a futuras modificaciones; el modelo espiral es más enfocado a un desarrollo guiado por riesgos, donde existe una alta concurrencia de usuarios donde puede generarse un crecimiento incremental del sistema.

Entre las principales metodologías del enfoque tradicional tenemos a RUP y MSF, en donde su enfoque está en llevar una documentación exhaustiva del proyecto en su totalidad y centran su atención en el cumplimiento riguroso del plan de proyecto, y una vez definido todo esto, en la fase inicial del desarrollo del proyecto.

Además, el alto costo cuando se suscita un cambio, constituye otra de las características de estas metodologías, que no las hace eficientes en proyectos donde el entorno es cambiante.

Las metodologías tradicionales o formales se enfocan en la documentación, la planificación y procesos, las plantillas, las técnicas de administración, las revisiones, entre otros artefactos. (Figueroa, Solís, & Cabrera, 2008).

En conclusión, las metodologías ágiles proporcionan un desarrollo estructurado y controlado, y se pueden emplear en entornos de desarrollo dónde los requerimientos, el problema, y la solución misma son conocidos en su totalidad y no es necesario un cambio continuo, disponen de una documentación exhaustiva y además, para poder desarrollar un sistema con metodologías ágiles, es necesario haber tenido experiencia en el desarrollo con metodologías tradicionales.

1.2.2. Metodologías ágiles

Debido a la agilización y aceleración continua de la tecnología, y sus continuos cambios, es necesario ir a la par con esta aceleración, generando competitividad en el desarrollo de sistemas, es por esto que se ha visto la necesidad de la implementación de metodologías ágiles, permitiendo potenciar el desarrollo de software a gran escala.

Según Canós, las metodologías ágiles constituyen un conjunto de técnicas y buenas prácticas para la gestión de proyectos, no sólo de software sino también de otros campos.

Las metodologías ágiles se basan en un concepto de desarrollo iterativo e incremental, involucrando los requerimientos con el desarrollo, aportando mucha efectividad en entornos muy volátiles, y en donde se prioriza la reducción de tiempos de desarrollo pero con calidad.

Las metodologías ágiles se enmarcan en un documento denominado “Manifiesto ágil”, que en resumen describe 4 puntos:

- El individuo y las interacciones
- El software que funciona
- La colaboración con el cliente
- La respuesta al cambio

Los valores anteriores son características que diferencian un proceso ágil de uno tradicional. Las metodologías ágiles se diferencian de las tradicionales principalmente con toda la forma de gestionar el equipo y a su organización, además del proceso principal de desarrollo. (Canós et al., 2013)

El uso de metodologías ágiles no significa falta de documentación o control del proyecto, el objetivo de las mismas es reducir el empleo de artefactos que no son necesariamente imprescindibles para llegar al objetivo del proyecto, para con esto reducir tiempos, aumentar la eficiencia y reducir el coste, respondiendo activamente a cambios inesperados generados durante el desarrollo, lo cuál encaja dentro del contexto de nuestro proyecto, debido a que el proceso para postulación se ha venido cambiando constantemente, y aún continúa en modificaciones dentro del flujo de revisiones de las propuestas.

Cada metodología ágil tiene su propiedad que la caracteriza de otras, aunque todas se circunscriben al manifiesto descrito anteriormente. A continuación se muestran algunas de las metodologías ágiles más usadas en la actualidad, para determinar la más acoplable a nuestro proyecto.

1.2.2.1. Crystal Methodologies

Canós (2013) define a Crystal Methodologies como “Un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo y la reducción al máximo de número de artefactos producidos”.

En esta metodología, el desarrollo de software se gestiona como un juego colaborativo de invención y comunicación, y se aplica según los recursos a utilizar. El equipo de desarrollo constituye un factor primordial, por lo tanto el mejorar sus habilidades y destrezas es una de las principales preocupaciones y dónde se enfocan los esfuerzos, así como el desarrollo de políticas de trabajo en equipo bien definidas. Estas políticas por supuesto dependen del tamaño del equipo de desarrollo, y para esta gestión se establece una categorización por colores (Canós et al., 2013).

1.2.2.2. Dynamic Systems Development Method (DSDM)

Canós (2013) asegura que DSDM “Define el marco para desarrollar un proceso de producción de software. Sus principales características son: es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos”.

Esta metodología consta de cinco fases:

- Estudio de viabilidad
- Estudio del negocio
- Modelado funcional
- Diseño y construcción
- Implementación.

El modelado funcional, el diseño y construcción y la implementación son iterativas, asimismo debe existir una realimentación a todas las fases anteriores (Canós et al., 2013).

1.2.2.3. Adaptative Software Development (ASD)

Canós (2013) asegura de ASD: “Sus principales características son: iterativo, orientado a los componentes software más que a las tareas y tolerante a los cambios. El ciclo de vida que propone tiene tres fases esenciales: especulación, colaboración y aprendizaje”.

En la primera fase, especulación, se inicia el proyecto y se proyectan las características que tendrá el software; en la segunda fase, colaboración, se desarrollan las características y finalmente en la tercera fase, aprendizaje, se revisa la calidad y se entrega al cliente. Aquí

es necesaria la revisión de cada uno de los componentes para conseguir un aprendizaje de los errores y reiniciar el ciclo de desarrollo (Canós et al., 2013).

1.2.2.4. Feature-Driven Development (FDD)

Según Canós (2013), FDD: “Define un proceso iterativo de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software”. (Canós et al., 2013).

1.2.2.5. Lean Development (LD)

Esta metodología fue definida por Bob Charette's desde su experiencia en proyectos con la industria japonesa de automoviles en los años 1980 y ha sido usada en cuantiosos proyectos de telecomunicaciones de Europa.

Según Canós (2013) en LD: “los cambios se consideran riesgos, pero si se manejan adecuadamente se pueden convertir en oportunidades que mejoren la productividad del cliente. Su principal característica es introducir un mecanismo para implementar dichos cambios” (Canós et al., 2013).

1.2.2.6. KANBAN

Kanban es uno de los principales elementos del proceso de Toyotista, es una sencilla herramienta de control a través de la cual sacó sistema de producción (que es el eje de la producción Justo a Tiempo) se gestiona. Un sistema de control de producción Kanban utiliza señales visuales simples, para vigilar el movimiento de materiales entre los centros de trabajo, así como la producción de nuevos materiales para reemplazar a los enviados al siguiente centro de trabajo. Además de la Kanban que ayuda en el proceso de seguimiento, para controlar la calidad de los artefactos generados.

Kanban es ampliamente utilizado también en la industria del software, esto tiene sus principales conceptos adaptados para que coincida con el proceso de desarrollo de software, la obra de construcción de una nueva funcionalidad a un sistema sólo se genera a partir del momento en que una funcionalidad anterior ya tiene puesto en marcha.

Kanban, en este contexto, busca mejorar los procesos, equipos y proyectos. Es útil para las empresas que buscan mejorar constantemente sus procesos, además de mejorar su productividad y su relación con clientes (Souza, L'Erario, Fabri, & Gonçalves, 2016).

Según Ahmad, el principal objetivo de esta metodología es evaluar los trabajos en curso, denominado WIP (Work in Progress). Según los autores de esta evaluación es la propuesta

para mostrar cuando un componente de software se puede Architected, codificado y probado (M. O. Ahmad, Markkula, & Oivo, 2013).

1.2.2.7. Extreme Programming (XP)

Según afirma Figueroa, Extreme Programming fue formulada por Kent Beck, ésta metodología se centra en potenciar las relaciones entre el equipo, se preocupa por la capacitación y el continuo aprendizaje de los desarrolladores generando un buen ambiente de trabajo.

Se diferencia de las demás principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.

Sus principales características son:

- Desarrollo iterativo e incremental
- Pruebas unitarias contiuas
- Programación por parejas
- Frecuente interacción del equipo
- Corrección de errores
- Refactorización del código
- Propiedad del código compartida
- Simplicidad en el código.

Sus fuertes son la simplicidad y la comunicación. Con más comunicación resulta más fácil identificar qué se debe y qué no se debe hacer (Figueroa et al., 2008).

1.2.2.8. SCRUM

Canós afirma que “Scrum define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años. Especialmente indicada para proyectos con un rápido cambio de requisitos”, como es el caso de nuestro proyecto.

En SCRUM, el desarrollo se realiza en forma iterativa e incremental, cada ciclo o iteración termina con una pieza de software ejecutable que incorpora una nueva funcionalidad.

Se enfoca en priorizar el trabajo en función del valor que tenga para el negocio, maximizando la utilidad de lo que se construye y el retorno de inversión. Su conjunto de reglas está basado en los principios de inspección continua, adaptación, autogestión e innovación.

Sus principales características se pueden sintetizar en dos. Primero, el software se desarrolla mediante iteraciones denominadas *sprints*, que duran un máximo de 30 días. El resultado de cada *sprint* es un incremento al software, una característica ejecutable que se presenta al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, donde la reunión diaria de 15 minutos del equipo constituye una vital importancia, puesto que se discute las características y problemas encontrados en el desarrollo (Canós et al., 2013).

Como ya se mencionó, en Scrum, cada iteración se llama *Sprint*, aunque generalmente su duración la decide el equipo, el *Sprint* debe finalizar con un prototipo operativo. Lo que se va a implementar en el Sprint, la funcionalidad del prototipo, proviene de la Pila del Producto (Product Backlog), que contiene un conjunto de items, normalmente, requisitos funcionales o historias de usuario.

En Scrum, la figura del Propietario del Producto (Product Owner), que vendría a ser el usuario o el cliente, es el responsable de gestionar el Product Backlog y de crear las historias de usuario.

Una vez seleccionadas las historias de usuario que se van a desarrollar en el Sprint, el equipo técnico las descompone en tareas, las cuales forman la Pila del Sprint (Sprint Backlog), que será inamovible durante el Sprint.

La figura 1 muestra el resumen del proceso que abarca Scrum.

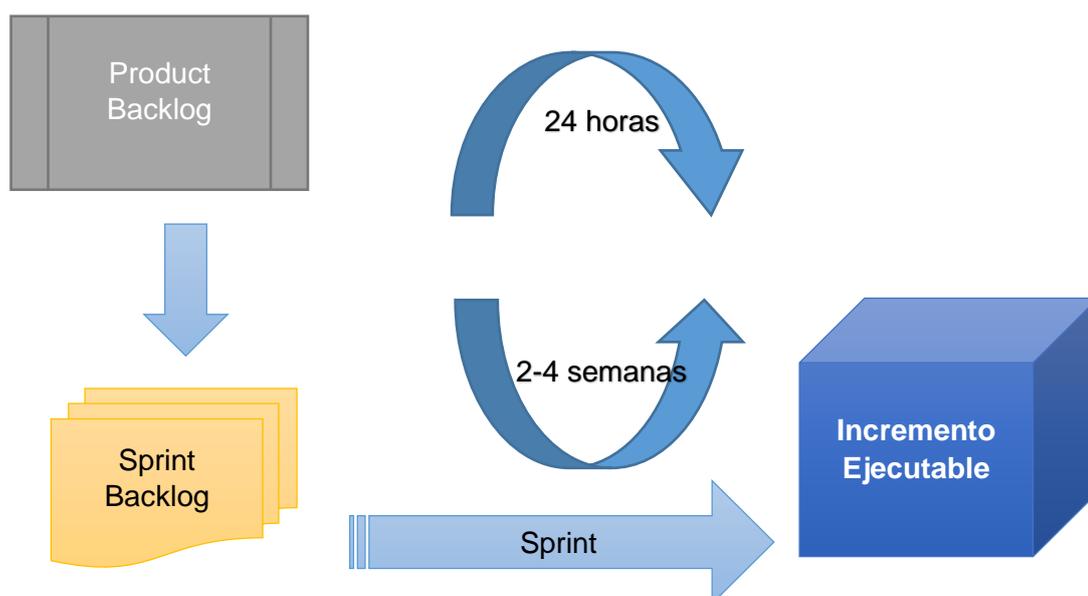


Figura 1. Proceso Global Scrum
Fuente: El autor.

Elaboración: El autor.

Scrum presenta un esquema de reuniones, que constituyen los artefactos de esta metodología.

El libro Metodología de Scrum de Gallego (2012) describe las reuniones y roles:

- **Reunión de planificación del Backlog**

En esta reunión se especificará un documento en el que se mostrarán los requisitos del sistema por prioridades. Se concretará también la planificación del primer sprint denominado Sprint 0, en donde se decidirá los objetivos y el trabajo que hay que realizar para esta iteración. Se generará además en esta reunión un Sprint Backlog, que constituye la lista de tareas y es el objetivo más importante del Sprint.

- **Reunión de seguimiento del Sprint**

Consiste en reuniones diarias en las que las 3 preguntas principales para evaluar el avance de las tareas serán:

- ✓ “¿Qué trabajo se realizó desde la reunión anterior?”
- ✓ “¿Qué trabajo se hará hasta una nueva reunión?”
- ✓ “Inconvenientes que han surgido y qué hay que solucionar para poder continuar”.

- **Reunión de revisión del Sprint**

Al finalizar el Sprint se efectuará una revisión del incremento o funcionalidad que se ha generado. Se mostrarán los resultados finales y una demo o versión, mejora la retroalimentación con el cliente.

Ahora describiremos los roles según el mismo libro:

- **Product Owner:** Es la persona encargada de tomar las decisiones, y conoce el negocio del cliente y lo que quiere desarrollar. Se encarga de escribir y conceptualizar las ideas del cliente, ordenarlas por prioridad y colocarlas en el Product Backlog.
- **Scrum Master:** Es la persona encargada de justificar que el modelo y la metodología funcionan. Se encargará de eliminar todos los obstáculos que ocasionen que el proceso no fluya y estará en constante contacto con el cliente y con los interesados.
- **Equipo de desarrollo:** Es un equipo pequeño, de preferencia de unas 5-9 personas y tienen jurisdicción para organizar y tomar decisiones para conseguir su objetivo. El

equipo está directamente involucrado en la estimación del esfuerzo, de tiempos de desarrollo y prioridad de las tareas del Backlog.

- **Usuarios:** Es el receptor final del producto.
- **Stakeholders:** Son las personas involucradas directa o indirectamente con el proyecto, a las que les producirá un beneficio. Los Stakeholders deben participar en las reuniones de revisión del Sprint.
- **Managers:** Los managers son encargados de tomar las decisiones finales y participan en la selección de los objetivos y de los requerimientos (Gallego, 2012).

Para nuestro proyecto se ha considerado la vinculación con DevOps, que se detalla posteriormente.

1.2.2.9. DevOps

Siguiendo la línea de metodologías ágiles y el desarrollo ágil de sistemas para entornos volátiles tenemos la metodología DevOps.

IBM en su página oficial define DevOps como: “Un conjunto de principios, prácticas y productos que ayudan a que las organizaciones entreguen software de alta calidad al mercado con mayor rapidez, al tiempo que minimizan costos y riesgos.”

El enfoque DevOps acelera y brinda apoyo a la innovación de software que esté planificando, desarrollando, probando y entregando. Independientemente de si su enfoque está en el desarrollo de aplicaciones móviles, alojamiento en nube o análisis de grandes datos, continuamente podrá liberar mejor software y servicios más rápido, con un costo más bajo y menor riesgo.

IBM DevOps funciona comprometiendo y alineando a todos los participantes de los equipos de negocios software del ciclo de vida del software; arquitectos, desarrolladores y probadores: y operaciones y producción de TI, alrededor de una meta única y compartida; innovación sostenida, impulsada por la entrega continua y conformada por la retroalimentación continua.



Figura 2. Ciclo de vida DevOps
 Fuente: <https://www.ibm.com/developerworks/community/blogs/>
 Elaboración: IBM

DevOps se basa en cuatro principios:

1. Cultura de aprendizaje colaborativo

La sola idea de desplegar cientos de cambios todos los días al tiempo que se mantiene simultáneamente un entorno de producción estable, confiable y seguro los desanima. Por eso, cultivar una cultura de aprendizaje continuo y experimentación libre de riesgos es necesario para que DevOps se implemente.

El aprendizaje continuo, combinado con el conocimiento que se comparte en forma amplia y transparente, ayuda a que los equipos identifiquen y repitan patrones exitosos.

La experimentación libre de riesgos permite que los equipos prueben cosas nuevas, aprendan de los fracasos y continuamente incorporen innovaciones en el ciclo de vida del software.

En una cultura de aprendizaje y experimentación, los equipos comparten descubrimientos y buenas prácticas libremente, crean rituales que premian la toma de riesgos y se dan tiempo para mejorar sus contribuciones todos los días.

2. Métodos de agilidad y automatización para acelerar la innovación.

DevOps utiliza las siguientes prácticas ágiles:

- Desarrollar
- Probar
- Desplegar
- Validar
- Ajustar

Éstas prácticas ágiles son la estructura y la disciplina que debe mantener todo el ciclo de producción del software, y las mismas se desarrollaran en un entorno de prueba casi igual al de producción ya que son una parte fundamental del acogimiento de DevOps, y aseguran un entrega de software iterativo según el cliente lo requiera.

Además DevOps automatiza el despliegue del software. Se enfoca en tratar de eliminar actividades y conductas que generan errores, por esto para eliminar estos errores manuales se automatiza el despliegue, haciendo que el software esté disponible más rápido.

En DevOps, predefinimos los procesos tratando que sean re aplicables para cada iteración y entrega de software, por lo tanto establecemos los procesos para llevar un control de las versiones y ubicar fácilmente los errores. Mantenemos una homogeneidad en los procesos para que no existan despliegues fallidos.

3. Los ciclos de retroalimentación reducen el tiempo hasta la retroalimentación.

La retroalimentación continua estimula el mejoramiento continuo — tanto para el software como para los procesos de entrega de software. En un flujo DevOps, el tiempo hasta la retroalimentación es breve, de modo que los ajustes pueden hacerse en etapas más tempranas y de manera más económica. A través del ciclo de vida, los equipos de desarrollo y de entrega controlan la calidad operacional de manera continua (en pos de la integración, funcionalidad, rendimiento y seguridad) para validar el software. Una vez que este se encuentra en producción, las mediciones capturan la experiencia del cliente.

Además, la retroalimentación se amplifica, de tal forma que todos los contribuyentes al ciclo de vida del software pueden aprender de esta retroalimentación —tanto interna como externa— y potencialmente asumir nuevos riesgos en respuesta (porque la cultura apoya la experimentación como algo fundamental para la innovación).

A fin de comprender y amplificar la retroalimentación, todos — líneas de negocios, desarrollo, aseguramiento de la calidad, seguridad, arquitectura y operaciones de TI — necesitan acceso a las mediciones que se recopilan. Y estas mediciones deben

estar en una forma que pueda consumirse y sobre la que pueda actuarse con facilidad.

4. El sistema entero moldea las metas.

IBM DevOps pone énfasis en el rendimiento colaborativo del sistema como un todo, en lugar de en el rendimiento y el resultado de contribuyentes o equipos individuales.

En un flujo de trabajo tradicional, los requisitos se identifican (lo hacen los equipos de negocios), se construyen (lo hacen los equipos de desarrollo y de pruebas) y se pasan a operaciones de TI para que los desplieguen y los entreguen a los usuarios. Cada equipo funciona en su propio silo, a menudo con herramientas diferentes y metas que compiten internamente. Los equipos de negocios quieren soluciones que sean rentables (controlar los costos); los equipos de desarrollo y pruebas quieren soluciones que traten tantas funciones y defectos nuevos como sea posible (maximizar los cambios); operaciones de TI quiere soluciones que sean estables, seguras y no disruptivas (minimizar los cambios).

En un flujo de trabajo DevOps, los equipos de negocios se involucran con los clientes a menudo y desde el principio para modelar y remodelar los requisitos. Los equipos de desarrollo y pruebas trabajan en colaboración con los equipos de operaciones y utilizan metas y procesos compartidos para desarrollar soluciones estables y fáciles, para que operaciones de TI las entregue y las mantenga. Por ejemplo, se les hace seguimiento a las tareas de entrega en un solo lugar, la integración continua y las construcciones oficiales se unifican, y se usa la misma herramienta de despliegue para todos los entornos de desarrollo y de prueba, por eso cualquier error se detecta y se soluciona enseguida.

Todo el sistema contribuye a que las entregas sean rápidas y exitosas. En un flujo de trabajo DevOps, la calidad (o el valor de negocio) de una solución mejora continuamente a medida que cada equipo aporta su contribución.

Buenas prácticas para aplicar IBM DevOps:

- **Planificar, hacer seguimiento y versionar todo**

La planificación continua de los negocios garantiza la transparencia. Tanto si se trata de una aplicación totalmente nueva o de un dispositivo para cumplir con el requisito de un cliente, — o de un cambio en un parche para el sistema operativo, un cambio en un parámetro de middleware, un caso de prueba actualizado o una versión depurada; — todo debe estar ligado a un requisito y un elemento de trabajo. Esta

asociación permitirá planificar las actividades subsiguientes, comunicarlas y saber cuándo se finalizan.

- **Poner todo el en panel de instrumentos**

El desarrollo colaborativo habilita la visibilidad. El estado de las aplicaciones, el rendimiento de las aplicaciones, el desarrollo de calidad y estatus, los problemas de producción, la eficiencia del equipo y los cuellos de botellas deben ser visibles a través de los paneles de instrumentos para que operaciones de TI sepa lo que se viene y pueda prepararse de la manera adecuada.

- **Automatizar todo**

La automatización continua de la gestión de entregas garantiza la repetitividad. Las hojas de cálculo que inducen a error y los procesos legados pueden entorpecer a su equipo. Reducimos tiempo, errores y costos automatizando el despliegue de aplicaciones, la configuración de middleware y los cambios en la base de datos en entornos de desarrollo, prueba y producción. Y con la nube, hasta podrá automatizar la provisión y el despliegue de las máquinas virtuales, middleware y códigos de aplicación.

- **Hacer pruebas de todo**

Las pruebas continuas garantizan la calidad. Toda la automatización del mundo no le será de ayuda si no podemos estar seguros de que las cosas funcionan. Necesitamos probar continuamente las configuraciones de VM, la instalación de middleware, los scripts de despliegue y, por supuesto, las aplicaciones, el software o el servicio.

Hay que mantener los scripts de despliegue, las definiciones de infraestructura para todos los entornos, casos de prueba y código en un sistema SCM para permitir una iteración rápida y libre de errores.

- **Monitorear y auditar todo**

El monitoreo continuo garantiza la responsabilidad. Los agentes monitorean las aplicaciones; los registros de auditoría capturan las acciones de despliegue; los elementos de trabajo instrumentan las actividades del equipo. («IBM developerWorks en español», 2015).

1.3. Lenguajes y herramientas de programación

A continuación se presentan los frameworks de programación más destacados y acordes a nuestro proyecto.

1.3.1. PHP

A pesar de ser de los lenguajes con ya menor proyección y uso en la actualidad para el desarrollo web, consideré aún útil detallar uno de los frameworks con mayor agilidad, de los muchos que se han desarrollado para tratar de mantener en pie este lenguaje, y aún darle uso para que pueda ser competitivo con las demás nuevas tecnologías.

PHP es un lenguaje de código abierto, que nació para ser utilizado con HTML.

PHP fue en el 2010 uno de los lenguajes más interesantes para crear scripts del lado del servidor. Esto se debía a que muchos proveedores ofrecían PHP, y además a precios económicos, y se pueden enlazar con bases de datos MySQL u ODBC de forma muy sencilla y segura. Además, PHP no es difícil de aprender. (Spona, 2010).

El framework descrito a continuación es de los considerados más útiles para el desarrollo ágil, que considera el alcance de nuestro proyecto.

1.3.1.1. Zend

Aguirre define a Zend: “Zend framework considerado open source para PHP5 desarrollado por Zend, empresa encargada de la mayor parte de las mejoras hechas a PHP, por lo que se podría decir que es el framework oficial” (Aguirre & Esquivel, 2012).

Zend es un framework modular, liviano, y extensible, su diseño minimalista permite libertad, proporcionando distribuciones, arreglos y plantillas de códigos reutilizables.

Características

- Orientado a objetos al 100%.
- Tiene soporte para localización e internacionalización de aplicaciones.
- Se puede configurar los proyectos por línea de comandos.
- Sus componentes con bajo acoplamiento permiten usarlos de manera independiente.
- Tiene un módulo llamado Zend_Test, que facilita las pruebas de aplicaciones.
- Tiene soporte con adaptadores para varios tipos de bases de datos.
- Dispone de mecanismos para la autorización y autenticación de usuarios, envío de correos electrónicos, memoria cache en varios formatos, creación de servicios web, entre otros (Aguirre & Esquivel, 2012).

1.3.2. Ruby

En la página oficial de Ruby, define al lenguaje como: *“Un lenguaje de programación dinámico y de código abierto enfocado en la simplicidad y productividad. Su elegante sintaxis se siente natural al leerla y fácil al escribirla.”*

En la misma página describe los componentes más importantes de este lenguaje, entre ellos los que más hemos podido destacar son:

- Ruby es totalmente libre. No sólo gratis, sino también libre para usarlo, copiarlo, modificarlo y distribuirlo.
- En Ruby, todo es un objeto. Se le puede asignar propiedades y acciones a toda información y código.
- Ruby es considerado un lenguaje flexible, ya que permite a sus usuarios alterarlo libremente. Las partes esenciales de Ruby pueden ser quitadas o redefinidas a placer.
- A diferencia de otros lenguajes de programación orientada a objetos, Ruby se caracteriza por su intencional herencia simple. Sin embargo, Ruby incorpora el concepto de módulos (llamados categorías en Objective-C), que son colecciones de métodos.

Ahora, para el desarrollo con Ruby disponemos de dos frameworks con los que podemos desarrollar, éstos son Rails y Sinatra, ambos con cualidades que las caracterizan en el desarrollo ágil, y se describen a continuación.

1.3.2.1. **Framework Rails**

Rails es un framework diseñado para trabajar con el lenguaje Ruby.

Carrillo en su proyecto define a Rails: Ruby on Rails, también conocido como RoR o Rails, es un framework de aplicaciones web de código abierto escrito en el lenguaje de programación Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). “Trata de combinar la simplicidad con la posibilidad de desarrollar aplicaciones del mundo real escribiendo menos código que con otros frameworks y con un mínimo de configuración. El lenguaje de programación Ruby permite la metaprogramación, de la cual Rails hace uso, lo que resulta en una sintaxis que muchos de sus usuarios encuentran muy legible”.

La filosofía Rails incluye dos grandes principios:

- Don't Repeat Yourself: se lo conoce como DRY, es un principio de desarrollo de software que establece que "Cada pieza de conocimiento debe tener una única representación autorizada, sin ambigüedades, dentro de un sistema." Para no escribir la misma información una y otra vez.
- Convention Over Configuration: Rails tiene opiniones sobre la mejor manera de hacer muchas cosas en una aplicación web, y los valores predeterminados para este conjunto de convenciones, antes que se requiera que se especifique cada detalle a través de archivos de configuración sin fin (Pico Cobra, 2016).

1.3.2.2. Sinatra

Según el libro de Alan Harris, define a Sinatra como: *"Sinatra es un lenguaje de dominio específico, para construir sitios web, servicios web y aplicaciones web en Ruby. Hace hincapié en un enfoque minimalista para el desarrollo, ofreciendo sólo lo que es esencial para manejar peticiones HTTP y entregar respuestas a los clientes"*.

Otra de las características que define el mismo libro, es que la sintaxis de Sinatra es simple y sencilla. Si se desea un despliegue rápido de una API, construir un sitio con el mínimo escándalo y configuración, o crear un servicio web basado en Ruby, Sinatra puede facilitar mucho.

También recalca que Sinatra no es un Framework. Ya que se encuentra sin herramientas ORM incorporadas, no hay archivos de configuración prefabricados y que ni siquiera recibirá una carpeta de proyecto a menos que se la cree uno mismo. Pero que aunque no lo parezca ahora, esto puede ser muy liberador. Las aplicaciones Sinatra son muy flexibles por naturaleza, por lo general no más grandes de lo que se necesita y pueden ser fácilmente distribuidas como gemas (Harris & Haase, 2011).

Además se destaca que Sinatra no obliga al usuario a usar el patrón MVC (Modelo Vista Controlador) que se ve en otros frameworks. En su lugar, Sinatra se enfoca en la "rápida creación de aplicaciones web en Ruby con el mínimo esfuerzo.

1.3.2.3. Comparativa

Entre foros de desarrolladores que han trabajado en proyectos con ambos marcos y enfoques, se destaca que Sinatra, con su enfoque minimalista, es extremadamente útil para proyectos pequeños, ideal para el estilo micro, sin embargo si se va más allá, Rails le ganará a Sinatra.

Konstantin Haase es el encargado actual de Sinatra y siente que ambos atienden a diferentes tipos de aplicaciones: “Ambos están resolviendo un conjunto diferente de problemas, a pesar de que existe una superposición. Si bien Rails es un marco centrado en la creación de aplicaciones web modelo impulsado, Sinatra es una biblioteca para hacer frente a HTTP desde el lado del servidor. Si usted piensa en términos de HTTP peticiones / respuestas, Sinatra es la herramienta ideal. Si usted necesita la plena integración y tanto repetitivo como sea posible, Rails es el camino a seguir”.

Además, un gran número considera que al usar Sinatra, y tener mayor libertad de elegir la arquitectura, lo mismo conlleva a que el desarrollador invierta más tiempo definiendo el modelo y la misma arquitectura, característica que Rails ya la dispone lista para su uso.

1.3.3. Python

Según Marzal (2003): “Python es un lenguaje de programación creado por Guido Van Rossum, con una sintaxis muy limpia, ideado para enseñar a la gente a programar bien. Se trata de un lenguaje interpretado o de script”

En su libro, Marzal, describe las principales características del desarrollo con Python:

- Python es un lenguaje muy expresivo, es decir, los programas Python son muy compactos: un programa Python suele ser bastante más corto que su equivalente en lenguajes como C. (Python llega a ser considerado por muchos un lenguaje de programación de muy alto nivel.)
- Python es muy legible. La sintaxis de Python es muy elegante y permite la escritura de programas cuya lectura resulta más fácil que si utilizáramos otros lenguajes de programación.
- Python ofrece un entorno interactivo que facilita la realización de pruebas y ayuda a despejar dudas acerca de ciertas características del lenguaje. El entorno de ejecución de Python detecta muchos de los errores de programación que escapan al control de los compiladores y proporciona información muy rica para detectarlos y corregirlos.
- Python puede usarse como lenguaje imperativo procedimental o como lenguaje orientado a objetos.
- Posee un rico juego de estructuras de datos que se pueden manipular de modo sencillo. (Marzal & Gracia, 2003)

1.3.3.1. Framework Django

Condori, en su artículo de revista, nos da una perspectiva de lo que abarca Django: “Aparte de las ventajas que tiene por ser framework, Django promueve el desarrollo rápido, se construyen aplicaciones en cuestión de días y con el conocimiento suficiente esos días se pueden reducir a horas”.

Este framework promueve el desarrollo de código limpio al mantener buenas prácticas de desarrollo web en el desarrollo de aplicaciones, utiliza el principio conocido como DRY (Dont Repeat Yourself) que significa también como Una vez y sólo una.

Además Django usa una alteración de la arquitectura MVC (Model View Controller), denominada MTV (Model Template View), que significa Modelo Plantilla Vista, haciendo pragmática la forma de trabajar.

La siguiente analogía ayuda a entender la relación para empezar a trabajar con MTV.

- En Django el modelo continúa siendo el encargado de la estructura de datos.
- En Django la vista se llama Template (Plantilla).
- En Django el controlador se llama View (Vista).

La figura 3 nos hará entender mejor esta relación:

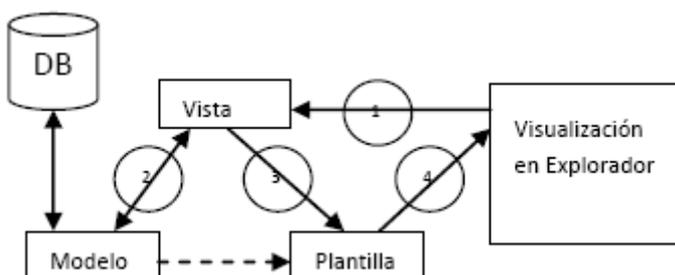


Figura 3. Esquema relación de Django

Fuente. Artículo “Python - DjangoFramework de desarrollo web para perfeccionistas Basado en el Modelo MTV”
Elaboración: Condori Ayala

1. El Navegador envía una solicitud
2. La vista (view) se comunica con el modelo (model) para realizar peticiones de datos.
3. La vista invoca a la plantilla.
4. La plantilla renderiza en lenguaje template de Django la respuesta a la solicitud del navegador.

Veamos que hace cada uno de ellos con un poco más de detalle y algunos conceptos adicionales.

El modelo

El modelo especifica los datos, se encuentra en forma de clases de Python, cada tipo de dato se define mediante una variable con ciertos parámetros, también posee funciones. Con esto nos permite definir y controlar el acceso comportamiento de los datos.

La vista

La vista se presenta en forma de métodos o funciones en Python, sirve para definir el acceso y visualización de los datos, entre otras cosas más. Con el ORM (Object Relational Mapping) de Django escribimos código Python en lugar de sentencias SQL para realizar las consultas que necesitamos en la vista. La vista también realiza tareas comunes, como el envío de mails, la autenticación local o con servicios externos y la validación de datos de formularios. La vista no tiene nada que ver con el estilo de presentación de los datos, sólo se encarga de la gestión de los datos, puesto que la tarea de presentación es responsabilidad exclusiva de la plantilla.

La plantilla

La plantilla es una página HTML que permite predefinir plantillas base reutilizables y contiene algunas etiquetas propias de Django, en sí no solamente crea contenido en HTML sino también XML, CSS, Javascript, CSV, entre otros.(Condori Ayala, 2012).

1.3.3.2. Framework Flask

Flask es un "microframework" dirigido principalmente a pequeñas aplicaciones con requisitos simples.

Flask es un framework minimalista escrito en Python y basado en la especificación WSGI de Werkzeug y el motor de templates Jinja2. Flask es muy flexible gracias a todos sus plugins disponibles que harán escalar, crecer la aplicación y que sea posible añadirle cualquier funcionalidad que deseemos. (Domínguez Purificación, 2015)

Flask, a diferencia de Django, no trae módulos para afrontar las tareas más comunes en el desarrollo web, más bien su prioridad es proporcionar lo mínimo necesario para poner a funcionar una aplicación básica en cuestión de minutos. Perfecto, por ejemplo, para el prototipado rápido de proyectos.

1.3.4. Node.js

IBM, en su página oficial, define a Node.js: Node es un intérprete Javascript del lado del servidor que cambia la noción de cómo debería trabajar un servidor. Su meta es permitir a

un programador construir aplicaciones altamente escalables y escribir código que maneje decenas de miles de conexiones simultáneas en una sólo una máquina física.

Es decir, tendremos un programa de servidor. Los beneficios, y el fuerte de Node.js es su capacidad de manejar los cuellos de botella que presentan los programas de servidor actuales, que usan Java o PHP. Éstos generan un hilo de conexión, que ocupa memoria, al servidor por cada petición del cliente, Node.js cambia este procedimiento y en lugar de eso, dispara un evento desde la misma conexión, por tanto puede manejar miles de conexiones simultáneas.

Esta arquitectura es altamente recomendada para aplicaciones de alta concurrencia de usuarios generando peticiones al servidor, y está pensado para nunca dejar caer el sistema por colapso en la capacidad de conexiones.

1.3.4.1. Framework Express

Castrelo, explica lo que es éste Framework para node.js: Express es un framework de desarrollo de aplicaciones web minimalista y flexible para Node.js.

Ofrece, entre otras características, un router de URL (get, post, put), el cual vamos a usar para capturar los diferentes eventos que se puedan producir, desde cambiar de página, hasta hacer operaciones con la base de datos. Además, Express nos ofrece la posibilidad de crear un proyecto predefinido con las funcionalidades básicas y a partir del cual empezar a desarrollar nuestra aplicación siguiendo los fundamentos del framework. Mencionar también que muchas de las librerías para node.js (como socket.io) son compatibles con Express, de modo que su uso también se simplifica bastante. (Castrelo Cid, 2014)

1.3.5. Java

Francisco Pérez Rodríguez, en su trabajo describe a Java: “Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos y basado en clases que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible”. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como WORA, o "write once, run anywhere"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra.

Los cinco objetivos de java en su creación fueron:

1. Usar el paradigma de POO (programación orientada a objetos).

2. Permitir la ejecución de un mismo programa en múltiples plataformas y sistemas operativos.
 3. Incluir soporte para trabajo en red.
 4. Diseñarse para ejecutar código en sistemas remotos de forma segura.
 5. Ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos, como C++.
- (Pérez Rodríguez, 2013).

Puedo destacar de este lenguaje su estructuración completa, un lenguaje dónde todo está bien definido y se puede estructurar, por tanto al aprender java, se tiene una comprensión completa sobre programación orientada a objetos y a la organización de un proyecto mediante clases y herencia.

Ahora, debido al alcance de nuestro proyecto, necesitamos el uso de Java para programación web. Java Server Pages (JSP) es la herramienta de programación de Java orientada al desarrollo web

1.3.5.1. JSP

Álvarez (2002) define a JSP: “JSP es un acrónimo de Java Server Pages, que en castellano vendría a decir algo como Páginas de Servidor Java. Es, pues, una tecnología orientada a crear páginas web con programación en Java”.

JSP nos permite desarrollar aplicaciones web que se produzcan en variados servidores web, de múltiples plataformas, por ser Java un lenguaje multiplataforma. Las páginas JSP se componen de código HTML/XML combinado con etiquetas de JSP para programar scripts de servidor en sintaxis de lenguaje Java. Así que, las JSP podremos escribirlas con nuestro editor HTML/XML de costumbre. El motor de las páginas JSP está basado en los servlets de Java, que son programas en Java desarrollados para ejecutarse en el servidor, aunque el número de desarrolladores que pueden afrontar la programación de JSP es considerable, puesto que es mucho más sencillo de aprender que los servlets. En JSP se crean páginas de manera similar a como se crean en ASP o PHP. Se generan archivos con extensión .jsp que incluyen las sentencias Java a ejecutar en el servidor dentro de la estructura de etiquetas HTML. Antes de que puedan funcionar los archivos, el motor JSP realiza una fase de traducción de esa página en un servlet, generado en un archivo class (Byte codes de Java). Esta fase de traducción se realiza rutinariamente cuando se recibe la primera solicitud de la página .jsp, aunque se puede optar por la opción de pre compilar en código para evitar el tiempo de espera la primera vez que un cliente solicita la página. (Alvarez, Lazaro, & Mendez, 2002)

1.3.5.2. Spring mvc

Javier Sevilla nos explica más a detalle este framework: “Spring es un framework de código abierto, es una estructura de soporte definida, mediante la cual otro proyecto de software puede ser organizado y desarrollado y al ser de código abierto es de distribución libre. Su creador intentaba reducir la complejidad de las aplicaciones empresariales”.

Una de las principales finalidades de Spring es que mediante JavaBeans sencillos, es posible crear funcionalidades antes sólo alcanzables para los EJB. Otro carácter especial que tiene este framework es que no ha de estar de manera obligada al lado servidor, así también aplicaciones más sencillas se pueden beneficiar.

La inyección de dependencia así como el soporte para la programación orientada a aspectos son los conceptos principales para la definición del framework.

Spring está dividido en una serie de módulos bien definidos y que juntos hacen que se pueda desarrollar aplicaciones empresariales bien definidas. Todos los módulos se construyen sobre el contenedor del núcleo de forma modular, haciendo completamente opcional el uso de cada módulo (Javier Sevilla Sánchez, s. f.).

Mayor Martín (2014), además nos dice que: “Spring MVC es un framework basado en peticiones, además define el patrón Strategy que da a las interfaces todas las responsabilidades que debería tener un framework moderno basado en peticiones. El objetivo de cada interfaz debe ser simple y claro para que sea más fácil para los usuarios de Spring MVC escribir sus propias implementaciones”.

El patrón MVC determina el camino para generar un código limpio y reutilizable. Todas las interfaces están vinculadas a la API Servlet, con esta relación se mantiene las características de la API de Servlet disponibles para los desarrolladores, al mismo tiempo que se entrega un alto nivel de abstracción para hacer más sencillo el trabajo con esta API. La clase DispatcherServlet constituye el controlador principal de la estructura y es el responsable de la delegación del control a las variadas interfaces durante las fases de ejecución de una solicitud HTTP.

1.3. Bases de datos

Ahora se describirán los motores de base de datos pertinentes a nuestro proyecto.

1.4.1. Base de datos NoSQL

Por el avance del Cloud Computing (CC), no sólo es necesario la adaptación en frameworks de programación, sino que, para ir a la par con éste rápido avance, que prioriza el

dinamismo de datos y transacciones para crear sistemas escalables y adaptables a entornos cambiantes, también requerimos una gestión de transaccionalidad de datos que haga frente a los mismos.

Para esto se han creado las bases de datos NoSQL, o bases de datos no relacionales.

A. Martín en su investigación explica: “El termino NoSQL hace referencia a sistemas de Base de Datos que no son DMBS tradicionales. Los conceptos fundamentales de este tipo de Bases de Datos NoSQL son la escalabilidad, la distribución y el manejo de datos no estructurados”. Estas características son primordiales frente al gran avance de CC, y a los múltiples servicios que requieren replicación distribuida. De este modo se puede asegurar que la información en el Cloud siempre esté disponible, y además gestionar el gran crecimiento de información y su variabilidad de formatos. Las bases de datos NoSQL son sistemas de almacenamiento de información que no cumplen con el esquema entidad-relación. Tampoco utilizan una estructura de datos en forma de tabla donde se van almacenando los datos sino que para el almacenamiento hacen uso de otros formatos.

Entre las ventajas de las NoSql podemos encontrar:

- No admite consultas complejas o join por lo cual las operaciones son simples
- Se ejecutan en máquinas con pocos recursos, a diferencia de los sistemas basados en SQL, ya que requieren de poco cómputo por lo tanto se pueden montar en máquinas de un coste más reducido.
- Escalabilidad horizontal: es decir tiene una arquitectura que posee la capacidad de distribuir y cargar los datos lo más uniformemente posible, en tantos servidores como sea viable, con una arquitectura shared-nothing.
- Pueden manejar gran cantidad de datos: ya que utiliza una estructura distribuida, en muchos casos mediante tablas Hash.
- No genera cuellos de botella: el principal problema de los sistemas SQL es que necesitan transcribir cada sentencia para poder ser ejecutada, y cada sentencia compleja requiere además de un nivel de ejecución aún más complejo, lo que constituye un punto de entrada en común, que ante muchas peticiones puede hacer al sistema cada vez más lento. (Martín, Chávez, Murazzo, Rodríguez, & Valenzuela, 2015)

1.4.2. MongoDB

A. Martín, también nos da una explicación de este esquema de base de datos no relacional: “MongoDB guarda la estructura de los datos en documentos tipo JSON con un esquema dinámico llamado BSON, no existe un esquema predefinido”.

Aquí los elementos de datos se llaman *documentos* y se guardan en *colecciones*.

Cada colección puede tener un indefinido número de documentos.

Comparando con una base de datos relacional, se puede decir que las colecciones son como tablas y los documentos son los archivos.

La diferencia es que en una base de datos relacional cada archivo en una tabla tiene la misma cantidad de campos, mientras que en MongoDB cada documento en una colección puede tener diferentes campos. En un documento, se pueden agregar, eliminar, modificar o renombrar nuevos campos en cualquier momento, ya que no hay un esquema predefinido. MongoDB soporta la búsqueda por campos, consultas de rangos y expresiones regulares.

El sistema permite al usuario personalizar una consulta en tiempo real y pueden devolver un campo específico del documento pero también puede ser una función definida por el usuario. Trabaja bajo un sistema de indexación similar a los de las base de datos relacionales. Soporta el tipo de replicación maestro-esclavo. El maestro puede ejecutar comandos de lectura y escritura. El esclavo puede copiar los datos del maestro y sólo se puede usar para lectura o para copia de seguridad, pero no se pueden realizar escrituras. El esclavo tiene la habilidad de poder elegir un nuevo maestro en caso de que se caiga el servicio con el maestro actual. MongoDB se ejecuta en múltiples servidores, balanceando la carga y/o duplicando los datos para poder mantener el sistema funcionando en el caso que exista un fallo de hardware.

Utiliza la función MapReduce para procesamiento de datos por lotes y operaciones de agregación. Incluso permite la utilización de índices geoespaciales, que si bien no es necesario para las operaciones normales o de desarrollo de aplicaciones, puede ser útil para solucionar problemas y para una mayor comprensión.

Incluso MongoDB se puede utilizar con un sistema de archivos, por su capacidad para el balanceo de carga y replicación de datos utilizando múltiples servidores para el almacenamiento de archivos, mediante el uso de GridFS. Cuando los archivos exceden el tamaño límite del documento BSON que es de 16 MB, divide el archivo y almacena cada uno de esos trozos en un documento aparte.

MongoDB tiene drivers oficiales para los siguientes lenguajes de programación: C, C++, C#, Erlang, Haskell, Java, Java Script, Lisp, node.JS, Perl, PHP, Python, Ruby, Scala. El código binario está disponible para los sistemas operativos Windows, Linux, OS X y Solaris.

Un aspecto importante cuando se decide instalar un ambiente cloud es contar con infraestructura de almacenamiento sólida y escalable de almacenamiento que ofrezca disponibilidad, eficiencia y protección de los datos. (Martín et al., 2015)

1.5. Servidores Cloud

1.5.1. Heroku

Heroku es una plataforma desarrollada por desarrolladores para desarrolladores. Debido a la importancia para el desarrollador sobre qué tan ágil se puede implementar su aplicación y que esta llegue al mercado, así mismo la rapidez con la que la misma se pueda modificar y ser productiva.

Heroku permite subir una aplicación usando su plataforma en lugar de configurar un servidor particular, en dónde el desarrollador se encarga de gestionar el versionamiento de sistema operativo, las dependencias, librerías, bases de datos, seguridad, etc. Heroku se encarga de esta gestión.

En Heroku no existen servidores, Heroku dispone de una capa de abstracción y ofrece una gran cantidad de bases de datos como servicio a los que el desarrollador puede conectarse, por lo que el desarrollador no tiene que preocuparse acerca de los detalles sobre como el servidor de base de datos está configurado y no necesita preocuparse sobre como los servidores están actualizados y en línea. Heroku es una plataforma que se encarga de todas estas cosas y permite al desarrollador integrar con Heroku como desee. (Middleton, So, & Schneeman, 2013).

Los lenguajes que actualmente soporta Heroku son ocho: Ruby, Java, Clojure, Python, Nodejs, Scala, Play, y PHP.

Heroku permite al desarrollador hacer despliegues de su aplicación usando el Software de control de versiones Git. Una de las ventajas es el poder hacer rollbacks a un commit anterior desde el panel de control de Heroku, para regresar a una versión anterior del sistema sin perder tiempo buscando errores en código.

También provee la agilidad de ver logs de cada componente de la aplicación, y de cada proceso, por tanto se puede monitorear una aplicación desde una terminal para visualizar el tráfico.

Heroku permite escalar, hacer crecer cada componente de una aplicación independientemente sin afectar funcionalidad ni el rendimiento.

Una de las ventajas de Heroku es el pago por uso, es decir, al escalar la aplicación Heroku cobra por el tiempo que el Dyno esté en uso, y no por la tasa mensual total.

La desventaja principal al usar la versión Free, es que el Dyno pasa a un estado de inactivo, y la primera petición de procesamiento por un usuario provoca una ralentización inicial de la aplicación en el primer acceso de un usuario.

1.5.2. Softlayer

SoftLayer, una empresa de IBM, proporciona infraestructuras en la nube como servicio desde un número creciente de centros de datos y puntos de presencia en la red distribuidos por todo el mundo. Sus clientes van desde startups de Internet a empresas globales.

Sus productos y servicios incluyen servidores virtuales y dedicados, entornos de red, soluciones Big Data, soluciones de nube privada, etc. Sus ventajas exclusivas incluyen la primera topología de "red dentro de una red" del sector, para un acceso real desde la web, un portal de clientes fácil de usar y una API robusta para el acceso remoto completo a todas las opciones de gestión de productos y servicios.

SoftLayer fue fundada en el año 2005 y sus oficinas centrales están en Dallas, Texas. Fue adquirida por IBM en julio de 2013.

SoftLayer aporta la infraestructura de nube de mayor rendimiento disponible. Una plataforma que abarca centros de datos por todo el mundo, formados por la más amplia gama de opciones de informática en la nube, y que lo integra y automatiza todo. (IBM, s. f.)

1.5.3. Azure

Azure ofrece modelos de Infraestructura como Servicio (IaaS) y Plataforma como servicio (PaaS) que permiten construcciones flexibles, despliegue y administración de aplicaciones, a través de una red global de data centros administrados de Microsoft.

Microsoft Azure soporta muchos diferentes lenguajes de programación, herramientas y frameworks, incluyendo los específicos de Microsoft y software y sistemas de terceros. (Souidi, Boccio, Mierzwa, & Aguilar, 2015)

Microsoft Azure, es bastante similar a Heroku, en cuanto a su instalación y despliegue de aplicaciones, pues utiliza los mismos requisitos de instalación y realiza los despliegues a través de comandos git.

De la misma manera ofrece una suscripción gratuita, y permite escalar aplicaciones a medida que va creciendo y se va incrementando la capacidad de procesamiento necesaria para que la aplicación funcione correctamente. Una vez que se sobrepasa la tasa de instancias gratuitas que ofrece Azure, permite escalar y establece una suscripción de pago por hora.

1.5.4. Google App Engine

De acuerdo con Kevin Gibbs, que es el líder técnico de App Engine, Google App Engine es "un sistema que expone varios elementos de la infraestructura escalable de Google para que pueda escribir aplicaciones de servidor encima de ellos". Simplemente se trata de una plataforma que permite a los usuarios ejecutar y alojar sus aplicaciones web en la infraestructura de Google.

Estas aplicaciones son fáciles de construir, fáciles de mantener y fáciles de escalar cuando el tráfico y el almacenamiento de datos necesarios. Al utilizar Google App Engine, no hay servidores para mantener y no se necesitan administradores. La idea es que el usuario sólo sube su aplicación y está listo para servir a sus propios clientes. El usuario puede elegir entre si su producto va a ser servido por el dominio gratuito appspot.com o para permitir que Google Apps lo sirva desde el dominio elegido por el cliente. Google también proporciona al usuario la opción de limitar el acceso de la aplicación dentro de los miembros de su propia organización o de compartirla con el resto del mundo. El paquete de salida es gratuito y obligación adicional. Todo lo que el usuario tiene que hacer es registrarse en una cuenta gratuita, y luego desarrollar y publicar su propia aplicación. El paquete inicial incluye hasta 500 MB de almacenamiento y suficiente potencia de CPU y ancho de banda para servir 5 millones de páginas vistas por mes.

Con este nuevo servicio proporcionado por Google, es realmente fácil crear aplicaciones confiables que funcionan bajo carga pesada y que utilizan grandes cantidades de datos. Varias características clave se incluyen en el entorno:

- Servicio web dinámico, con soporte completo para tecnologías web comunes.
- Almacenamiento persistente con consultas, clasificación y transacciones.
- Escalado automático y balanceo de carga.
- APIs para autenticar usuarios y enviar correo electrónico utilizando Cuentas de Google.
- Un entorno de desarrollo local completo que simula Google App Engine en el equipo del usuario. (Zahariev, 2009)

1.6. Requerimientos

Según el libro *“The Software Requirements: Memory Jogger”* se necesitan seguir los siguientes pasos para determinar una correcta elicitación de los requerimientos desde los usuarios e interesados y superar las muchas dificultades inherentes a la elicitación y análisis de requerimientos.

1. Seleccionar y planear las técnicas de elicitación de requerimientos
2. Establecer metas y expectativas
3. Elicitar los requerimientos
4. Verificar y corregir los hallazgos.
5. Repetir los pasos 1-4 profundizando la comprensión de requisitos por parte del equipo.

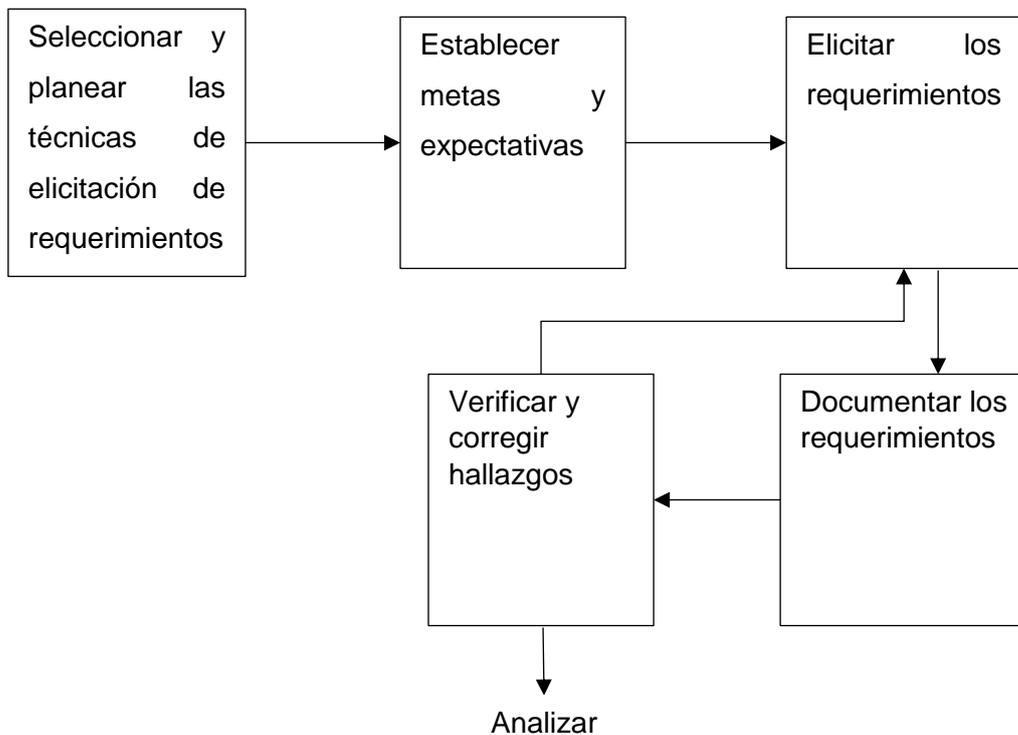


Figura 4. Pasos para elicitación de requerimientos.
Fuente: The Software Requirements Memory Jogger.
Elaboración: El autor.

Las herramientas y técnicas asociadas a cada paso son las siguientes:

Tabla 1. Herramientas para elicitación

Para	Crear
Identificar fuentes de requerimientos	Lista de fuentes de requerimientos
Identificar interesados del proyecto	Categorías de interesados
Describir necesidades y criterios de aceptación de interesados	Perfiles de interesados
Elegir técnicas de elicitación	Combinaciones de técnicas de elicitación identificadas: entrevistas, prototipos exploratorios, talleres facilitados, grupos focales, análisis de tareas de usuario, observación, estudio de la documentación existente
Planear un enfoque de elicitación	Un plan de elicitación de interesados.

Fuente: The Software Requirements Memory Jogger.
Elaboración: El autor.

1.6.1. Herramientas y técnicas

1.6.1.1. *Lista de fuentes de requerimientos*

Es un inventario de las personas, documentos específicos y fuentes de información externa que se necesitará para extraer requerimientos.

Esta lista nos ayuda a identificar fuentes de documentación potencial de requerimientos y permite analizar, revisar, documentar y verificar información de los requerimientos con los interesados.

Para realizarlo se requieren los siguientes tres pasos:

1. Identificar los interesados relevantes de los cuáles se debe obtener requerimientos
2. Identificar cualquier documentación que se pueda usar como fuente de información de requerimientos
3. Identificar fuentes externas de información.

1.6.1.2. Categorización de interesados (Stakeholders)

Son arreglos estructurados de grupos o de individuos que tienen un interés establecido en el producto de software que se está desarrollando.

Se lo realiza para entender quien tiene interés o influencia en el proyecto, quién usará el software y sus salidas, y a quién afectará el producto en alguna manera.

Se encarga de realizar las siguientes tareas:

- Especifica los tipos de personas que tienen requerimientos y necesitan ser involucrados o representados en el proceso de elicitación de requerimientos
- Diferencia los clientes del producto y sus usuarios
- Clarifica que personas y agencias externas se debería consultar.
- Asegura que el equipo considere involucrar a personas que con frecuencia se pasan por alto.

Preguntas clave que se debe responder para su correcta realización:

- ¿Quién afecta o es afectado por el sistema?
- ¿Quién o qué interactúa con el sistema?
- ¿Quién tiene conocimiento relevante para los requerimientos?

Existen tres categorías de interesados:

- Clientes.- Responsables por aceptar o pagar el producto de software
 - Sponsors: Quien autoriza o legitima el desarrollo del producto, hace las contrataciones y paga por el proyecto.
 - Product champions: Asegura que el software se alinee con las necesidades de múltiples comunidades de usuarios. Identifica los usuarios que participarían en el desarrollo de requerimientos.
- Usuarios.- Entran en contacto con el producto de software, o son afectados por éste de alguna manera.
 - Usuarios directos: Son las partes que interactúan directamente con el software (personas, organizaciones, componentes de sistemas o dispositivos).
 - Usuarios indirectos: No interactúan directamente con el sistema pero pueden entrar en contacto con productos generados por el sistema (reportes, facturas, bases de datos y otros bienes tangibles).

- Otros interesados.- tienen conocimiento acerca del producto o un interés en su mantenimiento y desarrollo.
 - Asesores: tienen información relevante sobre el producto de software. Pueden incluir expertos en la materia, personal operacional de soporte, personal de desarrollo y marketing de producto, administradores de sistemas, administradores de datos, personal legal, agencias regulatorias, auditores, entrenadores, personal de recursos humanos y personal de mejora de desempeño.
 - Proveedores: diseñan y producen el software, transformando los requerimientos en el producto final. Se incluyen los miembros del equipo del proyecto (analistas, diseñadores, desarrolladores, testers, gerentes de proyecto).

Para realizarlo se requieren los siguientes pasos:

1. Identificar los interesados, ya sea como clientes, usuarios u otros interesados.
2. Revisar la lista de categorías de interesados con los interesados del proyecto para asegurar que el listado está completo y preciso.
3. Revisar la lista según sea necesario y compartirla con el equipo entero.

1.6.1.3. Matriz de perfiles de interesados

Consiste en la elaboración de una matriz que describe una descripción que caracteriza cada interesado y explica su relación con el proyecto.

Se usa para entender los intereses, preocupaciones y criterios de aceptación del producto para cada interesado del sistema, para descubrir fuentes potenciales de requerimientos, conflictos entre interesados, y para resaltar requerimientos de temas que pueden necesitar tiempo adicional de atención. También puede revelar obstáculos potenciales para la implementación exitosa del producto, y ayudar a definir qué tan involucrado está cada interesado en la elicitación de requerimientos.

Esta herramienta se encarga de:

- Educar al equipo acerca de las expectativas de los interesados
- Proveer al equipo un alto nivel de comprensión de las necesidades de los usuarios
- Descubrir intereses contradictorios de los interesados tempranamente en el proyecto
- Resaltar obstáculos potenciales para la aceptación del sistema de los interesados.

Preguntas clave que esta herramienta responderá:

- ¿Qué son las responsabilidades clave con respecto al desarrollo del sistema y a la implementación de cambios?
- ¿Qué motivaciones, deseos y esperanzas tienen los interesados por el producto de software?
- ¿Qué características y capacidades del software deben ser representadas por cada interesado para ver el producto como un éxito?
- ¿Qué obstáculos, restricciones, o factores limitantes previenen los interesados que pueden poner en peligro la implementación exitosa?
- ¿Qué nivel de comodidad tienen los interesados con la tecnología?
- ¿Existe algún trabajo especial o condiciones ambientales que puedan impactar la capacidad de los interesados para usar efectivamente el sistema?

Pasos para realizarlo:

1. Escribir un breve perfil para cada interesado. Describir
 - a. Rol
 - b. Responsabilidades
 - c. Intereses
 - d. Criterios de aceptación
 - e. Preocupaciones
 - f. Capacidad técnica
 - g. Características y restricciones del entorno de trabajo
2. Incluir los perfiles de interesados en el documento de requerimientos de usuario (si se usa) y en el documento de especificación de requerimientos.

Con esta herramienta podemos vincular con otros modelos. Las responsabilidades de interesados pueden describir a los actores. Las capacidades listadas como “intereses” y “criterios de aceptación” pueden sugerir potenciales casos de uso.

1.6.1.4. Combinación de técnicas de obtención de requerimientos

Se puede combinar una variedad de técnicas de elicitación para asegurar una obtención apropiada de requerimientos de todos los interesados relevantes. Algunas de las técnicas más comunes son:

- Entrevistas con los interesados
- Talleres facilitados
- Prototipos exploratorios
- Grupos enfocados

- Observación
- Análisis de tareas de usuarios
- Estudio de documentación existente
- Encuestas

Las técnicas aplicables al presente proyecto, consideradas las más relevantes acordes a nuestro contexto son: las entrevistas, prototipos exploratorios y el estudio de documentación existente. Estas serán descritas en los capítulos 3 y 4 en la definición y elaboración del plan de elicitación.

1.6.1.5. Entrevistas con los interesados

Son encuentros cara a cara, dónde el entrevistador hace preguntas para obtener información del entrevistado. Pueden ser no estructuradas (con preguntas no predefinidas) o estructuradas (con preguntas definidas por adelantado).

Se la realiza para recolectar información general sobre las necesidades de los interesados, para preguntar a usuarios y clientes el estado de sus necesidades, y para ayudar a descubrir requerimientos conflictivos del software.

Los pasos para realizarla son los siguientes:

1. Identificar a las personas a entrevistar.
2. Preparar las preguntas de la entrevista
3. Programar las entrevistas y organizar la logística para las reuniones.
4. Realizar la entrevista
5. Documentar los resultados

1.6.1.6. Prototipos exploratorios

Son una versión parcial o preliminar del software, creados para explorar o validar requerimientos. También denominados Mock-Up y StoryBoard.

Se hacen para permitir a los usuarios dar una retroalimentación temprana en el proyecto y cooperar activamente en el co-desarrollo con analistas.

Algunas de las funciones que nos facilitan los prototipos son:

- Proveer una versión preliminar parcial del software como una maqueta usando papel, pizarras o herramientas de software.
- Una demostración sencilla de un subconjunto de las funcionalidades del producto, navegación de usuario o interfaces entre sistemas.

- Hacer conceptos abstractos más concretos y requerimientos más tangibles.
- Proveer un producto de trabajo compartido en la cual personas técnicas y de negocios pueden colaborar.

Luego de crear un prototipo se lo puede descartar (prototipo usar-desechar) o usarlo como base para desarrollar el sistema final (prototipo evolutivo). Los prototipos usar-desechar son menos costosos y rápidos de crear que los evolutivos, su propósito es generar información. Los evolutivos son más costosos en cuanto a tiempo, ya que se crean con un fundamento de arquitectura sólido que será retenido para diseño e implementación.

Los pasos para realizar los prototipos son:

1. Seleccionar una porción del alcance del producto para maquetar.
2. Determinar si se debe crear un prototipo de usar-desechar o un prototipo evolutivo.
3. Diseñar y construir el prototipo.
4. Presentar el prototipo y evaluar con usuarios.
5. Documentar resultados.

1.6.1.7. Estudio de la documentación existente

Es una inspección de fuentes de documentación existentes para descubrir información para requerimientos.

Se realiza para descubrir o verificar requerimientos usando una técnica de bajo costo. Este estudio permite al equipo definir características proporcionadas en un software competidor y descubrir requerimientos. También se puede proporcionar información de requerimientos cuando se está reemplazado un sistema existente.

Algunas de sus beneficios son:

- Reusar documentación de software relacionado para proporcionar un punto de inicio para la funcionalidad que debería incluir el producto.
- Permite una ingeniería inversa de requerimientos y otros entregables de software desde los documentos existentes.
- Revela funcionalidad y atributos de calidad necesarios.
- Descubre reglas de negocio que el software podría necesitar cumplir.

Los pasos para realizar el estudio son:

1. Identificar las fuentes de documentación apropiadas a usar.
2. Revisar y analizar la documentación.

3. Crear modelos de análisis de proyecto.

1.6.1.8. Encuestas

Es un método para recolectar información anónimamente de un gran número de usuarios. Puede ser en formato abierto o cerrado.

Se la realiza para obtener una muestra de bajo costo de las reacciones de los usuarios hacia un producto existente o requerimientos propuestos. Permiten analizar rápidamente respuestas y discretamente obtener requerimientos de usuarios que generalmente son inaccesibles. Pueden ayudar a obtener información subjetiva e información sobre la importancia relativa a varias características.

Los pasos para realizar una encuesta son:

1. Identificar el propósito de la encuesta.
2. Determinar el grupo de muestra y el método de recolección.
3. Diseñar las preguntas de encuesta.
4. Evaluar la encuesta antes de distribuirla.
5. Administrar la encuesta.
6. Analizar y documentar los datos.

1.6.1.9. Plan de elicitación de interesados

Es un plan que considera la importancia de las varias necesidades de los interesados y sus contribuciones para el desarrollo del proceso de requerimientos.

Se realiza para decidir quién debería estar involucrado en las diversas actividades de requerimientos y como sería su contribución. Se desarrolla como una estrategia para ayudar a evitar pasar por alto a interesados y requerimientos faltantes. Además ayuda a ganar compromiso de los interesados en su tiempo e involucramiento.

Los pasos para realizar el plan son:

1. Clasificar la importancia de cada interesado en las categorías de interesados.
Usar un esquema de clasificación como MoSCoW.
 - a. Must (M): Esencial para éxito.
 - b. Should (S): Muy importante para recolectar y entender requerimientos de interesados.
 - c. Could (C): Es bueno tener este interesado involucrado, pero menos importante.

- d. Won't (W): No debe ser considerado.
2. Determinar cómo involucrarse con cada interesado categorizado como un M, S, o C. Considerar:
 - a. Grado de implicación.
 - b. Método de participación: Activo, pasivo, indirecto.
 - c. Frecuencia de participación.
 3. Guardar el plan de elicitación en una tabla u otro documento. (Gottesdiener, 2005)

1.7. Servicios Web

Es necesario analizar los servicios web (Web Services) que disponemos ahora en la web para poder establecer una comunicación efectiva entre sistemas.

1.7.1. SOAP

SOAP (Simple Object Access Protocol) es considerado como el formato para definir el intercambio de datos XML entre dos usuarios, independiente de la plataforma o lenguaje de programación de una forma simple y ligera mediante un modelo de empaquetado de datos modular y una serie de mecanismos de codificación de datos. Su estructura cuenta con variadas especificaciones y extensiones como son la seguridad, formato de entrega, procesamiento del mensaje, enrutamiento, etc. Es considerado junto con el lenguaje de definición de servicios WSDL, un estándar, completamente dependiente del formato XML para la codificación de datos suponiendo una sobrecarga de trabajo para su procesado, también para la transmisión de datos el protocolo HTTP (Hiper Text Transport Protocol), diseñado para trabajar bajo el esquema RPC, invocando funciones remotas.

Las tecnologías que implementa SOAP hace que sea muy difícil ser adaptado a un ambiente móvil porque hasta ahora se conoce que son limitados y consumen muchos recursos, el encapsulado del mensaje que envía hace que consuma un mayor ancho de banda, requiere de más memoria y en este caso en procesamiento de un dispositivo.

Para crear este modelo distribuido se considera algunos objetivos que permiten cumplir con el marco de trabajo con las que SOAP es considerado un modelo estándar e independiente.

Estos son:

- Establecer un protocolo estándar de invocación a servicios remotos que esté basado en protocolos estándares de uso frecuente en Internet, como son HTTP (Hiper Text

Transport Protocol) para la transmisión y XML (eXtensible Markup Language) para la codificación de los datos.

- Independencia de plataforma hardware, lenguaje de programación e implementación del servicio Web.

La utilidad que presta SOAP es considerablemente útil por los protocolos ligeros y estándares que este utiliza para la conexión y transmisión de datos (Hidalgo Macas, Acaro, & Edison, 2016).

1.7.2. REST

Un estilo de arquitectura para sistemas distribuidos de hipermedia está teniendo un avance dentro del desarrollo de aplicaciones web es el modelo de Transferencia de Estado Representacional o REST (RESTFULL como implementación de REST), conocido como un servicio orientado a recursos, ayudando a mejorar y facilitar el trabajo en la web.

El termino REST fue implementado por primera vez por Roy Fielding en una conferencia en el Universidad de California, al tratar de principios arquitectónicos distribuidos. Este tipo de servicios web exponen datos y funcionalidades mediante recursos identificados por URI, los clientes interactúan con los recursos mediante métodos de ingresos. Para Fielding REST es un estilo arquitectónico que consiste en clientes y servidores. Los clientes generan solicitudes a los servidores, estos la procesan, generando una respuesta apropiada. Las solicitudes y respuestas se construyen alrededor de la transferencia de representaciones de los recursos. Un recurso puede ser esencialmente cualquier concepto que pueda ser tratado. Una representación de un recurso es típicamente un documento que captura el estado actual o previsto de un recurso (Fielding & Taylor, 2002, p. 82). A diferencia del Protocolo SOAP, REST consume un poco menos el ancho de banda porque no se analiza el documento XML como lo hace SOAP, además de que no requiere de cabeceras en el mensaje. Se puede decir que un servicio web RESTFULL es un diseño basado en la arquitectura REST direccionada a construir aplicaciones distribuidas, orientada a publicar e identificar recursos, utilizando de manera explícita las operaciones del protocolo HTTP y transfiriendo recurso XML y JSON.

Según Hidalgo Macas (2016) REST se define como un estilo arquitectónico, no es un estándar aunque hace uso de varios estándares como son HTTP, XML, URL, HTML entre otros. El diseño de sistemas basados en REST generalmente se denomina RESTFULL y satisfacen los siguientes principios.

- Interfaz uniforme para la identificación de recursos

- Utilización de métodos estándar de HTTP
- Comunicación sin mantener estados
- Recursos con múltiples representaciones

1.8. Análisis

A continuación se detalla un análisis objetivo para la selección correcta de la arquitectura, lenguaje de programación, motor de base de datos y servidor para la implementación de la solución.

Se elabora un esquema de valoración para cada componente, para poder valorar objetivamente cada aspecto. En el esquema se definen criterios aplicables al contexto del proyecto.

La calificación se realiza en una escala de 1 – 5, y se define un criterio de calificación para cada calificación cualitativa.

1.8.1. Arquitectura

La arquitectura a implementar es uno de los aspectos más fundamentales para inicial el desarrollo de la solución.

La arquitectura se evalúa en función a atributos de calidad: disponibilidad, escalabilidad, flexibilidad, rendimiento, seguridad y eficiencia.

La tabla 2 define los atributos y criterios de evaluación del esquema de calificación para evaluar las dos arquitecturas de desarrollo estudiadas: arquitectura cloud computing y cliente servidor.

Tabla 2. Esquema de valoración de arquitectura

Característica	Criterio Calificación	Calificación
Disponibilidad	La gestión de la disponibilidad y recuperación ante fallos depende completamente de la contratación de componentes técnicos y requiere un alto nivel de conocimientos en administración y monitorización de sistemas y un personal de soporte siempre disponible.	1
	La gestión de la disponibilidad y recuperación ante fallos depende del Acuerdo de Nivel de Servicios (SLA) del proveedor que se contrate, pero requiere personal de administración y equipo de soporte disponible.	2
	La gestión de la disponibilidad y recuperación ante fallos depende del Acuerdo de Nivel de Servicios (SLA) del proveedor que se contrate, éste se encarga de la administración pero requiere equipo de soporte	3

	disponible.	
	La arquitectura facilita la configuración para que el sistema esté disponible y se gestionen los fallos manualmente bajo un costo, con un equipo de soporte disponible.	4
	La arquitectura del sistema gestiona la disponibilidad y la recuperación ante los fallos y continúa siendo accesible todo el tiempo.	5
Escalabilidad	La arquitectura no permite agregar nuevas características al sistema de manera sencilla, y no permite aumentar el volumen de datos sin tener que comprar o contratar más componentes.	1
	La arquitectura permite agregar nuevas características al sistema modificando varios elementos de la arquitectura pero permite incrementar el volumen de datos.	2
	La arquitectura permite agregar nuevas características al sistema modificando varios elementos de la arquitectura pero no permite incrementar el volumen de datos.	3
	La arquitectura permite agregar nuevas características al sistema fácilmente pero no permite incrementar el volumen de datos.	4
	La arquitectura permite agregar nuevas características al sistema fácilmente y permite incrementar el volumen de datos sin modificar elementos de la arquitectura.	5
Rendimiento	El tiempo de respuesta del sistema ante una petición y el procesamiento con esta arquitectura es prolongado.	1
	Los tiempos de respuesta del sistema dependerán de la capacidad del servidor o núcleos que se contrate.	2
	Los tiempos de respuesta del sistema son gestionados por la arquitectura pero tienen un costo por procesamiento.	3
	Los tiempos de respuesta del sistema son rápidos para una suscripción gratuita, pero el primer acceso al sistema es lento mientras se levanta el núcleo.	4
	Los tiempos de respuesta del sistema son gestionados automáticamente por la arquitectura.	5
Seguridad	La probabilidad de acceso y daño a los datos y componentes es alta.	1
	El acceso a los datos y componentes requiere una alta configuración manual.	2
	La probabilidad de acceso y daño a los datos y componentes es mediana y requiere una baja configuración.	3

	La probabilidad de acceso y daño a los datos y componentes es baja.	4
	El acceso a los datos y componentes de la arquitectura está completamente protegido por la misma arquitectura.	5
Eficiencia	La arquitectura compromete un gran uso de recursos a un alto costo.	1
	La arquitectura compromete un gran uso de recursos a un costo mediano.	2
	La arquitectura utiliza pocos recursos a un costo bajo demanda.	3
	La arquitectura utiliza pocos recursos a un bajo costo.	4
	La arquitectura provee recursos justos para el correcto funcionamiento y aumenta los recursos aumentando el costo.	5

Fuente: El autor.
Elaboración. El autor.

A continuación las arquitecturas son evaluadas según el esquema de evaluación y los criterios ya descritos en la tabla 3.

Tabla 3. Evaluación de arquitecturas

Atributo \ Arquitectura	Peso %	Cliente - Servidor	Valoración Ponderada	Cloud Computing	Valoración Ponderada
Disponibilidad	10	2	0.2	5	0.5
Escalabilidad	20	2	0.4	5	1
Rendimiento	20	2	0.4	4	0.8
Seguridad	20	2	0.4	5	1
Eficiencia	30	3	0.9	5	1.5
Total	100		2.3		4.8

Fuente: El autor.
Elaboración. El autor.

La arquitectura con mayor puntaje es “Cloud Computing”. Por tanto se considera este paradigma cómo más apto para la implementación del presente proyecto. Los servicios cloud se analizan más adelante bajo un esquema de calificación similar.

1.8.2. Lenguaje de programación

La siguiente tabla 4 describe el esquema de valoración y el criterio para evaluar el lenguaje de programación a utilizar.

Tabla 4. Esquema de valoración lenguajes de programación.

Característica	Criterio Calificación	Calificación
Curva de aprendizaje	Su curva de aprendizaje es alta. Requiere un esfuerzo y lectura considerables	1
	Su curva de aprendizaje es baja al inicio y asciende rápidamente. El aprendizaje de la sintaxis básica es sencillo, pero su paradigma requiere un amplio esfuerzo de aprendizaje	2
	Su curva de aprendizaje es baja al inicio y empinada al final. Su aprendizaje es fácil inicialmente pero se complica la implementación de nuevas características.	3
	Su curva de aprendizaje es baja al inicio y no incrementa mucho. Su aprendizaje es fácil inicialmente y se mantiene así en implementación de nuevas características excepto en algunas.	4
	Posee una curva de aprendizaje baja. Es fácil y rápido de aprender durante todo el ciclo de aprendizaje	5
Documentación	No cuenta con mucha documentación, y no tiene soporte ni retroalimentación por los desarrolladores.	1
	Su documentación es escasa, pero tiene soporte y retroalimentación en el idioma inglés.	2
	Su documentación es escasa, pero tiene soporte y retroalimentación en el idioma inglés y español.	3
	Tiene documentación considerable. Tiene soporte en el idioma inglés.	4
	Cuenta con documentación extensa, soporte y actualización por los desarrolladores y foros de soluciones por los programadores en varios idiomas.	5

Rendimiento	Es un lenguaje compilado, su tiempo de ejecución es lento y difícil de depurar.	1
	En el lenguaje toda la carga de procesamiento la realiza el servidor y necesita un servidor obligatoriamente para funcionar.	2
	El lenguaje es únicamente orientado a objetos, permite utilización de patrones de diseño para mejorar el rendimiento.	3
	El lenguaje puede usarse como procedimental u orientado a objetos.	4
	Es un lenguaje interpretado, su ejecución es rápida, es multiplataforma y fácil de depurar.	5
Soporte UTP	No existe personal en UTP que desarrolle en este lenguaje de programación	1
	Existe personal con conocimiento básico para dar soporte a este lenguaje en UTP.	2
	Existe personal con conocimiento avanzado para dar soporte a este lenguaje en UTP.	3
	Existen escasas aplicaciones implementadas con este lenguaje en UTP.	4
	Existen varias aplicaciones y personal calificado para dar soporte a este lenguaje en UTP.	5

Fuente: El autor.
Elaboración. El autor.

Y la tabla 5 describe la valoración de cada lenguaje de programación según el esquema descrito.

Tabla 5. Evaluación de lenguajes de programación

Atributo \ Arquitectura	Peso %	Java	Valoración Ponderada	Python	Valoración Ponderada	Ruby	Valoración Ponderada	PHP	Valoración Ponderada	Node.js	Valoración Ponderada
Curva de aprendizaje	30	2	0.6	5	1.5	5	1.5	4	1.2	2	0.6
Documentación	20	5	1	5	1	4	0.8	5	1	4	0.8
Rendimiento	20	2	0.4	4	0.8	4	0.8	2	0.4	5	1
Soporte UTPL	30	4	1.2	4	1.2	2	0.6	2	0.6	2	0.6
Total	100		3.2		4.5		3.7		3.2		3

Fuente: El autor.
Elaboración. El autor.

El lenguaje con mayor valoración es Python, que es considerado el lenguaje para la implementación de la solución del proyecto.

1.8.3. Servidor

La selección del servidor es una de las partes más importantes para el despliegue de la solución. Es necesario considerar los despliegues continuos según los requerimientos cambiantes de la solución, para que cada característica esté disponible para los usuarios en el menor tiempo posible.

Tabla 6. Esquema de valoración servicios cloud.

Característica	Criterio Calificación	Calificación
Costos	Su precio base excede a \$0.10/hora por procesamiento de CPU virtual y no dispone de una suscripción Free.	1
	Su precio base excede a \$0.10/hora por procesamiento de CPU virtual pero dispone una suscripción Free.	2
	Su precio base es de \$0.05/hora por procesamiento de CPU virtual pero dispone una suscripción Free.	3
	Su precio es menor a \$0.05/hora por procesamiento de CPU virtual y dispone también de una suscripción Free.	4
	Dispone de una versión Free con las características necesarias de memoria y procesamiento para el funcionamiento correcto de la aplicación.	5
Soporte Django	No dispone implementación ni soporte para el framework Django	1
	Su implementación con Django está en sus inicios o se prevé que se implementará en un futuro próximo.	2
	Dispone implementación con Django pero existe falta de recursos y documentación.	3
	Dispone implementación con Django y su documentación y soporte es razonable.	4
	Soporta completamente la implementación con Django y dispone de una documentación y soporte amplio.	5
Soporte Postgres	No dispone implementación ni soporte para el motor de base de datos Postgres	1
	Dispone implementación con Postgres bajo un costo de utilización.	2
	Dispone implementación con Postgres pero existe falta de recursos y documentación.	3

	Dispone implementación con Postgres y su documentación y soporte es razonable.	4
	Soporta completamente la implementación con Postgres y dispone de una documentación y soporte amplio.	5
Facilidad de configuración	Su instalación, configuración y despliegue son un proceso engorroso y no está bien documentado.	1
	Su instalación, configuración y despliegue es sencillo pero no hay suficiente documentación para todos los frameworks y motores de bases de datos.	2
	Su instalación, configuración y despliegue son un proceso rápido, sencillo pero no cuenta con la suficiente documentación en caso de errores.	3
	Su instalación, configuración y despliegue son un proceso rápido, sencillo y tiene retroalimentación por parte de una comunidad de desarrolladores en caso de errores.	4
	Su instalación, configuración y despliegue son un proceso rápido, sencillo y cuenta con la suficiente documentación.	5
Convenios UTPL	La UTPL no cuenta con licencias y soporte para este servicio Cloud.	1
	La UTPL dispone de una licencia básica para este servicio Cloud, pero son pocas las aplicaciones dónde se ha implementado el mismo.	2
	La UTPL dispone de una licencia básica para este servicio Cloud, pero son pocas las aplicaciones dónde se ha implementado el mismo y personal con conocimiento suficiente.	3
	La UTPL dispone de una licencia básica para este servicio Cloud, con varias aplicaciones dónde se ha implementado el mismo.	4
	La UTPL dispone de una licencia Enterprise para este servicio Cloud y personal capacitado en la configuración y despliegue del mismo.	5

Fuente: El autor.
Elaboración. El autor.

La tabla 7 evalúa los servicios cloud analizados los evalúa para seleccionar el servidor más viable.

Tabla 7. Evaluación servicios cloud.

Atributo \ Servicio Cloud	Peso %	Microsoft Azure	Valoración Ponderada	SoftLayer	Valoración Ponderada	Heroku	Valoración Ponderada	Google App Engine	Valoración Ponderada
Costos	20	4	0.8	1	0.2	5	1	4	0.8
Soporte Django	30	3	0.9	1	0.3	5	1.5	5	1.5
Soporte Postgres	30	3	0.9	3	0.9	5	1.5	2	0.6
Facilidad de configuración	10	5	0.5	2	0.2	5	0.5	5	0.5
Convenios UTPL	10	4	0.4	2	0.2	1	0.1	1	0.1
Total	100		3.5		1.8		4.6		3.5

Fuente: El autor.
Elaboración. El autor.

El servicio cloud que tiene una mayor valoración es “Heroku” por tanto se considera la utilización de este servicio para la implementación de la solución de este proyecto.

1.8.4. Base de datos

El motor de base de datos constituye también uno de los elementos primordiales a implementar en la solución.

Para la correcta selección del mismo, se evalúa el rendimiento del motor de base de datos, costo, integración con el servidor más probable de implementación y el soporte para el motor en UTPL.

La tabla 8 describe el esquema de valoración para el motor de base de datos.

Tabla 8. Esquema de valoración motores de bases de datos.

Característica	Criterio Calificación	Calificación
Soporte de objetos nativamente	No soporta ningún objeto nativo (dominio, cursores, triggers, funciones, procedimientos).	1
	Sólo soporta triggers pero ningún otro objeto nativo.	2
	Sólo soporta triggers y otros pocos objetos nativos.	3
	Soporta todos los objetos nativos excepto dominio.	4
	Soporta todos los objetos nativos (dominio, cursores, triggers, funciones, procedimientos).	5
Costo	Base de datos de paga. La licencia de su versión básica excede un monto de \$1000 dólares.	1
	Base de datos de paga. La licencia de su versión básica comprende un monto entre \$500 a \$1000 dólares.	2
	Base de datos de paga. La licencia de su versión básica comprende un monto entre \$200 a \$500 dólares.	3
	Base de datos de paga. La licencia de su versión básica es un monto menor a \$200. O se puede usar la licencia de instalación de UTPL.	4
	Base de datos gratuita.	5
Integración con Cloud	Los servicios Cloud analizados no soportan la implementación de este motor de base de datos.	1
	Los servicios Cloud analizados soportan este motor de base de datos a un determinado precio, y sólo disponible en las regiones de Europa y Estados Unidos.	2

	Los servicios Cloud analizados soportan este motor de base de datos a un determinado precio, y es un motor ya implementado en nuestra región.	3
	Los servicios Cloud analizados soportan este motor de base de datos gratuitamente por un tiempo, y es un motor ya implementado en nuestra región.	4
	Los servicios Cloud analizados soportan este motor de base de datos gratuitamente.	5
Soporte UTPL	La UTPL no cuenta con licencias y soporte para este motor de base de datos.	1
	La UTPL dispone de una licencia básica para este motor de base de datos, pero son pocas las aplicaciones dónde se ha implementado el mismo.	2
	La UTPL no cuenta con implementaciones de este motor de base de datos, pero es un motor de base de datos relacional y existe personal capacitado en este tipo de base de datos.	3
	La UTPL dispone de una licencia básica para este motor de base de datos, con varias aplicaciones dónde se ha implementado el mismo.	4
	La UTPL dispone de una licencia empresarial para este motor de base de datos y su personal tiene experiencia en el desarrollo del mismo.	5

Fuente: El autor.
Elaboración. El autor.

Los motores de base de datos utilizados en UTPL fueron tomados del trabajo de Efrén Narváez (2017), dónde describe la heterogeneidad de tecnologías de almacenamiento de datos en la UTPL. (Narvaez, Efrén, 2017)

La tabla 9 muestra la evaluación de los diferentes motores de bases de datos.

Tabla 9. Evaluación motores de bases de datos.

Atributo\MotorBD	Peso %	Oracle	Valoración Ponderada	SQLite	Valoración Ponderada	MySql	Valoración Ponderada	Postgre SQL	Valoración Ponderada	Mongo DB	Valoración Ponderada
Soporte de objetos nativamente	10	5	0.5	2	0.2	4	0.4	5	0.5	5	0.5
Costo	30	4	1.2	5	1.5	4	1.2	5	1.5	5	1.5
Integración con Cloud	25	3	0.75	1	0.25	5	1.25	5	1.25	2	0.5
Soporte UTP	35	5	1.75	1	0.35	2	0.7	3	1.05	1	0.35
Total	100		4.2		2.3		3.55		4.3		2.85

Fuente: El autor.
Elaboración. El autor.

El motor de base de datos con mayor valoración es “PostgreSQL” por tanto se considera la implementación de este motor en la solución del proyecto.

1.8.5. Resultados del análisis.

En base al análisis de las metodologías y arquitecturas anteriormente descritas, se ha considerado los siguientes criterios, que se vinculan con nuestro proyecto y se considera la aplicación de las siguientes características y componentes para el desarrollo de la solución.

Tabla 10. Resultados del análisis.

Característica / Componente.	Implementación en solución.
Extracción de requerimientos	Se aplicará las técnicas y herramientas descritas en la sección 1.6
Arquitectura	Para la implementación se utilizará una arquitectura Cloud, por su escalabilidad y facilidad de despliegues de la aplicación solución, que se integra con la metodología ágil a utilizar.
Metodología	Se utilizará Scrum y DevOps para el desarrollo de nuestro proyecto, debido al entorno cambiante de requisitos y la necesidad de la implementación de un desarrollo ágil para la aplicación. Se usará de Scrum los artefactos para la descripción de requerimientos (Product BackLog), y para la entrega y prueba de características (Sprints). La cultura DevOps deberá estar siempre presente durante todo el desarrollo de la solución.
Lenguaje de Programación	El lenguaje a aplicar en la solución es Python, por su flexibilidad, rápido desarrollo y facilidad de aprendizaje.
Framework	Se utilizará el framework Django, por motivos de agilización en el desarrollo, ya que nos define ya el esquema de directorios, nos facilita el modelado de datos y nos provee clases predefinidas por el framework para su implementación directa, para centrarnos únicamente en la implementación de la lógica de la programación una vez dominado el framework.
Motor de Base de Datos	Se iniciará la configuración de la solución con el motor de base de datos PostgreSQL, por su integración con

	el framework Django y su implementación con servicio Cloud.
Servicio Alojamiento Cloud	Se utilizará Heroku por su integración sencilla con el lenguaje Python y el Framework Django, y por su sencilla implementación con el motor de base de datos PostgreSQL.
Modelado	El modelado de la arquitectura de software se desarrollará usando las partes más relevantes del modelo 4+1 de Krutchen, que describen con mayor precisión aspectos más relevantes de nuestro proyecto y que servirán como documentación complementaria a la metodología a ser utilizada.

Fuente: El autor.

Elaboración: El autor.

La vinculación de todas las tecnologías, su acoplamiento y pasos a seguir, se detallarán en el capítulo 3, una vez analizado el proceso de negocio en el siguiente capítulo.

CAPÍTULO 2: PROCESO DE NEGOCIO

Una vez descritas y analizadas las herramientas pertinentes para el desarrollo del presente proyecto, este capítulo se enfoca en la comprensión en su totalidad del proceso de trabajos de titulación para las titulaciones de: “Sistemas Informáticos y Computación” e “Ingeniería Informática”.

Para iniciar correctamente el desarrollo de un sistema de software, se debe tener una comprensión completa del proceso al cual se va a implementar el software, por tanto en esta sección se describe y analiza el proceso concerniente a los trabajos de titulación en la Universidad Técnica Particular de Loja, la problemática asociada, para luego continuar con los pasos para la correcta extracción de requerimientos de las fuentes asociadas identificadas.

En esta sección categorizamos a los interesados en el proyecto, luego según sus intereses los clasificamos y establecemos sus perfiles de Interesados. Todo esto se realiza en combinación con las técnicas: entrevistas, prototipos exploratorios, estudio de documentación existente y al final se analiza y plantea una solución para proceder a su desarrollo en el capítulo tres.

2.1. Proceso de trabajos de titulación UTPL

El proceso para la presentación, aprobación y seguimiento de trabajos de titulación, se encuentra descrito en el documento: “*Lineamientos Generales para Postulaciones de T.T. (v 1.2)*”, desarrollado por el coordinador de titulación y el equipo de calidad del área técnica de las titulaciones *Ingeniería en Sistemas Informáticos y Computación* y *Computación e Ingeniería Informática*. El mismo que se encuentra en el anexo 13.

A continuación se presentan los contenidos más importantes para comprender el proceso, centrados específicamente en la propuesta, revisión, evaluación, aprobación, postulación y notificación de los trabajos de titulación, que corresponde al alcance del proyecto.

2.1.2. Proceso

El documento de lineamientos generales para postulación de trabajos de titulación, anexo 13, define las características que debe contener un proyecto, una vez que cumple con las mismas se inicia el proceso de postulación.

El proceso para la presentación de propuestas de trabajos de titulación comprende varias etapas, descritas en el documento anteriormente mencionado, que van desde la elaboración de propuestas, hasta la postulación y asignación de un proyecto.

El proceso se encuentra descrito completamente de forma resumida en la figura 5 dónde se detalla cada una de las etapas concernientes al mismo.



Figura 5. Proceso General de Postulación de Trabajos de Titulación.
Fuente: Lineamientos generales para postulación de Trabajos de Titulación v1.2.
Elaboración: El autor.

2.1.2.1. *Elaboración de Propuesta.*

En esta etapa del proyecto, docentes y estudiantes pueden elaborar una propuesta para trabajo de titulación y notificarla a la sección departamental correspondiente a su tema, considerando las condiciones establecidas en la correspondiente Convocatoria de Trabajos de Titulación.

En el caso de la elaboración por parte de estudiantes la sección departamental asigna un docente para su acompañamiento.

Oportunidad de mejora.

En esta etapa es necesaria la existencia de una Convocatoria de Trabajos de Titulación.

La solución de este proyecto comprende la elaboración digital de una Convocatoria, la cual comprende cinco actividades y cada actividad dispone de un plazo definido por una fecha de inicio y fecha final. Estas actividades son: Propuesta, Revisión, Validación, Autorización y Postulación.

2.1.2.2. *Validación de Propuesta.*

En esta etapa, el responsable de la sección departamental asigna uno o varios revisores a las propuestas para trabajo de titulación que han llegado a su sección. Los revisores utilizan la rúbrica o esquema de evaluación definido por su titulación para revisar y validar la propuesta.

Una vez validada por los revisores de la sección, en equipo la sección departamental y el coordinador de titulación verifican la correspondencia de las propuestas con las opciones de titulación y el perfil de egreso.

Oportunidad de mejora.

El proyecto contempla la elaboración digital de un esquema de evaluación para la titulación por parte del coordinador de titulación. Este paso se lo deberá realizar luego de ofertada la convocatoria para trabajos de titulación.

El esquema de evaluación debe permitir definir los ítems con los que se evaluará una propuesta por parte de un revisor, cada ítem tener cinco criterios para calificar

cualitativamente el ítem y debe tener un porcentaje que representa el peso que tiene dicho ítem en la calificación de la propuesta.

Cada esquema de evaluación deben pertenecer a una convocatoria al igual que una propuesta, por lo tanto la solución presentará al revisor el esquema de evaluación activo para la convocatoria a la que pertenece la propuesta.

La solución también involucra la aprobación por parte de responsable de sección y coordinador de titulación, de modo que una vez emitidas las calificaciones de los revisores para una propuesta el responsable de sección y el coordinador deben aprobar la misma.

La solución también debe permitir solicitar una recalificación, solicitar modificación al autor o rechazar la propuesta directamente.

2.1.2.3. *Aprobación de propuesta.*

En esta etapa el consejo de departamento conoce y aprueba el listado general de la oferta de trabajos de titulación. Aquí se emite un acta de aprobación por la sección, de la oferta general de trabajos de titulación.

Las propuestas de trabajo de titulación autorizadas por el consejo pasan al tutor de Gestión Productiva para que sean puestas a disposición de los estudiantes que esperan asignación de tema.

La sección asigna un equipo de acompañamiento al trabajo de titulación, conformado por un director y uno o más docentes de apoyo.

Oportunidad de mejora.

La solución realizará la autorización digital de las propuestas de trabajos de titulación validados por el responsable de sección y coordinador de titulación.

La autorización la realiza el director de departamento, seleccionando las propuestas validadas a autorizar.

Una vez autorizadas el sistema solución permite la asignación de equipo de acompañamiento conformado por director y docentes de apoyo por parte del responsable de sección. Y los trabajos de titulación (T.T) con estudiante asignados podrán ser notificados por parte de la secretaria de la titulación.

2.1.2.4. Postulación de propuesta

Cada docente carga su T.T autorizados por la sección departamental al Sistema de Gestión de Trabajos de Titulación del Vicerrectorado Académico (S.V.A).

Oportunidad de mejora.

Los T.T autorizados por la sección departamental se enlazarán directamente con el S.V.A.

Una vez los T.T han sido autorizados, estos proyectos aparecerán en el S.V.A para que los estudiantes matriculados en los niveles de Prácticum respectivos puedan seleccionar el tema de propuesta de su interés y entrevistarse con el docente.

2.1.2.5. Selección e inscripción de estudiantes

Los estudiantes matriculados en los niveles de Prácticum respectivos, seleccionan entre las propuestas de T.T un tema de su interés.

El docente autor del trabajo de titulación selecciona a uno o los estudiantes que requiera el T.T, de los estudiantes que han postulado a dicho trabajo.

La sección conoce y aprueba el listado final de trabajos de titulación asignados para el semestre con estudiante y equipo de acompañamiento.

Oportunidad de mejora.

Una vez inscritos los estudiantes en el S.V.A, la solución permite consultar los estudiantes inscritos y aceptados para el T.T.

Esta consulta la realiza la secretaria de la titulación y las propuestas pasan a estar en estado de ejecución.

2.1.2.6. Notificación de designación

La titulación notifica la asignación a los docentes del equipo de apoyo y al estudiante la asignación oficial en el T.T.

Oportunidad de mejora.

La solución permite notificar vía correo electrónico al equipo de acompañamiento (director y docentes de apoyo) y al estudiante la designación oficial en el trabajo de titulación.

2.1.2.7. Trabajos de titulación terminados

Oportunidad de mejora.

Esta etapa no está contemplada en el proceso de lineamientos generales para la postulación de trabajos de titulación, creado por la titulación de Sistemas Informáticos y Computación puesto que este proceso llega hasta la notificación de designación según el diagrama del anexo 1.

En la solución propuesta, la aplicación permitirá consultar del S.V.A. los T.T terminados por el estudiante para que el docente director pueda emitir un informe de finalización sobre el trabajo y se pueda proceder a la asignación de tribunal para revisión de trabajo.

2.1.2.8. Asignación de tribunal

Oportunidad de mejora.

Esta etapa tampoco está contemplada en el proceso de lineamientos generales para la postulación de trabajos de titulación. La solución permite la asignación de tribunal para revisión del T.T, basándose en el equipo de acompañamiento.

Además permitirá a la secretaria de la titulación notificar al tribunal la revisión de la memoria escrita mediante un *esquema de evaluación para trabajos completados* definida por la titulación.

Este esquema de evaluación también puede ser definido luego de creada la convocatoria, y contempla la evaluación de aspectos del T.T, cada aspecto tendrá un porcentaje que corresponde al peso que le corresponde en la calificación. El esquema de evaluación tendrá una calificación máxima y una calificación mínima.

2.1.2.9. Calificación de trabajo de titulación completado

Oportunidad de mejora.

La secretaria de la titulación puede notificar al tribunal designado la asignación como tribunal a un T.T, para que el mismo pueda revisar la memoria escrita y calificarla.

En la solución una persona será la encargada de emitir la calificación del T.T completado y será el presidente de tribunal.

El presidente podrá calificar el T.T completado según los aspectos definidos en el esquema de evaluación activo de la convocatoria definida por la titulación.

Una vez calificado y aprobado el T.T, éste pasa a un estado de finalizado.

2.1.3. Matriz de responsabilidades

Para la descripción de las responsabilidades de los involucrados directos en el proceso, el proyecto hará el uso de la matriz de responsabilidades denominada RACI.

RACI proviene de una sigla en inglés:

- “R” (Responsible)**: es quien ejecuta una tarea. Su función es “HACER”.
- “A” (Accountable)**: es quien vela porque la tarea se cumpla, aún sin tener que ejecutarla en persona. Su función es “HACER HACER”.
- “C” (Consulted)**: indica que una persona o área debe ser consultada respecto de la realización de una tarea.
- “I” (Informed)**: indica que una persona o área debe ser informada respecto de la realización de una tarea. (Longarini, 2011).

Con la matriz RACI, tenemos una perspectiva más específica, sobre las responsabilidades y los involucrados directos con el proceso que involucra el alcance de la investigación.

En la tabla 11, encontramos la representación de las responsabilidades del proceso de postulación de trabajos de titulación, estructurado en una matriz de responsabilidades RACI, para una mejor comprensión del rol que desempeñan los diferentes actores en el proceso y en sus diferentes etapas.

Actividad	Estudiante	Docente	Secretaria	Director de TT	Tutor de G.P.	Sección Departamental	Consejo de Departamento	Coordinador de Titulación	Vicerrectorado académico
------------------	-------------------	----------------	-------------------	-----------------------	----------------------	------------------------------	--------------------------------	----------------------------------	---------------------------------

Tabla 11. Matriz de Responsabilidades del Proceso .

Elaborar propuesta	R	R			C	A		I	I
Analizar propuesta		R			I	A		I	C
Validar pertinencia		C, I			A	R	C	I	I
Asignar revisores		C			I	R	A	I	I
Revisar propuesta		R				A, C, I	I		
Aprobar proyecto		I			A	C	R	I	I
Notificar proyectos aprobados	I	I	R		A	I	I	C	I
Asignar T.T. a estudiante	I	A		I	R	I	C	C	
Asignar director y tribunal evaluador	I	I		I	A	I	R	C	I
Presentar avance T.T	R			C	A	I	I	I	I
Cambiar alcance T.T.	R			C	A	I	C	C	I

Fuente: Lineamientos generales para postulación de Trabajos de Titulación v1.2.

Elaboración: El autor.

2.2. Categorización de interesados (Stakeholders)

A continuación se define la tabla con la categorización de interesados descrita en la sección 1.6.1.2 del presente documento.

Tabla 12. Categorización de Interesados.

Clientes		Usuarios		Otros Interesados	
Sponsor	Product Champion	Directos	Indirectos	Asesores	Proveedores
Titulación “Sistemas Informáticos y Computación”	Director de Trabajo de Titulación	Docente	Secretaria	Coordinadores de otras titulaciones	Gerente de Proyecto o Product Master (Scrum).
		Estudiante Presencial.	Vicerrectorado UTPL	Departamento de Procesos UTPL	Desarrollador
		Estudiante a Distancia.	Sistema Vicerrectorado Académico (S.V.A)	Vicerrectorado académico	
		Director de Trabajo de Titulación	Equipo de acompañamiento a Trabajo de Titulación.		
		Presidente de Trabajo de Titulación	Tribunal de trabajo de Titulación (vocal 1 y vocal 2)		
		Responsable de Sección Departamental	Tutor de Gestión Productiva		
		Revisor de sección departamental.			
		Director Consejo de Departamento			

		Coordinador de Titulación			
--	--	---------------------------	--	--	--

Fuente: Lineamientos generales para postulación de Trabajos de Titulación v1.2.
Elaboración: El autor.

2.3. Matriz de perfiles de interesados

A continuación se desarrolla la matriz de perfiles de interesados descrita en el capítulo 1 sección 1.6.1.3. Para esto se hace uso de las herramientas y técnicas también descritas en el capítulo y se describe su implementación.

2.3.1. Entrevistas

Las entrevistas se elaboraron siguiendo el proceso detallado en la sección 1.6.1.5

Los criterios de aceptación y preocupaciones se determinaron en base a entrevistas con un grupo de interesados:

Personas a entrevistar:

- Secretaria Sistemas Informáticos y Computación modalidad presencial.
- Secretaria Informática modalidad abierta.
- Responsable de sección inteligencia artificial.
- Tutor principal Gestión Productiva 4.1 modalidad presencial.
- Tutor principal Gestión Productiva 4.2 modalidad presencial.
- Tutor principal Trabajo de Titulación décimo ciclo modalidad abierta.
- Coordinador de titulación Sistemas Informáticos y Computación modalidad presencial.
- Coordinador de titulación Electrónica y Telecomunicaciones modalidad presencial.
- Coordinador de titulación Ingeniería Civil modalidad presencial.

Preguntas para la entrevista:

Las preguntas se desarrollaron en con el objetivo de determinar el proceso que se lleva en cada titulación, el rol que desempeñan cada uno de los entrevistados y la problemática asociada al proceso. Dichas preguntas encuentran en el anexo 2.

Parte de los resultados se adhieren a la matriz de perfiles de interesados descritos en la tabla 13, además se describen textualmente en la sección 3.1.3 Identificación de necesidades.

2.3.2. Prototipos exploratorios

Los prototipos se desarrollaron según el proceso descrito en la sección 1.6.1.6.

Porción del alcance del producto para maquetar.

Se seleccionó dos porciones a maquetar:

- Propuesta de trabajo de titulación
- Validación de Propuesta

Selección del tipo de prototipo

Para la construcción se seleccionó un prototipo usar-desechar, puesto que el prototipo se presentará una única vez a una porción de interesados, para su retroalimentación implementar en el software en construcción.

Diseño y construcción del prototipo

Los dos prototipos implementados se encuentran en los anexos. Se los diseñó utilizando una herramienta de software llamada “*Just in mind Prototyper*”, usando su versión *Trial*.

Presentación del prototipo y evaluación con usuarios.

El prototipo fue presentado a un docente, un responsable de sección y un coordinador de titulación. Obteniendo como retroalimentación el uso de los estilos definidos por la Universidad Técnica Particular de Loja para la implementación del mismo en el software en construcción.

2.3.2.3. Estudio de la documentación existente

Cada titulación de la Universidad Técnica Particular de Loja se atiene al reglamento de régimen académico y a la Ley Orgánica de Educación Superior en donde se detalla el índice terminal de titulación para cada titulación de tercer nivel.

El proceso para la presentación de propuestas de trabajos de titulación comprende varias etapas, descritas en el documento anteriormente mencionado, que van desde la elaboración de propuestas, hasta la postulación y asignación de un proyecto

La Figura 5 muestra el proceso asumiendo que la propuesta ha tenido el flujo normal para la postulación, cabe recalcar que si la propuesta no califica con la rúbrica, o no aprueba en la

revisión posterior por el Consejo Departamental, puede ser sometida a cambios y nueva revisión o rechazarse.

Una vez aplicadas las técnicas y herramientas anteriormente descritas se procede a desarrollar la matriz de perfiles de interesados que se muestra a continuación en la tabla 13.

Tabla 13. Matriz de perfiles de interesados

Interesado	Roles	Responsabilidades	Intereses	Criterios de Aceptación	Preocupaciones	Capacidad Técnica/Limitaciones del entorno de trabajo
Titulación Sistemas Informáticos y Computación	Sponsor	Autorizar el desarrollo del sistema	Gestionar adecuadamente el proceso de postulación de trabajos de titulación.	Herramienta que gestione todo el proceso y genere reportes de estados de las propuestas de trabajos de titulación.	Integración con los esquemas de datos actuales de la UTPL.	N/A
Vicerrectorado académico	Asesor	Verificar la pertinencia del sistema.	<p>Reducir los índices terminales de titulación</p> <p>Mejorar la eficiencia de los procesos para titulación de los estudiantes</p> <p>Racionalizar las operaciones de negocio</p>	<p>Titulación de estudiantes en plazos establecidos.</p> <p>Reportes del estado de los trabajos de titulación en curso</p>	<p>Integración con el Reglamento de Régimen académico vigente.</p> <p>Actualización del sistema según cambios a la normativa vigente.</p> <p>Integración de sistemas actuales vigentes.</p>	N/A
Director de Trabajo Titulación	Product Champion	Acompañar el desarrollo de un trabajo de titulación	Conocer el estado de sus trabajos de titulación dirigidos	Reporte de estado de sus trabajos de titulación		Familiarizado con proceso actual de UTPL

	Usuario directo	Revisar avances de un trabajo de titulación Realizar correcciones de un trabajo de titulación	Notificar modificaciones a los estudiantes que desarrollan sus trabajos de titulación			
Docente	Usuario directo	Elaborar y subir una propuesta para trabajo de titulación Validar aptitudes de estudiantes candidatos a postular para su trabajo de titulación	Propuestas correctamente validadas	Notificación a tiempo vía email sobre plazos para postulación de propuestas. Notificación a tiempo vía email sobre correcciones en sus propuestas para trabajo de titulación Notificación sobre estudiantes aceptados e inscritos para su trabajo de titulación	Tener que elaborar nuevas credenciales para el sistema Retrasos en revisiones de sus propuestas para trabajo de titulación Pérdida del estado de propuestas postuladas	Docente usa el sistema SGTT en vigencia y Entorno Virtual de Aprendizaje de UTP
Estudiante modalidad abierta.	Usuario directo	Elaborar y subir una propuesta para trabajo de titulación	Propuestas correctamente validadas.	Notificación a tiempo sobre plazos para postulación de propuestas.	Pérdida del estado de propuestas postuladas	Estudiante usa el sistema SGTT en vigencia.

		<p>Entrevistarse con docente que ha subido una propuesta de trabajo de titulación</p> <p>Actualizar avances en su trabajo de titulación</p>	<p>Asignación a una propuesta de trabajo de titulación.</p>	<p>Notificación a tiempo sobre correcciones en sus propuestas para trabajo de titulación</p> <p>Notificación de asignación a un trabajo de titulación ofertado</p>	<p>Propuesta no validada correctamente.</p> <p>Calificación errónea de trabajo de titulación finalizado</p>	
Estudiante modalidad presencial.	Usuario directo	<p>Entrevistarse con docente que ha subido una propuesta de trabajo de titulación</p> <p>Actualizar avances en su trabajo de titulación</p>	<p>Propuestas correctamente validadas.</p> <p>Asignación a una propuesta de trabajo de titulación</p>	<p>Notificación a tiempo sobre plazos para postulación de propuestas.</p> <p>Notificación a tiempo sobre correcciones en sus propuestas para trabajo de titulación.</p> <p>Notificación de asignación a un trabajo de titulación ofertado</p>	<p>Calificación errónea de trabajo de titulación finalizado</p>	Estudiante usa el sistema SGTT en vigencia.

Responsable de Sección Departamental	Usuario directo	<p>Solicitar modificaciones a autor de propuesta.</p> <p>Solicitar recalificación a revisor.</p> <p>Aprobar/Rechazar una propuesta de un trabajo de titulación</p> <p>Asignar roles de revisores de sección departamental</p>	<p>Disponibilidad de listado actualizado de revisores.</p> <p>Notificación correcta y a tiempo a revisores</p> <p>Comentarios sobre modificaciones recibidos a tiempo.</p>	<p>Notificaciones correctas a tiempo a revisores sobre Nuevas propuestas a revisar</p> <p>Notificaciones correctas a docentes o estudiantes que han subido propuestas, sobre el estado o comentarios sobre su propuesta</p>	<p>Retrasos en notificaciones.</p> <p>Pérdida del estado de propuestas en revisión</p>	Usa el actual sistema SGTT UTPL vigente
Revisor de sección departamental.	Usuario directo	<p>Calificar mediante rúbrica una propuesta de un trabajo de titulación</p> <p>Emitir un comentario sobre la rúbrica para mejora de una propuesta.</p>	<p>Disponibilidad de las rúbricas actualizadas para calificar una propuesta</p> <p>Emitir comentarios sobre las secciones que necesitan corrección de una propuesta</p>	<p>Comentarios correctamente enviados a responsable de sección y a estudiante/docente que sube propuesta</p> <p>Rúbrica actualizada</p>	<p>Retraso en recibir notificaciones sobre propuestas a revisar</p> <p>Retraso en envío de notificaciones sobre comentarios para correcciones</p>	Usa el actual sistema SGTT UTPL vigente

		Notificar a docente/estudiante sobre modificaciones requeridas en su propuesta				
Director Consejo de Departamento	Usuario directo	Aprobar/Rechazar una propuesta de un trabajo de titulación Asignar Tribunal a un Trabajo de Titulación	Tener el listado de las propuestas correctamente revisadas y aprobadas.	Listado de propuestas revisadas y aprobadas	Retrasos en revisiones de sus propuestas para trabajo de titulación	Usa el actual sistema SGTT UTPL vigente
Coordinador de titulación	Usuario indirecto	Establecer convocatoria para postulación de propuestas para trabajo de titulación Establecer convocatoria para calificación de rúbricas de propuestas. Asignar roles de responsables de sección.	Subida de propuestas en convocatoria establecido. Revisiones de propuestas en convocatoria establecida. Terminación de trabajos de titulación en plazos establecidos en el trabajo de titulación	Reporte de estado de trabajos de titulación en curso	Retrasos en la culminación de trabajos de titulación	Usa el actual sistema SGTT UTPL vigente

		<p>Revisar las propuestas para trabajo de titulación revisadas y aprobadas</p> <p>Revisar estado de trabajos de titulación en curso.</p> <p>Definir plantilla para trabajos de titulación aplicables a su titulación.</p>	<p>Trabajos de titulación pertinentes al perfil profesional y competencias adquiridas por el estudiante durante la carrera.</p>			
<p>Coordinadores de otras titulaciones</p>	<p>Asesor Usuarios directos.</p>	<p>Definir plantilla para trabajos de titulación aplicables a su titulación.</p> <p>Definir rúbrica vigente para revisión de trabajos de titulación.</p>	<p>Plantilla con campos actualizados para subir propuestas de trabajo de titulación</p> <p>Actualización de rúbricas vigentes para revisión de trabajos de titulación.</p>	<p>Ingreso de campos necesarios para la nueva plantilla.</p> <p>Ingreso de campos necesarios para la nueva rúbrica.</p>	<p>Plantilla para los ofertantes de propuestas fuera de vigencia.</p> <p>Rúbrica desactualizada para calificar trabajo de titulación a los revisores.</p>	<p>Hace las asignaciones a revisores y revisiones manualmente.</p>
<p>Equipo de acompañamiento</p>	<p>Revisor</p>	<p>Apoyar en la definición académico – científico del trabajo.</p>	<p>Cumplimiento de requisitos académicos.</p>	<p>El trabajo cumpla con los objetivos planteados y que se desarrollen competencias del perfil profesional para graduación de ingeniero en</p>	<p>Calidad de la documentación.</p> <p>Software/aplicación aplicable.</p>	<p>Calificación del trabajo.</p> <p>Recomendar mejoras.</p>

				Sistemas.		
Sistema Vicerrectorado Académico (S.V.A) – Sistema Actual Vigente	Usuario Indirecto	Oferta trabajos de titulación para postulación de estudiantes. Realiza seguimiento a avances en un trabajo de titulación puesto en marcha.	Integración a nivel de datos de aplicaciones.	Integración de formatos de propuestas, con los campos e identificadores que cada modelo de datos tiene configurado en su sistema. Seguridad Accesos	Inconsistencia de datos con la integración de sistemas.	Desarrollador de software. Experto en el proceso actual de postulación. Experto en modelo de datos de UTPL.
Área soluciones UGTI	Asesor	Gestionar privilegios y accesos a infraestructura y datos de UTPL.	Permitir a los desarrolladores el acceso necesario a los datos que se necesiten.	Acceso sólo a datos pertinentes.	Mal uso de privilegios de datos privados. Seguridad de los datos.	Experto en gestión de procesos y gestión de datos en infraestructura UTPL.

Fuente: Lineamientos generales para postulación de Trabajos de Titulación v1.2.

Elaboración: El autor.

2.4. El problema

Las reformas en el proceso, la falta de control del mismo en las titulaciones, la información dispersa y principalmente la falta de un sistema que gestione y homogeneice el proceso para toda la UTPL constituyen la principal problemática de nuestro contexto.

En vista a optimizar la gestión de trabajos de titulación, con el fin de alinear el índice terminal de titulación con el establecido en el reglamento, se ha realizado una evaluación objetiva los problemas asociados, utilizando la diagramación mediante un diagrama de espina de pescado (diagrama de Ishikawa) o diagrama causa-efecto para identificar las causas raíces y plantear una solución a la misma.

Se han identificado cinco problemas principales:

- Falta de información de los Trabajos de Titulación.
- Estudiantes que reprueban o desertan de sus Trabajos de Titulación.
- Retraso en la postulación de propuestas para Trabajos de Titulación.
- Retraso en las revisiones de propuestas para Trabajos de Titulación.
- Dificultad en notificar a los involucrados en alguna parte del proceso.

A continuación se presenta los diagramas causa-efecto para los cinco problemas y se realiza un análisis y solución a la causa raíz.

En el análisis y solución se realiza un listado de todas las causas identificadas por categoría en el diagrama causa-efecto y se establecen los siguientes criterios para evaluar cada causa:

- ¿Es un factor que lleva al problema?
- ¿Esto ocasiona directamente el problema?
- ¿Si esto es eliminado se corregiría el problema?
- ¿La implementación de la solución es factible?
- ¿Se puede medir si la solución funcionó?
- ¿Es compleja la implementación de la solución?

Luego se establece una escala de calificación/peso para cada criterio en un rango de 1 a 3, asumiendo que 1 significa que el criterio no es muy determinante, 2 es medianamente y 3 es sumamente determinante. Seguidamente se elabora la matriz de evaluación de causas.

Seguido de esto se elabora un histograma de causas por cada matriz de análisis, para identificar las causas e inconvenientes en los que se debe centrar el desarrollo de la solución del proyecto.

2.4.1. Falta de información de los Trabajos de Titulación (T.T.)

La Figura 6 muestra el diagrama causa-efecto para este problema.

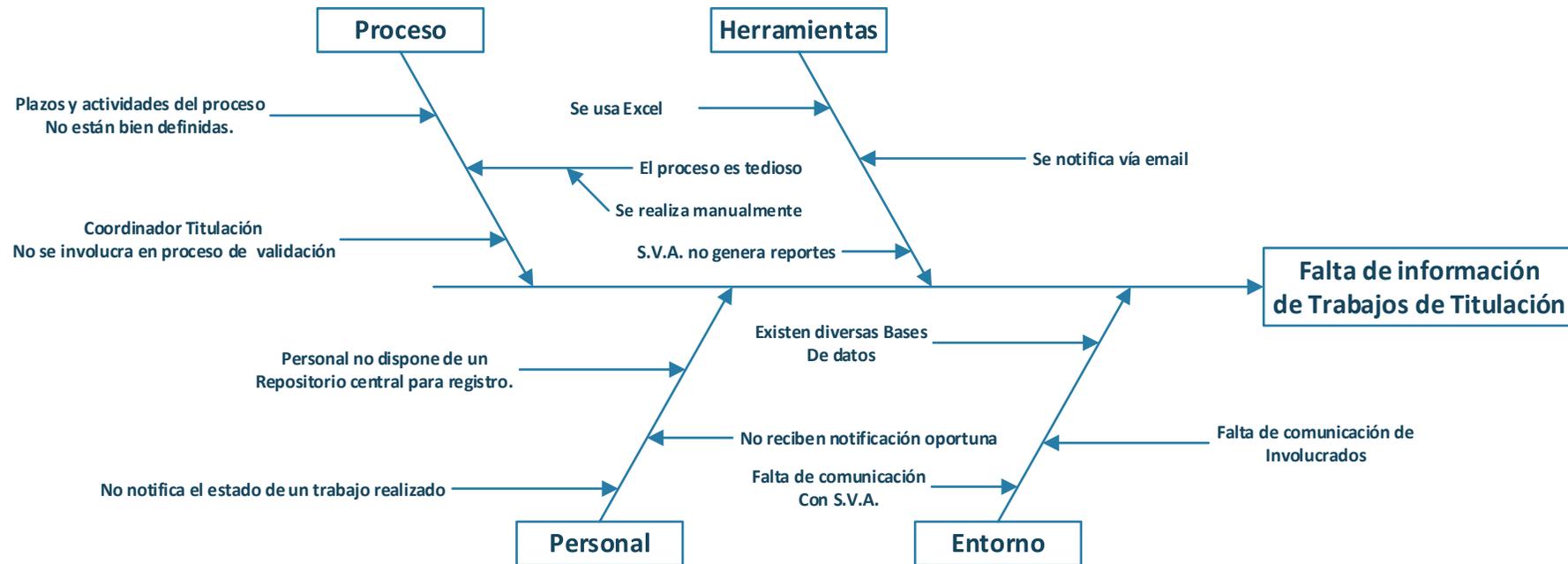


Figura 6. Diagrama causa efecto – Falta de información de T.T.

Fuente: El autor.

Elaboración: El autor.

2.4.1.1. Análisis y solución a causa raíz

Se procede a elaborar la matriz de análisis y se muestra en la tabla 14.

Tabla 14. Análisis: “Falta de información de T.T.”

CAUSAS	SOLUCIONES	CRITERIOS					TOTALES
Proceso	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Plazos y actividades no están bien definidas	Herramienta para creación de actividades y plazos que pertenezcan a una convocatoria.	2	1	2	3	3	11
El proceso se realiza manualmente.	Creación de una herramienta para automatización y gestión del proceso	3	3	3	3	3	15
Coordinador de titulación no se involucra en el proceso de validación.	Validación obligatoria de Coordinador en la herramienta.	1	1	2	3	3	10
Herramientas	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Se usa Excel.	Creación de una herramienta para automatización y gestión del proceso	1	2	3	3	3	12
Se notifica vía email.	Notificación automática en la herramienta.	2	2	3	2	1	10
S.V.A. no genera reportes	Generación de reportes en la herramienta.	2	3	3	3	3	14
Personal	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Personal no dispone de un repositorio central para registrar la información.	Creación de una herramienta para automatización y gestión del proceso	3	3	3	3	3	15
No reciben notificación oportuna	Notificación automática en la herramienta.	1	1	2	2	1	7
No notifican el estado de un trabajo realizado.	Registro obligatorio de un proceso en la herramienta.	2	2	3	3	3	13
Entorno	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Existen diversas bases de datos	Crear una herramienta que integre todos los sistemas.	2	1	3	1	1	8
Falta de comunicación de involucrados.	Creación de un módulo de notificaciones en la herramienta.	1	2	3	3	1	10
Falta de comunicación con S.V.A.	Enlazar la herramienta con S.V.A para generar reportes con sus datos.	2	2	3	2	2	11

Fuente: El autor.

Elaboración: El autor.

2.4.1.2. *Histograma.*

Ahora se describe histograma. Para esto es necesario extraer el grado de incidencia de la matriz de análisis. La figura 7 describe el histograma para el análisis del problema “Falta de información de Trabajos de Titulación”.

Falta de información de Trabajos de Titulación

Causa	Grado de incidencia
Proceso manual	15
No hay repositorio	15
S.V.A. no genera reportes	14
No se envía notificación	13
Se usa Excel.	12
Plazos no definidos	11
Comunicación con SVA.	11
Coordinador excluido	10
Se notifica vía email.	10
Falta comunicación	10
Diversas BD	8
No recibe notificación	7
	136

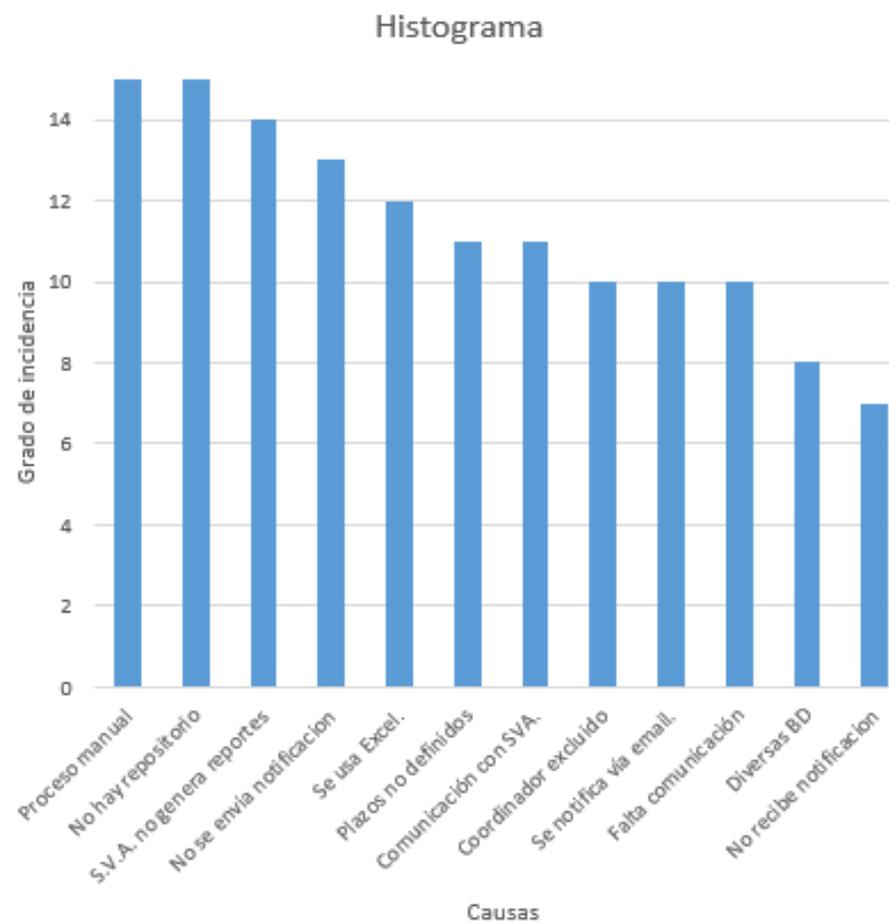


Figura 7. Histograma análisis– Falta de información de T.T.

Fuente: El autor.

Elaboración: El autor.

Como se puede apreciar, si atacamos las principales causas se puede resolver el problema principal de la falta de información de los Trabajos de Titulación.

2.4.2. Estudiantes reprobaban o desertaban de los T.T.

La figura 8 muestra el diagrama causa-efecto para este problema.

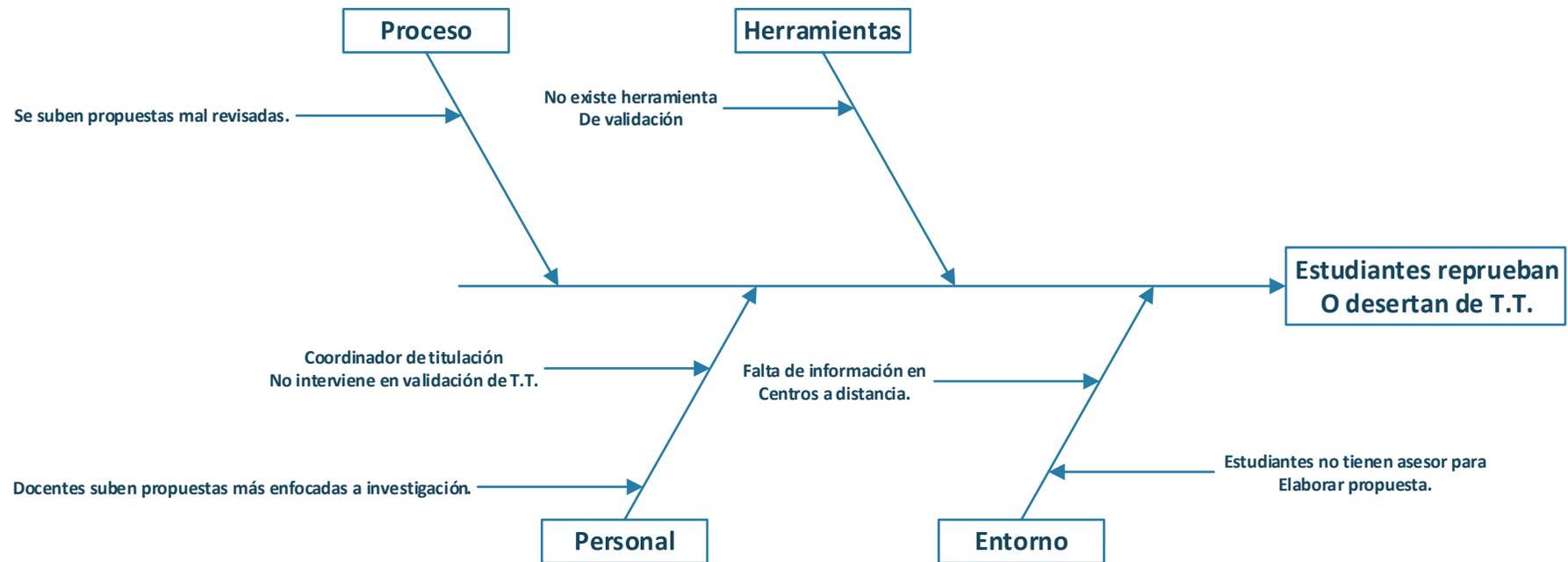


Figura 8. Diagrama causa efecto – Estudiantes reprobaban o desertaban de T.T.

Fuente: El autor.

Elaboración: El autor.

2.4.2.1. Análisis y solución a causa raíz

Se procede a elaborar la matriz de análisis y se muestra en la tabla 15.

Tabla 15. Análisis: “Estudiantes reprueban o desertan T.T.”

CAUSAS	SOLUCIONES	CRITERIOS					TOTALES
		FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Proceso	Solución						
Se suben propuestas mal revisadas	Aprobación de Coordinador Titulación y Responsable Sección obligatoria en la herramienta.	3	3	3	3	3	15
Herramientas	Solución						
No existe herramienta de validación de propuestas.	Módulo de calificación de propuestas en la herramienta	2	2	3	3	3	13
Personal	Solución						
Coordinador Titulación no interviene en validación de T.T.	Aprobación de Coordinador Titulación obligatoria en la herramienta.	2	1	3	3	3	12
Docentes suben propuestas más enfocadas a investigación	Módulo de calificación de propuestas en la herramienta	1	1	2	3	3	10
Entorno	Solución						
Falta de información en centros a distancia	Notificación de nuevas convocatorias a la titulación en la herramienta.	1	1	2	2	1	7
Estudiantes no tienen asesor para elaborar propuesta.	Asignación de director acompañante a un estudiante que suba una propuesta en la herramienta.	1	1	2	3	2	9

Fuente: El autor.

Elaboración: El autor.

2.4.2.2. Histograma.

Ahora se describe histograma. Para esto es necesario extraer el grado de incidencia de la matriz de análisis. La figura 9 describe el histograma para el análisis del problema “Falta de información de Trabajos de Titulación”.

2.4.3. Retraso en postulación de propuestas para T.T.

La figura 10 muestra el diagrama causa-efecto para este problema.

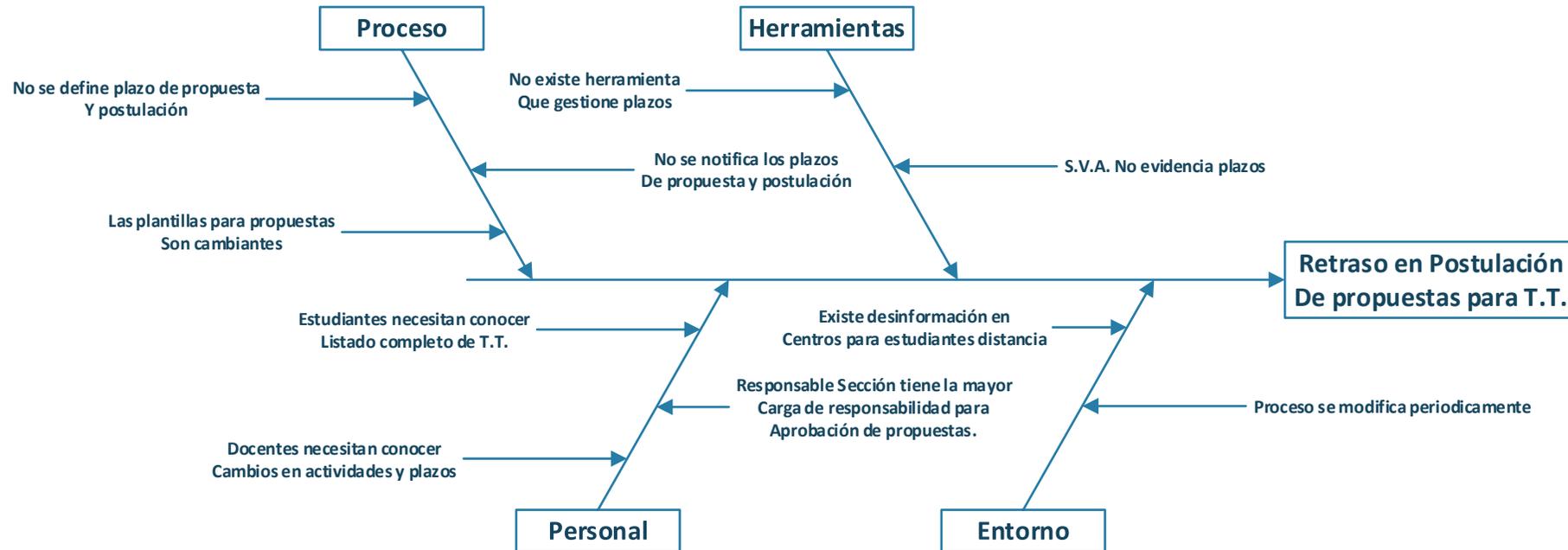


Figura 10. Diagrama causa efecto – Retraso en postulación de T.T.

Fuente: El autor.

Elaboración: El autor.

2.4.3.1. Análisis y solución a causa raíz

Se procede a elaborar la matriz de análisis y se muestra en la tabla 16.

Tabla 16. Análisis: “Retraso en postulaciones de propuestas T.T.”

CAUSAS	SOLUCIONES	CRITERIOS					TOTALES
Proceso	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
No se define plazo de propuesta y postulación	Creación de actividades de una convocatoria en la herramienta.	3	3	3	3	3	15
No se notifica plazos de propuesta y postulación	Notificación de creación de convocatoria en la herramienta.	2	2	2	3	1	10
Las plantillas para propuestas son cambiantes.	Creación de plantillas personalizadas en la herramienta.	2	1	2	3	1	9
Herramientas	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
No existe herramienta que gestione los plazos	Creación de actividades de una convocatoria en la herramienta.	2	1	2	3	2	10
S.V.A. no evidencia plazos	Definición de plazos por actividad en la herramienta.	1	1	1	2	1	6
Personal	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Estudiantes necesitan conocer listado completo de T.T.	Subida automática de propuestas aprobadas a S.V.A.	2	3	3	3	2	13
Responsable Sección tiene la mayor carga de responsabilidad para aprobación de propuestas.	Asignación de revisores que realicen el trabajo de revisión de propuestas en la herramienta.	2	2	3	3	2	12
Docentes necesitan conocer cambios en actividades y plazos.	Notificación de cambio de plazos en las actividades en la herramienta.	1	1	3	2	2	9
Entorno	Solución	FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Existe desinformación en centros para estudiantes a distancia.	Notificación de nuevas convocatorias a la titulación en la herramienta.	1	1	2	2	1	7
Proceso se modifica periódicamente.	Mantener un proceso de calificación sólo modificable por coordinador en la herramienta.	1	2	3	3	2	11

Fuente: El autor.

Elaboración: El autor.

2.4.3.2. Histograma.

Ahora se describe histograma. Para esto es necesario extraer el grado de incidencia de la matriz de análisis. La figura 11 describe el histograma para el análisis del problema “Retraso en postulación de propuestas para T.T.”.



Figura 11. Histograma análisis– Retraso en postulación.
Fuente: El autor.
Elaboración: El autor.

De la misma manera, al atacar y resolver las principales causas se puede resolver el problema de que se retrase la propuesta y postulación de propuestas

2.4.4. Retraso en revisiones de propuestas para T.T.

La figura 12 muestra el diagrama causa-efecto para este problema.

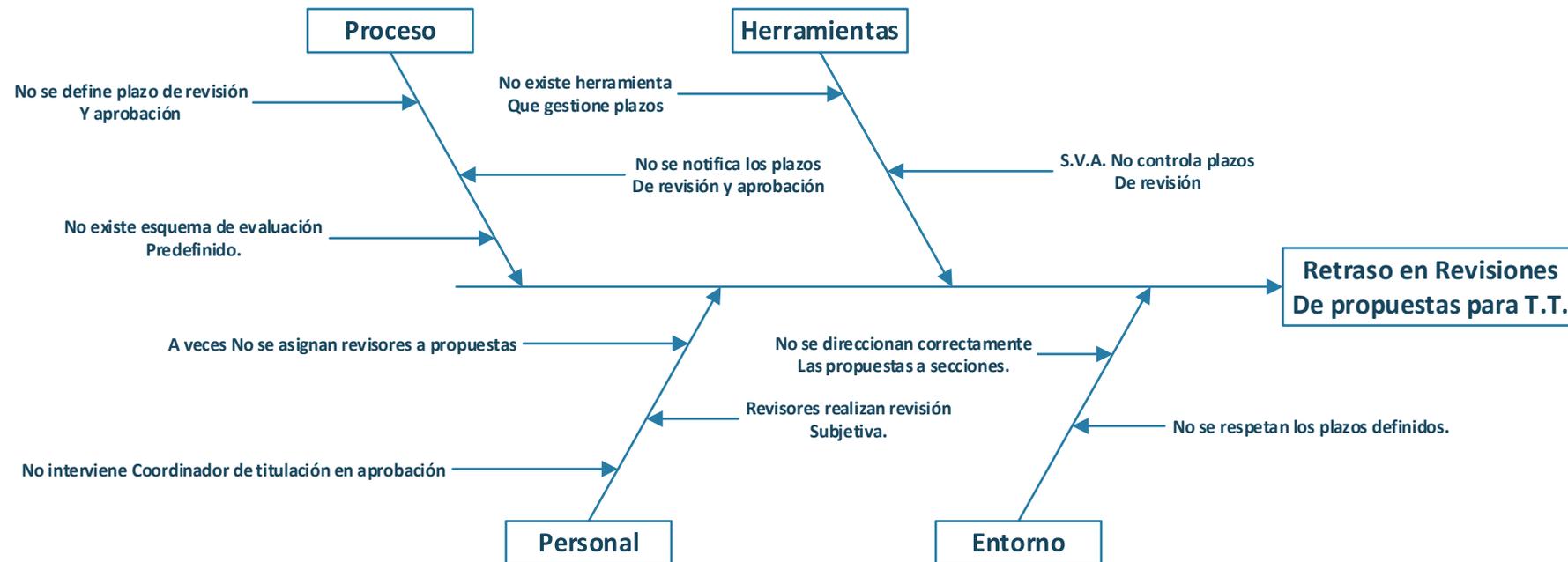


Figura 12. Diagrama causa efecto – Retraso en revisiones de T.T.

Fuente: El autor.

Elaboración: El autor.

2.4.4.1. Análisis y solución a causa raíz

Se procede a elaborar la matriz de análisis y se muestra en la tabla 17.

Tabla 17. Análisis: “Retraso en revisión de propuestas T.T.”

CAUSAS	SOLUCIONES	CRITERIOS					TOTALES
		FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Proceso	Solución						
No se define plazo de revisión y aprobación	Creación de actividades de una convocatoria en la herramienta.	3	3	3	3	3	15
No se notifica plazos de revisión y aprobación	Notificación de creación de convocatoria en la herramienta.	2	2	2	3	1	10
No existe un esquema de evaluación predefinido.	Creación de esquemas de evaluación en la herramienta.	3	2	2	3	3	13
Herramientas	Solución						
No existe herramienta que gestione los plazos	Creación de actividades de una convocatoria en la herramienta.	3	1	3	2	3	12
S.V.A. no controla plazos de revisión.	Definición de plazos por actividad en la herramienta.	1	1	1	3	1	7
Personal	Solución						
A veces no se asigna revisores a propuestas.	Asignación obligatoria de al menos un revisor diferente al responsable de sección en la herramienta.	2	1	2	3	1	9
Revisores realizan revisiones subjetivas.	Revisión según los parámetros especificados en el esquema de evaluación definido por la titulación en la herramienta	2	2	2	3	3	12
No interviene Coordinador de titulación en aprobación	Aprobación de Coordinador Titulación obligatoria en la herramienta.	2	1	2	3	1	9
Entorno	Solución						
No se direccionan correctamente las propuestas a las secciones departamentales.	Direccionamiento directo a responsable de sección correspondiente en la herramienta.	1	1	2	3	1	8
No se respetan los plazos definidos.	La herramienta restringe la ejecución de procesos fuera de los plazos establecidos.	3	2	2	3	3	13

Fuente: El autor.

Elaboración: El autor.

2.4.4.2. Histograma.

Ahora se describe histograma. Para esto es necesario extraer el grado de incidencia de la matriz de análisis. La figura 13 describe el histograma para el análisis del problema “Retraso en revisión de propuestas para T.T.”.

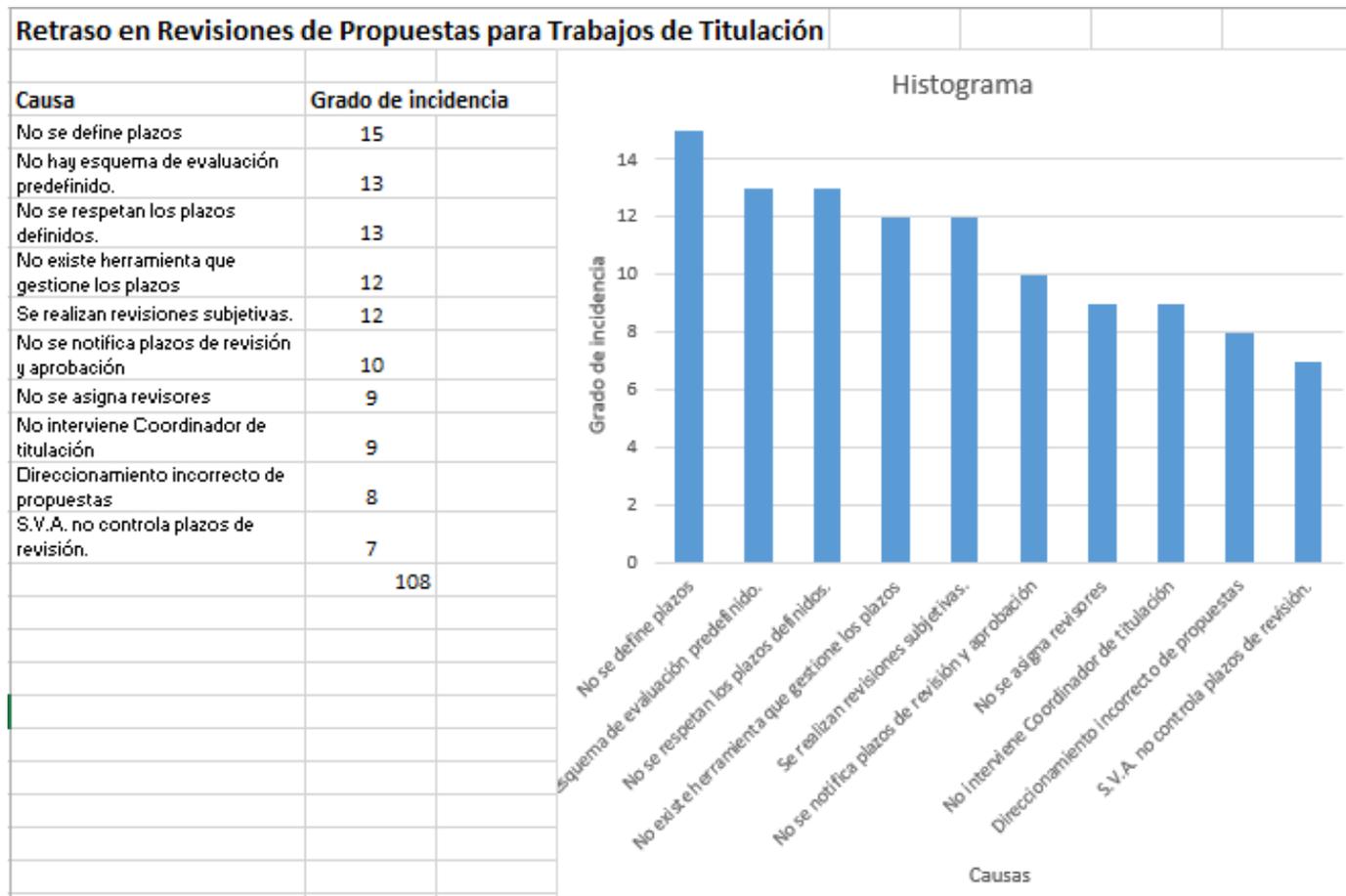


Figura 13. Histograma análisis– Retraso en revisión.
Fuente: El autor.
Elaboración: El autor.

De la misma manera, al atacar y resolver las principales causas se puede resolver el problema de que se retrase la revisión y aprobación de propuestas para T.T.

2.4.5. Dificultad para notificaciones.

La figura 14 muestra el diagrama causa-efecto para este problema.

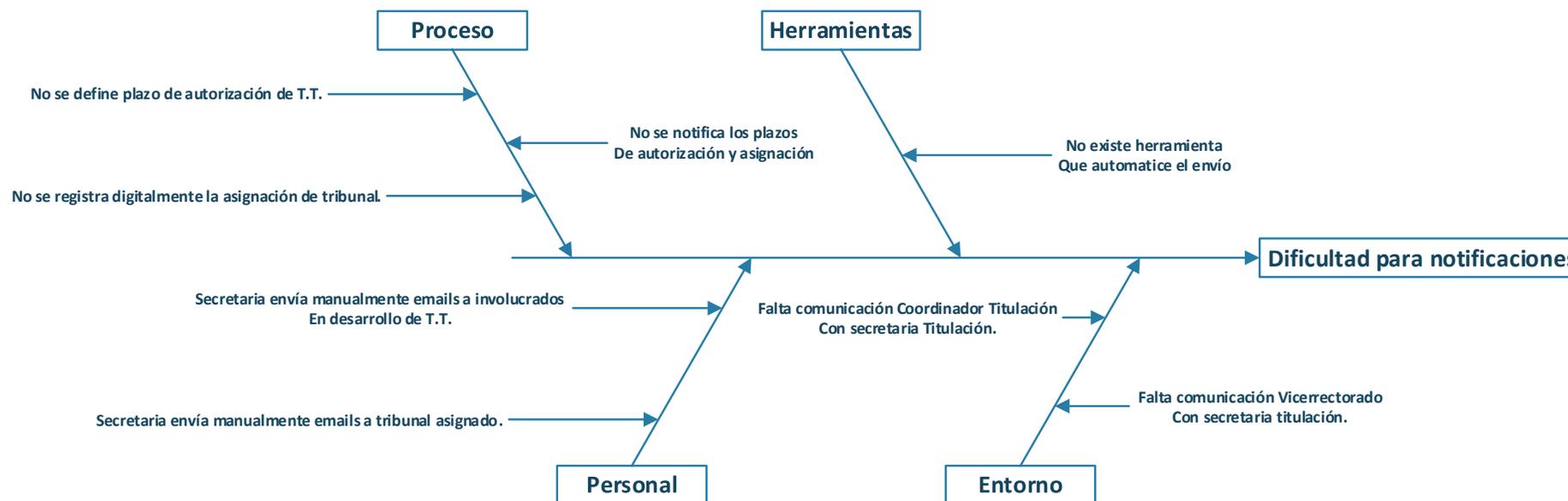


Figura 14. Diagrama causa efecto – Dificultad para notificaciones

Fuente: El autor.

Elaboración: El autor.

2.4.5.1. Análisis y solución a causa raíz

Se procede a elaborar la matriz de análisis y se muestra en la tabla 18.

Tabla 18. Análisis: “Dificultad para notificaciones.”

CAUSAS	SOLUCIONES	CRITERIOS					TOTALES
		FACTOR	CAUSA DIRECTA	SOLUCION	FACTIBLE	MEDIBLE	
Proceso	Solución						
No se define plazo de autorización de T.T.	Creación de actividades de una convocatoria en la herramienta.	3	3	3	3	3	15
No se notifica los plazos de autorización y asignación	Notificación de creación de convocatoria en la herramienta.	2	2	2	3	1	10
No se registra digitalmente la asignación de tribunal.	Asignación de tribunal a una propuesta completada en la herramienta.	2	1	1	3	2	9
Herramientas	Solución						
No existe herramienta que automatice el envío	Notificación automática en la herramienta.	2	3	3	3	1	12
Personal	Solución						
Secretaria envía manualmente emails a involucrados en desarrollo de T.T.	Envío automático de emails a los involucrados en cada parte del proceso.	1	2	2	2	1	8
Secretaria envía manualmente emails a tribunal asignado.	Envío automático de emails a los involucrados en cada parte del proceso.	1	2	2	2	1	8
Entorno	Solución						
Falta comunicación Coordinador Titulación con secretaria Titulación.	Notificación automática en la herramienta cuando el coordinador defina plazos.	1	1	1	2	1	6
Falta comunicación Vicerrectorado con secretaria titulación.	Notificación a vicerrectorado de la creación de una convocatoria.	1	1	1	2	1	6

Fuente: El autor.

Elaboración: El autor.

2.4.5.2. Histograma.

Ahora se describe histograma. Para esto es necesario extraer el grado de incidencia de la matriz de análisis. La figura 15 describe el histograma para el análisis del problema “Dificultad para notificaciones”.

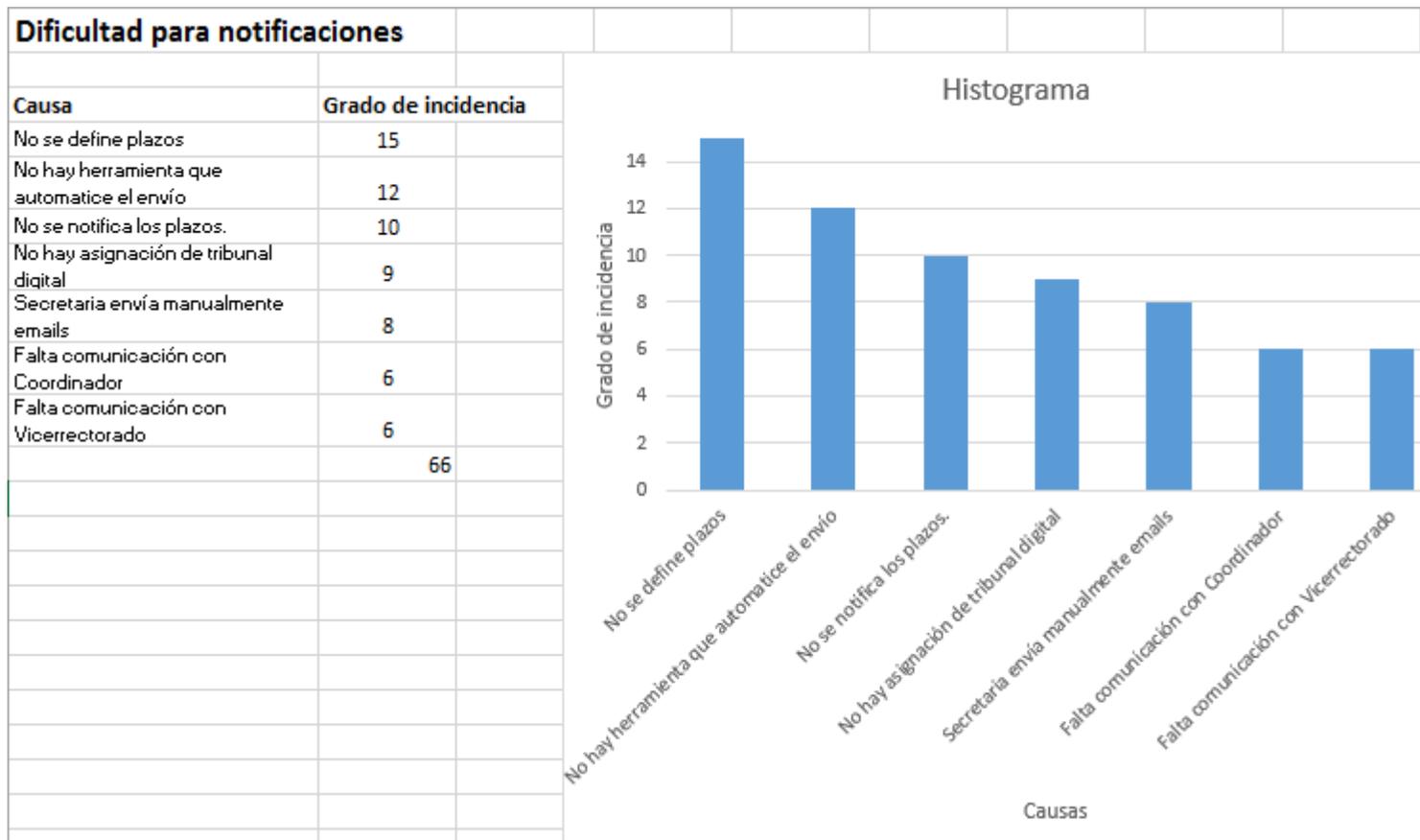


Figura 15. Histograma análisis– Dificultad para notificaciones.
Fuente: El autor.
Elaboración: El autor.

De la misma manera, al atacar y resolver las principales causas se puede resolver el problema de que se dificulta notificar a los involucrados en una parte del proceso de la postulación de T.T.

2.5. Propuesta de Solución

En base a las causas principales identificadas en el análisis para cada problema del proceso de postulación de trabajos de titulación, la siguiente matriz describe cada problema, las causas principales a atacar y la solución a implementar en este proyecto.

Tabla 2.5. Matriz problema-solución.

Problema	Causas	Solución
Falta de información de los T.T.	<ul style="list-style-type: none"> • Proceso manual. • No hay repositorio central. • S.V.A. no genera reportes de T.T. 	Gestión del proceso mediante una herramienta en línea, que dispondrá una base de datos centralizada con los T.T para poder consultar y generar reportes.
Reprobación y abandono de T.T.	<ul style="list-style-type: none"> • Propuestas mal revisadas • No existe una herramienta para revisión y validación de propuestas. • El coordinador de titulación no interviene en la aprobación de propuestas. 	Implementación de un módulo de calificación en la herramienta, en donde se pueda calificar mediante una rúbrica definida por la titulación cada propuesta de T.T y sea obligatoria la aprobación del Coordinador de Titulación para aprobar una propuesta.
Retraso en postulación de propuestas para T.T.	<ul style="list-style-type: none"> • No se definen plazos de propuesta y postulación. • Estudiantes necesitan conocer el listado completo de T.T. • Responsable de Sección es el principal actor. 	Creación de un módulo de convocatorias en la herramienta, donde se definirán las actividades y plazos para propuesta y para postulación. La herramienta restringirá el acceso a estas actividades si se encuentran fuera de los plazos. La herramienta también

		delegará la revisión a docentes revisores, siendo el Responsable de Sección únicamente el validador final de las calificaciones realizadas por los revisores a una propuesta de T.T.
Retraso en revisiones de propuestas para T.T.	<ul style="list-style-type: none"> • No se definen plazos de revisión y aprobación. • No hay un esquema de evaluación de propuestas predefinido para la titulación. • No se respetan los plazos definidos por la titulación. 	<p>En el módulo de convocatorias de la herramienta se podrá definir el plazo para las actividades de revisión y aprobación, y se restringirá el acceso a estas actividades si están fuera del plazo.</p> <p>La herramienta podrá definir un esquema de evaluación de propuestas activo por convocatoria, con este esquema de evaluación los docentes revisores podrán calificar una propuesta.</p>
Dificultad para notificaciones	<ul style="list-style-type: none"> • No se definen plazos de autorización. • No hay una herramienta que automatice el envío. • No se respetan los plazos definidos por la titulación. 	<p>La herramienta notificará automáticamente vía email la asignación de docentes a un proyecto, y a los involucrados en una actividad realizada.</p> <p>Así mismo se notificará a los involucrados la creación de una nueva convocatoria con las actividades y sus plazos definidos.</p>

La figura 16 representa un esquema de la solución propuesta para la implementación del software e integración con las metodologías y herramientas propuestas.

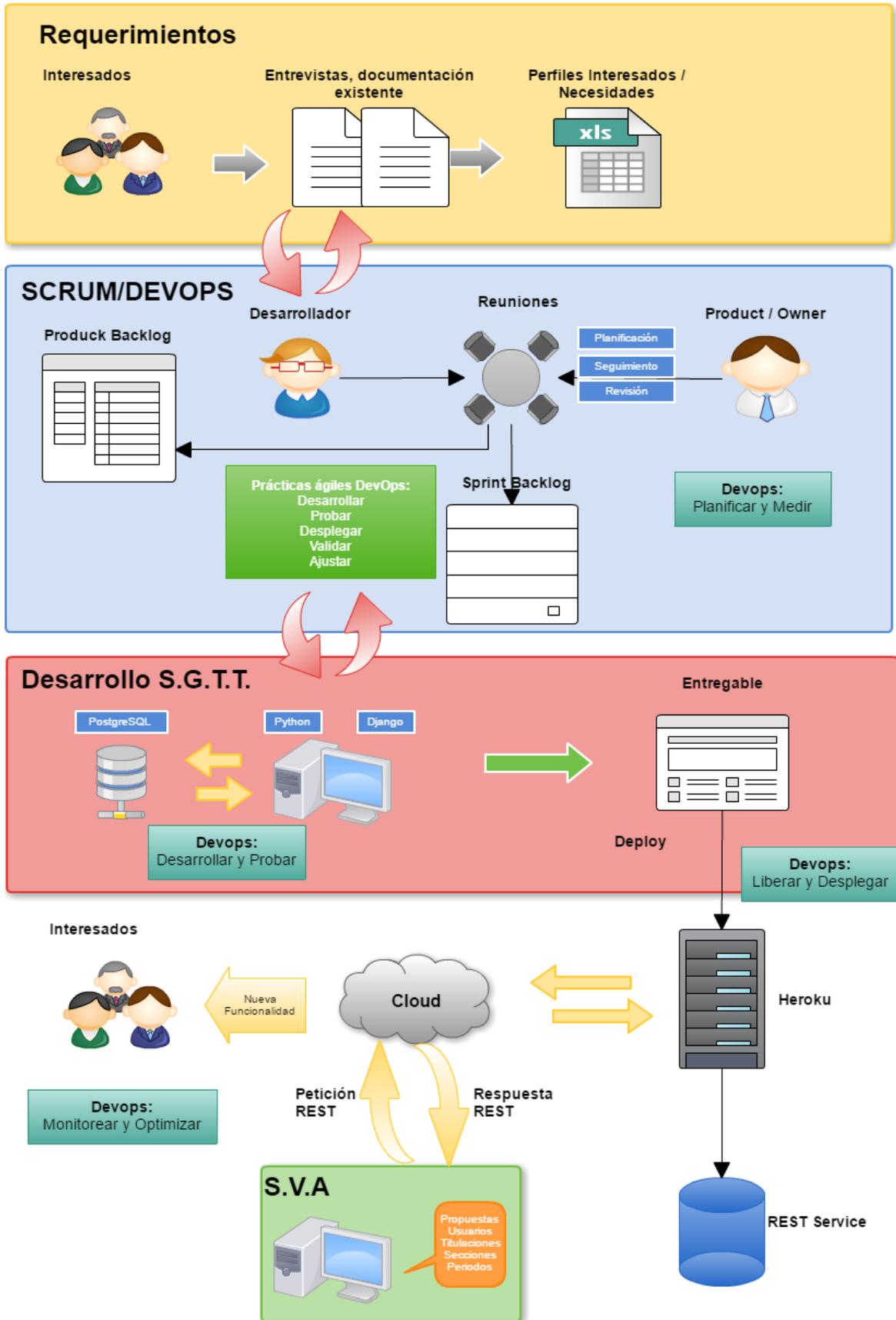


Figura 16. Diagrama Propuesta de solución.
Fuente: El autor.
Elaboración: El autor.

El presente proyecto requiere de un proceso de ingeniería de requerimientos, para lo cual se implementa el uso de un conjunto de técnicas y buenas prácticas para la elicitación de requerimientos. Se usan herramientas, técnicas y modelos que se usa para extraer, analizar, especificar validar y administrar los requerimientos de software según (Gottesdiener, 2005).

Además, se realizará el modelado UML del sistema, usando las técnicas de modelado UML como las habilidades de comunicación necesarias para alcanzar el éxito de un proyecto descritas por (Bruegge & Dutoit, 2010).

Para el desarrollo de la aplicación web solución, se utilizará el lenguaje Python en conjunto con el framework Django consideradas como mejor arquitectura de software según el análisis realizado en la sección 1.3.3 y en la sección 1.8. Se utiliza este lenguaje por su facilidad de aprendizaje y este framework por su desarrollo rápido, ordenado y limpio, y porque dispone de un ORM que puede gestionar las bases de datos candidatas para este proyecto.

El proyecto pretende integrar esta aplicación con el actual “Sistema de Gestión de Trabajos de Titulación” del Vicerrectorado Académico, por lo tanto se implementará un servicio REST de solicitud/respuesta para su acoplamiento.

La aplicación se enfoca en los puntos del 1 al 4, descritos en la Figura 5, desde la elaboración de la propuesta hasta la postulación de la propuesta. La siguiente etapa, la inscripción y selección de estudiantes, la realiza el Sistema de Vicerrectorado Académico (S.V.A.), y se comunicará con éste mediante un servicio REST. Es decir, la aplicación web mostrará las propuestas de Trabajos de Titulación autorizados por el Consejo de Departamento en un formato consumible para el S.V.A y que este las ponga a disposición de los estudiantes.

Además, se desarrollará bajo la metodología de desarrollo Scrum, considerada la más acorde para cubrir las necesidades del proyecto, y se mantendrá los principios ágiles de DevOps. Se realizarán reuniones semanales con el Product Owner (Coordinador de Titulación de Sistemas Informáticos) para definir y priorizar las historias de usuario y fijar los criterios de aceptación para cada entregable. En estas reuniones se ejecuta la “Reunión de revisión” y “Reunión de seguimiento” para aceptar un entregable, y seguidamente hacer la “Reunión de planificación” para definir las historias de usuario a desarrollar para el siguiente entregable.

Además de las historias de usuario como documentación de la metodología se necesita un modelado que proporcione robustez al sistema para su comprensión e implementación, por tanto se vinculará con documentación de metodologías tradicionales usando para el

modelado de la arquitectura de software las partes más relevantes del modelo 4+1 de Krutchen.

Se utilizará el servicio de alojamiento en la nube Heroku, para las entregas continuas de la metodología Scrum y el enfoque DevOps de despliegues funcionales continuos. Por tanto, al finalizar el desarrollo de cada entregable, este se desplegará el servicio Heroku para que pueda ser revisado por el Product Owner u otros interesados y poder realizar la revisión del mismo.

Se utilizará un esquema de base de datos relacional PostgreSQL, ya que para la implementación con Heroku este motor es el que permite por defecto utilizar la versión gratuita de este servicio.

Cada nueva funcionalidad implementada y aceptada por el Product Owner es necesario que sea validada por otros usuarios directos del sistema, para definir cambios en el nuevo Sprint e implementar las nuevas características o funcionalidades que estos soliciten.

La aplicación deberá estar disponible en línea las 24 horas, y parametrizada con los formatos de comunicación para el S.V.A.

CAPÍTULO 3: DESARROLLO DE LA SOLUCIÓN

3.1. Introducción

Una vez descritas y analizadas las herramientas pertinentes para el desarrollo del presente proyecto, este capítulo se enfoca en el desarrollo de la solución propuesta en la sección 2.5, con una selección óptima de las herramientas según el análisis en la sección 1.8 y una correcta permutación entre las mismas que permita una integración eficiente según el alcance y los objetivos del proyecto.

Se sigue el flujo de desarrollo recomendado en el libro *“Prentice – Object Oriented Software”* para elaborar el modelado del sistema, obtener una mejor comprensión y detalle desde el inicio del sistema hasta su solución final. Se inicia el desarrollo con la definición del problema, los objetivos, requerimientos funcionales, requerimientos no funcionales e integrando con la metodología Scrum, elaborando el Product BackLog con los actores identificados, las características del sistema que necesitan y los escenarios de aceptación de cada uno. Cada paso de este flujo va relacionado con las vistas del modelo 4+1 de Krutchen, y se especificarán los diagramas concernientes a cada vista.

Luego de elaborado el Product BackLog se vio necesaria la especificación detallada de casos de uso que recomienda el libro, para definir atributos, restricciones y flujos alternos que el Product BackLog no contempla en su totalidad, y esta especificación detallada sirve como documentación complementaria a la metodología Scrum para el proyecto.

Este esquema de desarrollo va vinculado con las entregas continuas de Scrum y las prácticas ágiles de DevOps (Desarrollar, probar, desplegar, validar, ajustar). Una vez definido el Product BackLog y los casos de uso se desarrolla la primera iteración o Sprint utilizando el tablero de la metodología Kanban.

Se utilizan las herramientas descritas en la sección 2.3 para extraer correctamente los requerimientos y documentarlos de manera completa según las plantillas suministradas por los libros analizados y a la documentación sugerida por el estándar UML.

El diseño e implementación se realiza con el lenguaje Python más el framework Django. El framework utiliza el patrón de diseño Modelo Plantilla Vista o MTV (Model, Template, View) analizado en la sección 1.3.3.1.

Para la implementación y despliegues con Heroku es necesario utilizar el motor de base de datos PostgreSQL, debido a que la versión gratuita de este servicio de alojamiento sólo permite este motor de base de datos. Por consiguiente el framework se configura para este motor de base de datos.

3.2. Requerimientos

3.2.1. Problema.

Partimos del enunciado del problema inicial, que constituye el objetivo global de este trabajo.

“Implementación de una aplicación para gestionar los procesos de elaboración, valoración postulación y asignación de trabajos de titulación par la carrera de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática”.

El problema abarca a todas las titulaciones de la UTPL, al estar sujeta cada titulación al Reglamento de Régimen Académico, cada titulación necesita llevar un control del proceso de postulación de trabajos de titulación. Este proceso comprende desde la elaboración de una propuesta para trabajo de titulación, su revisión, calificación, aprobación, autorización, postulación, asignación, evaluación, hasta la finalización del trabajo de titulación por el estudiante.

Actualmente, cada titulación de la UTPL maneja su propio esquema de evaluación y proceso de postulación de trabajos de titulación, ateniéndose al reglamento, pero con esquemas de evaluación diferentes.

Esta solución permitirá integrar el proceso, creando un sistema centralizado, homogéneo, dónde cada titulación de la UTPL pueda definir un esquema de evaluación y que éste pueda ser reutilizado por otras titulaciones, facilitando además la gestión y el seguimiento de todas las propuestas de trabajos de titulación para los coordinadores de las titulaciones de la UTPL.

3.2.2. Objetivos.

Los objetivos de la aplicación son:

- Proveer una infraestructura para operar con las diferentes titulaciones de la UTPL, que incluya la oferta de convocatorias, creación de esquemas de evaluación de propuestas y de esquemas de evaluación para trabajos de titulación finalizados, subida de propuestas de trabajos de titulación, asignación de revisores a una propuesta de trabajo de titulación, calificación de una propuesta mediante el esquema de evaluación de propuesta definido, aprobación por responsables de sección y coordinadores de titulación, postulación de estudiantes, asignación de equipo de acompañamiento a un trabajo de titulación, asignación de tribunal a un trabajo de titulación, calificación de un trabajo de titulación completado y notificación a los involucrados en un trabajo de titulación.

- Proveer un sistema integrado con el actual Sistema de Gestión a Trabajos de Titulación del Vicerrectorado Académico (S.V.A.).
- Proveer una infraestructura centralizada y homogénea al proceso de postulación de trabajos de titulación, dónde se pueda consultar y dar seguimiento a los trabajos de titulación por parte de los coordinadores de las titulaciones.
- Proveer un esquema de calificación y aprobación de propuestas de trabajos de titulación, dónde las propuestas no se puedan aprobar si no alcanzan la calificación mínima definida por la titulación.
- Proveer un sistema donde los autores de las propuestas de trabajos de titulación puedan conocer el estado de sus propuestas.
- Proveer un sistema con un esquema de evaluación fácil para los revisores de cada sección departamental.
- Proveer un sistema con un esquema de asignación de equipos y de tribunal sencillo para los responsables de sección y coordinadores de titulación.
- Proveer un sistema de notificación de asignaciones a los involucrados en un trabajo de titulación sencillo para las secretarías de las titulaciones.
- Proveer un sistema con un modelo de calificación sencillo para los presidentes de un trabajo de titulación completado según el esquema de evaluación definido por la titulación.

3.2.3. Requerimientos Funcionales.

Los requerimientos funcionales (RF) se identifican como las características y servicios que debe proporcionar la solución, contemplando el comportamiento que debe tener la misma según las entradas particulares de un usuario.

En el contexto del presente proyecto, los RF son extraídos usando la metodología Scrum y descritos en forma de historias de usuario (HU), en donde se describe la característica que desea cada usuario y el criterio de aceptación que este tiene sobre el comportamiento que tendrá la solución.

Las HU pasan a formar parte del Product BackLog que se describe en la sección 3.2.4.

3.2.4. Product BackLog.

El Product BackLog de la metodología Scrum se desarrolla a partir de las entrevistas que constan en el anexo 8, y de la aplicación de prototipos exploratorios del anexo 12. Se describen las historias de usuario que constituyen el Product BackLog. Cada historia tiene un identificador, una prioridad, rol, característica, razón y criterios de aceptación de cada usuario directo.

El Product BackLog se muestra en el anexo 8. Este es el Product BackLog inicial, que describe los requerimientos al inicio del desarrollo de este proyecto. Hay que considerar que se irán implementando nuevas características con cada entrega funcional, las cuáles pasarán a formar parte del Sprint BackLog.

3.2.5. Identificación de actores y responsabilidades

Los actores se representan a continuación en base al análisis desarrollado en el capítulo anterior en la categorización de interesados tabla 12, de las entrevistas y del Product BackLog desarrollado. Se han identificado 10 actores, uno por cada tipo de usuario descrito en la tabla 13 perfiles de interesados.

En la siguiente tabla 19 se describe los actores identificados, conjuntamente con las tareas y responsabilidades que desempeña cada uno.

Esta tabla sirve para posteriormente detallar los casos de uso, las interfaces a las que tiene acceso cada usuario y la configuración en el framework Django con estos nombres, para restringir los accesos y privilegios de cada actor.

Tabla 19. Identificación de actores

Actor	Tareas/Responsabilidades
Estudiante Distancia	<ul style="list-style-type: none">• Subir propuesta de trabajo de titulación (PropuestaTrabajoTitulacion)• Postula a una PropuestaTrabajoTitulacion.
Estudiante Presencial	<ul style="list-style-type: none">• Postula a una PropuestaTrabajoTitulacion.
Docente	<ul style="list-style-type: none">• Subir propuesta de trabajo de titulación (PropuestaTrabajoTitulacion)• Asigna estudiantes que han postulado a su PropuestaTrabajoTitulacion.

	<ul style="list-style-type: none"> • Ver e imprimir proyectos asignados como director, que incluya estudiantes, docentes de apoyo y plazos.
Director Trabajo Titulación	<ul style="list-style-type: none"> • Emite informe de finalización de trabajo de titulación cuando está en estado de “ejecución”, para pasar a estado de “completado”.
Presidente Trabajo Titulación	<ul style="list-style-type: none"> • Emite calificación de trabajo de titulación cuando está en estado “Revisión Tribunal” para pasar a estado “Finalizado”.
ResponsableSeccion	<ul style="list-style-type: none"> • Asigna revisores a PropuestaTrabajoTitulacion • Modifica asignación de revisores. • Aprueba / Rechaza PropuestaTrabajoTitulacion calificados. • Solicita modificación a autor de propuesta. • Solicita recalificación a revisor de propuesta. • Asigna Equipo de apoyo: Director y docentes, a TrabajoTitulacion.
RevisorSeccion	<ul style="list-style-type: none"> • Visualiza listado de PropuestaTrabajoTitulacion a revisar asignadas a su persona. • Califica PropuestaTrabajoTitulacion mediante Rubrica activa vinculada a la propuesta.
CoordinadorTitulacion	<ul style="list-style-type: none"> • Administrar Periodos académicos (Crear, Editar, Eliminar). • Administrar Convocatorias (Crear, Editar, Eliminar). • Administrar esquemas de evaluación (rúbricas) para revisión de PropuestaTrabajoTitulacion. (Crear, Editar, Eliminar). • Administrar esquemas de evaluación (rúbricas) para revisión de Trabajos de titulación terminados por el estudiante. • Aprueba / Rechaza revisión de PropuestaTrabajoTitulacion. • Solicita modificación a autor de propuesta. • Solicita recalificación a revisor de propuesta. • Asigna Tribunal (Presidente y vocales) a PropuestaTrabajoTitulacion.

	<ul style="list-style-type: none"> • Revisa propuestas subidas por estudiantes. • Revisa reportes de estado de las propuestas.
DirectorDepartamento	<ul style="list-style-type: none"> • Editar convocatoria para modificar plazos de revisión de propuestas. • Autoriza el desarrollo de un TrabajoTitulacion una vez asignados Presidente, Tribunal (vocal1 y volcal2) y estudiante. • Envía propuestas autorizadas a Sistema Vicerrectorado Académico (S.V.A.). • Emite acta con trabajos de titulación autorizados.
Secretaria	<ul style="list-style-type: none"> • Formaliza la inscripción de una PropuestaTrabajoTitulacion con su Director, tribunal y estudiante. • Notifica a equipo de acompañamiento (Director y docentes) y a estudiantes mediante documento formal la aprobación de un Trabajo de Titulación para iniciar su desarrollo y pasa a estado "Ejecución". • Notifica a Tribunal la asignación y revisión de Trabajo Final para calificación.

Fuente: El autor.

Elaboración: El autor.

3.2.6. Identificación de necesidades.

Para la elaboración del primer Sprint BackLog de Scrum, es necesario establecer cuál historia de usuario y necesidad se debe desarrollar primero. Para esto se desarrolla la matriz de identificación de necesidades, dónde cada necesidad está vinculada con una historia de usuario del Product BackLog y nos permite establecer el orden de desarrollo de las mismas según su prioridad y complejidad.

Las necesidades se detallan en la siguiente matriz 20, asignándoles un identificador, la historia de usuario del Product BackLog a la que se enlaza, el origen, la prioridad y complejidad asociados a las mismas.

Tabla 20. Matriz de identificación de necesidades.

ID	HU	Necesidad	Origen	Prioridad	Complejidad
----	----	-----------	--------	-----------	-------------

N001	HU-003	Un docente necesita subir su propuesta para un trabajo de titulación y que la misma sea revisada y autorizada para la posterior postulación de los estudiantes.	Documento de lineamientos para postulación de trabajos de titulación.	Media	Baja
N002	HU-011	La titulación necesita poder realizar una convocatoria para el inicio de proceso de postulación de propuestas para trabajos de titulación	Documento de lineamientos para postulación de trabajos de titulación.	Alta	Media
N003	HU-017	La titulación necesita poder crear un esquema de validación o rúbrica, para definir los parámetros de calificación y los ítems a calificar de una propuesta para trabajo de titulación.	Documento de lineamientos para postulación de trabajos de titulación.	Alta	Alta
N004	HU-018	La titulación necesita poder crear un esquema de validación o rúbrica, para definir los parámetros de calificación y los ítems a calificar de un trabajo de titulación terminado por el estudiante.	Coordinador titulación Ingeniería Civil.	Alta	Alta
N005	HU-008	El responsable de una Sección Departamental necesita asignar uno o varios revisores a cada propuesta que un docente haya subido para su sección.	Responsable Sección: Inteligencia Artificial	Media	Baja
N006	HU-005	El revisor de una Sección Departamental necesita poder calificar una propuesta asignada a su persona por el Responsable de Sección, según el esquema de evaluación o rúbrica definida por la titulación. Y emitir su criterio si la propuesta es aprobada, rechazada o requiere	Coordinador Titulación: Sistemas Informáticos y computación.	Media	Alta

		modificaciones.			
N007	HU-013	El Responsable de Sección Departamental y el Coordinador de Titulación necesitan poder visualizar una revisión de una propuesta, desglosando la calificación de cada revisor asignado, para poder aprobar la revisión, solicitar recalificación o rechazar la propuesta.	Coordinador Titulación: Arquitectura.	Alta	Baja
N008	HU-019	El consejo de departamento necesita poder visualizar las propuestas calificadas y aprobadas por responsables de sección y coordinador, para poder autorizar la ejecución de estos proyectos.	Coordinador Titulación: Sistemas Informáticos y computación.	Media	Baja
N009	HU-019	Titulación necesita poder enviar los proyectos autorizados al Sistema de Vicerrectorado Académico para que los estudiantes puedan visualizar las propuestas y postular a las mismas.	Coordinador Titulación: Sistemas Informáticos y computación.	Alta	Alta
N010	HU-020	Sistema Vicerrectorado Académico (S.V.A.) necesita poder consumir las propuestas autorizadas por cada titulación en el formato definido por su esquema de datos, para proceder a mostrarlas para los estudiantes de modalidad presencial matriculados en el periodo académico correspondiente.	Coordinador Titulación: Sistemas Informáticos y computación.	Alta	Alta
N011	HU-021	Los estudiantes de modalidad a distancia necesitan visualizar las propuestas ofertadas para su titulación para poder postular a las	Coordinador Titulación: Sistemas Informáticos y computación.	Baja	Baja

		mismas.			
N012	HU-022	El Responsable de Sección necesita poder asignar el equipo de acompañamiento (director, docentes) a las propuestas autorizadas.	Coordinador titulación Ingeniería Civil.	Media	Baja
N013	HU-023	La Secretaria necesita poder extraer del S.V.A las propuestas con estudiante, para notificar al estudiante y al director asignado.	Secretaria Sistemas Informáticos y Computación.	Media	Media
N014	HU-024	La titulación necesita poder consultar del S.V.A las propuestas terminadas.	Coordinador Titulación: Sistemas Informáticos y computación.	Alta	Alta
N015	HU-025	El coordinador de titulación necesita poder asignar el tribunal a un trabajo de titulación terminado.	Coordinador Titulación: Sistemas Informáticos y computación.	Media	Baja
N016	HU-026	La secretaria necesita poder notificar al tribunal para que revisen un trabajo de titulación terminado.	Coordinador Titulación: Sistemas Informáticos y computación.	Media	Baja
N017	HU-027	El docente asignado como presidente de tribunal necesita poder visualizar y calificar un trabajo de titulación terminado según el esquema de evaluación definido por la titulación.	Coordinador Titulación: Sistemas Informáticos y computación.	Alta	Media
N018	HU-017	Cada titulación necesita poder visualizar las rúbricas del área para poder reutilizar dicha rúbrica en una convocatoria.	Coordinador Titulación Arquitectura.	Baja	Media
N019	HU-028	El coordinador de titulación necesita poder visualizar un reporte de todas las propuestas de su titulación con su estado y poder filtrarlas	Coordinador Titulación: Sistemas Informáticos y computación.	Baja	Baja

		para realizar búsquedas personalizadas.			
--	--	---	--	--	--

Fuente: El autor.

Elaboración: El autor.

3.2.7. Identificación de casos de uso

A continuación se desarrolla el caso de uso global que abarque todas las funcionalidades del sistema con sus actores identificados, para posteriormente detallar cada caso de uso textualmente.

El caso de uso global se muestra en el anexo 4. La figura contempla los actores y las funcionalidades específicas de cada uno.

A continuación se detalla cada caso de uso en el anexo 5, donde se encuentra una descripción del nombre del caso de uso, descripción, el flujo básico, flujos alternativos, requerimientos especiales, precondiciones, post condiciones y puntos de extensión.

Esta documentación es complementaria a las historias de usuario de Scrum que hemos seleccionado, y sirve de respaldo para tener una concepción más a fondo de los requerimientos del sistema.

3.2.8. Requerimientos No Funcionales

Los requerimientos no funcionales van surgiendo desde variadas fuentes durante el proceso de extracción de requerimientos. Según se va definiendo los escenarios y describiendo el detalle de cada caso de uso van apareciendo requisitos de calidad que necesitan los usuarios del sistema. La tabla 21 describe los requerimientos no funcionales esenciales.

Tabla 21. Requerimientos no funcionales.

Categoría	Requerimientos No Funcionales
Usabilidad	<ul style="list-style-type: none"> • El sistema debe ser accesible desde un navegador web. • El sistema debe presentar una pantalla de inicio con las tareas específicas que le corresponden al usuario. • El sistema debe mostrar mensajes de alerta cuando el usuario intente acceder a funcionalidades no permitidas. • El sistema debe funcionar correctamente si se accede

	desde un navegador de un dispositivo móvil.
Confiabilidad	<ul style="list-style-type: none"> • El sistema debe descartar una transacción de datos cuando ocurra un error en el sistema, es decir, no guardar datos subyacentes que dependan de un dato inicial. Ejemplo. No guardar actividades que dependan de una convocatoria, o la convocatoria sin actividades si el sistema deja de funcionar. • El sistema no debe permitir eliminar un objeto que ya haya sufrido una modificación o haya sido utilizado para otro proceso por otro usuario en el sistema.
Rendimiento	<ul style="list-style-type: none"> • El sistema debe soportar el acceso de múltiples usuarios. • El sistema debe responder en el mínimo de tiempo ante las consultas que se realicen entre sistemas. Ejemplo. Consulta del S.V.A. de propuestas autorizadas.
Eficiencia	<ul style="list-style-type: none"> • Toda transacción de negocio debe responder al usuario en menos de 5 segundos. • El sistema debe ser capaz de operar adecuadamente con múltiples transacciones. • Los usuarios deben poder ver los cambios reflejados en una propuesta en no más de 5 segundos.
Seguridad	<ul style="list-style-type: none"> • Todos los formularios enviados por el sistema deben tener una protección contra ataques de falsificación de petición en sitios cruzados, de usuarios malintencionados fuera del sistema. • Cada usuario únicamente debe poder modificar los datos correspondientes a las responsabilidades de su rol. • El acceso y manipulación de datos desde el panel de administración del framework estará disponible sólo para el desarrollador encargado del mantenimiento del sistema. • Los datos del servidor, base de datos, y configuración del servidor en sí, no deben estar visibles en caso de fallo en el servidor de producción.

Fuente: Prentice Object Oriented Software.

Elaboración: El autor.

3.2.9. Diagramas de secuencia para cada caso de uso.

Ahora se detalla los diagramas de secuencia. Estos diagramas se muestran en el anexo 6, dónde se muestra un diagrama de secuencias para cada caso de uso.

Estos diagramas tienen relación con los diagramas solicitados por la vista lógica de la arquitectura 4+1 que se detalla en la siguiente sección.

Este proyecto utiliza las partes más relevantes de esta arquitectura y las relaciona con los requerimientos y la metodología de nuestro proyecto como documentación de apoyo para la comprensión del proceso global.

3.2.10. Vista de escenarios 4+1.

El modelo 4+1 en esta vista se encarga de modelar usando el diagrama UML de casos de uso. Esta vista trata de modelar la funcionalidad del sistema y describir las interacciones entre los sistemas y actores.

La vista de escenarios se enlaza con la etapa de requerimientos actual, dónde se describió los casos de uso como documentación complementaria en la sección 3.2.7.

El modelado de casos de uso se presenta en el anexo 4, y la especificación de casos de uso se describe en el anexo 5.

3.3. Análisis

Siguiendo el flujo de desarrollo, en esta sección se identifican entidades participantes, límites y objetos de control, y se los refina agregando atributos y asociaciones al análisis del modelo de objetos. Se identifican relaciones de herencia y se consolida el análisis del modelo de objetos.

3.3.1. Identificación de objetos entidad, límites, y objetos de control

En esta sección identificamos cada objeto de entidad extrayendo los sustantivos de los casos de uso, usamos los atributos de éstos como propiedades que pueden ser de tipos simples de datos, los pronombres que se refieren a colecciones son asociaciones.

3.3.1. Modelado de interacción entre objetos.

Luego se representan los objetos identificados en un diagrama de secuencias, describiendo las interacciones que ocurren durante un caso de uso para identificar asociaciones adicionales y atributos.

El modelado en diagramas de secuencia en patrones MVC y MTV sirve también para definir la comunicación entre el modelo de datos, la vista y la presentación en la plantilla en el caso del framework Django.

Los diagramas de secuencia se desarrollan para cada caso de uso, y se encuentran descritos en el anexo 6.

3.3.6. Vista Lógica 4+1

En lo concerniente al modelo 4+1 de Krutchen, esta vista se encarga principalmente de la abstracción para el modelado que apoye la comprensión de las características funcionales del sistema y de mecanismos y elementos de diseño que se van a utilizar.

Esta vista utiliza los diagramas de clases y de secuencia descritos en esta sección.

3.4. Diseño

En esta sección identificamos los objetivos de diseño, partiendo de los requerimientos no funcionales previamente descritos, para luego diseñar una descomposición de subsistemas inicial.

El modelado del diseño nos facilita la implementación en el framework Django, al dividir el sistema en subsistemas que pueden ser administrados individualmente.

3.4.1. Identificación de subsistemas

La identificación de subsistemas parte de los requerimientos funcionales identificados.

Podemos identificar los siguientes subsistemas:

- Gestión de Usuarios
- Gestión de Propuestas
- Gestión de Convocatorias
- Gestión de Esquemas de Evaluación
- Gestión de Revisiones
- Notificaciones

Modelamos la interacción de estos subsistemas utilizando un diagrama de componentes. Este diagrama se muestra en el anexo 9.

3.4.2. Modelado de componentes en Django

Para facilitar la tarea del programador, se realiza un modelado de componentes considerando el patrón de diseño MTV que utiliza Django.

El framework permite crear subsistemas que denomina “aplicación” (apps). Cada *app* que se crea el framework genera un esquema de archivos, principalmente implementa un archivo de lógica llamado “*view*” y un archivo para escribir el modelo de datos llamado “*model*”.

En el anexo 10, se describe el modelado de estos componentes como subsistemas (apps) para el framework Django.

3.4.3. Vista de Procesos 4+1

El modelo 4+1 en esta vista representa los procesos negocio, su secuencia y comunicación entre cada uno de estos procesos. Para el modelado esta vista puede incluir el diagrama de actividad o el diagrama de estado.

Este modelado tiene correspondencia con la fase de diseño actual, y su diagramación ayuda a la comprensión del comportamiento del sistema para el programador.

Para el contexto de la aplicación, el proceso central de negocio lo constituyen los trabajos de titulación. Por consiguiente se modelará los estados que atraviesa una “propuesta de trabajo de titulación (T.T.)” utilizando el diagrama de estados UML.

El diagrama de estados de una Propuesta de T.T. se encuentra modelada en el anexo 11.

3.5. Implementación

Para la implementación del sistema se toma en cuenta todos los objetos e interacciones identificados para desarrollarlos en el framework Django, que fue el seleccionado para el desarrollo de la aplicación.

Django ejecuta sus sentencias directamente desde consola usando comandos Python.

Para la implementación inicial de Django se requiere los siguientes pasos:

1. Instalación de gestor de paquetes *pip*.
2. Instalar Django: *pip install django*.
3. Crear proyecto Django con el comando: *django-admin.py startproject nombreProyecto*
4. Correr migraciones: *python manage.py migrate*
5. Crear apps: *python manage.py startapp nombreApp*

Cada app de Django corresponde a un componente en el anexo 10.

El framework se configura para trabajar con el motor de base de datos Postgres.

Una vez definidas las *apps* en Django, iniciamos el mapeo de datos según las características identificadas a lo largo del diseño. El mapeo de datos se realiza en el archivo *model* de cada *app*.

Para esto es imperativo contar con el diagrama de datos. El diagrama de datos se especifica en la siguiente sección.

3.5.1. Diagrama de Datos

En este proyecto, el diagrama de datos constituye una parte elemental, puesto que en el framework Django el modelado de datos se define mediante lenguaje Python especificado mediante clases y variables, por lo tanto cada relación y campo debe estar correctamente identificado y diagramado, para poder hacer el seguimiento cada vez que ocurra una modificación al esquema de datos.

El modelo de datos se muestra en el anexo 7, dónde figura el diagrama final especificado luego de haber ejecutado las migraciones que han impuesto cada cambio en el modelo de datos en el avance del desarrollo del proyecto.

Cada tabla del diagrama de datos va vinculada con la *app* que le corresponde, tanto a la tabla con el nombre de la *app* como a las tablas relacionadas por una llave foránea.

Una vez identificadas las tablas que corresponde a cada *app*, se realiza el mapeo mediante código Python.

3.5.2. Despliegue en Heroku

Como parte de la fase de implementación y siguiendo el esquema de Scrum y los principios de DevOps: desarrollar, probar, desplegar; es necesario configurar el sistema para realizar los despliegues a Heroku, que fue la el servicio cloud seleccionado para el despliegue de la aplicación.

En el anexo 14 se describe el proceso de instalación y configuración de Heroku utilizado para el presente proyecto.

Al realizar el despliegue la consola debe mostrar al final un mensaje con el estado del despliegue satisfactorio.

Es necesario realizar un despliegue cada vez que se finalice un sprint o cuando se requiera desplegar una característica necesaria para la validación del Product Owner o de un usuario.

La URL de la aplicación configurada es utpl-sggt.herokuapp.com, mediante la cual se puede acceder a la herramienta y autenticarse en la misma para el uso de características según el rol que le corresponda.

3.5.3. Sprint BackLog

El framework se va configurando y desarrollando la aplicación según las historias de usuario que se vayan solicitando a medida que se realicen las reuniones con el Product Owner.

Para la asignación de Historias de Usuario y llevar un control del desarrollo se necesita el uso del tablero Kanban descrito en la sección 1.2.2.6, y para esto se ha utilizado la herramienta en línea <https://kanban-chi.appspot.com/>, que nos permite visualizar el progreso, asignar desarrolladores, sub tareas, ítems de desarrollo y vinculación de tarjetas de Historias de Usuario.

Esta herramienta nos permite crear un tablero que dispone de tres columnas: “To Do”, “Doing” y “Done”. Las mismas que son personalizables y se pueden agregar más columnas.

Para involucrar al Product Owner en la revisión directa del tablero, función que esta herramienta lo permite, se agrega una columna más antes de “Done” llamada “A verificar”. En esta columna se colocan las historias de usuario terminadas para ser verificadas por el Product Owner.

La herramienta permite crear historias de usuario (HU) fácilmente en cada columna, y posteriormente editar y agregar características a la HU como: etiqueta, color, cards relacionadas, prioridad, asignar a usuario, fecha inicio, fecha fin, estimación, tiempo, check list, adjuntos y comentarios.

La siguiente figura 17 muestra el tablero Kanban de la herramienta con la creación de una historia de usuario.

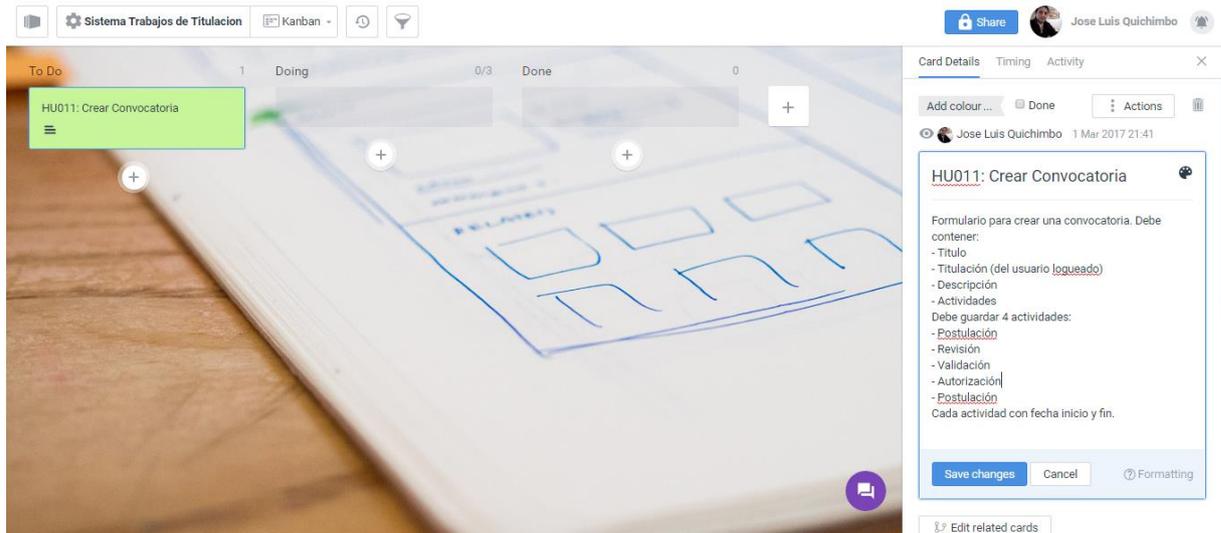


Figura 17. Tablero Herramienta Kanban AppSpot.
Fuente: Metodología Kanban.
Elaboración: El autor.

Las historias de usuarios son seleccionadas por el Product Owner según su prioridad y complejidad descritas en la tabla 20. Una vez elegidas se elabora el primer Sprint BackLog que constituye el primer entregable.

A continuación se detallan los Sprints que se han desarrollado secuencialmente en una matriz que detalla el plan de iteraciones.

3.5.3.1. Plan de iteraciones

Para iniciar las pruebas unitarias siguiendo los principios DevOps “Probar y Verificar” es necesario realizar una parametrización inicial de datos.

Esta parametrización inicial comprende la creación manual de entidades para fines de pruebas y validación. Estas tareas de creación son:

- Crear Áreas.
- Crear Modalidades.
- Crear titulaciones.
- Crear usuarios.
- Asignar coordinadores de titulación, secretarías.
- Crear secciones y sus responsables de sección.

Además se requiere crear un módulo de autenticación para el ingreso al usuario según sus privilegios.

Estas historias de usuario pasan a formar parte del Sprint 1 en el plan de iteraciones que se muestra en la matriz 22.

Tabla 22. Plan de Iteraciones.

PILA DEL SPRINT									Total Horas
BackLog ID	Tarea	Tipo	Estado	Responsable	Prioridad	Complejidad	Horas		
SPRINT1		Parametrización Inicial	Análisis - Diseño	Terminada	José Quichimbo	Alta	Baja	6	
	HU001	Login	Desarrollo	Terminada	José Quichimbo	Media	Media	12	
	HU011	Crear Convocatoria	Desarrollo	Terminada	José Quichimbo	Alta	Media	8	
	HU011	Crear Actividades Convocatoria	Desarrollo	Terminada	José Quichimbo	Alta	Media	4	
	HU014	Editar Convocatoria	Desarrollo	Terminada	José Quichimbo	Media	Media	4	
	HU011	Eliminar Convocatoria	Desarrollo	Terminada	José Quichimbo	Alta	Baja	2	
	Release: Convocatorias							36	36
SPRINT2	HU017	Crear esquema evaluación propuesta	Desarrollo	Terminada	José Quichimbo	Alta	Alta	48	
	HU018	Crear esquema evaluación trabajo completado	Desarrollo	Terminada	José Quichimbo	Alta	Alta	48	
	HU017	Editar esquema evaluación	Desarrollo	Terminada	José Quichimbo	Alta	Media	72	
	HU017	Editar esquema evaluación trabajo completado	Desarrollo	Pendiente	José Quichimbo	Alta	Media	72	
	HU017	Eliminar Esquemas evaluación	Desarrollo	Terminada	José Quichimbo	Media	Baja	6	

	Release: Esquemas Evaluación						246	282	
SPRINT3	HU003	Subir propuesta de T.T.	Desarrollo	Terminada	José Quichimbo	Media	Baja	48	
	HU004	Conocer estado de propuesta de T.T.	Desarrollo	Terminada	José Quichimbo	Media	Baja	8	
	Release: Subida y visualización de propuestas						56	338	
SPRINT4	HU008	ResponsableSeccion asigna docentes revisores.	Desarrollo	Terminada	José Quichimbo	Media	Baja	8	
	HU005	Revisor califica propuesta asignada a su usuario.	Desarrollo	Terminada	José Quichimbo	Media	Alta	72	
	Release: Asignación de revisores y calificación de revisor.						80	418	
SPRINT5	HU013	ResponsableSeccion y CoordinadorTitulacion visualizan calificaciones.	Desarrollo	Terminada	José Quichimbo	Alta	Baja	12	
	HU013	Aprobación y rechazo de revisión	Desarrollo	Terminada	José Quichimbo	Alta	Baja	8	
	HU013	Solicitar recalificación a revisor.	Desarrollo	Terminada	José Quichimbo	Alta	Media	8	
	HU013	Solicitar modificación a autor.	Desarrollo	Terminada	José Quichimbo	Alta	Media	8	
	HU013	Rechazo de propuesta.	Desarrollo	Terminada	José Quichimbo	Alta	Baja	8	
	Release: Aprobación de revisión - ResponsableSeccion y CoordinadorTitulacion.						44	462	

SPRINT6	HU019	Visualizar listado de propuestas calificadas y aprobadas.	Desarrollo	Terminada	José Quichimbo	Media	Baja	12	
	HU019	Selección de propuestas y autorización	Desarrollo	Terminada	José Quichimbo	Alta	Media	8	
	HU020	Definición de formato de consumo con desarrollador S.V.A.	Análisis - Diseño	Terminada	José Quichimbo	Alta	Alta	8	
	HU020	Envío de propuestas a servicio REST para consumo del S.V.A.	Desarrollo	Terminada	José Quichimbo	Alta	Alta	48	
	Release: Autorización de propuestas por parte del Consejo de Departamento								76
SPRINT7	HU022	Visualizar listado de propuestas calificadas y aprobadas.	Desarrollo	Pendiente	José Quichimbo	Media	Baja	4	
	HU022	Asignación de equipo de acompañamiento a T.T. (director y docentes de apoyo)	Desarrollo	Terminada	José Quichimbo	Media	Alta	12	
	Release: Asignación de equipo de acompañamiento								16
SPRINT8	HU021	Listado de propuestas aprobadas para estudiantes modalidad a distancia.	Desarrollo	Pendiente	José Quichimbo	Media	Baja	4	
	HU021	Postulación de estudiantes a Distancia	Desarrollo	Terminada	José Quichimbo	Media	Alta	12	

	HU023	Definición de formato de consumo con desarrollador S.V.A.	Desarrollo	Terminada	José Quichimbo	Alta	Alta	4	
	HU023	Consulta de estudiantes de S.V.A. para asignar estudiante a una propuesta.	Desarrollo	Terminada	José Quichimbo	Alta	Alta	24	
	HU023	Notificación a estudiante mediante correo electrónico de la asignación	Desarrollo	Pendiente	José Quichimbo	Media	Media	8	
	Release: Consulta y notificación a estudiante							52	590
SPRINT9	HU024	Consulta de estudiantes de S.V.A. para actualizar estado a completado.	Desarrollo	Pendiente	José Quichimbo	Alta	Media	48	
	HU024	Emisión de informe a T.T terminados del S.V.A.	Desarrollo	Terminada	José Quichimbo	Alta	Baja	8	
	Release: Consulta de estado de propuesta, informe y notificación a tribunal							56	646
SPRINT10	HU025	Asignación de tribunal (presidente, vocal1 y vocal2) a T.T.	Desarrollo	Terminada	José Quichimbo	Alta	Media	12	
	HU026	Notificación a estudiante y tribunal mediante correo electrónico de la asignación.	Desarrollo	Pendiente	José Quichimbo	Alta	Media	12	
	Release: Asignación y notificación a tribunal a T.T.							24	670

SPRINT11	HU027	Creación de rúbrica para trabajo completado.	Desarrollo	Terminada	José Quichimbo	Alta	Media	72	
	HU027	Calificación de trabajo terminado por parte de presidente.	Desarrollo	Terminada	José Quichimbo	Alta	Media	24	
	Release: Calificación a T.T completado							96	766
SPRINT11	HU028	Creación de página de inicio con enlaces a sus actividades según los roles del usuario.	Desarrollo	Terminada	José Quichimbo	Baja	Media	48	
	HU028	Creación de reporte en la página de inicio para Coordinador de Titulación.	Desarrollo	Pendiente	José Quichimbo	Media	Media	48	
	HU028	Módulo para filtrado de T.T para generar reportes.	Desarrollo	Pendiente	José Quichimbo	Alta	Media	72	
	Release: Página de inicio y reportes.							168	934
SPRINT12	HU017	Filtrado de esquemas de evaluación de otras titulaciones.	Desarrollo	Pendiente	José Quichimbo	Baja	Baja	48	
	HU017	Copidado de esquemas de evaluación a convocatoria.	Desarrollo	Pendiente	José Quichimbo	Baja	Media	48	
	Release: Calificación a T.T completado							96	1030

Fuente: El autor.

Elaboración: El autor.

Una vez finalizado el desarrollo de una historia esta se ubica en la columna “A verificar” y se realiza el despliegue al servicio Heroku, para la respectiva validación del Product Owner.

3.5.4. Integración con servicio REST

Las historias de usuario HU-020 y HU-023 específicamente, comprenden la vinculación con el actual sistema del vicerrectorado académico (S.V.A.) de trabajos de titulación.

La comunicación con este sistema está presente en tres procesos:

1. Envío de propuestas para trabajos de titulación autorizadas por consejo de departamento.
2. Extracción de estudiante asignado a trabajo de titulación por parte de secretaria de la titulación.
3. Extracción de estado de la propuesta cuando esta haya sido finalizada por el estudiante.

Para realizar esta comunicación fue necesario acordar mediante reunión con el desarrollador del S.V.A. un protocolo de comunicación que permitiese el envío y consumo de datos entre ambos sistemas.

Se acordó utilizar el protocolo HTTP para la comunicación, levantando un servicio REST en cada aplicación con el formato que pueda consumirse por la otra aplicación.

El formato de datos elegido es el formato JSON, y ambas aplicaciones deben levantar un servicio REST con este formato.

Para la implementación del servicio REST en Django fue necesaria la instalación de un nuevo módulo llamado “*djangoRESTframework*”, y una vez instalado se utilizaron serializadores para la consulta de propuestas. Luego el módulo se encarga de la presentación de propuestas según la URL configurada en el proyecto.

A continuación se explica los esquemas de datos utilizados para cada proceso.

3.5.4.1. Envío REST de propuestas autorizadas.

Una vez autorizadas las propuestas por el director de departamento el sistema pasa estas propuestas a estado de “Postulada”, para que se transfiera estas propuestas al S.V.A a la espera de estudiantes que postulen a cada T.T.

Éstas propuestas pasan a formar parte del servicio REST levantado por la aplicación.

La UTPL utiliza un identificador único para sus datos denominado GUID, por lo que fue necesario agregar este campo a: usuario, titulación, sección y periodo.

Los datos que debe consumir el S.V.A para la presentación de este T.T a los estudiantes son:

- ID de esta aplicación (entero).
- Estado de la propuesta (cadena de caracteres).
- ID del usuario (entero).
- GUID del usuario (cadena de caracteres).
- GUID del periodo académico (cadena de caracteres).
- GUID de la sección (cadena de caracteres).
- Tema del T.T (cadena de caracteres).
- Descripción del T.T (cadena de caracteres).
- Fecha de registro del T.T (cadena de caracteres).
- Fecha de validación del T.T (cadena de caracteres).
- Objetivo general (cadena de caracteres).
- Objetivos específicos (cadena de caracteres).

La siguiente figura 18 muestra la salida del servicio REST de la aplicación con un T.T autorizado.



Figura 18. Salida servicio REST formato JSON.

Fuente: Herramienta en producción.

Elaboración: El autor.

Con esto se determina que cada iteración ha sido desarrollada y desplegada en los tiempos acordados con el Product Owner, y cada despliegue constituye una característica que queda lista para su utilización en el servidor Heroku.

En el siguiente capítulo se describe un plan de pruebas para la validación exitosa del sistema con varios usuarios directos.

CAPÍTULO 4: PLAN DE PRUEBAS Y VALIDACIÓN.

4.1. Introducción

Es imperativo para el contexto del presente proyecto la realización de pruebas en un ambiente de producción real, puesto que la metodología con la que se ha venido desarrollando la aplicación exige un despliegue de una nueva característica funcional cada vez que se finaliza el desarrollo de un sprint.

Las pruebas unitarias se realizan para mantener la alineación con las prácticas ágiles de DevOps que constituyen parte del desarrollo del proyecto: desarrollar, probar, desplegar, validar y ajustar.

Las pruebas de rendimiento se realizan para comprobar la correspondencia con los requerimientos no funcionales identificados para la aplicación.

El principal objetivo de esta realización de pruebas es demostrar que la solución desarrollada puede realizar todas las funciones identificadas y esperadas por los usuarios, y que puede dar el soporte necesario al proceso de postulación de trabajos de titulación en tiempos aceptables. De igual manera para poder detectar de forma temprana errores e incidentes que puedan ser identificados por los usuarios directos, así como agregar nuevas características, corregir y desarrollar nuevas funcionalidades acorde a la metodología para la satisfacción de los usuarios.

Las pruebas se ejecutan con usuarios directos identificados para la aplicación.

Se requirió la presencia de representantes de la titulación de Sistemas informáticos y Computación.

Y de la titulación las personas con responsabilidades para los roles de:

- Coordinador de titulación.
- Responsable de una sección de la titulación.
- Docentes revisores.
- Docentes que ofertan una propuesta de trabajo de titulación.
- Secretaria de titulación

4.2. Ejecución de Pruebas

En esta sección se ejecutarán las pruebas en cada escenario. Cada escenario corresponde a un caso de uso, y debe mantener los flujos, requerimientos y condiciones descritos en el mismo.

Los escenarios a validar se determinan según el flujo analizado en todo el trabajo, que van desde la creación de convocatoria hasta la calificación de trabajo completado. Los escenarios son:

- Creación de convocatoria.
- Creación de esquema de evaluación para propuesta.
- Subida de propuesta
- Asignación de revisores.
- Calificación de propuesta
- Validación de revisiones.
- Autorización de propuestas.
- Envío de propuestas al sistema de vicerrectorado académico (S.V.A.).
- Asignación de equipo de acompañamiento.
- Consumo de estudiantes del S.V.A.
- Notificación a estudiante y equipo de acompañamiento.
- Consumo de propuestas terminadas.
- Informe de director de T.T.
- Asignación de tribunal.
- Calificación de presidente de tribunal.

Una vez el usuario a probado la funcionalidad del sistema correspondiente al escenario, deberá responder una encuesta que contiene una validación cuantitativa para cada etapa del proceso (Muy satisfactorio, satisfactorio, medianamente satisfactorio, insatisfactorio).

Esta validación evalúa 4 parámetros:

- Soporte al proceso de negocio
- Usabilidad de la aplicación
- Tiempos de respuesta
- Confiabilidad

4.2.1. Creación de convocatoria.

La creación de convocatoria pertenece a la historia de usuario HU011 y al caso de uso CU001.

Es iniciada por el Coordinador de titulación y comprende la creación de una convocatoria con actividades y sus plazos. La siguiente figura 19 muestra el formulario para la creación de una convocatoria.

Nueva Convocatoria

Título

Titulación

Activa

Descripción

Responsable

Persona a quien remitirse por dudas

Propuesta:
Elaboración y subida de una propuesta.
 Hoy | Hoy

Revisión:
Revisión por pares de los miembros de sección.
 Hoy | Hoy

Validación:
Validación por parte de Responsable Sección y Coordinador Titulación.
 Hoy | Hoy

Autorización:
Autorización por parte del consejo de departamento.
 Hoy | Hoy

Postulación:
Postulación de estudiantes a las propuestas subidas.
 Hoy | Hoy

Figura 19. Formulario creación de convocatoria.
Fuente: Herramienta en producción.
Elaboración: El autor.

Esta convocatoria da paso a la creación de esquemas de evaluación.

La validación de este escenario la realiza el Coordinador de Titulación de Sistemas Informáticos y Computación.

Una vez realizada la validación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.2. Creación de esquema de evaluación para propuesta.

El esquema de evaluación de propuestas para T.T pertenece a la historia de usuario HU017 y al caso de uso CU002.

Es realizada por el Coordinador de Titulación y comprende la creación de un esquema de evaluación con una calificación máxima, mínima e ítems a evaluar. La siguiente figura 20 muestra el formulario para la creación de un esquema de evaluación de propuestas.

Nuevo Esquema Evaluación Propuesta:

[Regresar](#)

Convocatoria

Nombre

Comentario

Calificación máxima

 Calificación sobre la que se calificará el esquema de evaluación (rúbrica).

Calificación mínima

 Calificación mínima que debe tener una propuesta para ser aprobada.

Ítems:
 Marque los campos predefinidos de la propuesta base que desee evaluar:

Ítem	Deficiente (1punto)	Regular (3puntos)	Bueno (4puntos)	Excelente (5puntos)	Peso %
<input checked="" type="checkbox"/> TEMA	<input type="text"/> Criterio de evaluación % de la nota máxin	<input type="text"/> Criterio de evaluación % de la nota máxin	<input type="text"/> Criterio de evaluación % de la nota máxin	<input type="text"/> Criterio de evaluación % de la nota máxin	<input type="text"/> Peso ítem
<input type="checkbox"/> PLAZO					
<input type="checkbox"/> COMPONENTE TECNICO					
<input type="checkbox"/> COMPONENTE INVESTIGACIÓN					
<input type="checkbox"/> COMPONENTE INNOVACIÓN					
<input type="checkbox"/> INTEGRANTES					
<input type="checkbox"/> DESCRIPCIÓN DEL TRABAJO					
<input type="checkbox"/> DESCRIPCIÓN DEL PROBLEMA					
<input type="checkbox"/> JUSTIFICACIÓN					
<input type="checkbox"/> OBJETIVOS					

Figura 20. Formulario creación de esquema de evaluación.

Fuente: Herramienta en producción.

Elaboración: El autor.

La validación de este escenario la realiza el Coordinador de Titulación de Sistemas Informáticos y Computación.

Una vez realizada la validación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.3. Subida de propuesta de trabajo de titulación.

El esquema de evaluación de propuestas para T.T pertenece a la historia de usuario HU003 y al caso de uso CU004.

Es realizada por un Docente y comprende la elaboración y subida de una propuesta para T.T con los ítems base de la plantilla más los ítems definidos por la titulación en la creación del esquema de evaluación. El formulario para subida de propuesta se muestra en la figura 21.

Nueva Propuesta

Periodo	<input type="text"/>
Tema	<input type="text"/>
Plazo	<input type="text"/>
Plazo en meses	<input type="text"/>
Número de estudiantes	<input type="text"/>
Perfil estudiante	<input type="text"/>
Línea Estratégica	<input type="text"/>
Sección	<input type="text"/>
Línea Investigación	<input type="text"/>
Descripción	<input type="text"/>
Justificación	<input type="text"/>

Propuesta deriva

Propuesta de la que deriva la presente propuesta.

Objetivo general

Bibliografía

Objetivos Específicos

+ Objetivo

Objetivo

Campos de Rubrica:

Subir Propuesta

Figura 21. Formulario subida de propuesta.
Fuente: Herramienta en producción.
Elaboración: El autor.

La propuesta subida será dirigida al Responsable de Sección de la sección correspondiente.

La validación de este escenario la realizan docentes de Titulación de Sistemas Informáticos y Computación.

Una vez realizada la validación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.4. Asignación de revisores.

La asignación de revisores a una propuesta para T.T pertenece a la historia de usuario HU008 y al caso de uso CU005.

Es realizada por un Responsable de Sección y comprende la selección de uno o más docentes revisores de la titulación para que califiquen la propuesta. El formulario para asignación de revisores se muestra en la figura 22.

Asignar revision: "Implementación de una aplicación para gestionar los procesos de elaboración, valoración postulación y asignación de trabajos de titulación par la carrera de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática"

Regresar

Revisores

Daniel Alejandro Guaman Coronel - 2 revisiones asignadas.
 Nelson Oswaldo Piedra Pullaguari - 2 revisiones asignadas.

Mantenga presionada la tecla Ctrl para seleccionar varios revisores

Asignar

Figura 22. Formulario asignación de revisores.

Fuente: Herramienta en producción.

Elaboración: El autor.

Los docentes seleccionados recibirán una nueva propuesta a calificar.

La validación de este escenario la realiza el responsable de una sección de la Titulación de Sistemas Informáticos y Computación.

Una vez realizada la asignación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.5. Calificación de propuesta de trabajo de titulación.

La calificación de una propuesta para T.T pertenece a la historia de usuario HU005 y al caso de uso CU006.

Es realizada por un Docente Revisor de una Sección y comprende la calificación a una propuesta mediante el esquema de evaluación definido por el Coordinador de Titulación. El formulario para calificación de propuesta se muestra en la figura 23.

Calificacion con Rubrica:

Calificación: 2.9

Item	Calificación	Descripción	Valor	Peso %	Calif.	Comentario
Objetivos	Excelente	Objetivos claros y medibles, muy relacionados con el tema y problema. Se vislumbra posibilidades de solución/implementación.	5	10	0.5	
Tema	Bueno	El tema es claro y específico pero tiene poca relación con las competencias del perfil profesional.	4	40	1.6	
Plazo	Regular	El plazo planteado no alcanza el tiempo mínimo de duración del proyecto y tampoco se evidencia el uso de al menos 400 horas.	2	40	0.8	
Bibliografía			20		

Terminar

Aprobada ● Requiere Modificaciones ● Rechazada ●

Figura 23. Formulario calificación de propuesta.

Fuente: Herramienta en producción.

Elaboración: El autor.

La calificación se guarda para que puedan visualizarla el Responsable de Sección y el Coordinador de Titulación.

La validación de este escenario la realizan los docentes revisores de la Titulación de Sistemas Informáticos y Computación.

Una vez realizada la calificación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.6. Validación de revisiones.

La calificación de una propuesta para T.T pertenece a la historia de usuario HU013 y a los casos de uso CU007 y CU008.

Es realizada por el Responsable de Sección y el Coordinador de Titulación, y comprende la visualización de las calificaciones emitidas por los revisores para poder aprobar, rechazar, solicitar recalificación o solicitar modificaciones a la propuesta. La interfaz para visualización de calificaciones se muestra en la figura 24.

Calificación: "Minería de datos - Tecnologías Avanzadas de la Web. Por mpabad"

Calificaciones de Revisores

[Regresar](#)

Id	Revisor	Fecha	Calif.	Estado	Items			
					Nombre	Calificacion	Valor	Comentario
52	nopiedra	3 de Febrero de 2017	3,40	Requiere Modificacion	Tema - 40%	Bueno	4	
					Plazo - 40%	Bueno	4	
					Centros de salud aplicables - 20%	Deficiente	1	Agregar mas centros de salud.

Calificación mínima: 4

Calificación promedio: 3,40

No alcanza la calificación mínima.
No se puede aprobar.

Recalificar

Modificar Propuesta

Rechazar

Figura 24. Interfaz visualización de calificaciones

Fuente: Herramienta en producción.

Elaboración: El autor.

El Responsable de Sección y el Coordinador de Titulación pueden emitir su criterio final sobre la propuesta. Es necesario que el Responsable de Sección haya validado la revisión primero.

La validación de este escenario la realizan el Responsable de Sección y el Coordinador de la Titulación de Sistemas Informáticos y Computación.

Una vez realizada la calificación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.7. Autorización y envío de propuestas a S.V.A.

La calificación de una propuesta para T.T pertenece a las historias de usuario HU019 y HU020, y al caso de uso CU009.

Es realizada por el Director de Departamento, y comprende la selección de propuestas que ya han sido aprobadas para registrarlas como formato consumible para el S.V.A y que este las pueda poner a disposición de los estudiantes. La interfaz para visualización de propuestas aprobadas se muestra en la figura 25.

Trabajos Titulación Aprobados

Convocatoria: Convocatoria 1 Sistemas Presencial

Fecha: Fecha aprobación

Hoy

Autorizar y Enviar a S.V.A.

Subido por	Tema	Revisado por	Promedio Calif.	Equipo
<input type="checkbox"/> Marco Patricio Abad	Data Mining	nopiedra, djaramillo,	5,00	Marco Patricio Abad Nelson Oswaldo Piedra Pullaguari Maria Soledad Segarra Morales Ramiro Leonardo Ramirez Coronel
<input type="checkbox"/> Ramiro Leonardo Ramirez Coronel	Cloud Computing	daguaman,	4,45	Ramiro Leonardo Ramirez Coronel Daniel Alejandro Guaman Coronel

Figura 25. Interfaz visualización de calificaciones
Fuente: Herramienta en producción.
Elaboración: El autor.

El Director de Departamento marca las propuestas y selecciona “Autorizar y Enviar”.

Sólo se muestran las propuestas que han tenido la aprobación del Responsable de Sección y Coordinador de Titulación.

El mensaje de confirmación se muestra en la siguiente figura 26.

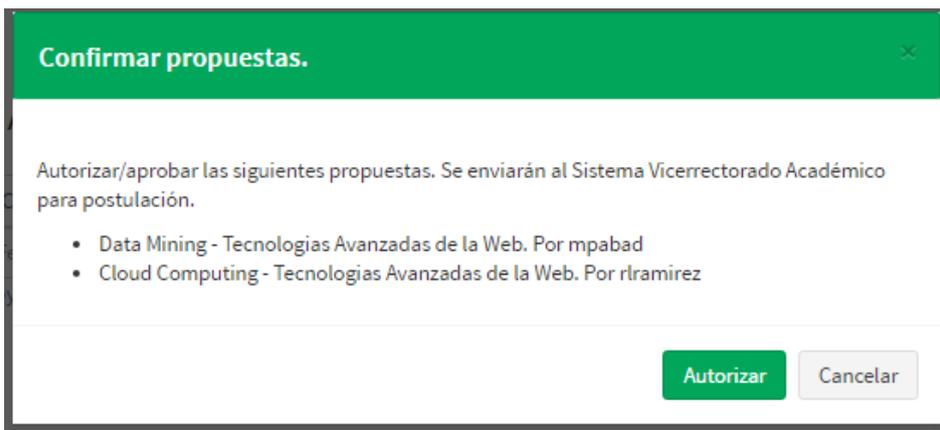


Figura 26. Confirmación de autorización propuestas.

Fuente: Herramienta en producción.

Elaboración: El autor.

Estas propuestas pasan a formar parte del servicio REST de la aplicación para que sean consumidas por el S.V.A.

La validación de este escenario la realiza el Director de un Departamento de la Titulación de Sistemas Informáticos y Computación.

Una vez realizada la calificación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

4.2.8. Consumo de estudiante para propuesta de S.V.A.

El consumo de estudiantes para propuestas pertenece a la historia de usuario HU023 y al caso de uso CU011.

Es realizada por la Secretaria de Titulación, y comprende la visualización del listado propuestas de T.T autorizadas y postuladas para poder consultar el estudiante en los trabajos que aún no lo tengan. La interfaz para visualización de trabajos se muestra en la figura 27.

Proyectos Autorizados

Id	Tema	Estado	Subida por	Director	Equipo	Estudiante	Notificar
23	Cloud Computing	Autorizada	rramirez	rramirez	daguaman,	Extraer de S.V.A	Notificar

Figura 27. Interfaz visualización de propuestas autorizadas y postuladas.

Fuente: Herramienta en producción.

Elaboración: El autor.

La secretaria puede extraer del S.V.A el estudiante en caso de que ya estuviere asignado, caso contrario el sistema muestra un mensaje de confirmación o de error. La secretaria puede notificar al estudiante y al equipo de acompañamiento la asignación al T.T.

La validación de este escenario la realiza la Secretaria de la Titulación de Sistemas Informáticos y Computación.

Una vez realizada la calificación debe responder las preguntas correspondientes a este escenario la misma que consta en el anexo 15.

CAPÍTULO 5: RESULTADOS

5.1. Introducción

Para la validación del Sistema de Gestión para Trabajos de Titulación se realizó una capacitación a un grupo de docentes de la titulación de Sistemas Informáticos y Computación, los mismos que empezaron a trabajar con la herramienta subiendo las propuestas para trabajo de titulación y elaborando la calificación respectiva.

Se realizó pruebas para cada escenario descrito en la sección 4.2, seguido de esto se solicitó llenar la encuesta que contiene todos los escenarios descrita en el anexo 15. En la encuesta se solicita inicialmente al usuario seleccionar el rol con el que desempeña sus funciones en el sistema.

A continuación se describe la validación del sistema por parte del grupo de docentes de la titulación de Sistemas Informáticos y Computación, específicamente de la sección de inteligencia artificial, los mismos que han utilizado el sistema haciendo uso de los roles como responsable de sección, docentes revisores y docentes proponentes.

5.2. Resultados de la encuesta

De cada escenario se evaluó 4 aspectos:

- Soporte al proceso de negocio
 - Muy satisfactorio
 - Satisfactorio
 - Poco satisfactorio
 - Insatisfactorio
- Complejidad
 - Muy sencillo
 - Sencillo
 - Complicado
 - Muy complicado
- Velocidad
 - Muy rápido
 - Rápido
 - Lento
 - Muy lento
- Confiabilidad

- Datos correctos (sí)
- Datos incorrectos (no)

La siguiente figura 28 muestra el porcentaje de usuarios que respondieron la encuesta según su rol, con un total de 12 respuestas: 11 respuestas completas y 1 respuesta parcial donde el usuario había respondido únicamente respondió las preguntas del primer escenario.

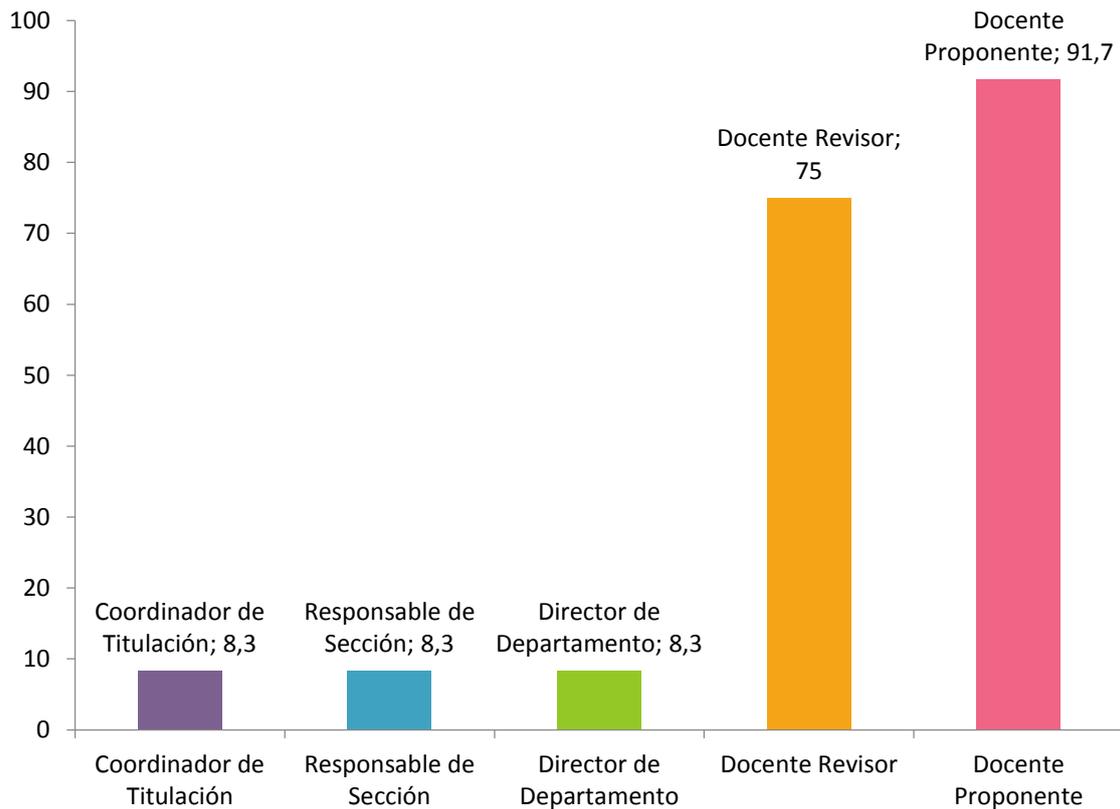


Figura 28. Porcentaje personas encuestadas.
 Fuente: El autor.
 Elaboración: El autor.

De cada escenario se mostraron 5 preguntas de opción múltiple obligatorias con respecto a los 4 aspectos y un campo opcional para que el usuario agregue comentarios sobre el escenario.

A continuación se detalla, representa gráficamente y analiza las respuestas recolectadas para cada escenario.

5.2.1. Oferta de convocatorias

A continuación se muestran los resultados correspondientes a la descripción del plan de pruebas detallado en la sección 4.2.1.

Soporte al proceso de negocio

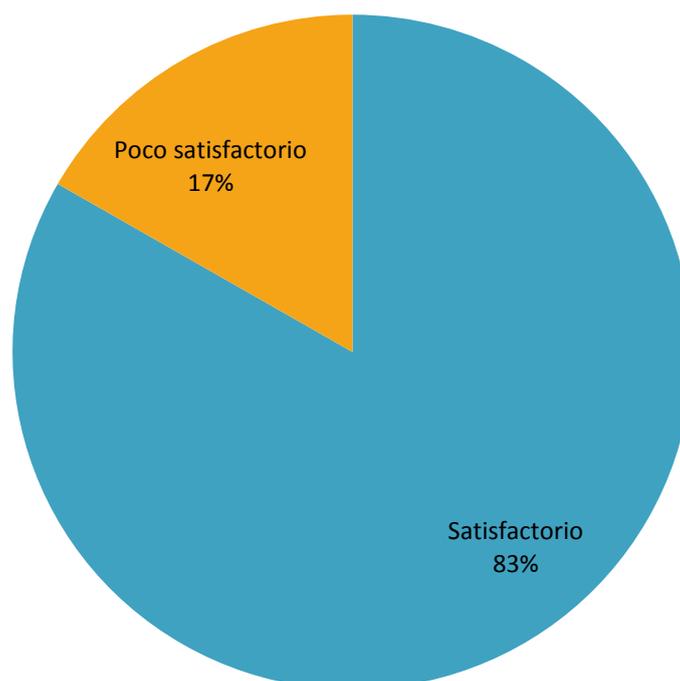


Figura 29. Gráfico circular – Soporte al proceso de negocio “Oferta de convocatoria”.

Fuente: El autor.

Elaboración: El autor.

Tabla 23. Valores encuesta - Soporte al proceso de negocio “Oferta de convocatoria”.

Valor	Porcentaje	Respuestas
Satisfactorio	83.3%	10
Poco satisfactorio	16.7%	2
	Total	12

Fuente: El autor.

Elaboración: El autor.

Complejidad

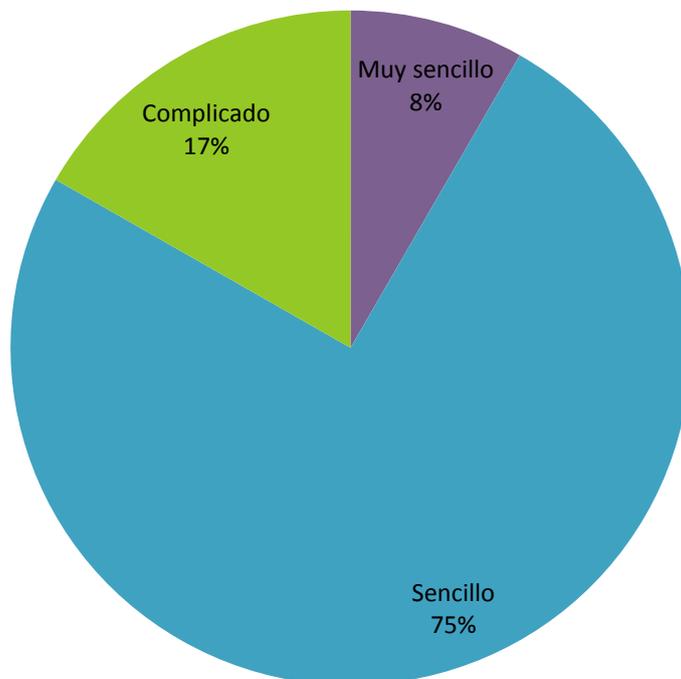


Figura 30. Gráfico circular – Complejidad “Oferta de convocatoria”.

Fuente: El autor.

Elaboración: El autor.

Tabla 24. Valores encuesta – Complejidad “Oferta de convocatoria”.

Valor	Porcentaje	Respuestas
Muy sencillo	8.3%	1
Sencillo	75.0%	9
Complicado	16.7%	2
	Total	12

Fuente: El autor.

Elaboración: El autor.

Velocidad

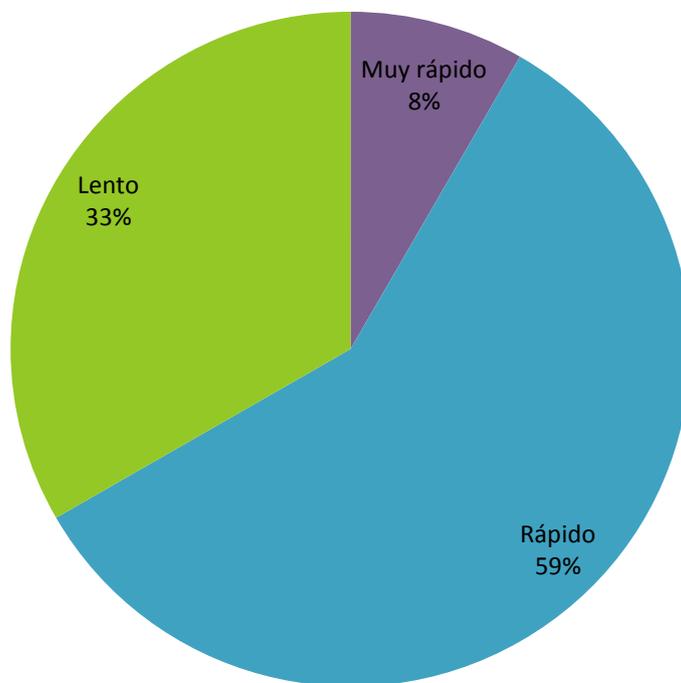


Figura 31. Gráfico circular – Velocidad “Oferta de convocatoria”.

Fuente: El autor.

Elaboración: El autor.

Tabla 25. Valores encuesta – Velocidad “Oferta de convocatoria”.

Valor	Porcentaje	Respuestas
Muy rápido	8.3%	1
Rápido	58.3%	7
Lento	33.3%	4
	Total	12

Fuente: El autor.

Elaboración: El autor.

Confiabilidad

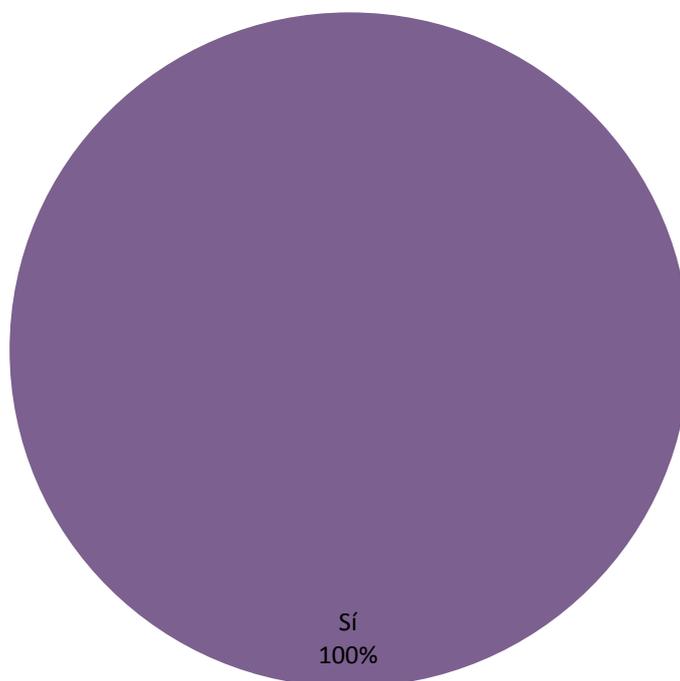


Figura 32. Gráfico circular – Confiabilidad “Oferta de convocatoria”.

Fuente: El autor.

Elaboración: El autor.

Tabla 26. Valores encuesta – Confiabilidad “Oferta de convocatoria”.

Valor	Porcentaje	Respuestas
Sí	100.0%	12
	Total	12

Fuente: El autor.

Elaboración: El autor.

Análisis

El primer escenario evaluado por los docentes suministra los siguientes resultados mayoritarios con respecto a la oferta de convocatorias según los cuatro aspectos evaluados:

Tabla 27. Análisis encuesta - Escenario “Oferta de convocatoria”.

Aspecto evaluado	Resultado	Porcentaje
Soporte al proceso de negocio	Satisfactorio	83.3%
Complejidad	Sencillo	75%
Velocidad	Rápido	59%
Confiabilidad	Confiable	100%

Fuente: El autor.

Elaboración: El autor.

Se puede observar que para los docentes la oferta de convocatorias soporta al proceso de negocio de la titulación, es sencillo de crear, se oferta rápidamente y los datos guardados son los correctos. Por lo tanto este escenario en la herramienta es aceptado por la titulación.

5.2.2. Creación de esquema de evaluación (rúbrica)

A continuación se muestran los resultados correspondientes a la descripción del plan de pruebas detallado en la sección 4.2.2.

Soporte al proceso de negocio

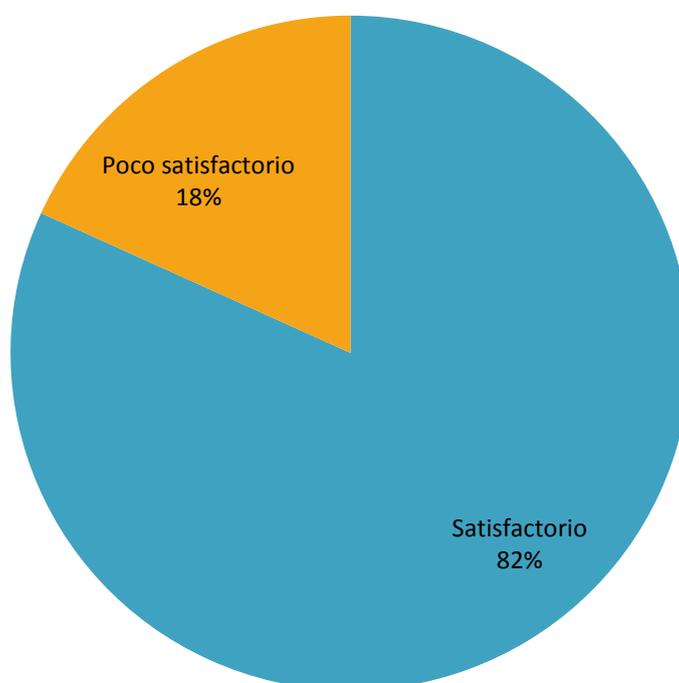


Figura 33. Gráfico circular – Soporte al proceso de negocio “Esquemas de evaluación”.

Fuente: El autor.

Elaboración: El autor.

Tabla 28. Valores encuesta - Soporte al proceso de negocio “Esquemas de evaluación”.

Valor	Porcentaje	Respuestas
Satisfactorio	81.8%	9
Poco satisfactorio	18.2%	2
	Total	11

Fuente: El autor.

Elaboración: El autor.

Complejidad

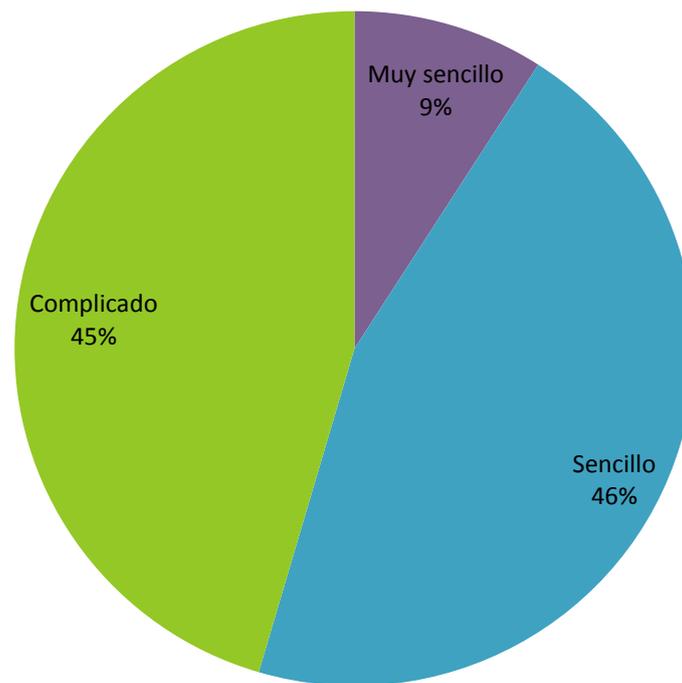


Figura 34. Gráfico circular – Complejidad “Esquemas de evaluación”.

Fuente: El autor.

Elaboración: El autor.

Tabla 29. Valores encuesta – Complejidad “Esquemas de evaluación”.

Valor	Porcentaje	Respuestas
Muy sencillo	9.1%	1
Sencillo	45.5%	5
Complicado	45.5%	5
	Total	11

Fuente: El autor.

Elaboración: El autor.

Velocidad

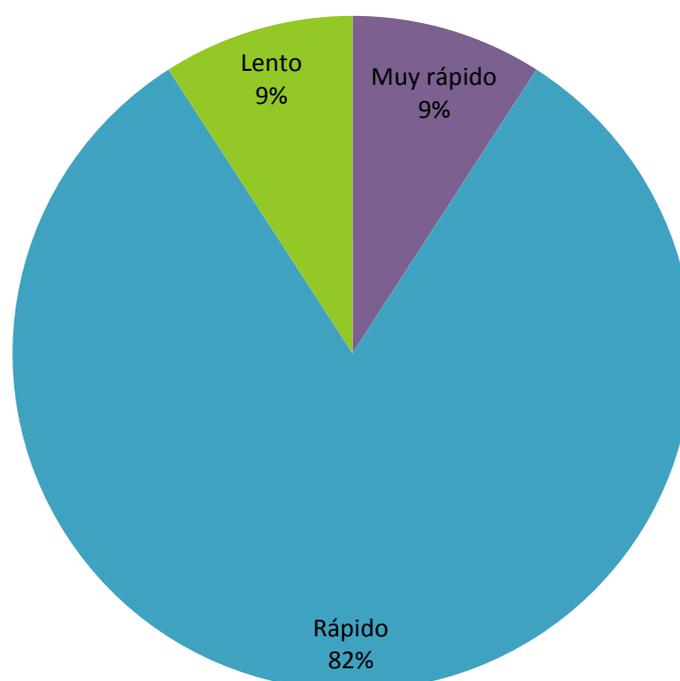


Figura 35. Gráfico circular – Velocidad “Esquemas de evaluación”.

Fuente: El autor.

Elaboración: El autor.

Tabla 30. Valores encuesta – Velocidad “Esquemas de evaluación”.

Valor	Porcentaje	Respuestas
Muy rápido	9.1%	1
Rápido	81.8%	9
Lento	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

Confiabilidad

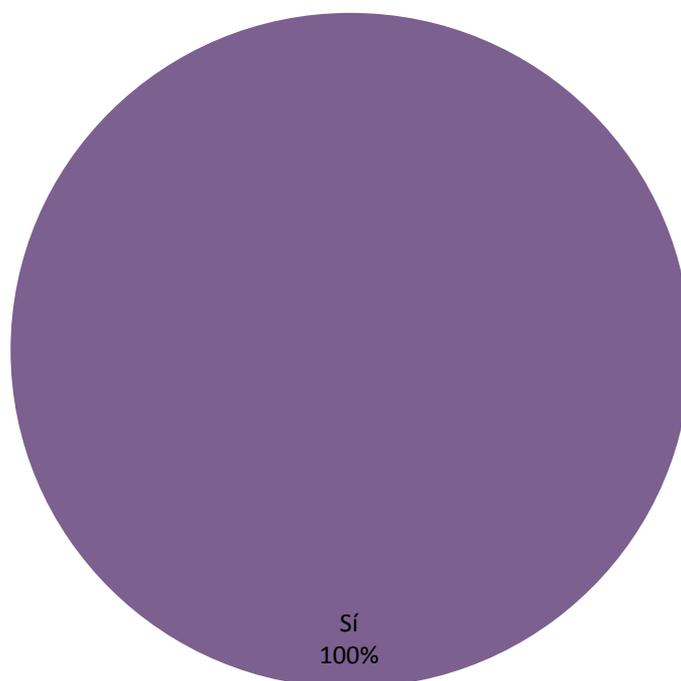


Figura 36. Gráfico circular – Confiabilidad “Esquemas de evaluación”.

Fuente: El autor.

Elaboración: El autor.

Tabla 31. Valores encuesta – Confiabilidad “Esquemas de evaluación”.

Valor	Porcentaje	Respuestas
Sí	100.0%	11
	Total	11

Fuente: El autor.
 Elaboración: El autor.

Análisis

El segundo escenario evaluado por los docentes suministra los siguientes resultados mayoritarios con respecto a la creación de esquemas de evaluación según los cuatro aspectos evaluados:

Tabla 32. Análisis Encuesta - Escenario “Esquemas de evaluación”.

Aspecto evaluado	Resultado	Porcentaje
Soporte al proceso de negocio	Satisfactorio	82%
Complejidad	Sencillo	46%
Velocidad	Rápido	82%
Confiabilidad	Confiable	100%

Fuente: El autor.
 Elaboración: El autor.

Se puede observar que para los docentes la creación de esquemas de evaluación soporta al proceso de negocio de la titulación, es sencillo de crear a pesar que un porcentaje lo considera complicado, se crea rápidamente y los datos guardados son los correctos. Por lo tanto este escenario en la herramienta es aceptado por la titulación.

5.2.3. Subida de propuestas

A continuación se muestran los resultados correspondientes a la descripción del plan de pruebas detallado en la sección 4.2.3.

Soporte al proceso de negocio

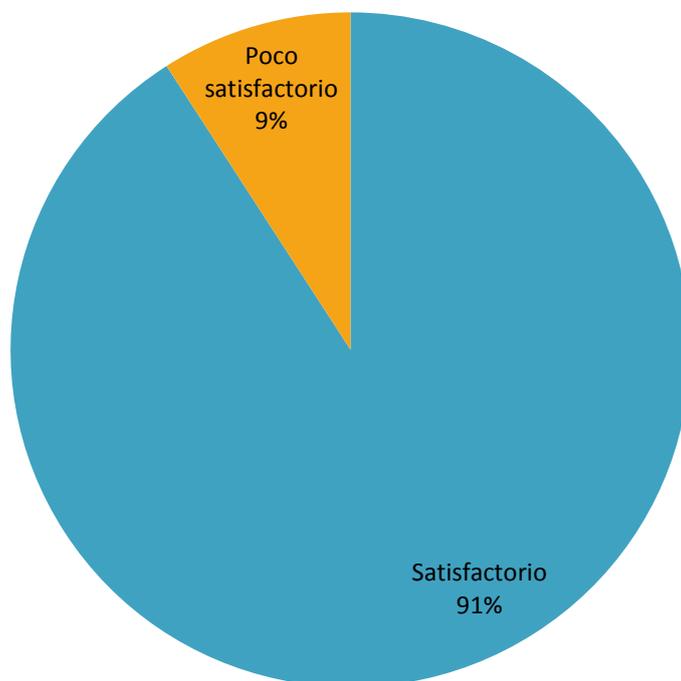


Figura 37. Gráfico circular – Soporte al proceso de negocio “Subida de propuestas”.
Fuente: El autor.
Elaboración: El autor.

Tabla 33. Valores encuesta - Soporte al proceso de negocio “Subida de propuestas”.

Valor	Porcentaje	Respuestas
Satisfactorio	90.9%	10
Poco satisfactorio	9.1%	1
	Total	11

Fuente: El autor.
Elaboración: El autor.

Complejidad

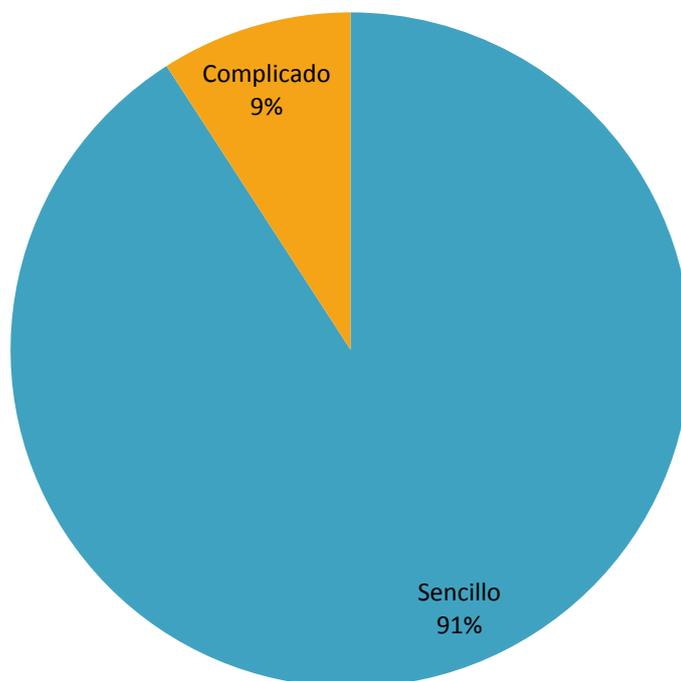


Figura 38. Gráfico circular – Complejidad “Subida de propuestas”.

Fuente: El autor.

Elaboración: El autor.

Tabla 34. Valores encuesta – Complejidad “Subida de propuestas”.

Valor	Porcentaje	Respuestas
Sencillo	90.9%	10
Complicado	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

Velocidad

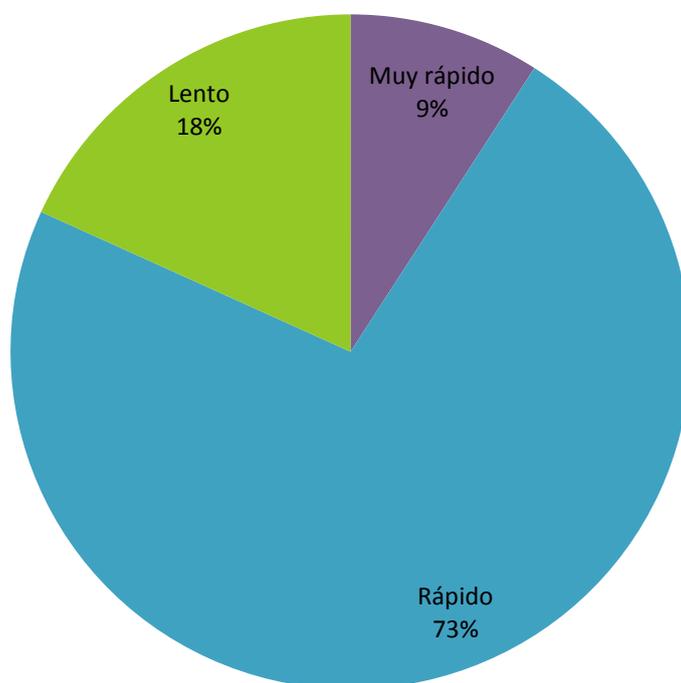


Figura 39. Gráfico circular – Velocidad “Subida de propuestas”.

Fuente: El autor.

Elaboración: El autor.

Tabla 35. Valores encuesta – Velocidad “Subida de propuestas”.

Valor	Porcentaje	Respuestas
Muy rápido	9.1%	1
Rápido	72.7%	8
Lento	18.2%	2
	Total	11

Fuente: El autor.

Elaboración: El autor.

Confiabilidad

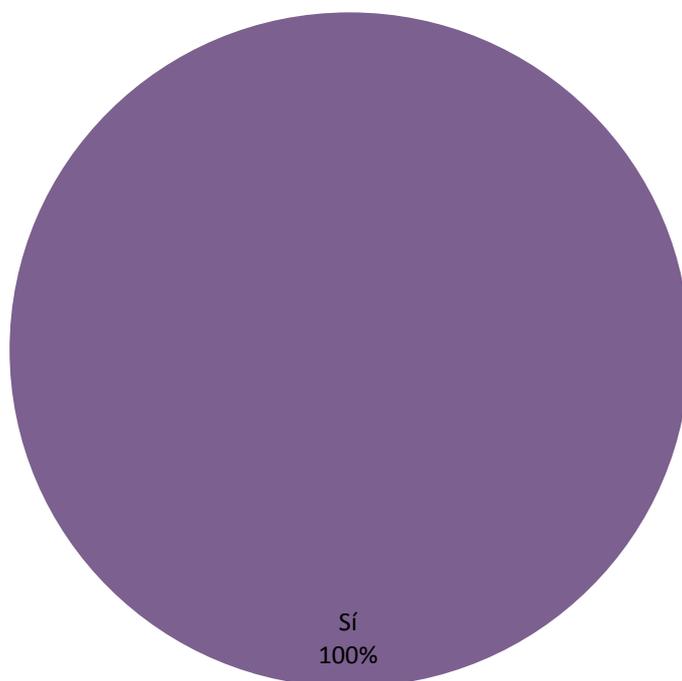


Figura 40. Gráfico circular – Confiabilidad “Subida de propuestas”.

Fuente: El autor.

Elaboración: El autor.

Tabla 36. Valores encuesta – Confiabilidad “Subida de propuestas”.

Valor	Porcentaje	Respuestas
Sí	100.0%	11
	Total	11

Fuente: El autor.

Elaboración: El autor.

Análisis

El tercer escenario evaluado por los docentes suministra los siguientes resultados mayoritarios con respecto a la subida de propuestas según los cuatro aspectos evaluados:

Tabla 37. Análisis Encuesta - Escenario “Subida de propuestas”.

Aspecto evaluado	Resultado	Porcentaje
Soporte al proceso de negocio	Satisfactorio	91%
Complejidad	Sencillo	91%
Velocidad	Rápido	73%
Confiabilidad	Confiable	100%

Fuente: El autor.

Elaboración: El autor.

Se puede observar que para los docentes la subida de propuestas soporta al proceso de negocio de la titulación, es sencillo de crear, se sube rápidamente y los datos guardados son los correctos. Por lo tanto este escenario en la herramienta es aceptado por la titulación.

5.2.4. Calificación de propuesta

A continuación se muestran los resultados correspondientes a la descripción del plan de pruebas detallado en la sección 4.2.5.

Soporte al proceso de negocio

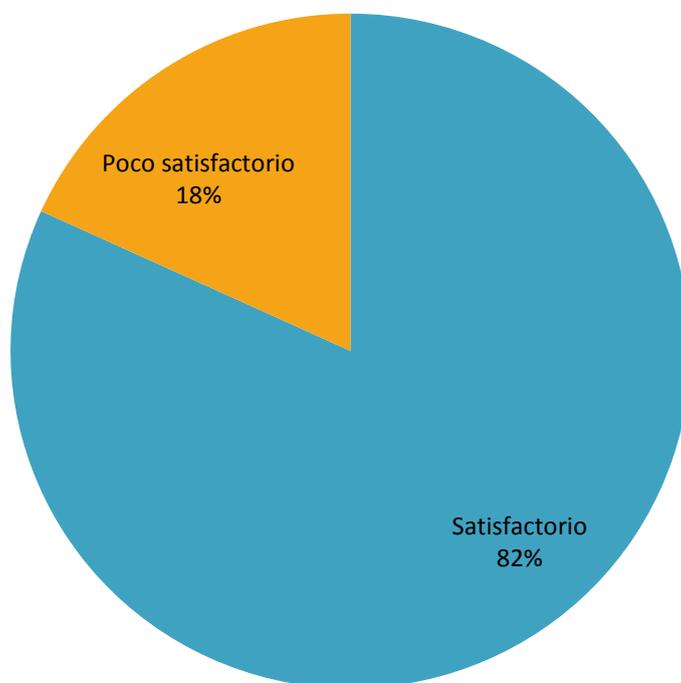


Figura 41. Gráfico circular – Soporte al proceso de negocio “Calificación de propuesta”.

Fuente: El autor.

Elaboración: El autor.

Tabla 38. Valores encuesta - Soporte al proceso de negocio "Calificación de propuesta".

Valor	Porcentaje	Respuestas
Satisfactorio	81.8%	9
Poco satisfactorio	18.2%	2
	Total	11

Fuente: El autor.

Elaboración: El autor.

Complejidad

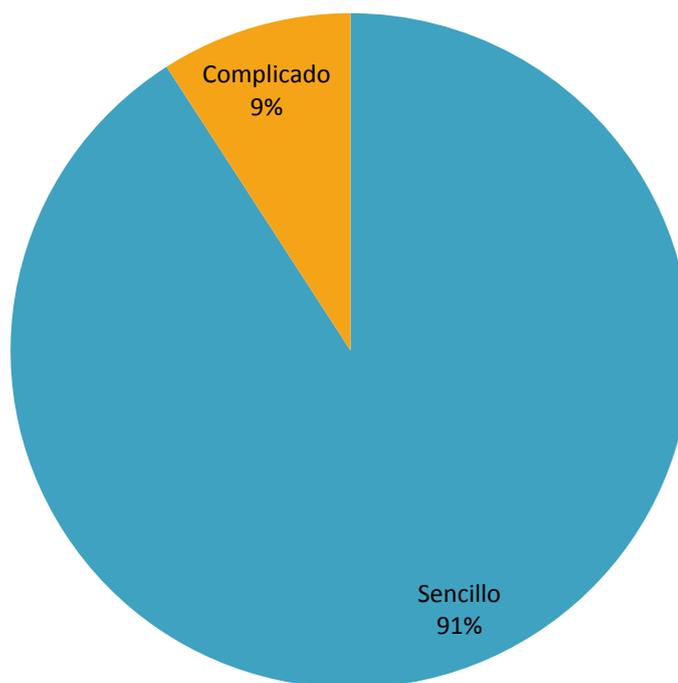


Figura 42. Gráfico circular – Complejidad "Calificación de propuesta".

Fuente: El autor.

Elaboración: El autor.

Tabla 39. Valores encuesta – Complejidad “Calificación de propuesta”.

Valor	Porcentaje	Respuestas
Sencillo	90.9%	10
Complicado	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

Velocidad

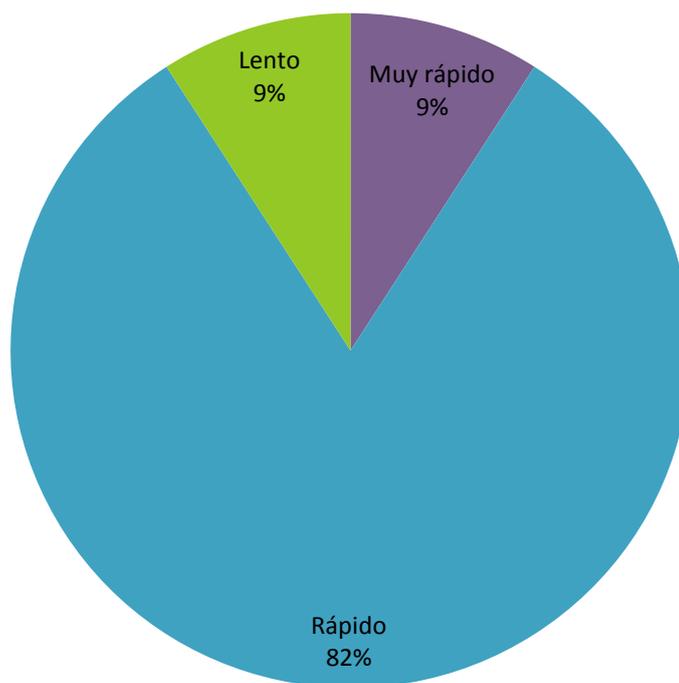


Figura 43. Gráfico circular – Velocidad “Calificación de propuesta”.

Fuente: El autor.

Elaboración: El autor.

Tabla 40. Valores encuesta – Velocidad “Calificación de propuesta”.

Valor	Porcentaje	Respuestas
Muy rápido	9.1%	1
Rápido	81.8%	9
Lento	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

Confiabilidad

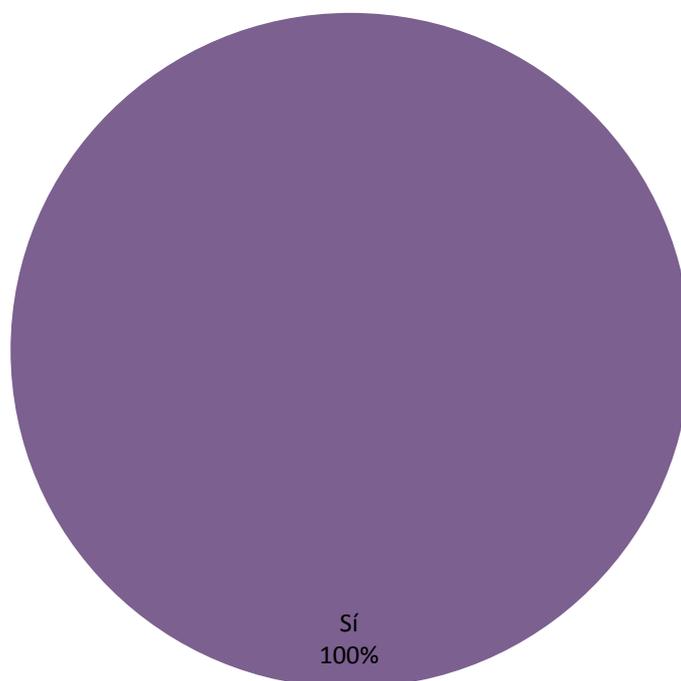


Figura 44. Gráfico circular – Confiabilidad “Calificación de propuesta”.

Fuente: El autor.

Elaboración: El autor.

Tabla 41. Valores encuesta – Confiabilidad “Calificación de propuesta”.

Valor	Porcentaje	Respuestas
Sí	100.0%	11
	Total	11

Fuente: El autor.
Elaboración: El autor.

Análisis

El cuarto escenario evaluado por los docentes suministra los siguientes resultados mayoritarios con respecto a la calificación de propuestas según los cuatro aspectos evaluados:

Tabla 42. Análisis Encuesta - Escenario “Calificación de propuesta”.

Aspecto evaluado	Resultado	Porcentaje
Soporte al proceso de negocio	Satisfactorio	82%
Complejidad	Sencillo	91%
Velocidad	Rápido	82%
Confiabilidad	Confiable	100%

Fuente: El autor.
Elaboración: El autor.

Se puede observar que para los docentes la calificación de propuesta soporta al proceso de negocio de la titulación, es sencillo de calificar, se guarda rápidamente y los datos guardados son los correctos. Por lo tanto este escenario en la herramienta es aceptado por la titulación.

5.2.5. Validación de revisiones

A continuación se muestran los resultados correspondientes a la descripción del plan de pruebas detallado en la sección 4.2.6.

Soporte al proceso de negocio

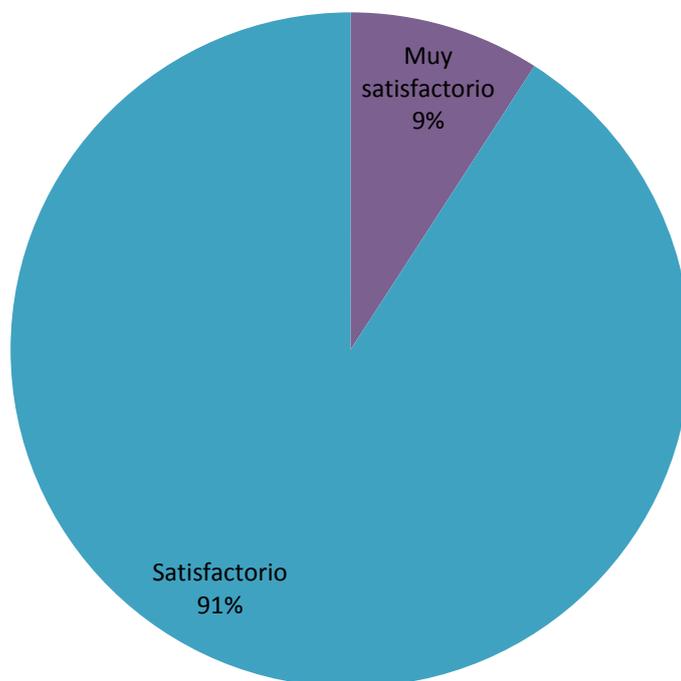


Figura 45. Gráfico circular – Soporte al proceso de negocio “Validación de revisiones”.
Fuente: El autor.
Elaboración: El autor.

Tabla 43. Valores encuesta - Soporte al proceso de negocio “Validación de revisiones”.

Valor	Porcentaje	Respuestas
Muy satisfactorio	9.1%	1
Satisfactorio	90.9%	10
	Total	11

Fuente: El autor.
Elaboración: El autor.

Complejidad

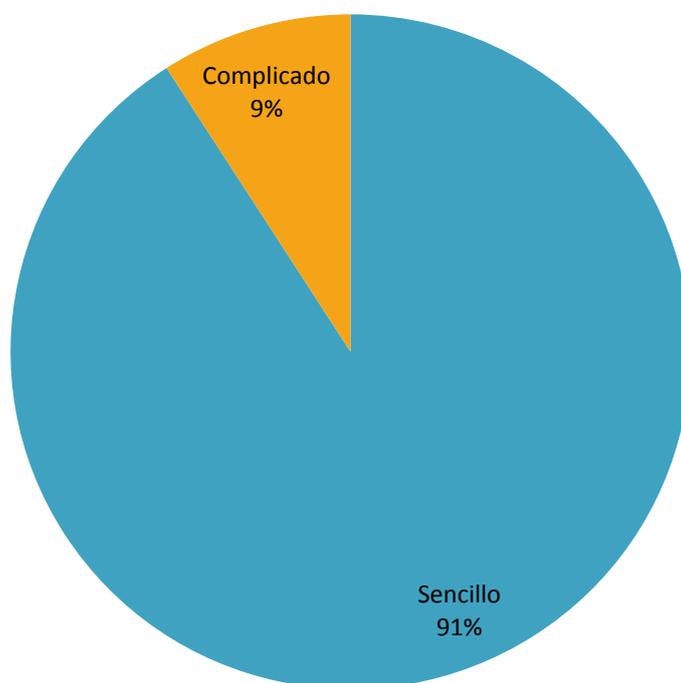


Figura 46. Gráfico circular – Complejidad “Validación de revisiones”.

Fuente: El autor.

Elaboración: El autor.

Tabla 44. Valores encuesta – Complejidad “Validación de revisiones”.

Valor	Porcentaje	Respuestas
Sencillo	90.9%	10
Complicado	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

Velocidad

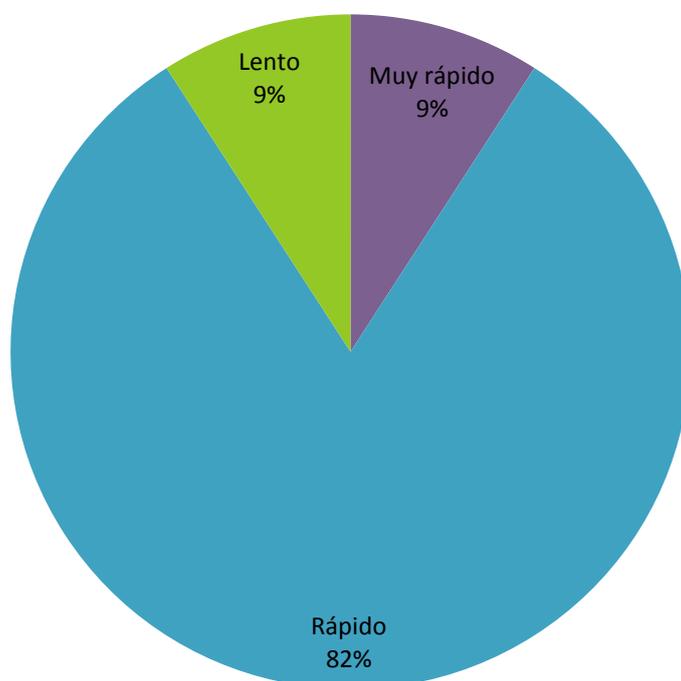


Figura 47. Gráfico circular – Velocidad “Validación de revisiones”.

Fuente: El autor.

Elaboración: El autor.

Tabla 45. Valores encuesta – Velocidad “Validación de revisiones”.

Valor	Porcentaje	Respuestas
Muy rápido	9.1%	1
Rápido	81.8%	9
Lento	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

Confiabilidad

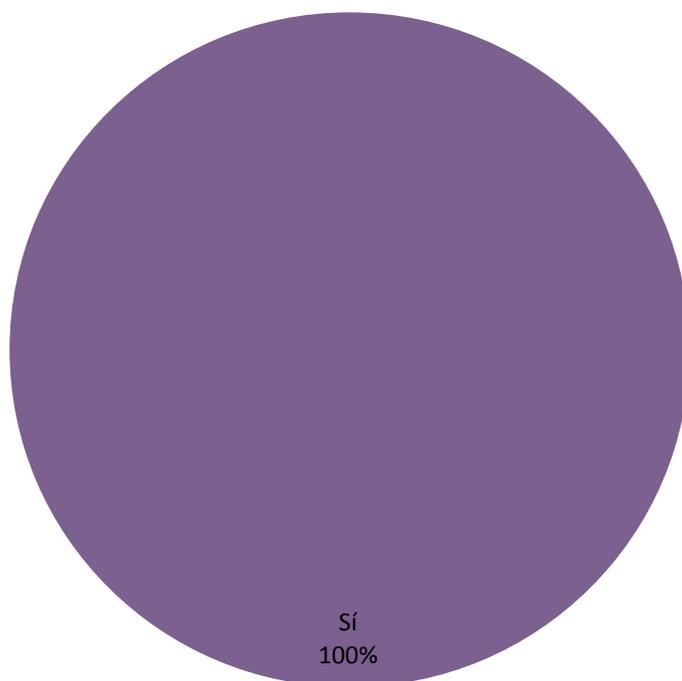


Figura 48. Gráfico circular – Confiabilidad “Validación de revisiones”.

Fuente: El autor.

Elaboración: El autor.

Tabla 46. Valores encuesta – Confiabilidad “Validación de revisiones”.

Valor	Porcentaje	Respuestas
Sí	100.0%	11
	Total	11

Fuente: El autor.

Elaboración: El autor.

Análisis

El quinto escenario evaluado por los docentes suministra los siguientes resultados mayoritarios con respecto a la validación de revisiones según los cuatro aspectos evaluados:

Tabla 47. Análisis Encuesta - Escenario “Validación de revisiones”.

Aspecto evaluado	Resultado	Porcentaje
Soporte al proceso de negocio	Satisfactorio	91%
Complejidad	Sencillo	91%
Velocidad	Rápido	82%
Confiabilidad	Confiable	100%

Fuente: El autor.
Elaboración: El autor.

Se puede observar que para los docentes la validación de revisiones soporta al proceso de negocio de la titulación, es sencillo de validar, se hace rápidamente y los datos guardados son los correctos. Por lo tanto este escenario en la herramienta es aceptado por la titulación.

5.2.6. Autorización de propuestas

A continuación se muestran los resultados correspondientes a la descripción del plan de pruebas detallado en la sección 4.2.7.

Soporte al proceso de negocio

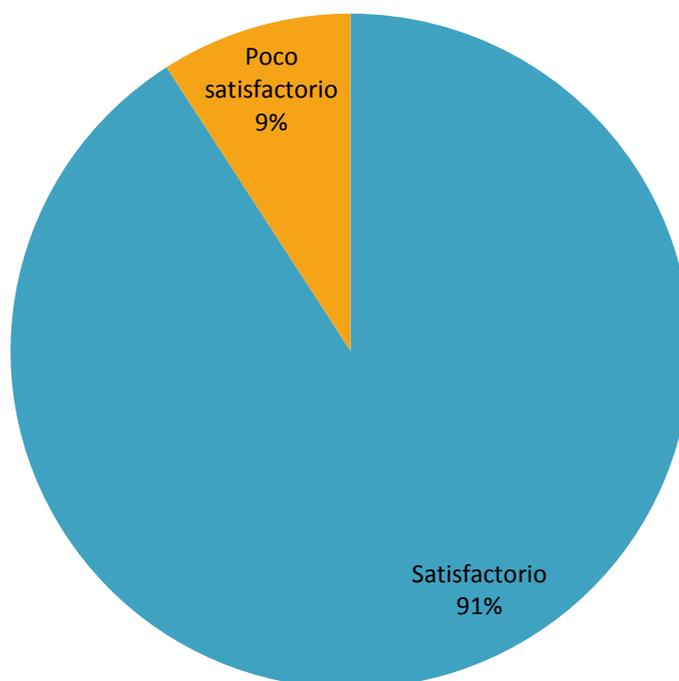


Figura 49. Gráfico circular – Soporte al proceso de negocio “Autorización y envío de propuestas”.
Fuente: El autor.
Elaboración: El autor.

Tabla 48. Valores encuesta - Soporte al proceso de negocio "Autorización y envío de propuestas".

Valor	Porcentaje	Respuestas
Satisfactorio	90.9%	10
Poco satisfactorio	9.1%	1
	Total	11

Fuente: El autor.
Elaboración: El autor.

Complejidad

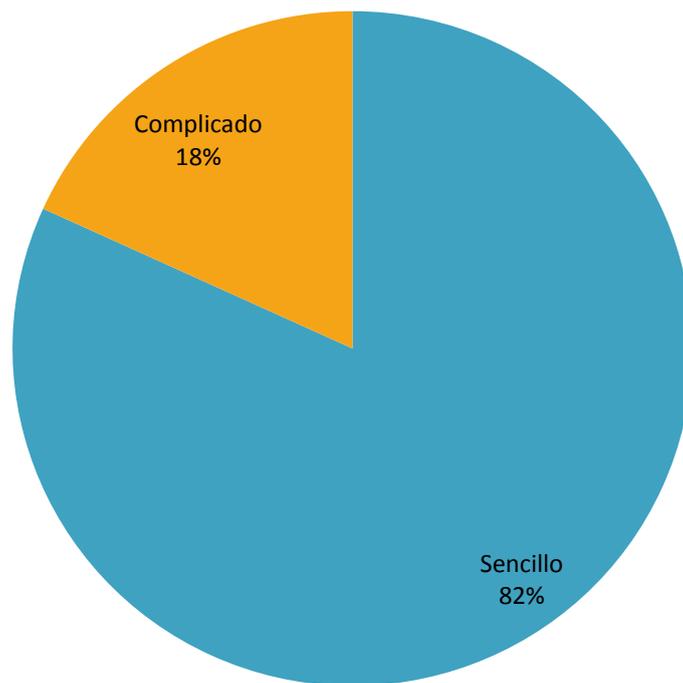


Figura 50. Gráfico circular – Complejidad "Autorización y envío de propuestas".

Fuente: El autor.
Elaboración: El autor.

Tabla 49. Valores encuesta – Complejidad “Autorización y envío de propuestas”.

Valor	Porcentaje	Respuestas
Sencillo	81.8%	9
Complicado	18.2%	2
	Total	11

Fuente: El autor.
Elaboración: El autor.

Velocidad

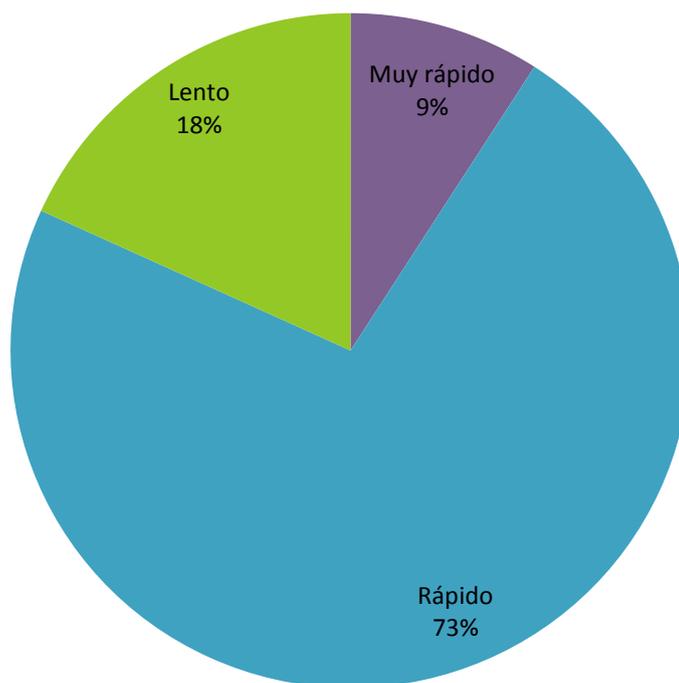


Figura 51. Gráfico circular – Velocidad “Autorización y envío de propuestas”.

Fuente: El autor.
Elaboración: El autor.

Tabla 50. Valores encuesta – Velocidad “Autorización y envío de propuestas”.

Valor	Porcentaje	Respuestas
Muy rápido	9.1%	1
Rápido	72.7%	8
Lento	18.2%	2
	Total	11

Fuente: El autor.

Elaboración: El autor.

Confiabilidad

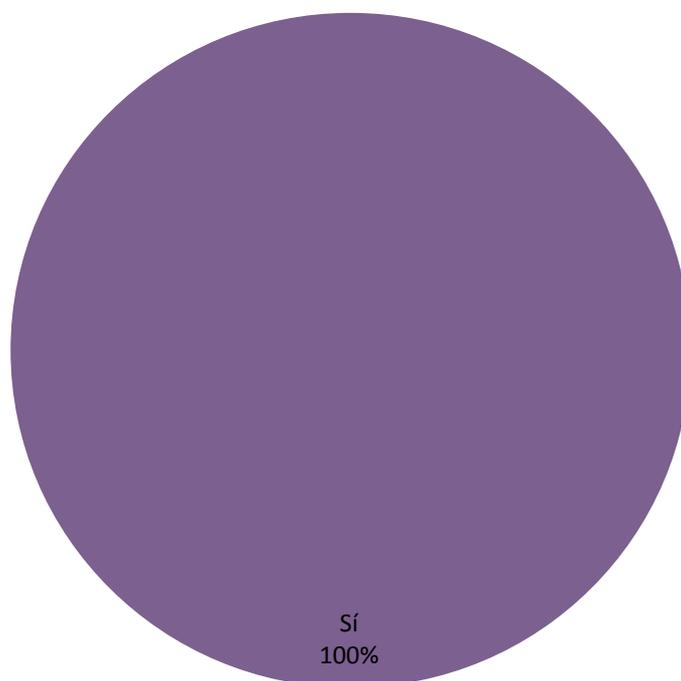


Figura 52. Gráfico circular – Confiabilidad “Autorización y envío de propuestas”.

Fuente: El autor.

Elaboración: El autor.

Tabla 51. Valores encuesta – Confiabilidad “Autorización y envío de propuestas”.

Valor	Porcentaje	Respuestas
Sí	100.0%	11
	Total	11

Fuente: El autor.
Elaboración: El autor.

Análisis

El quinto escenario evaluado por los docentes suministra los siguientes resultados mayoritarios con respecto a la Autorización y envío de propuestas según los cuatro aspectos evaluados:

Tabla 52. Análisis Encuesta - Escenario “Autorización y envío de propuestas”.

Aspecto evaluado	Resultado	Porcentaje
Soporte al proceso de negocio	Satisfactorio	91%
Complejidad	Sencillo	82%
Velocidad	Rápido	73%
Confiabilidad	Confiable	100%

Fuente: El autor.
Elaboración: El autor.

Se puede observar que para los docentes la Autorización y envío de propuestas soporta al proceso de negocio de la titulación, es sencillo de validar, se hace rápidamente y los datos guardados son los correctos. Por lo tanto este escenario en la herramienta es aceptado por la titulación.

5.3. Cómo la herramienta resuelve los problemas identificados

La solución desarrollada se enfoca en resolver los problemas identificados en la sección 2.4. La creación de un repositorio centralizado para los trabajos de titulación, en los cuales se les pueda dar seguimiento con una aplicación que de soporte al proceso completo y que se convierta como una herramienta de uso principal para las titulaciones de la UTPL, iniciando con las titulaciones de Sistemas Informáticos y Computación e Ingeniería Informática, constituye un gran paso para una mejora de procesos eficiente para la UTPL.

A continuación se presenta un análisis por parte de los docentes que hicieron uso de la herramienta, para desde su perspectiva evaluar si la herramienta constituye una solución para los cinco problemas principales identificados en este trabajo.

Se evaluó los cinco problemas con tres respuestas objetivas para cada problema:

- Lo resuelve completamente
- Lo resuelve parcialmente
- No lo resuelve

5.3.1. Falta de información de propuestas para trabajos de titulación.

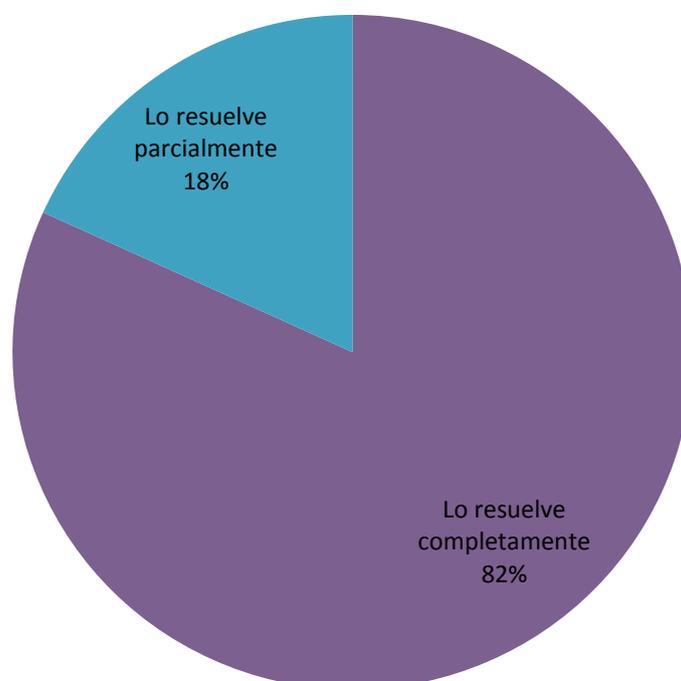


Figura 53. Gráfico circular – Resolución del problema "Falta de información de T.T."

Fuente: El autor.

Elaboración: El autor.

Tabla 53. Valores encuesta – Resolución del problema "Falta de información de T.T."

Valor	Porcentaje	Respuestas
Lo resuelve completamente	81.8%	9

Lo resuelve parcialmente	18.2%	2
	Total	11

Fuente: El autor.
Elaboración: El autor.

5.3.2. Estudiantes reprobaban o desertaban de los trabajos de titulación.

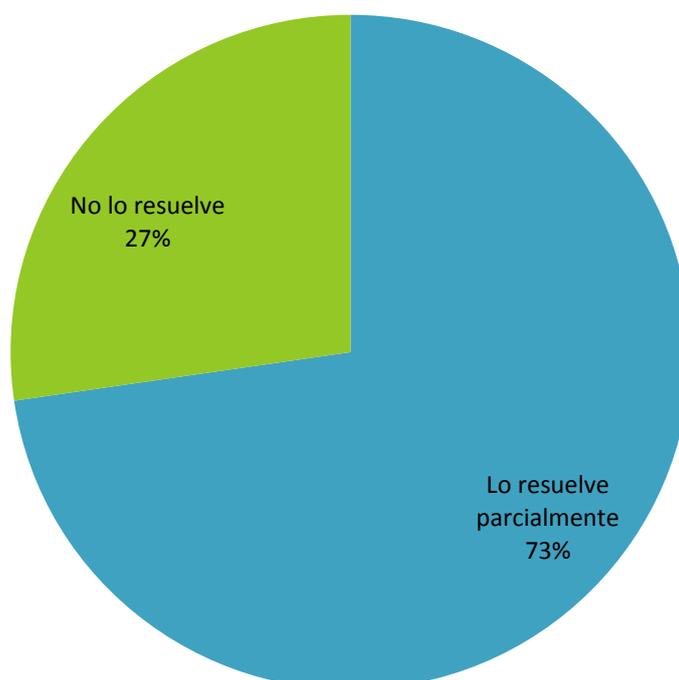


Figura 54. Gráfico circular – Resolución del problema "Reprobación y abandono de estudiantes a T.T."

Fuente: El autor.
Elaboración: El autor.

Tabla 54. Valores encuesta – Resolución del problema "Reprobación y abandono de estudiantes a T.T."

Valor	Porcentaje	Respuestas
Lo resuelve parcialmente	72.7%	8
No lo resuelve	27.3%	3

	Total	11
--	-------	----

Fuente: El autor.
Elaboración: El autor.

5.3.3. Retraso en propuesta y postulación de trabajos de titulación.

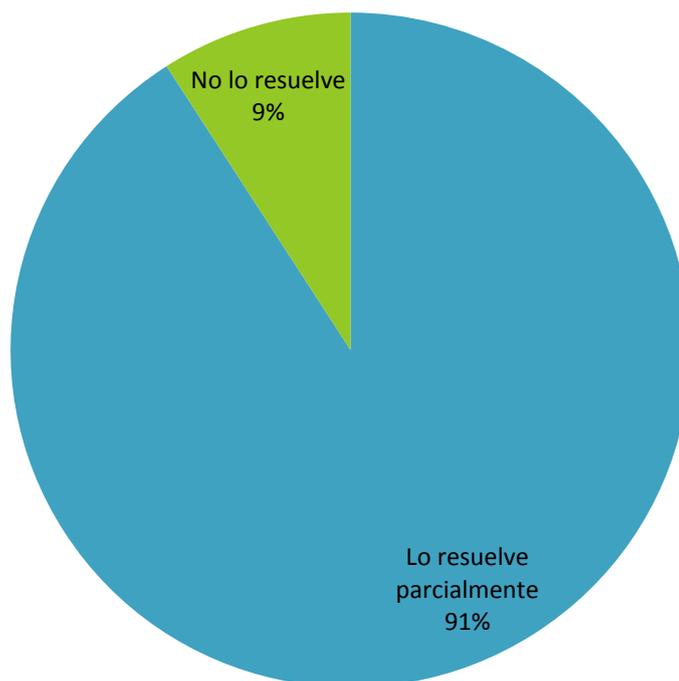


Figura 55. Gráfico circular – Resolución del problema "Retraso en propuesta y postulación de T.T."
Fuente: El autor.
Elaboración: El autor.

Tabla 55. Valores encuesta – Resolución del problema "Retraso en propuesta y postulación de T.T."

Valor	Porcentaje	Respuestas
Lo resuelve parcialmente	90.9%	10
No lo resuelve	9.1%	1
	Total	11

Fuente: El autor.
Elaboración: El autor.

5.3.4. Retraso en revisiones de propuestas para trabajos de titulación.

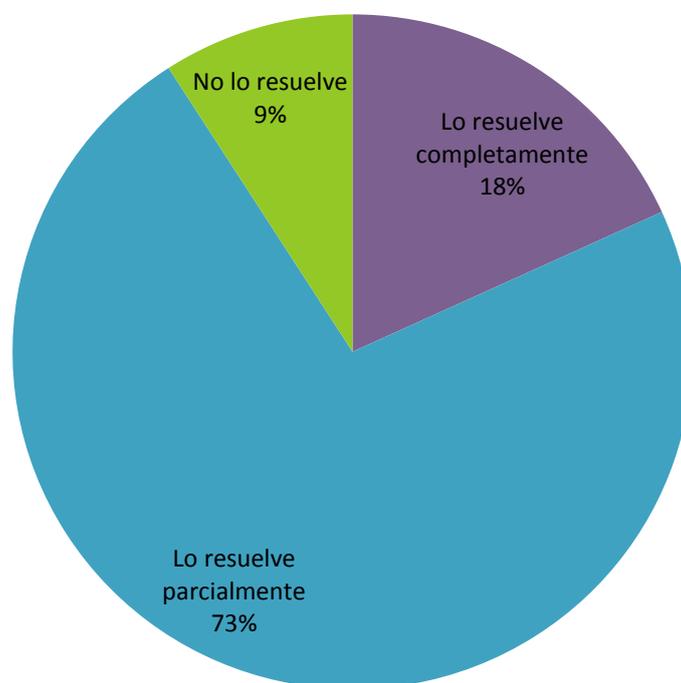


Figura 56. Gráfico circular – Resolución del problema "Retraso en revisiones de T.T."

Fuente: El autor.

Elaboración: El autor.

Tabla 56. Valores encuesta – Resolución del problema "Retraso en revisiones de T.T."

Valor	Porcentaje	Respuestas
Lo resuelve completamente	18.2%	2
Lo resuelve parcialmente	72.7%	8
No lo resuelve	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

5.3.5. Dificultad para notificación a involucrados.

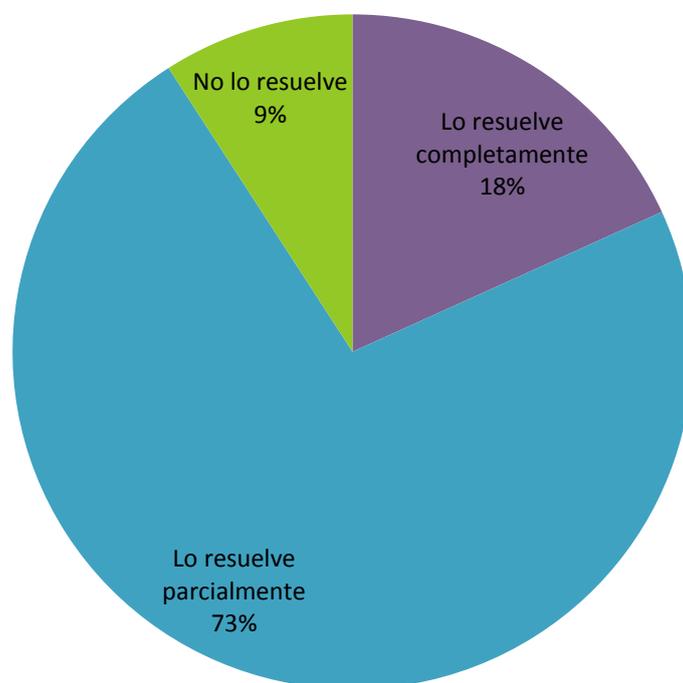


Figura 57. Gráfico circular – Resolución del problema "Dificultad para notificaciones."

Fuente: El autor.

Elaboración: El autor.

Tabla 57. Valores encuesta – Resolución del problema "Dificultad para notificaciones"

Valor	Porcentaje	Respuestas
Lo resuelve completamente	18.2%	2
Lo resuelve parcialmente	72.7%	8
No lo resuelve	9.1%	1
	Total	11

Fuente: El autor.

Elaboración: El autor.

5.3.6. Análisis.

Al analizar las respuestas de los docentes sobre si la herramienta resuelve los principales problemas de la titulación identificados en cuanto al proceso concerniente a los trabajos de titulación obtenemos los siguientes resultados mayoritarios.

Tabla 58. Análisis Encuesta - Escenario "Autorización y envío de propuestas".

Problema	Resultado	Porcentaje
Falta de información de Trabajos de Titulación	Lo resuelve completamente	82%
Reprobación o abandono por parte de estudiantes a Trabajos de Titulación.	Lo resuelve parcialmente	73%
Retraso en propuesta y postulación de Trabajos de Titulación.	Lo resuelve parcialmente	91%
Retraso en revisiones de Trabajos de Titulación.	Lo resuelve parcialmente	73%
Dificultad para notificación a involucrados en cada parte del proceso.	Lo resuelve parcialmente	73%

Fuente: El autor.

Elaboración: El autor.

CONCLUSIONES

Al finalizar el presente trabajo se ha podido concluir lo siguiente:

- Es necesario el desarrollo y la implementación de una herramienta unificando los criterios de cada involucrado en el proceso de postulación de trabajos de titulación, que mantenga el proceso centralizado y alineado a los lineamientos oficiales definidos por la titulación y que se pueda medir y controlar cada etapa del proceso, puesto que la interpretación y el cumplimiento del proceso por parte de todos los involucrados se realiza de manera diferente para cada uno. Cada involucrado realiza sus actividades según lo descrito en los lineamientos y plantillas pero con su interpretación personal.
- Scrum suministra muchas ventajas frente a metodologías tradicionales, debido a los criterios de iteratividad para el desarrollo incremental, sin embargo, los procesos complejos y altamente formales requieren que se complemente con una documentación usual en otro tipo de metodologías, lo que aporta un alto grado de comprensión del proceso y facilita el desarrollo al tener todo el escenario completamente comprendido.
- En cuanto al modelado de requerimientos, la diagramación y especificación de casos de uso vinculados con las historias de usuario han otorgado robustez al proceso de desarrollo y permitido comprender cada flujo que requería cada historia de usuario, manteniendo las entregas continuas de la metodología ágil.
- La Infraestructura como Servicio (IaaS) en cuanto a seguridad es un modelo que independiza de la gestión al equipo y al gerente de proyecto, pues el equipo de desarrollo sólo se encarga de la seguridad a nivel de código, y el acceso a la infraestructura es gestionada y garantizada por el proveedor de servicio.
- Los tiempos de respuesta para el servicio de infraestructura Cloud implementado en el proyecto (Heroku) son óptimos aun cuando la suscripción utilizada sea gratuita, 500 a 1200 ms en esta aplicación desde la petición inicial hasta la carga completa de la respuesta tanto en la presentación como en el guardado de una transacción, comparado con el promedio normal de una aplicación web cliente-servidor dónde sus tiempos de respuesta está entre los 5000 a 11000 ms (tiempo de respuesta de www.miamisportloja.com).
- La opción más viable de integración de sistemas es la utilización de servicios bajo el protocolo HTTP que pueden ser SOAP o REST, en nuestro caso REST constituye un

servicio más rápido de implementar porque puede suministrar los datos específicos necesarios para la aplicación y permite la integración de sistemas heterogéneos que usualmente es una tarea complicada, puesto que varios sistemas son desarrollados bajo distintos lenguajes de programación, infraestructura y motores de base de datos.

RECOMENDACIONES

Durante el proceso de elaboración del presente proyecto técnico se han considerado las siguientes recomendaciones técnicas:

- Aunque el desarrollo ágil no utiliza una documentación exhaustiva frente al desarrollo tradicional, es muy recomendable definir un esquema de documentación para usar artefactos de las metodologías tradicionales que apoyen el desarrollo eficiente y faciliten la comprensión completa de un requerimiento para el programador. Los artefactos más recomendados son: El diagrama de casos de uso, la especificación de casos de uso, el diagrama de secuencias y el modelo de datos.
- La implementación con el framework Django, y con la mayoría de frameworks utilizan migraciones mediante sus propios ORM (Object Relational Mapping) implementados para poder definir sus datos mediante el lenguaje de programación y no sentencias propias del motor de base de datos. Por lo tanto es muy recomendable que como parte de la documentación siempre se mantenga actualizado el modelo de datos de la aplicación, para gestionar de manera sencilla los cambios en el esquema de datos y poder dar un seguimiento correcto a los cambios solicitados.
- Para la integración de sistemas que están desarrollados en distintos lenguajes de programación y motores de bases de datos, se recomienda la implementación de un servicio REST en ambas aplicaciones, y definir los esquemas de datos que necesita cada aplicación para consumir de la otra.
- Para la correcta validación de cada entregable, es recomendable identificar usuarios directos clave para el sistema e involucrarlos en la validación desde los inicios. Estos usuarios son quienes disponen un alto grado de conocimiento sobre el proceso, y pueden originar un cambio de requerimientos sustancial si no se valida la aplicación con ellos desde los primeros entregables.

BIBLIOGRAFÍA

- Aguirre, C. B. E., & Esquivel, P. (2012). Comparativa de Frameworks para el desarrollo de aplicaciones con php. Recuperado a partir de <http://dspace.uazuay.edu.ec/bitstream/datos/2125/1/08920.pdf>
- Ahmad, M. O., Markkula, J., & Oivo, M. (2013). Kanban in software development: A systematic literature review. En *Software Engineering and Advanced Applications (SEAA), 2013 39th EUROMICRO Conference on* (pp. 9–16). IEEE. Recuperado a partir de <http://ieeexplore.ieee.org/abstract/document/6619482/>
- Ahmad, S., & Kumar, P. S. (2016). An efficient privacy-preserving multi-keyword ranked search over encrypted data in cloud computing. En *2016 IEEE Annual India Conference (INDICON)* (pp. 1-6). <https://doi.org/10.1109/INDICON.2016.7838916>
- Alvarez, M. A., Lazaro, J. M., & Mendez, N. (2002). Introducción a los lenguajes del web. *DesarrolloWeb.com*.< [http://www.ipereda.com/descargas/manuales/php/1.-Manual%20de%20Introducci%C3%B3n%20Lenguajes%20Web%20-%2022%20pag.pdf](http://www.desarrolloweb.com/manuales/27>(26/02/2003). Recuperado a partir de <a href=)
- Bruegge, B., & Dutoit, A. H. (2010). *Object-Oriented Software Engineering*. Prentice Hall. Recuperado a partir de <http://danaih50portfolio.synthasite.com/resources/Bruegge%20-%20Object-Oriented%20Software%20Engineering.pdf>
- Camacho, E., Cardeso, F., & Nuñez, G. (2004). *Arquitecturas de Software. Guía de estudio.[En línea]*.
- Canós, J., Letelier, P., & Penadés, M. C. (2013). Metodologías Ágiles en el desarrollo de Software. *Universidad Politécnica de Valencia, Valencia*. Recuperado a partir de http://www.carlosfau.com.ar/nqi/nqifiles/XP_Agil.pdf
- Castrelo Cid, A. (2014). MMO de navegador en tiempo real con Node.js y WebSockets. Recuperado a partir de <http://diposit.ub.edu/dspace/handle/2445/59452>
- Chavez, S., Martín, A., Rodríguez, N. R., Murazzo, M. A., & Valenzuela, A. (2012). Metodología AGIL para el desarrollo SaaS. Presentado en XIV Workshop de Investigadores en Ciencias de la Computación. Recuperado a partir de <http://hdl.handle.net/10915/18977>

- Condori Ayala, J. L. (2012). Python-DjangoFramework de desarrollo web para perfeccionistas Basado en el Modelo MTV. *Revista de Información, Tecnología y Sociedad*, 36.
- Domínguez Purificación, A. (2015). Desarrollo de una red social musical para Android. Recuperado a partir de <http://upcommons.upc.edu/handle/2099.1/26630>
- Figueroa, R. G., Solís, C. J., & Cabrera, A. A. (2008). Metodologías Tradicionales vs. Metodologías Ágiles. *Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación. (En línea)*, Disponible en: <http://adonisnet.files.wordpress.com/2008/06/articulo-metodologia-de-sw-formato.doc>. Recuperado a partir de <http://tg-tatiana-oquendo.googlecode.com/svn/trunk/articulo-metodologia-de-sw-formato.doc>
- Gallego, M. T. (2012). Metodología Scrum. *Gestión de Proyectos Informáticos*, <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>. Recuperado a partir de <http://www.quimbiotec.gob.ve/sistem/auditoria/pdf/ciudadano/mtrigasTFC0612memoria.pdf>
- Gottesdiener, E. (2005). *The Software Requirements: Memory Jogger: a Pocket Guide to Help Software and Business Teams Develop and Manage Requirements*. GOAL/QPC.
- Harris, A., & Haase, K. (2011). *Sinatra: Up and Running*. O'Reilly Media, Inc.
- Hidalgo Macas, L. N., Acaro, J., & Edison, M. (2016). *Estudio Comparativo de los Servicios Web Restfull Jersey y SOAP JAX-WS para el Desarrollo de una Aplicación Android con Wikitude Aplicada a la Gestión de Información Geolocalizada del Turismo de la Provincia de Chimborazo*. (B.S. thesis). Escuela Superior Politécnica de Chimborazo. Recuperado a partir de <http://dspace.esPOCH.edu.ec/handle/123456789/4733>
- IBM. (s. f.). SoftLayer | Cloud Servers, Storage, Big Data, & More IAAS Solutions. Recuperado 8 de febrero de 2017, a partir de <http://www.softlayer.com/es>
- IBM developerWorks en español: DevOps: Visión general. (2015, octubre 1). [CT512]. Recuperado 27 de agosto de 2015, a partir de <http://www.ibm.com/developerworks/ssa/rational/devops/>

- Javier Sevilla Sánchez. (s. f.). *Inyección de dependencia y primeros pasos con Spring*. Recuperado a partir de http://spring-uah-study.googlecode.com/svn-history/r26/trunk/TFC_Memoria.doc
- Kruchten, Philippe. (s. f.). Planos Arquitectónicos: El Modelo de 4+ 1 Vistas de la Arquitectura del Software. 1995, 42-50.
- Martín, A., Chávez, S., Murazzo, M. A., Rodríguez, N. R., & Valenzuela, A. (2015). MongoDB en ambiente Cloud Híbrido con OpenStack. En *XVII Workshop de Investigadores en Ciencias de la Computación (Salta, 2015)*. Recuperado a partir de <http://sedici.unlp.edu.ar/handle/10915/45569>
- Marzal, A., & Gracia, I. (2003). Introducción a la programación con Python. *Universitat Jaume I*. Recuperado a partir de <http://miavbe.net/documentos/python/python.pdf>
- Mayor Martín, D. (2014). Evaluación de Spring MVC. Recuperado a partir de <http://dspace.uah.es/dspace/handle/10017/20742>
- Middleton, N., So, N. M., & Schneeman, R. (2013). *Heroku: Up and Running*. O'Reilly Media, Inc.
- Narvaez, Efren. (2017). Diseño e implementación de una arquitectura basada en web services RESTful para garantizar la interoperabilidad semántica e integridad de datos académicos. Recuperado a partir de <http://dspace.utpl.edu.ec/handle/123456789/16650>
- Pérez Rodríguez, F. (2013). Aplicación web para compartir coche, mediante las tecnologías JAVA, SPRING y JPA. Recuperado a partir de <http://diposit.ub.edu/dspace/handle/2445/48608>
- Pico Coba, A. F. (2016). *Análisis descriptivo de la tecnología Ruby on Rails para el desarrollo de páginas web*. (B.S. thesis). Quito: UCE. Recuperado a partir de <http://www.dspace.uce.edu.ec:8080/handle/25000/6358>
- Rodriguez, I. P., Pettoruti, J. E., Chichizola, F., & De Giusti, A. E. (2011). Despliegue de un Cloud Privado para entornos de cómputo científico. En *XVII Congreso Argentino de Ciencias de la Computación*. Recuperado a partir de <http://sedici.unlp.edu.ar/handle/10915/18648>
- Sommerville, I. (2005). *Ingeniería del software*. Pearson Educación.

- Souidi, S., Boccio, D., Mierzwa, S., & Aguilar, J. (2015). The feasibility of using Microsoft Azure infrastructure for a monitoring and evaluation system solution in sub-Saharan Africa. En *2015 IEEE Global Humanitarian Technology Conference (GHTC)* (pp. 226-232). <https://doi.org/10.1109/GHTC.2015.7343977>
- Souza, V. F. de, L'Erario, A., Fabri, J. A., & Gonçalves, J. A. (2016). Model for monitoring in distributed projects: An experiment using Kanban and Business Process Modeling Notation (BPMN). En *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1-5). <https://doi.org/10.1109/CISTI.2016.7521385>
- Spona, H. (2010). *Programación de bases de datos con MYSQL y PHP*. Marcombo. Recuperado a partir de https://books.google.com.ec/books?hl=es&lr=lang_es&id=y11L7pQfdRsC&oi=fnd&pg=PA1&dq=php&ots=5xj-IAHZWY&sig=ou95miSn9q4QyqkMQXqVNocfJyY
- Torbacki, W. (2008). SaaS—direction of technology development in ERP/MRP systems. *Archives of Materials Science*, 58, 58.
- Vaquero, L. M., Roderó-Merino, L., Cáceres, J., & Lindner, M. (2008). A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1), 50–55.
- Zahariev, A. (2009). Google app engine. *Helsinki University of Technology*, 1–5.

ANEXOS

ANEXO 1. Diagrama de flujo de proceso de postulación

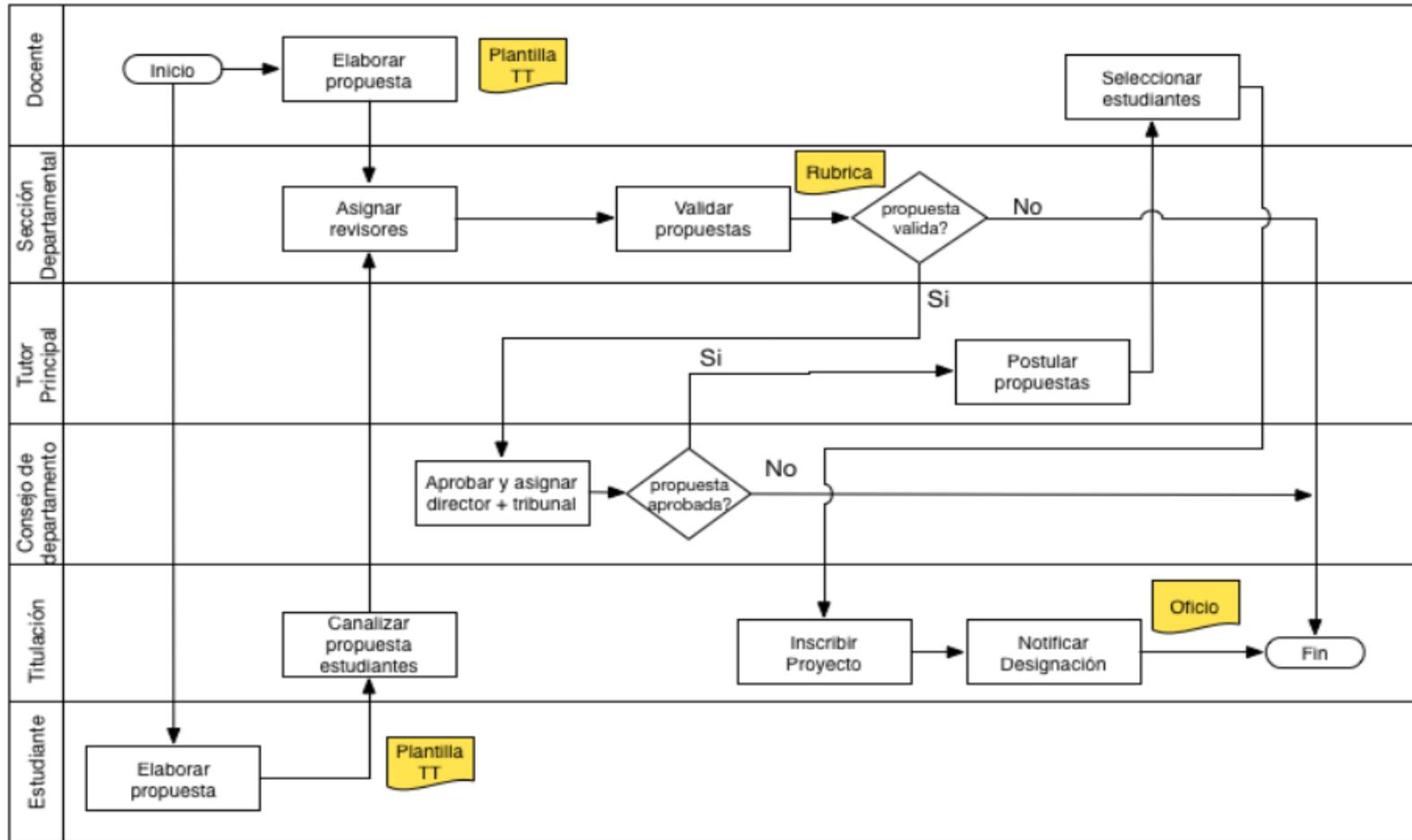


Figura 58. Diagrama de Flujo proceso postulación de trabajos de titulación
 Fuente: Lineamientos generales para postulación de Trabajos de Titulación v1.2.
 Elaboración: Titulación Sistemas Informáticos y Computación.

ANEXO 2. Preguntas para entrevista

En base al proceso de “Postulación, revisión, validación, aprobación, y asignación de trabajos de titulación en su titulación” sírvase responder las siguientes preguntas:

1. ¿Cuál es el proceso oficial si existe?
2. Existe documentación oficial sobre el proceso.
3. ¿Cómo es su participación en el proceso?
 - a. Qué roles cumple
 - b. Qué decisiones toma
4. ¿Qué inconvenientes ha encontrado en los procesos de postulación y seguimiento?
 - a. ¿En qué tarea específica?
5. ¿Cuáles cree que serían las pautas para resolver estos inconvenientes?
6. ¿Con qué otros procesos o aplicaciones se relaciona el proceso?
7. ¿Qué otros aspectos se debería considerar antes de proponer una solución?
8. En caso de implementarse la herramienta para gestionar el proceso. ¿Qué otros aspectos debería considerarse?
9. ¿Qué información considera usted importante de obtener con la herramienta/aplicación?

ANEXO 3. Reporte de entrevistas

Entrevista

UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA

06 mayo/2016

Reporte de Entrevista

Mg. María Soledad Segarra

COORDINADOR DE TITULACIÓN INGENIERÍA CIVIL

1. ¿Cuál es el proceso oficial si existe?

Los docentes proponen los temas pero también se da cabida a que los estudiantes propongan los temas con el respaldo de algún docente.

- a. El docente hace la postulación
- b. El responsable de sección valida esa postulación. Al hacer esta validación el responsable de sección debería verificar que esta esté afín a las líneas de investigación, que el tema propuesto esté respaldado por las competencias ganadas dentro de la formación del estudiante, puesto que si no se alinea al perfil profesional puede suceder que el estudiante no pueda conseguir cumplir con el proyecto, o durante el adiestramiento en el tema la tesis no va a cumplir el cronograma, por tanto se empiezan a alargar los temas.
- c. Responsable de sección los lleva a consejo de departamento y se aprueban los temas. Todo el proceso se realiza manualmente.
- d. Los docentes que tengan los temas aprobados suben sus temas a la plataforma actual de trabajos de titulación.
- e. Los temas son nuevamente validados por el responsable de sección, habilitando a que el tema se lo muestre al estudiante. Esta validación debería considerar:
 - i. La propuesta debe estar acorde al perfil profesional.
 - ii. Debe estar acorde a la formación que ha tenido el estudiante
 - iii. Debe estar acorde al perfil de egreso.
 - iv. Debe estar acorde a las líneas de investigación que se están manejando.
 - v. Debe estar acorde a las necesidades del contexto, es decir, respondiendo a agendas zonales, respondiendo a necesidades u objetivos definidos en el plan nacional del buen vivir.

2. ¿Existe documentación oficial sobre el proceso?

El Vicerrectorado Académico se basa en la Unidad de Titulación Especial y lo que el CES ha regulado mediante los reglamentos vigentes, la UTPL da cumplimiento a los mismos. Sin embargo hay algunos vacíos,

3. ¿Cómo es su participación en el proceso?**a. Qué roles cumple**

Participo como docente, puesto que los coordinadores no intervienen en la validación de temas ni cronogramas.

Sólo subo como docente mis temas para que los aprueben y los validen.

4. ¿Qué inconvenientes ha encontrado en los procesos de postulación y seguimiento?

- El coordinador en ningún momento entra a validación o revisión en el sistema actual de Trabajos de Titulación.
- La malla académica está ofertada de tal manera que al finalizar el décimo semestre los estudiantes se titulan. No está ofertada de tal manera que al finalizar la G.P 4.2 se pueda finalizar la tesis, y esto no se está cumpliendo.
- Las líneas de investigación pertenecen a las secciones y no a la titulación, es decir, en este momento las líneas de investigación están separadas de la titulación. Y la titulación hace formación en ciertas ramas pero muchas veces las líneas de investigación se van por otros caminos. Si tengo una línea de investigación en vialidad, transporte y tráfico, debe haber componentes que respalden la formación de los estudiantes.
- Las líneas de investigación no están respaldadas por componentes de la titulación. Por tanto los trabajos de tesis no pueden desarrollarse o crecer en esas líneas.
- Los estudiantes no se gradúan o no terminan en el tiempo que debe ser. La eficiencia terminal en la titulación es del 10%, y el CES está pidiendo una eficiencia terminal de las titulaciones de un 75%.
- El sistema debe evidenciar plazos que tiene el estudiante para titularse según el RRA que es de 3 semestres, es decir, la GP 4.1, GP4.2 y un semestre más. Por tanto los estudiantes desconocen a veces que su tiempo ya finalizó y que tienen que matricularse y asistir como oyentes a la GP 4.2.

5. ¿Cuáles cree que serían las pautas para resolver estos inconvenientes?

- Que el coordinador de titulación se involucre en la revisión, pues es el coordinador quien está en constante contacto con los estudiantes y sabe que temas presentan mayores problemas y toman más tiempo.
- El CES mismo ha pedido que las líneas de investigación pertenezcan a la titulación no a las secciones, para que haya una armonía entre las titulaciones y las líneas de investigación.
- Una vez que esté bien definido el proceso los estudiantes ya puedan postular a todos los temas y no esperar a las postulaciones con las fechas retrasadas.
- Debe existir un cronograma adecuado de manera que se cumpla el cronograma, en cuanto a revisiones y al desarrollo del trabajo de titulación.
- La intervención del coordinador de titulación va a velar que se trabaje por las líneas de investigación adecuadas y que esté afin a la formación del estudiante.

6. ¿Qué otros aspectos se debería considerar antes de proponer una solución?

- Considerar la inclusión de pares revisores con respecto a la información que va subiendo el estudiante y puedan hacer sugerencias antes que el proceso esté terminado.
- Considerarse los estudiantes que aplican por el examen complejo

7. En caso de implementarse la herramienta para gestionar el proceso. ¿Qué otros aspectos debería considerarse?

- La implementación de pares revisores a la propuesta
- La asignación de docentes de apoyo durante el desarrollo del trabajo de titulación.

8. ¿Qué información considera usted importante de obtener con la herramienta/aplicación?

- Los reportes, puesto que el CES está pidiendo reportes constantemente, sobre la aprobación o abandono de estudiantes en cada GP. Para actuar si hay que cambiar de tutor, de tema o cambiar el estudiante a la opción de examen complejo.

- Reporte
Mg. María
COORDINADORA
- El cumplimiento evidenciando el avance físico del trabajo de los estudiantes.
 - Dar una salida de estado del trabajo de los estudiantes.
 - La valoración objetiva con una rúbrica antes de que la propuesta se postule a los estudiantes.
 - La valoración objetiva con una rúbrica cuando el trabajo se ha finalizado.


Mg. María Soledad Segarra
COORDINADOR TITULACIÓN


José Luis Quichimbo Suárez
ESTUDIANTE

Figura 59. Reporte entrevista coordinador titulación
Fuente: El autor
Elaboración: El autor

Reporte de Entrevista

Ing. Guido Riofrío

RESPONSABLE SECCIÓN DEPARTAMENTAL

TITULACIÓN SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

1. ¿Cuál es el proceso oficial si existe?

Se está rediseñando este proceso. Pero por ejemplo los profesores de las G.P 4.1 nos envían a nuestra sección departamental en una plantilla cuáles son las propuestas y se evalúa en base a diferentes parámetros.

Un alumno o un docente hoy en día plantea un proyecto de tesis, a ese proyecto de tesis se le asigna un revisor a un determinado profesor, ellos revisan el tema y hay un documento de evaluación de ese proyecto en base a ciertos componentes, y al final eso da una evaluación total de ese ante proyecto que es máximo una nota de 5, el que tiene 5 será lo ideal.

Una vez con eso, el docente que hizo la revisión me devuelve a mí el ante proyecto y el documento que hizo él la revisión.

Eso se lo trae nuevamente a consejo de departamento para hacer una nueva revisión entre los miembros de consejo de departamento.

2. ¿Cómo es su participación en el proceso?

Actualmente en la asignación de propuestas de tesis, ya sea que salgan a partir de un alumno, o que salgan de un profesor, yo como coordinador de sección intervengo en la parte de asignarle, por ejemplo, reviso brevemente el tema y si es algo que lo maneja más cierto profesor pues le digo a él que lo revise, y así mismo si lo maneja más otros profesores pues les asigno a ellos, les entrego el documento físico que me envían o a veces un correo electrónico, y les asigno a ellos para que ellos lo revisen. Y después me dan eso, obviamente se le da al coordinador de titulación que después hace una nueva revisión.

3. ¿Qué inconvenientes ha encontrado en los procesos de postulación y seguimiento?

Los envíos de avances de trabajos de titulación, es complicado revisarlos y darles seguimiento. Ahora por ejemplo cuando un alumno envía un avance, el profesor en algunos casos lo tienen bien organizado, en otros casos se ponen a buscar en los correos, en otros casos dicen "¿dónde lo guardé?, envíame de nuevo el último avance" y esas cosas hacen que el proceso vaya disminuyendo su calidad, se vaya minando ese proceso.

Hay muchas revisiones en el proceso de aprobaciones de tesis.

Algo en lo que los coordinadores de sección coincidimos es en que el proceso es demasiado largo hoy en día, porque digamos yo le asigno como revisor a un docente y el revisa y da sus criterios, después de eso viene la junta de departamento y nuevamente se hace una revisión, entonces vimos que en el proceso habían como tres o cuatro revisiones. Entonces el proceso estaba muy largo, y a veces pasaba varias revisiones y al llegar a la revisión de consejo y se anulaba.

4. ¿Cuáles cree que serían las pautas para resolver estos inconvenientes?

Sería interesante tener un portal dónde en el cuál yo como coordinador de sección designe a unos 3 profesores que evalúen un tema de tesis y entonces en el sistema tener la disponibilidad que varios docentes hagan una evaluación y al final me saque un promedio. Porque a veces si yo le asigno una evaluación a una sola persona, él por lo general desde su criterio, desde su punto de vista, desde su perspectiva del tema que se está asignando lo revisa; pero si se tuviera una herramienta sería más fácil que los diferentes profesores entren al sitio y hagan una evaluación y al final sacar un promedio entre los tres sería algo más realista.

Para mí lo ideal sería potenciar la revisión de los ante proyectos en la sección, en lugar que sea un revisor, que sean unos dos o tres miembros de la sección departamental y darles unos tres días para que ellos revisen el tema y mediante una herramienta ellos suban ahí su calificación del ante proyecto.

En un software poner ahí las observaciones de porqué se lo negó, tal vez porqué se lo aceptó, en qué se mejore, es decir ese proceso de revisión hacerlo estrictamente en la sección departamental, porque los temas de tesis cubren temas propios de la sección. Todo proceso de revisión para mi criterio tendría que ser internamente en la sección departamental.

5. ¿Con qué otros procesos o aplicaciones se relaciona el proceso?

Orientado específicamente al tema de las tesis existe este sistema para ver el tema del plagio de las tesis. Que sometes la tesis a este sistema y te dice en qué porcentaje ha sido copiado, plagiado.

Todo esto nace del Sistema de Gestión Académica, pero pienso que es demasiado amplio enganchar este proyecto con el Sistema De Gestión Académica. Sería una tesis para tres o cuatro personas que esta herramienta se enganche con todos los sistemas que intervienen.

6. ¿Qué otros aspectos se debería considerar antes de proponer una solución?

Que en los trabajos de titulación se brinden facilidades operativas, por ejemplo, un sitio centralizado en los avances que va desarrollando el tesista. Porque a veces es complicado, como son muchos estudiantes de modalidad a distancia, alguno envía un avance, luego tienes que revisarlo, no sabes dónde te lo envió, te pones a buscarlo en el correo y se hace un poco difícil.

Sería interesante que el director tenga el repositorio en un sistema controlando las diferentes versiones que va enviando el alumno. Que en alguna herramienta te permita poner las observaciones del trabajo.

Considerar un versionamiento de los avances de tesis. Poder revisar en un nuevo avance las observaciones que se le hizo anteriormente y ver si ha mejorado, si está mal, si está bien y ver esa evolución.

7. En caso de implementarse la herramienta para gestionar el proceso. ¿Qué otros aspectos debería considerarse?

- En esa herramienta mientras se va revisando el documento se debe ir poniendo observaciones, y a partir de esa herramienta debe notificarse directamente al alumno o autor del ante proyecto diciendo cuáles son las observaciones que tiene que hacer, que el alumno agilice y reenvíe ese proyecto y de manera coordinada lo afinen al proyecto o lo rechacen si no es viable.
- Que se agilice el proceso desde que un estudiante o docente hace una propuesta de tesis hasta que por fin sea aceptado, hay que tomar en cuenta que cuando un alumno propone un tema, nunca va a estar al 100% correcta, siempre va a ser susceptible a cambios, y obviamente esos cambios según la agilidad del estudiante y profesor, hasta corregir esa propuesta para que finalmente sea aceptada.

8. ¿Qué información considera usted importante de obtener con la herramienta/aplicación?

- La agilización en cuanto a la asignación de revisores, para que el proceso no pase por tantas instancias, más bien potenciar que la revisión sea internamente en el departamento.
- Sería interesante me llegue a mí en una especie de bandeja de entrada un ante proyecto a revisar, y yo tengo la opción de asignar a los docentes que pertenecen a mi sección y automáticamente les hago un check indicando a quién está asignado, el sistema genera un correo electrónico notificándoles que se les ha asignado un ante proyecto de tesis y que por favor lo revisen. Entonces ellos obviamente ya saben que su tarea es que no van a encontrar en el correo electrónico el ante proyecto sino que tendrían que ir a esa herramienta, revisar, y en la herramienta hacer las observaciones, comentarios y poner todos los criterios; y que eso finalmente se una en un solo documento de los tres revisores y se envíe al alumno cuáles son las observaciones, a través de ese mismo software indicando cuáles son las cosas que tiene que mejorar, o indicando que el documento ya ha sido aceptado.
- También debería alertar al coordinador de titulación indicando que el ante proyecto que lo envió tal sección ya cumplió el proceso de revisión y que ya está listo para que se continúe con el proceso.

Ing. Guido Riofrío
DOCENTE
COORDINADOR SECCIÓN INTELIGENCIA A.

José Luis Quichimbo Suárez
ESTUDIANTE

Figura 60. Reporte entrevista responsable sección
Fuente: El autor
Elaboración: El autor

Reporte de Entrevista

Mg. María Belén Mora

TUTOR PRINCIPAL

TITULACIÓN INGENIERÍA EN INFORMÁTICA

1. ¿Cuál es el proceso oficial si existe?

Normalmente dentro de la malla de ingeniería en informática, hay dos materias donde ellos ya trabajan su proyecto de fin de titulación, la una materia es el seminario de fin de carrera que se da en noveno ciclo y la otra es el trabajo de titulación o Prácticum 4 que es la de decimo que la trabajan conmigo, entonces hasta ahora lo que se hacía era en el seminario de fin de carrera de noveno los estudiantes con el docente trabajan propuestas de proyectos ahí veían lo que es un tema de la metodología, de cómo desarrollar un proyecto de tesis, normas bibliográficas, como se desarrolla un estado del arte, en todo sentido ese tipo de temas que les ayuda a ellos a formular el proyecto, entonces ellos trabajaban durante el seminario de fin de carrera toda la propuesta y ellos como resultado de esa materia sacaban lo que es la propuesta en si del documento

Entonces cuando venían ya en mi materia de trabajo de titulación de 10mo ciclo, ellos me traían prácticamente ya aprobado o revisado por lo menos esa propuesta, entonces eso subía acá se revisaba la pertinencia del tema que esté acorde al perfil de la carrera y se asignaba los ante-revisores porque claro yo estoy de tutora y yo podre tener algún tipo de conocimiento en el área que trabajo pero por lo general hay otros temas que me llegan por ejemplo redes, ingeniería de software que no es el área que trabajo entonces normalmente lo que la titulación ha venido haciendo es asignando revisores.

Hay un primer filtro que se llama consejo de departamento inicialmente cuando empieza el ciclo y me llegan todos los proyectos tanto presencial como abierta presentan toda la propuesta del proyecto para el ciclo al concejo entonces el concejo es el que dice pasa o no pasa y generan estas hojas de evaluación entonces ese es un primer filtro que se hace

Entonces yo reviso la pertinencia del tema que este más bien alineado al perfil, si está alineado al perfil se señala, por ejemplo estos son de redes, entonces cada responsable de sección me asigna un revisor entonces ese revisor trabaja con el estudiante evalúa a nivel ya de línea de investigación si está el proyecto debidamente formulado, bueno ahora mismo en este ciclo se nos planteó una hoja de evaluación con rubricas donde dice esto cumple esto no cumple y nos genera un montón de observaciones, entonces hasta ahora lo que se hace esa hoja de rubrica es tener una calificación hay un rango, si el rango pasa de 3.5 creo para arriba se lo tiene aceptado como un cambio si es para abajo se rechaza el proyecto. Entonces cuando se rechaza obviamente al estudiante se le comunica que se lo rechaza y el estudiante tiene dos opciones volver a plantear por el mismo un proyecto o dos trabajar con algún docente específico en el área que quiera para que le ayude a formular otro proyecto, y cuando son aceptados y le sugieren cambios se le comunica igual al estudiante se le pasa los cambios ellos se ponen de acuerdo con los docentes revisores realizan los cambios y lo vuelven a entregar.

Cuando el proyecto está ya totalmente bien se notifica igual con la hoja diciendo que está aprobado eso pasa al consejo de departamento para que asigne director o tribunal de ahí si

se notifica y oficialmente y empieza el desarrollo, eso es hasta ahora como se ha llevado el proceso.

desde ahora octubre 2015-febrero 16, se ha optado un nuevo proceso que es el siguiente normalmente en presencial hay Un Prácticum 4.1 y un Prácticum 4.2, y ahí en la presencial se tiene claro lo que hacen en el 4.1 y 4.2 pero a nivel de la abierta no porque hay un solo Prácticum que es de 10mo ciclo entonces se va a utilizar ese seminario de fin de carrera para hacerlo como un Prácticum 1 y en donde ellos ya puedan ver las opciones de titulación que van a tener es que ahora como ya el régimen académico nos exige a todas las universidades tener al menos dos opciones para que los estudiantes se puedan titular la UTPL y específicamente la titulación de informática opto por dos opciones el estudiante puede o bien desarrollar un trabajo de titulación que es un proyecto de tesis o bien dar el examen complejo entonces en 9no ciclo en el seminario de fin de carrera a ellos se les va a presentar claramente en cómo se trabaja en cada una de las opciones si es bien trabajo de titulación o si es bien examen complejo se les va a explicar la metodología de trabajo de cada una de ellas y se les va a reforzar dentro de áreas temáticas es decir decidanse por cualquiera que sea de la opciones igual va a ver un área fuerte de software, una área fuerte de base de datos, de redes e inteligencia artificial entonces en ese ciclo los estudiantes van a tener claro esas dos opciones y van a tener un poco de recordar lo que tienen e cada área de conocimiento y cuando ya vengan a decimo ciclo yo me voy a encargar en el caso de que hayan escogido el trabajo de titulación como opción, yo como voy a llevar eso a encargarme de que es una metodología de investigación, como se desarrolla el estado del arte, como se hace la referencia bibliográfica hasta que ellos puedan terminar en ese ciclo con lo que es la propuesta aprobada de su trabajo es lo que vamos ahora aplicar desde octubre y la otra opción si escogen examen complejo va a ver otra persona que igual va a estar a cargo de trabajo complejo en donde claro con ellos trabajaran ya ciertos temas, digamos a nivel de examen complejo no le puedo aportar mucho porque yo no manejo ese tema.

2. ¿Cómo es su participación en el proceso?

El rol de nosotros como tutoras es de ver la pertinencia del tema alineado con al perfil y más bien dirigir al docente, esto es de inteligencia artificial, de redes de software y estar nosotros tutelado a nivel de ponernos de acuerdo con el docente revisor, con el estudiante y que ya digamos a nivel metodológico la propuesta quede como tiene que quedar.

3. ¿Qué inconvenientes ha encontrado en los procesos de postulación y seguimiento?

El principal problema que hemos tenido que como ya vinieron de noveno diciendo que les dijeron que esos proyectos ya están listos y cuando paso la primera revisión del concejo los proyectos fueron negados, entonces esa es una de la gran problemática que se tiene, que ahora mismo hemos tenido durante estos dos ciclos que vienen de noveno diciendo que ya están y vienen a 10mo y los proyectos son negados por el consejo, muchos son totalmente rechazados hasta habido un montón de demora que ahora mismo ya estamos a final de ciclo y sigo con estudiantes sin proyecto, trabajando con nuevas propuestas ese es el principal problema que se ha tenido.

¿En cuanto a notificación y comunicación?:

A nivel de notificación es rápido cuando el proyecto ya está aprobado prácticamente me lo envían ya está aprobado ya tienen que generarse una hoja de rúbrica, la hoja de rúbrica está

dentro del proceso que maneja la titulación donde hay ciertos criterios los docentes lo califican y el concejo de departamento como ya ven que esa evaluación está bien entonces ellos dicen perfecto eso se pasa y ellos asignan el director y el tribunal, cuando eso ya esta se pasa directamente a la secretaria a la señora Lidia en este caso que es la secretaria de la abierta ella es lo que hace es oficios de notificación al director y al estudiante, envía tiene tanto tiempo este es su director va a trabajar así y hasta ahí, de ahí es trabajo del estudiante en sí.

4. ¿Cuáles cree que serían las pautas para resolver estos inconvenientes?

Yo sí creo que puede ser importante tener es un control de cambios, un control de versión de todo lo que el estudiante va presentando porque por ejemplo yo le puedo decir ahora yo tengo 10 estudiantes matriculados y para sacar eso tengo que filtrar todos los correos para ver todo el proceso que ellos han ido trabajando, entonces eso sería creo yo importante, incluso para tener registro de la titulación entonces desde que el estudiante presenta su propuesta las revisiones que ha pasado, el proceso de aprobación digamos y ya, un historial de revisiones y un historial de cambios que se ha ido dando ahora hay otro tema al estudiante ya se lo aprueba se lo notifica, pero hay estudiantes que no realizan el tema, entonces claro la herramienta debería también registrar todo ese tipo de cosas, porque por ejemplo Lidia, nuestra secretaria, tiene un Excel de así por años donde por ciclos académicos ella tiene nombre del estudiante, director, fecha de aprobación, fecha de entrega de oficio, entonces eso no le da a ella notificaciones ni es algo automático ella obviamente como se trata de un Excel nada mas donde ella cuando ya le dice la gerente de área necesito reporte de graduados entonces ella ahí filtra y dice este ya está pasado y todo lo demás, pero claro si la herramienta me permite gestionar tiempos que yo digo el estudiante n/n se le notifico en octubre del 2015 y tiene hasta marzo del 2016 para que apruebe, es decir hasta abril al mes ya debería yo por lo menos tener algún tipo de notificación y registro de que ya hay esos cierres de que tiene que hacer como titulación, porque claro a nosotros como titulación nos golpea también el índice de graduación que tiene el estudiante es un índice de incidencia que uno tiene y que a nivel de ingeniera en informática abierta es bajísima, por ejemplo para los tramites de acreditación eso de ahí nos golpea hartísimo porque recuerdo que la otra vez nos sacaron recuerdo creo que teníamos el 3% de graduación era bajísimo, entonces eso a nivel de acreditación de carrera nos va a golpear luego, entonces claro eso es porque tal vez no tengo una herramienta que a mí me ayude a que yo pueda ir controlando eso, que Lidia lo tiene al Excel pero solo cuando a ella le piden reportes se acuerda ella no puede estar atrás de toditos entonces la titulación tiene automatizado eso, eso sí ayudaría mucho.

5. ¿Qué otros aspectos se debería considerar antes de proponer una solución?

Se supone que el proceso que ahora vamos a manejar tanto abierta y presencial tiene que ser el mismo, claro como le decía en presencial se tiene claramente identificado que es lo que se tiene en las dos GP pero acá vamos a utilizar el seminario de fin de carrera tal vez si es bueno que cuando se realice la aplicación se tenga anotado que actividades o que porcentajes el estudiante debe cumplir en cada uno de los niveles, hasta la 4.1 perfecto se debe entregar que digo yo la metodología al menos de la propuesta del trabajo de titulación, en la 4.2 al menos ya el 50% de desarrollo, debería tener algún tipo de tema que nos permita valorar no sé si es que será eso mucho porque claro ahí los tutores tenemos que entrar a la herramienta y poder evaluar porque hay un tema que es otro problema que se lo tiene eso es a nivel más un poco de presencial yo digo porque alguna vez también fui tutora de la 4.1,

cual es el tema, anteriormente se creía que el tutor era únicamente quien decía haber señor Director pásame el reporte de cómo está su tesista entonces nos venía el oficio diciendo que el estudiante tiene una calificación de 20/20 o 40/40 entonces nosotros como tutor ingresamos la nota y realmente el estudiante tenía un 15% o un 20% de tesis y eso es lo que siempre nos ha golpeado entonces por eso mismo ahora en cada gestión productiva hay cortes que nosotros tenemos de presentación de los proyectos que usted mismo va a tener que hacer, nosotros ya vamos viendo el desarrollo del proyecto de como el estudiante va adelantando entonces tiene que haber un corte en cada nivel que diga hasta aquí tiene que presentar el estudiante y el tutor nosotros como tutores podríamos tener en la herramienta un registro, de que quede registrado que nosotros este criterio cumplido perfecto ha cumplido este criterio y cada criterio debería tener una ponderación en porcentaje y en puntos porque ahora mismo nosotros no dejamos que el 100% de la nota sea la que nos pase el director no ahora nosotros también como tutores recibimos la nota de ellos pero también la evaluación nuestra se les ha tenido que realizar un proceso de actividades porque si no era algo ficticio eso que antes se hacía nosotros como tutores recibíamos una nota que era mentira y era el tema después nos golpeó eso mucho en ese tiempo a la escuela Ciencias de la Computación porque claro teníamos montón de estudiantes aprobado la GP 4 y toditos retirados de tesis y eso me acuerdo hace el 2008 o 2009 que hicimos una acreditación ya pequeña que nos pidieron ya el CEACES cuando empezó ya el Gobierno con esto de las universidades eso fue un verdadero problema aquí para nosotros entonces eso sí sería bueno que se limite y que cada tutor tengamos la opción de poder ir validando esos criterios para que se quede como un registro porque eso también necesita un registro, incluso por temas de acreditación los papeles se los lleva el viento pero si hay una base de datos en donde está toda esa información sería perfecto.

6. ¿Qué información considera usted importante de obtener con la herramienta/aplicación?

- Para mi es importante todo ese tema del historial del estudiante
- Segundo debería tener claro todos los reportes a nivel de presencial hay reportes por ejemplo Audrey y María del Carmen que llevan la GP en la presencial ellas tienen hojas de evaluaciones de los grupos de docentes revisores, eso tiene que estar registrado en el sistema esa valoración que se hace con observaciones y todo con fecha cuando hay la siguiente presentación, deberían tener si es que es un respaldo o deberían subirse esos documentos pero deberían estar ahí de la evaluación y de todo lo que se ha ido dando firmados si yo puedo recibir el documento de la evaluación perfecto está ahí escaneado con firmas.


Mg. María Belén Mora
DOCENTE TUTOR PRINCIPAL G.P.


José Luis Quichimbo Suárez
ESTUDIANTE

Figura 61. Reporte entrevista tutor de Gestión Productiva

Fuente: El autor

Elaboración: El autor

Reporte de Entrevista

Lic. Lady Sanmartín

SECRETARÍA TITULACIÓN DE INGENIERÍA EN SISTEMAS INFORMÁTICOS Y COMPUTACIÓN

1. ¿Cuál es el proceso oficial si existe?

Empieza por los alumnos baja el formato de la página de la universidad y presentan en servicios estudiantiles, luego nos llega a nosotros lo que tenemos que realizar un porcentaje de cada grupo de créditos, por ejemplo de genéricas y formación básica que estén el 100%, troncales en 90%, complementarias y libre en 80% y dentro de la GP haber aprobado la 3.1 y la 3.2.

Nosotros revisamos eso y pasamos al coordinador para que realice la inserción a la unidad de titulación que es el oficio, ellos manejan la plantilla para realizar el oficio,

Luego nosotros cargamos ese oficio a la herramienta de servicio estudiantiles donde adjuntamos diciendo al alumno que ya está insertado, luego de eso tenemos que revisar el tema, el ante proyecto que esté aprobado, por eso yo a ustedes no es he enviado todavía un oficio donde diga si, su tema esta aceptado, su director es tal persona, porque a mí no me han hecho llegar los ante-proyectos firmados, los ante-proyectos tienen que venir firmados por el jefe de sección y por el coordinador de titulación. Quien hace llegar el oficio es el tutor de proyecto de fin de carrera el de proyecto de titulación que se llama ahora.

Ese oficio se le grapa al ante proyecto y se lo archiva en su carpeta.

2. ¿Cómo es su participación en el proceso?

Una vez que me venga el ante proyecto realizar el oficio respectivo de aprobación de ante-proyecto para que legalmente ustedes desarrollen el tema que no hayan complicaciones después

3. ¿Qué inconvenientes ha encontrado en los procesos de postulación y seguimiento?

Inconvenientes que para la inserción algunos que no cumplen con este grupo de créditos pero ya están matriculados en proyecto 1, o sea ahí se pasó a operaciones que hasta ahora no dan ningún resultado, entiendo que hasta que el alumno desarrolle el tema se va ir igualando con los créditos por que otra forma no nos ha pasado de operaciones ya se comunicó oportunamente cuando se detectó este problema.

En cuanto a retrasos en el cronograma de postulación:

Manejan en cada sección o sea eso los docentes tiene que sacar los proyectos luego para publicarles a los estudiantes pero de eso yo ya no tengo conocimiento como hacen ellos.

4. ¿Gestiona usted todo su proceso con alguna aplicación o alguna herramienta?

Este ciclo tenemos la herramienta que la realizo el Ing. Patricio pero a más de eso donde el Ing. Revisa con el equipo de calidad y en sesión de consejo revisa su tema, entonces ahí

dice el señor José Luis Quichimbo va a tomar el tema tal, aprobado pero yo necesito además de eso el oficio, donde usted firmo un ante-proyecto le firmo su director y eso para que vaya a su carpeta.

Es un documento de Excel en línea, el portafolio de trabajos de trabajos de titulación, ese es el que estamos manejando, o sea al haber una aplicación para esto fuera perfecto.

5. ¿Con qué otros procesos o aplicaciones se relaciona el proceso?

O sea en una base que tenemos para ingresar a todos los que hayan sido aprobados, para la inserción a la UT es una base que se llama el SIEC. Este lo manejan allá en operaciones el que los capacita en esto es Santiago Ortega que es también Ing. en Sistemas o sea a más de esto tenemos la otra base.

6. En caso de implementarse la herramienta para gestionar el proceso. ¿Qué otros aspectos debería considerarse?

Al haber cómo poner una alerta cuando el alumno ya va dos meses, dos ciclos así de pronto.

7. ¿Qué información considera usted importante de obtener con la herramienta/aplicación?

Un reporte con los datos por ejemplo número de cédula, nombres completos del alumno, el tema, el director, cuando se aprobó el tema luego poner el tribunal que si esta en esta base del Ing. Patricio, eso también le decimos que una vez por ejemplo usted está desarrollando el tema se graduó una opción donde diga graduado, sí o no, en qué fecha se graduó de pronto la nota de grado.

De ahí se pueda sacar el oficio pero en este caso sería de aprobación del tema y regular en un tiempo determinado que nos hagan llegar también aquí los ante-proyectos, sería más o menos de establecer un formato o modelo de oficio para ya bajarlo de ahí mismo ingresando los demás datos


Lic. Lady Sanmartín
SECRETARÍA TITULACIÓN




José Luis Quichimbo Suárez
ESTUDIANTE

Figura 62. Reporte entrevista secretaria titulación Sistemas informáticos y computación

Fuente: El autor

Elaboración: El autor

Reporte de Entrevista

Lic. Lidia Villacís

SECRETARIA

TITULACIÓN INGENIERÍA EN INFORMÁTICA

1. ¿Cuál es el proceso oficial si existe?

En los centros se encargan de receptor las propuestas de trabajos de titulación porque los estudiantes no pueden hacerlo en línea, entonces dejan la petición hay el formato todo eso para que ellos sólo llenen, pero a veces hay confusión en los centros, no está claro, no les indican bien a los estudiantes entonces toca devolver al momento que llega a secretaría, o bien el momento en que suben en la herramienta trámites, porque ahí nos suben toda esa información, nosotros revisamos si está todo bien bajamos, vemos si está egresado.

Entonces le tenemos listo al coordinador en una base que el mismo nos dio que hagamos que llenemos, le pasamos a él luego nos reunimos.

Luego tenemos que llenar kardex, aunque no hemos llenado kardex hace 2 años y medio porque dijeron que del sistema iba a salir el kardex, que consiste en el expediente del estudiante, eso nos permite a nosotros ver si el estudiante ya es egresado o no, cuántas materias le faltan, los periodos en que aprueba, todo eso. Todo esto se hace manualmente y demanda mucho tiempo.

2. ¿Cómo es su participación en el proceso?

En el centro ingresan, yo adelanto viendo la herramienta, bajando el adjunto yo ya voy adelantando haciendo el cuadro para el Ing. Patricio, posiblemente nos reunimos, se hace el oficio, lo escaneamos, y en la herramienta mismo le contestamos adjuntando ese documento al estudiante.

Dependiendo de los casos mis tareas van:

- Si pide inserción a la Unidad de Titulación Especial (UTE) y que está desarrollando el tema de tesis y prórroga: entonces tengo que ver si el estudiante ha terminado ya la malla para poderle informar al ingeniero.
- Otros piden únicamente inserción a la Unidad de Titulación son de STS: yo todavía manejo lo que son asignaturas, por lo tanto ahí si se considera egresados mientras que en STS no tenemos la calidad de egresados. Entonces los que piden inserción a la UTE, más el proyecto de tesis tienen que estar egresados.
- O el examen complejo: revisar que tienen que estar egresados.

3. ¿Qué inconvenientes ha encontrado en los procesos de postulación y seguimiento?

- Que no registran bien en los centros el caso de cada alumno, no en todos.
- A cada base de datos se repiten muchas cosas.

4. ¿Cuáles cree que serían las pautas para resolver estos inconvenientes?

- Que los centros estén súper capacitados, no todos, pero en algunos como Quito por ejemplo, en temporadas altas se contrata personal de apoyo y no saben bien el proceso, entonces reciben como llegan las cosas. Entonces que el personal de planta maneje bien todas las cosas.

5. ¿Qué otros aspectos se debería considerar antes de proponer una solución?

Deberían unirse todas las bases y esos subsistemas y hacer uno sólo, porque eso es lo que nos quita tiempo, el tener que llenar una base, luego otra base, con el 50% de la una se llena la otra, y así con cada una. Entonces eso quita tiempo.

6. ¿Con qué otros procesos o aplicaciones se relaciona el proceso?

- La herramienta trámites, porque a través de eso nos llega el documento con la petición del alumno. Si es inserción a la UTE, prórroga u otro según el formato.
- En un Excel, donde ingresamos el número de cédula, si es Sr. Sra. O señorita, si tiene tema de tesis, que porcentaje de avance tiene, cuál es el director de tesis, todo esto. Este documento es personal no está compartido.
- Del sistema académico que se registran las notas y de allí vamos sacando kardex.
- Tenemos otra base de datos inmensa, dónde tenemos que registrar cuando le aceptamos la inserción, cuál es el tema, cuál es el director de tesis, si es complejo o tesis, el avance de la tesis. Esta base de datos está dentro del sistema SIEC.
- También tenemos que registrar en otra base de exalumnos, cuáles son egresados, cuáles son graduados, etc. Aunque aquí no van los de inserción en la UTE.

7. En caso de implementarse la herramienta para gestionar el proceso. ¿Qué otros aspectos debería considerarse?

- Para mí debería tener relación con el sistema académico, si el alumno hace la petición que quiere dar el examen complejo, el sistema automáticamente debería arrojar la información si está en condiciones de que sí o que no.

8. ¿Qué información considera usted importante de obtener con la herramienta/aplicación?

- Tendría que estar en línea para que cuando el estudiante ingrese la petición ya le arroje el resultado, si puede o no puede. El momento que el sistema dice que si puede acceder a la inserción más el examen complejo ya puede presentar la petición. Eso nos ayuda para nosotros no tener que revisar kardex, no tenemos que revisar nada más. Con eso gana el estudiante, ganamos nosotros y gana el coordinador porque todo estaría bloqueado, los filtros van ayudando, y al coordinador sólo le llegan los casos en que sí puede aplicar.



Figura 63. Reporte entrevista secretaria titulación Ingeniería Informática

Fuente: El autor

Elaboración: El autor

ANEXO 4. Caso de uso global

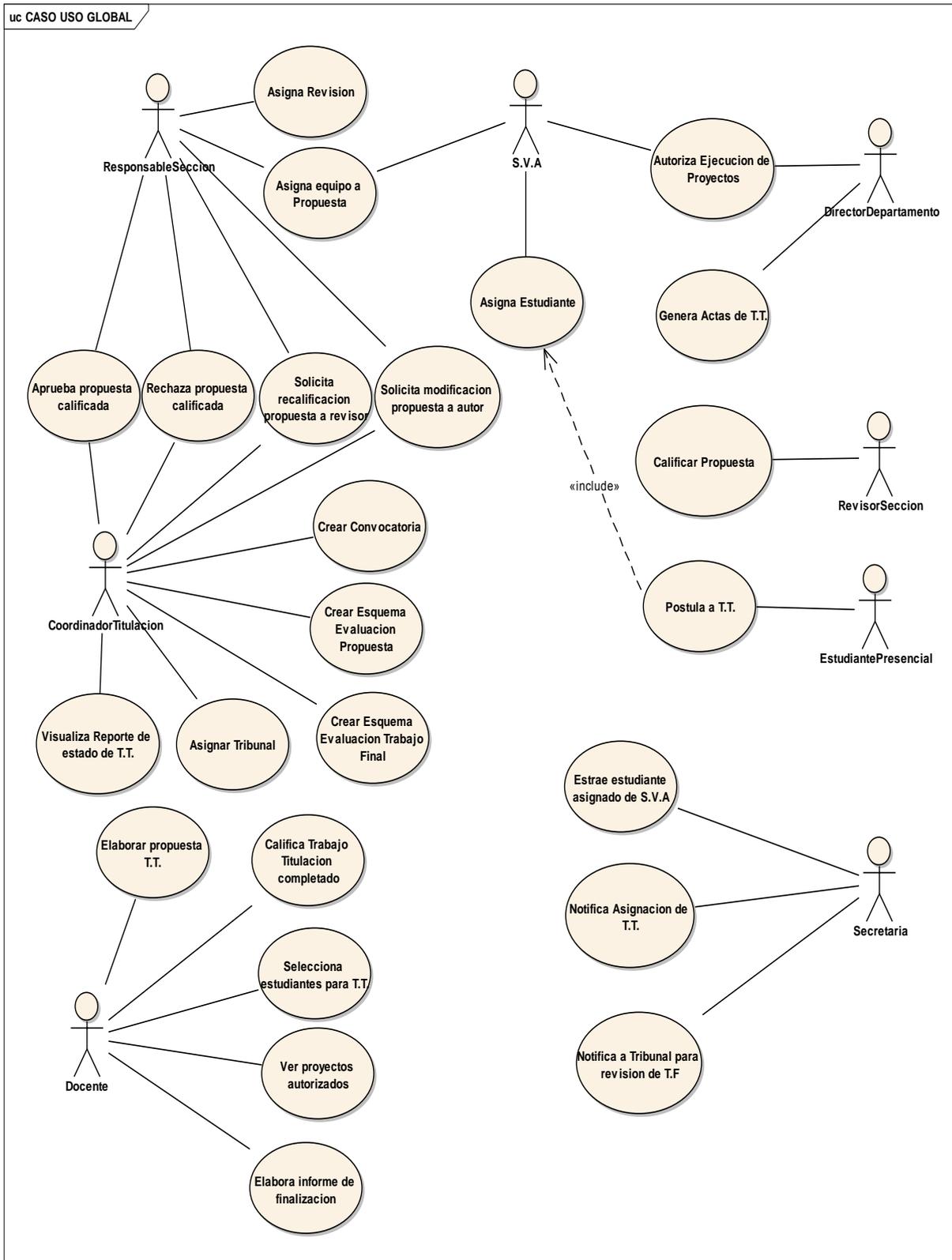


Figura 64. Caso de uso
 Fuente: Diagramación UML.
 Elaboración: El autor.

ANEXO 5. Especificación de casos de uso.

CU001. Crear convocatoria

Especificación de caso de uso: Crear convocatoria

1. Crear convocatoria

1.1 Descripción

El Coordinador Titulación crea una Convocatoria al iniciar un periodo académico, para definir los plazos para las actividades (Propuesta, Revisión, Validación, Autorización, Postulación) correspondientes a la postulación de propuestas para trabajos de titulación.

Esta Convocatoria habilita la creación de esquemas de evaluación (rúbrica).

2. Flujo de Eventos

2.1 Flujo Básico

1. El Coordinador Titulación ingresa a la opción: Convocatorias – Nueva.
2. El Coordinador Titulación ingresa la información correspondiente a la convocatoria: título, titulación, descripción, activa, responsable, plazo para propuesta, plazo para revisión, plazo para validación, plazo para autorización, plazo para postulación.
3. Coordinador Titulación llena el formulario.
4. El sistema guarda convocatoria con las actividades.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla Inicial

El Coordinador Titulación puede acceder al formulario desde su pantalla de inicio: Inicio – Nueva Convocatoria.

2.2.2 Ya existe una convocatoria activa para la titulación

El sistema presenta una alerta cuando ya se encuentra una convocatoria activa para la titulación seleccionada.

2.2.3 Usuario no pertenece a titulación

El sistema no contiene información sobre la titulación o titulaciones a las que pertenece el usuario. No permite guardar convocatoria sin este campo.

3. Requerimientos especiales

3.1.1 Rangos de fechas

En las fechas plazo de las actividades, la “Fecha Hasta” no debe ser menor a la “Fecha Desde”.

4. Precondiciones

- 4.1 El usuario debe estar autenticado en el sistema
- 4.2 El usuario debe tener el rol "CoordinadorTitulacion".

5. Post condiciones

- 5.1 Se permiten crear Esquemas de evaluación
- 5.2 Se notifica a docentes la creación de una nueva convocatoria.

6. Puntos de Extensión

- 6.1 Nuevo esquema de evaluación inicial para calificación de propuestas.

CU002. Crear esquema evaluación propuesta

Especificación de caso de uso: Crear esquema evaluación propuesta

1. Crear esquema evaluación propuesta

1.1 Descripción

El CoordinadorTitulacion crea un esquema de evaluación (rúbrica) para definir los ítems para evaluar una propuesta subida por un docente.

Este EsquemaEvaluacionPropuesta será usado por los revisores de sección para calificar una propuesta.

2. Flujo de Eventos

2.1 Flujo Básico

1. El CoordinadorTitulacion ingresa a: Esquemas Evaluación – Esquema Propuesta.
2. El CoordinadorTitulacion ingresa la información correspondiente al nuevo esquema de evaluación de propuesta: convocatoria, nombre, comentario, calificación máxima, calificación mínima, ítems, agregar nuevos ítems.
3. Al seleccionar cada ítem el sistema presenta los campos para definir un criterio y porcentaje de la nota máxima para cada calificación cualitativa (deficiente, regular, bueno, excelente) y un peso que tendrá el ítem con respecto al esquema de evaluación.
4. El sistema permite agregar nuevos ítems para evaluar una propuesta, y ofrece la posibilidad de agregar estos ítems como un nuevo campo requerido para subir una nueva propuesta por un docente.
5. El sistema guarda el EsquemaEvaluacionPropuesta con sus ítems.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla Inicial

El CoordinadorTitulacion puede acceder al formulario desde su pantalla de inicio: Inicio – Nuevo Esquema Propuesta.

2.2.2 No existen convocatorias.

Cuando no existen convocatorias creadas el sistema no permite la creación de esquema de evaluación para propuesta.

3. Requerimientos especiales

3.1.1 Porcentaje excedido de 100 en nota cualitativa.

El sistema no debe permitir que el usuario ingrese un porcentaje superior a 100% para cada porcentaje de cada nota cualitativa.

3.1.2 Calificación cuantitativa en base a porcentaje

Se debe calcular la nota cuantitativa según el porcentaje asignado para cada nota cualitativa con respecto a la nota máxima, para que si posteriormente el usuario modifica el esquema de evaluación y modifica la nota máxima el sistema calcule el nuevo valor cuantitativo.

3.1.3 Peso total de esquema diferente de 100%

El sistema no debe permitir guardar un esquema si la sumatoria de los pesos de sus ítems no es igual a 100.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "CoordinadorTitulacion".

4.3 Debe existir una convocatoria creada para las titulaciones a las que pertenece el usuario.

5. Post condiciones

5.1 Se permiten subir nuevas propuestas.

6. Puntos de Extensión

6.1 Nueva propuesta para trabajo de titulación.

CU003. Crear esquema evaluación trabajo final

Especificación de caso de uso: Crear esquema evaluación trabajo final

1. Crear esquema evaluación trabajo final

1.1 Descripción

El CoordinadorTitulacion crea un esquema de evaluación (rúbrica) para definir los ítems para evaluar un trabajo de titulación en estado “finalizado” por un estudiante.

Este esquema será usado por el presidente de tribunal para calificar un trabajo de titulación.

2. Flujo de Eventos

2.1 Flujo Básico

1. El CoordinadorTitulacion ingresa a: Esquemas Evaluación – Esquema T. Final.

2. El CoordinadorTitulacion ingresa la información correspondiente al nuevo esquema de evaluación para trabajo final: convocatoria, nombre, comentario, calificación máxima, calificación mínima, aspectos y agregar nuevos aspectos.

3. Cada aspecto contiene dos campos: aspecto y ponderación.

4. El sistema permite agregar nuevos aspectos para evaluar un trabajo de titulación finalizado. El sistema guarda el esquema de evaluación para propuesta con sus ítems.

2.2 Flujos Alternativos

2.2.1 No existen convocatorias.

Cuando no existen convocatorias creadas el sistema no permite la creación de esquema de evaluación para trabajo finalizado.

3. Requerimientos especiales

3.1.1 Aspectos agrupados

Los aspectos deben estar agrupados en dos grupos: Teórico – Técnico y Metodológico

3.1.2 Ponderación por grupo de aspectos diferente de 100%

El sistema no debe permitir guardar un esquema si la sumatoria de las ponderaciones de un grupo no es igual a 100.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol “CoordinadorTitulacion”.

4.3 Debe existir una convocatoria creada para las titulaciones a las que pertenece el usuario.

5. Post condiciones

5.1 Se permite a presidente de tribunal calificar un trabajo de titulación finalizado.

6. Puntos de Extensión

6.1 Nueva calificación para trabajo de titulación finalizado.

CU004. Elaborar propuesta de trabajo de titulación.

Especificación de caso de uso: Elaborar propuesta de trabajo de titulación.

1. Elaborar propuesta de trabajo de titulación

1.1 Descripción

El Docente elabora una Propuesta para trabajo de titulación según la plantilla definida por la titulación por la universidad.

2. Flujo de Eventos

2.1 Flujo Básico

1. El Docente ingresa a: Propuestas – Nueva.
2. El Docente ingresa la información correspondiente a la nueva propuesta: periodo, tema, plazo, número de estudiantes, perfil del estudiante, línea estratégica, sección , línea de investigación, descripción, justificación, propuesta de la que deriva, objetivo general, objetivos específicos.

2.2 Flujos Alternativos

2.2.1 No existe convocatoria activa.

Cuando no existen convocatorias creadas, el sistema no permite la elaboración de propuestas.

2.2.2 No existe esquema de evaluación activo.

Cuando no existen esquemas de valuación para propuesta creados, el sistema no permite la elaboración de propuestas.

2.2.3 Existen varias convocatorias activas.

Cuando existen varias convocatorias activas, el sistema no permite la elaboración de propuestas.

2.2.4 Existen varios esquemas de evaluación activos.

Cuando existen varios esquemas de evaluación activos, el sistema no permite la elaboración de propuestas.

2.2.5 El plazo para postulación ha finalizado.

Cuando el plazo para postulación definido en la creación de una convocatoria, el sistema no permite la elaboración de propuestas.

3. Requerimientos especiales

3.1.1 Líneas de investigación según sección

Las líneas de investigación deben pertenecer a una sección y mostrarse sólo las de la sección seleccionada.

4. Precondiciones

- 4.1 El usuario debe estar autenticado en el sistema
- 4.2 El usuario debe tener el rol "Docente".
- 4.3 Debe existir una convocatoria creada para las titulaciones a las que pertenece el usuario.
- 4.4 Debe existir un esquema de evaluación para propuesta creado para la convocatoria activa.
- 4.5 La fecha de elaboración debe estar dentro del plazo para postulación definido en la convocatoria activa.

5. Post condiciones

- 5.1 Se genera una Propuesta en estado "Elaborada".
- 5.2 Aparece una nueva Propuesta en el listado de asignaciones a ResponsableSeccion.
- 5.3 Se habilita la asignación de revisores a ResponsableSeccion.

6. Puntos de Extensión

Nueva asignación de revisores a propuesta.

CU005. Asignar revisión

Especificación de caso de uso: Asignar revisión

1. Asignar revisión

1.1 Descripción

El ResponsableSeccion asigna uno o varios revisores para que califiquen una propuesta subida a su Sección por un Docente.

2. Flujo de Eventos

2.1 Flujo Básico

1. El ResponsableSeccion ingresa a: Propuestas – Asignar Revisión.
2. El sistema muestra un listado de las propuestas que los usuarios con el rol Docente han subido y pertenecen a su Sección.
3. El ResponsableSeccion selecciona una propuesta del listado de propuestas subidas de su sección y selecciona “Asignar revisión” en una propuesta.
4. El ResponsableSeccion selecciona uno o varios revisores y asigna la revisión.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla Inicial

El ResponsableSeccion puede acceder a la asignación desde su pantalla de inicio:

Inicio – Nueva Revisión.

2.2.2 El plazo para revisión ha finalizado.

Cuando el plazo de revisión de la convocatoria con la que fue ofertada la propuesta ha finalizado, el sistema no permite crear la asignación de revisores.

2.2.3 La propuesta ya ha tenido una asignación.

Si la propuesta ya ha sido asignada a un revisor o revisores, el sistema ya no permite la asignación de revisores.

Para modificar la asignación el Responsable debe ingresar a Revisiones – Solicitadas, buscar la propuesta y seleccionar editar revisores.

3. Requerimientos especiales

3.1.1 Mostrar nombres completos de docentes con el rol RevisorSeccion.

El sistema debe mostrar los nombres completos de los docentes que tengan el rol RevisorSeccion.

3.1.2 Mostrar docentes RevisorSeccion de la titulación

El sistema debe filtrar a los docentes con el rol que pertenezcan RevisorSeccion a la titulación de la propuesta.

3.1.3 Mostrar el número de asignaciones que ya tiene cada RevisorSeccion.

El sistema debe mostrar el número de asignaciones que ya tiene cada RevisorSeccion para evitar asignar demasiadas revisiones a un solo docente RevisorSeccion.

3.1.4 Mostrar enlace a la visualización de la Propuesta.

El sistema debe mostrar un enlace para poder visualizar todos los campos de la Propuesta.

3.1.5 Excluir a ResponsableSeccion del listado de revisores si tiene también rol de Revisor.

Si el ResponsableSeccion, también tiene el rol de Revisor, el sistema debe excluirlo del listado de revisores para que no pueda realizarse un auto asignación.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "ResponsableSeccion".

4.3 La fecha actual debe estar dentro del plazo para revisión definido en la convocatoria activa.

5. Post condiciones

5.1 Se crea una nueva revisión que pertenece a la propuesta.

5.2 Los revisores asignados tienen una nueva propuesta a calificar.

5.3 Se habilita la calificación de propuesta.

6. Puntos de Extensión

Nueva propuesta a calificar.

CU006. Calificar Propuesta

Especificación de caso de uso: Calificar Propuesta

1. Calificar Propuesta

1.1 Descripción

El RevisorSeccion califica una Propuesta según el EsquemaEvaluacionPropuesta definido para la Convocatoria en la que se ofertó la Propuesta.

2. Flujo de Eventos

2.1 Flujo Básico

1. El RevisorSeccion ingresa a: Revisiones – Calificar Propuestas.
2. El sistema muestra un listado de las propuestas dónde un ResponsableSeccion ha asignado a su usuario como revisor.
3. El RevisorSeccion selecciona una propuesta del listado de propuestas subidas de su sección y selecciona “Calificar” en una propuesta.
4. El sistema muestra el detalle de la propuesta , la calificación total (0 inicialmente), a continuación los Ítems del EsquemaEvaluacionPropuesta activo perteneciente a la convocatoria en la que se ofertó la Propuesta y al final un estado seleccionable:
 - Aprobada
 - Requiere Modificaciones
 - Rechazada

Cada Ítem contiene:

- Ítem: Nombre del Ítem
- Calificación: Calificación cualitativa seleccionable: Deficiente, regular, bueno, excelente.
- Descripción: Criterio según el revisor seleccione la calificación.
- Valor: Valor cuantitativo según la calificación cuantitativa seleccionada.
- Peso: Peso definido en EsquemaEvaluacionPropuesta para el ítem.

- Calificación: Calificación del ítem calculada sacando el porcentaje que le corresponde al valor cuantitativo según el peso.
 - Comentario: Un comentario que puede emitir el revisor sobre el ítem.
5. El RevisorSeccion selecciona la calificación para cada Ítem que considere pertinente.
 6. El sistema muestra el criterio de la calificación seleccionada según el EsquemaEvaluacionPropuesta y calcula la calificación del ítem y la calificación total sumando la calificación individual de cada ítem.
 7. El RevisorSeccion selecciona el estado que considere para la revisión.
 8. El RevisorSeccion termina y registra su calificación.
 9. El sistema guarda la calificación que pertenece a una revisión.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla Inicial

El RevisorSeccion puede acceder a la asignación desde su pantalla de inicio:

Inicio – Nueva Calificación.

2.2.2 El plazo para revisión ha finalizado.

Cuando el plazo de revisión de la convocatoria con la que fue ofertada la propuesta ha finalizado, el sistema no permite calificar una propuesta.

2.2.3 La propuesta ya ha sido calificada.

Si la propuesta ya ha sido calificada por el revisor, el sistema ya no permite calificar nuevamente, a menos que el CoordinadorTitulacion o ResponsableSeccion soliciten una recalificación.

Para solicitar recalificación el Responsable o el Coordinador deben ingresar a Revisiones – Solicitadas, buscar la propuesta, visualizar la revisión y solicitar recalificación.

3. Requerimientos especiales

3.1.1 Calificación de ítem en tiempo real.

El sistema debe extraer en tiempo real el criterio y el valor cuantitativo del ítem que seleccione el revisor, y mostrar la calificación que le corresponde al ítem.

3.1.2 Calificación total en tiempo real

Por cada ítem que el revisor vaya calificando, el sistema debe ir sumando y mostrar la calificación total.

3.1.3 Promediar según el número de revisores.

El sistema debe promediar la calificación de la revisión según el número de revisores asignados.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "RevisorSeccion".

4.3 La fecha actual debe estar dentro del plazo de revisión de la convocatoria a la que pertenece la propuesta.

4.4 La propuesta debe estar en estado de "Elaborada".

5. Post condiciones

5.1 Se crea una nueva CalificacionPropuesta que pertenece a la Revisión.

5.2 ResponsableSeccion y CoordinadorTitulacion pueden visualizar la calificación en la sección "Revisiones – Solicitadas".

5.3 Cuando todos los revisores emiten su calificación la propuesta pasa a estado de "Calificada".

6. Puntos de Extensión

Visualización de revisiones.

CU007. Aprobar propuesta calificada

Especificación de caso de uso: Aprobar propuesta calificada – Responsable Sección

1. Aprobar propuesta calificada – Responsable Sección

1.1 Descripción

ResponsableSeccion aprueba una Propuesta que ha pasado el proceso de Revisión y tiene la calificación de todos los revisores asignados.

2. Flujo de Eventos

2.1 Flujo Básico

1. ResponsableSeccion ingresa a: Revisiones – Solicitadas.

2. El sistema muestra un filtro para facilitar la búsqueda de Revisiones. El filtro permite buscar por:

- Estado: estado de la revisión (Revisada, Parcialmente revisada o Sin revisar).
- Aprobado: Si la propuesta ha sido aprobada o aún no tiene aprobación.
- Fecha solicitud: Fecha en que se solicitó la revisión.
- Titulación: Titulaciones a las que pertenece el usuario.
- Convocatoria: Convocatorias en las que se han solicitado revisiones.

3. ResponsableSeccion selecciona una revisión del listado de propuestas subidas de su sección y selecciona “Ver”.

4. El sistema muestra el detalle de la Revisión:

- Calificaciones de revisores: Id, revisor, fecha de revisión, calificación, estado, ítems con su nombre, calificación cualitativa, valor cuantitativo y comentario emitido.
- Acciones sobre la revisión: Aprobar, Recalificar, Modificar Propuesta, Rechazar.
- Detalle de la propuesta.

5. ResponsableSeccion selecciona “Aprobar”.

6. El sistema presenta una alerta de confirmación de aprobación.

7. ResponsableSeccion guarda la aprobación.

8. El sistema guarda la aprobación de ResponsableSeccion.

2.2 Flujos Alternativos

2.2.1 Aprobación desde listado.

Luego de filtrar los datos, el sistema permite aprobar una revisión directamente desde el listado.

2.2.2 No se han realizado calificaciones de la propuesta.

Cuando los revisores aún no realizan ninguna calificación el sistema emite un mensaje de alerta informando que aún no se ha realizado ninguna revisión.

2.2.3 Faltan revisiones por parte de revisores.

Cuando el número de revisiones efectuadas es menor al número de revisores asignados, el sistema no permite ejecutar ninguna de las acciones cuando aún no hay revisado todos los revisores.

2.2.4 Promedio de calificación menor a calificación mínima.

Si el promedio de calificaciones de los revisores es menor a la calificación mínima, el sistema no permite realizar la aprobación, en lugar de la opción muestra un mensaje informando que no es posible realizar la aprobación.

3. Requerimientos especiales

3.1.1 Pedir confirmación de aprobación.

El sistema debe mostrar un mensaje de confirmación antes de guardar la aprobación de ResponsableSeccion.

3.1.2 Deshabilitar aprobación.

El sistema debe deshabilitar la aprobación si los todos los revisores aún no han emitido su calificación.

3.1.3 Deshabilitar acciones después de aprobación.

El sistema debe deshabilitar las acciones cuando el usuario ya ha aprobado la revisión.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "ResponsableSeccion".

4.3 La propuesta debe estar en estado de "Calificada".

5. Post condiciones

5.1 El sistema guarda la aprobación de ResponsableSeccion

5.2 La propuesta pasa a un estado de "Pertinente".

6. Puntos de Extensión

Aprobar propuesta calificada – Coordinador de Titulación.

CU008. Aprobar propuesta calificada

Especificación de caso de uso: Aprobar propuesta calificada – Coordinador Titulación

1. Aprobar propuesta calificada – Coordinador Titulación

1.1 Descripción

CoordinadorTitulacion aprueba una Propuesta que ha pasado el proceso de Revisión, tiene la calificación de todos los revisores asignados y tiene la aprobación de ResponsableSeccion.

2. Flujo de Eventos

2.1 Flujo Básico

1. CoordinadorTitulacion ingresa a: Revisiones – Solicitadas.

2. El sistema muestra un filtro para facilitar la búsqueda de Revisiones. El filtro permite buscar por:

- Estado: estado de la revisión (Revisada, Parcialmente revisada o Sin revisar).
- Aprobado: Si la propuesta ha sido aprobada o aún no tiene aprobación.
- Fecha solicitud: Fecha en que se solicitó la revisión.
- Titulación: Titulaciones a las que pertenece el usuario.

- Convocatoria: Convocatorias en las que se han solicitado revisiones.
3. CoordinadorTitulacion selecciona una revisión del listado de propuestas subidas de su sección y selecciona “Ver”.
 4. El sistema muestra el detalle de la Revisión:
 - Calificaciones de revisores: Id, revisor, fecha de revisión, calificación, estado, ítems con su nombre, calificación cualitativa, valor cuantitativo y comentario emitido.
 - Acciones sobre la revisión: Aprobar, Recalificar, Modificar Propuesta, Rechazar.
 - Detalle de la propuesta.
 5. CoordinadorTitulacion selecciona “Aprobar”.
 6. El sistema presenta una alerta de confirmación de aprobación.
 7. CoordinadorTitulacion guarda la aprobación.
 8. El sistema guarda la aprobación de CoordinadorTitulacion.

2.2 Flujos Alternativos

2.2.1 Aprobación desde listado.

Luego de filtrar los datos, el sistema permite aprobar una revisión directamente desde el listado.

2.2.2 No se han realizado calificaciones de la propuesta.

Cuando los revisores aún no realizan ninguna calificación el sistema emite un mensaje de alerta informando que aún no se ha realizado ninguna revisión.

2.2.3 Faltan revisiones por parte de revisores.

Cuando el número de revisiones efectuadas es menor al número de revisores asignados, el sistema no permite ejecutar ninguna de las acciones cuando aún no hay revisado todos los revisores.

2.2.4 Promedio de calificación menor a calificación mínima.

Si el promedio de calificaciones de los revisores es menor a la calificación mínima, el sistema no permite realizar la aprobación, en lugar de la opción muestra un mensaje informando que no es posible realizar la aprobación.

2.2.5 ResponsableSeccion no ha aprobado la revisión.

Si el ResponsableSeccion aún no ha realizado la aprobación, el sistema no permite aprobar la revisión.

3. Requerimientos especiales

3.1.1 Pedir confirmación de aprobación.

El sistema debe mostrar un mensaje de confirmación antes de guardar la aprobación de ResponsableSeccion.

3.1.2 Deshabilitar aprobación.

El sistema debe deshabilitar la aprobación si el ResponsableSeccion no ha realizado aún su aprobación.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "CoordinadorTitulacion".

4.3 La propuesta debe estar en estado de "Pertinente".

5. Post condiciones

5.1 El sistema guarda la aprobación de ResponsableSeccion

5.2 La propuesta pasa a un estado de "Válido".

6. Puntos de Extensión

Autorización de propuestas por Consejo de Departamento.

CU009. Autorizar ejecución de proyectos

Especificación de caso de uso: Autorizar ejecución de proyectos

1. Autorizar ejecución de proyectos

1.1 Descripción

DirectorDepartamento autoriza las propuestas que han pasado el proceso de revisión y aprobación por parte de ResponsableSeccion y CoordinadorTitulacion, y envía estas propuestas al Sistema de Vicerrectorado Académico (S.V.A.).

2. Flujo de Eventos

2.1 Flujo Básico

1. DirectorDepartamento ingresa a: Revisiones – Aprobadas.
2. El sistema muestra un filtro para facilitar la búsqueda de Revisiones - Aprobadas. El filtro permite buscar por: Convocatoria y Fecha de Aprobación.
3. DirectorDepartamento selecciona los filtros y realiza la búsqueda.
4. El sistema muestra un listado de Propuestas revisadas válidas con los atributos: Subida por, Tema, Revisada por, Promedio Calificación, Equipo, Tribunal.
5. DirectorDepartamento marca las propuestas que desee autorizar y enviar al S.V.A.
6. El sistema muestra una mensaje de confirmación detallando las propuestas que se van a enviar al S.V.A.
7. DirectorDepartamento selecciona “Autorizar”.
8. El sistema autoriza las propuestas y envía las propuestas a un servicio REST consumible por el S.V.A.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla Inicial

El DirectorDepartamento puede acceder al formulario desde su pantalla de inicio:

Inicio – Autorizar Propuestas Aprobadas

2.2.2 Propuestas ya autorizadas.

Las propuestas que ya han sido autorizadas y enviadas ya no se pueden volver a seleccionar.

2.2.3 El plazo para autorización ha finalizado.

Cuando el plazo de autorización de la convocatoria con la que fue ofertada la propuesta ha finalizado, el sistema no permite autorizar y enviar las propuestas.

3. Requerimientos especiales

3.1.1 Pedir confirmación de autorización.

El sistema debe mostrar un mensaje de confirmación con las propuestas, antes de autorizar y enviar al S.V.A.

3.1.2 Convocatorias de las titulaciones del docente.

El filtro debe mostrar sólo las convocatorias de las titulaciones a las que pertenece el DirectorDepartamento.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "DirectorDepartamento".

4.3 La propuesta debe estar en estado de "Válido".

5. Post condiciones

5.1 El sistema guarda la autorización de DirectorDepartamento

5.2 El sistema guarda las propuestas como consumibles para que el S.V.A pueda hacer la postulación de estudiantes.

5.3 La propuesta pasa a un estado de "Postulada".

6. Puntos de Extensión

Asignación de equipo a propuesta por ResponsableSeccion.

CU010. Asignar equipo a propuesta

Especificación de caso de uso: Asignar equipo a propuesta

1. Asignar equipo a propuesta

1.1 Descripción

ResponsableSeccion asigna equipo de acompañamiento (Director y docentes) a propuesta “postulada”.

2. Flujo de Eventos

2.1 Flujo Básico

1. ResponsableSeccion ingresa a: Revisiones - Aprobadas.
2. El sistema muestra un filtro para facilitar la búsqueda de Revisiones - Aprobadas. El filtro permite buscar por: Convocatoria y Fecha de Aprobación.
3. ResponsableSeccion selecciona los filtros y realiza la búsqueda.
4. El sistema muestra un listado de Propuestas revisadas válidas con los atributos: Subido por, Tema, Revisado por, Promedio Calificación, Equipo, Tribunal.
5. El sistema muestra un enlace para asignar equipo si la propuesta ha sido autorizada y enviada al S.V.A.
6. ResponsableSeccion selecciona la propuesta a asignar equipo
7. El sistema muestra los campos principales de la propuesta (Tema, sección departamental, descripción y plazo), y un formulario para asignación de equipo (Director, Equipo)
8. ResponsableSeccion selecciona los docentes y guarda el equipo
9. El sistema guarda el equipo para la propuesta.

2.2 Flujos Alternativos

2.2.1 Desde Revisiones - Solicitadas

El ResponsableSeccion puede acceder al formulario desde la sección Revisiones – Solicitadas, usar los filtros necesarios y las propuestas en estado “Postulada” permitirán la asignación de equipo.

2.2.2 Propuestas con equipo.

Las propuestas que ya tienen equipo muestran a los docentes del equipo en lugar del enlace a “asignar equipo”.

3. Requerimientos especiales

3.1.1 Extraer docentes de la titulación.

Los docentes que van a ser miembros del equipo deben ser los de la titulación de la propuesta, para que se puede elegir de diferentes secciones.

3.1.2 Nombres completos de docentes y sección.

El sistema debe mostrar los nombres completos de los docentes y la sección a la que pertenecen.

3.1.3 Preseleccionar autor como director.

El sistema debe pre seleccionar al autor de la propuesta como director de la misma.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol “ResponsableSeccion”.

4.3 La propuesta debe estar en estado de “Postulada”.

5. Post condiciones

5.1 El sistema guarda la asignación de Director y Docentes a la Propuesta.

5.2 La propuesta pasa a un estado de “Autorizada”.

6. Puntos de Extensión

Extracción de estudiantes del S.V.A. y notificación a Estudiante y Director por Secretaria.

CU011. Extraer estudiante asignado de S.V.A.

Especificación de caso de uso: Extraer estudiante asignado de S.V.A.

1. Extraer estudiante asignado de S.V.A.

1.1 Descripción

Secretaria extrae del Sistema de Vicerrectorado Académico (S.V.A.) el estudiante asignado a una propuesta.

2. Flujo de Eventos

2.1 Flujo Básico

1. Secretaria ingresa a: Trabajos de Titulación - Autorizados.

2. El sistema muestra un listado de Trabajos de Titulación - Autorizados con los atributos: Id, Tema, Subida por, Director, Equipo, Estudiante, Notificar.

3. El sistema muestra un enlace para consultar el estudiante aceptado de una propuesta en el S.V.A.

4. Secretaria selecciona "Consultar de S.V.A."

5. El sistema muestra un mensaje de estado mientras ejecuta la consulta.

6. El sistema muestra un mensaje de notificación si se encontró un estudiante aceptado.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla de Inicio

Secretaria puede acceder al listado desde Inicio-Trabajos Autorizados.

3. Requerimientos especiales

3.1.1 Extraer datos de estudiante.

El S.V.A. debe presentar los datos del estudiante: Nombres, identificador, cédula, email.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "Secretaria".

4.3 La propuesta debe estar en estado de "Autorizada".

5. Post condiciones

5.1 El sistema guarda la asignación de Estudiante a la Propuesta.

5.2 La propuesta pasa a un estado de "Ejecución".

6. Puntos de Extensión

Notificación a Estudiante y Equipo por Secretaria.

CU012. Notificación a estudiante y equipo por Secretaria.

Especificación de caso de uso: Notificación a estudiante y equipo por Secretaria.

1. Notificación a estudiante y equipo por Secretaria.

1.1 Descripción

Secretaria notifica a Estudiante, Director y Docentes miembros del equipo de acompañamiento la asignación para la ejecución de un Trabajo de Titulación.

2. Flujo de Eventos

2.1 Flujo Básico

1. Secretaria ingresa a: Trabajos de Titulación - Autorizados.
2. El sistema muestra un listado de Trabajos de Titulación - Autorizados con los atributos: Id, Tema, Subida por, Director, Equipo, Estudiante, Notificar.
3. El sistema muestra un enlace para Notificar.
4. Secretaria selecciona "Notificar".
5. El sistema guarda una notificación para mostrarla a Estudiante, Director y Docentes.
6. El sistema envía una notificación por correo electrónico a Estudiante, Director y Docentes.
7. El sistema muestra un mensaje de notificación de los usuarios que fueron notificados.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla de Inicio

Secretaria puede acceder al listado desde Inicio-Trabajos Autorizados.

3. Requerimientos especiales

3.1.1 Envío correos electrónicos.

El sistema debe consultar las direcciones de correo electrónico de los usuarios a los que se va a notificar y enviar correos electrónicos a los mismos.

3.1.2 Guardado de notificaciones

El sistema debe guardar la notificación para que cada usuario pueda visualizarla cuando ingrese al sistema.

4. Precondiciones

- 4.1 El usuario debe estar autenticado en el sistema
- 4.2 El usuario debe tener el rol "Secretaria".

4.3 La propuesta debe estar en estado de "Autorizada".

5. Post condiciones

5.1 La propuesta continúa en estado de "Ejecución".

6. Puntos de Extensión

Informe de Docente-Director y Asignación de tribunal.

CU013. Actualizar estado de propuesta de S.V.A.

Especificación de caso de uso: Actualizar estado de propuesta de S.V.A.

1. Actualizar estado de propuesta de S.V.A.

1.1 Descripción

Secretaria extrae del Sistema de Vicerrectorado Académico (S.V.A.) el estado de un Trabajo de Titulación para verificar que el trabajo ha sido terminado por el estudiante en S.V.A.

2. Flujo de Eventos

2.1 Flujo Básico

1. Secretaria ingresa a: Trabajos de Titulación - Completados.

2. El sistema muestra un listado de Trabajos de Titulación - Completados con los atributos: Id, Tema, Estado, Actualizar Estado, Presidente, Vocal 1, Vocal 2, Estudiante y Notificar.

3. El sistema muestra un enlace para consultar el estado del Trabajo de titulación en el S.V.A.

4. Secretaria selecciona "Consultar".

5. El sistema muestra un mensaje de estado mientras ejecuta la consulta.

6. El sistema muestra un mensaje de notificación con el estado del Trabajo de Titulación.

2.2 Flujos Alternativos

2.2.1 Desde Pantalla de Inicio

Secretaria puede acceder al listado desde Inicio-Trabajos Completados.

3. Requerimientos especiales

3.1.1 Extraer estado de Trabajo de Titulación.

El S.V.A. debe presentar el estado de la propuesta y el sistema actualizar el mismo a "Terminado SVA".

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "Secretaria".

4.3 La propuesta debe estar en estado de "Ejecución".

5. Post condiciones

5.1 La propuesta pasa a un estado de "Terminado en S.V.A.".

6. Puntos de Extensión

Notificación a Estudiante y Tribunal por Secretaria.

CU014. Informe de Finalización Docente - Director

Especificación de caso de uso: Informe de finalización Docente - Director

1. Informe de Docente - Director.

1.1 Descripción

Docente escribe un informe sobre el trabajo desarrollado por el estudiante.

2. Flujo de Eventos

2.1 Flujo Básico

1. Docente ingresa a: Propuestas – Mis propuestas.
2. El sistema muestra un listado de Propuestas con los atributos: Id, Tema, Estado, Subida por, Director, Convocatoria, Fecha Registro, Acciones.
3. El sistema muestra un enlace para Crear Informe.
4. Docente selecciona “Crear Informe”.
5. El sistema presenta los campos de la propuesta: Tema, Sección Departamental y Descripción, y un espacio para escribir un informe de la propuesta.
6. El sistema guarda el informe.

2.2 Flujos Alternativos

2.2.1 Desde Propuestas – Mis proyectos

Docente puede acceder al listado desde Propuestas – Mis proyectos.

3. Requerimientos especiales

4. Precondiciones

- 4.1 El usuario debe estar autenticado en el sistema
- 4.2 El usuario debe tener el rol “Docente”.
- 4.3 El usuario debe tener ser el autor o el director del Trabajo de Titulación.
- 4.4 La propuesta debe estar en estado de “Terminado S.V.A.”.

5. Post condiciones

- 5.1 La propuesta pasa a estado de “Completado”.

6. Puntos de Extensión

Asignación de Tribunal.

CU015. Asignación de Tribunal

Especificación de caso de uso: Asignación de Tribunal

1. Informe de Docente - Director.

1.1 Descripción

CoordinadorTitulacion asigna Tribunal a un Trabajo de Titulación completado.

2. Flujo de Eventos

2.1 Flujo Básico

1. CoordinadorTitulacion ingresa a: Revisiones - Aprobadas.

2. El sistema muestra un filtro para facilitar la búsqueda de Revisiones - Aprobadas. El filtro permite buscar por: Convocatoria y Fecha de Aprobación.

3. CoordinadorTitulacion selecciona los filtros y realiza la búsqueda.

4. El sistema muestra un listado de Propuestas revisadas válidas con los atributos: Subido por, Tema, Revisado por, Promedio Calificación, Equipo, Tribunal.

5. El sistema muestra un enlace para asignar Tribunal si la propuesta está en estado de "Completado".

6. CoordinadorTitulacion selecciona la propuesta a asignar Tribunal y selecciona "Asignar".

7. El sistema muestra los campos principales de la propuesta (Tema, sección departamental, descripción y plazo), y un formulario para asignación de Tribunal (Presidente, Vocal 1 y Vocal 2)

8. CoordinadorTitulacion selecciona los docentes y guarda el tribunal.

9. El sistema guarda el tribunal para la propuesta.

2.2 Flujos Alternativos

3. Requerimientos especiales

3.1.1 Preseleccionar director como vocal 2.

El sistema debe presentar ya seleccionado a director del Trabajo de Titulación como vocal 2.

3.1.2 Preseleccionar miembro de equipo como vocal 1.

El sistema debe presentar ya seleccionado a un docente del equipo de acompañamiento del Trabajo de Titulación como vocal 1.

3.1.3 Preseleccionar otro miembro de equipo como presidente.

El sistema debe presentar ya seleccionado a un docente del equipo de acompañamiento del Trabajo de Titulación como presidente.

3.1.4 Mostrar docentes de la titulación

Todos los campos seleccionables deben presentar los docentes de la titulación de la propuesta con la sección a la que pertenece.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "CoordinadorTitulacion".

4.3 La propuesta debe estar en estado de "Completado".

5. Post condiciones

5.1 La propuesta pasa a estado de "Revisión Tribunal".

6. Puntos de Extensión

Calificación de Trabajo de Titulación Completado.

CU016. Calificación de Trabajo de Titulación Completado

Especificación de caso de uso: Calificación de Trabajo de Titulación Completado

1. Calificación de Trabajo de Titulación Completado.

1.1 Descripción

Docente asignado como Presidente de un Trabajo de Titulación emite una calificación del trabajo de titulación finalizado, usando el esquema de evaluación definido por la titulación.

2. Flujo de Eventos

2.1 Flujo Básico

1. Docente ingresa a: Propuestas – Mis Proyectos.
2. El sistema muestra un listado de Propuestas con los atributos: Id, Tema, Estado, Subida por, Director, Convocatoria, Fecha Registro, Acciones.
3. El sistema muestra un enlace para Calificar.
4. Docente selecciona “Calificar”.
5. El sistema presenta los campos de la propuesta: Tema, Sección Departamental y Descripción, y los campos del esquema de evaluación para trabajo final definida por la titulación.
6. El sistema guarda la calificación del Trabajo de Titulación.

2.2 Flujos Alternativos

3. Requerimientos especiales

3.1.1 Mostrar la calificación máxima del esquema de evaluación.

El sistema debe mostrar sobre cuánto debe calificar el Docente mostrando la calificación máxima del esquema de evaluación para trabajo final.

3.1.2 El Docente no puede sobrepasar la calificación máxima.

El sistema restringir que el Docente no pueda escribir una calificación que sobrepase la calificación máxima.

3.1.3 Calcular Ponderación Otorgada por cada aspecto.

El sistema debe calcular la ponderación otorgada automáticamente después de que el Docente haya ingresado su calificación.

La ponderación otorgada por aspecto se calcula: calificación * ponderación / 100.

3.1.4 Calcular Calificación redondeada

El sistema debe calcular automáticamente el promedio de calificación de cada grupo.

3.1.5 Calcular Calificación Total

El sistema debe calcular automáticamente el promedio de calificación de los dos grupos.

4. Precondiciones

4.1 El usuario debe estar autenticado en el sistema

4.2 El usuario debe tener el rol "Docente".

4.3 El Docente debe ser el Presidente del Trabajo de Titulación.

4.4 La propuesta debe estar en estado de "Revisión Tribunal".

5. Post condiciones

5.1 La propuesta pasa a estado de "Finalizado".

6. Puntos de Extensión

Trabajo de Titulación Finalizado.

ANEXO 6. Diagramas de secuencia

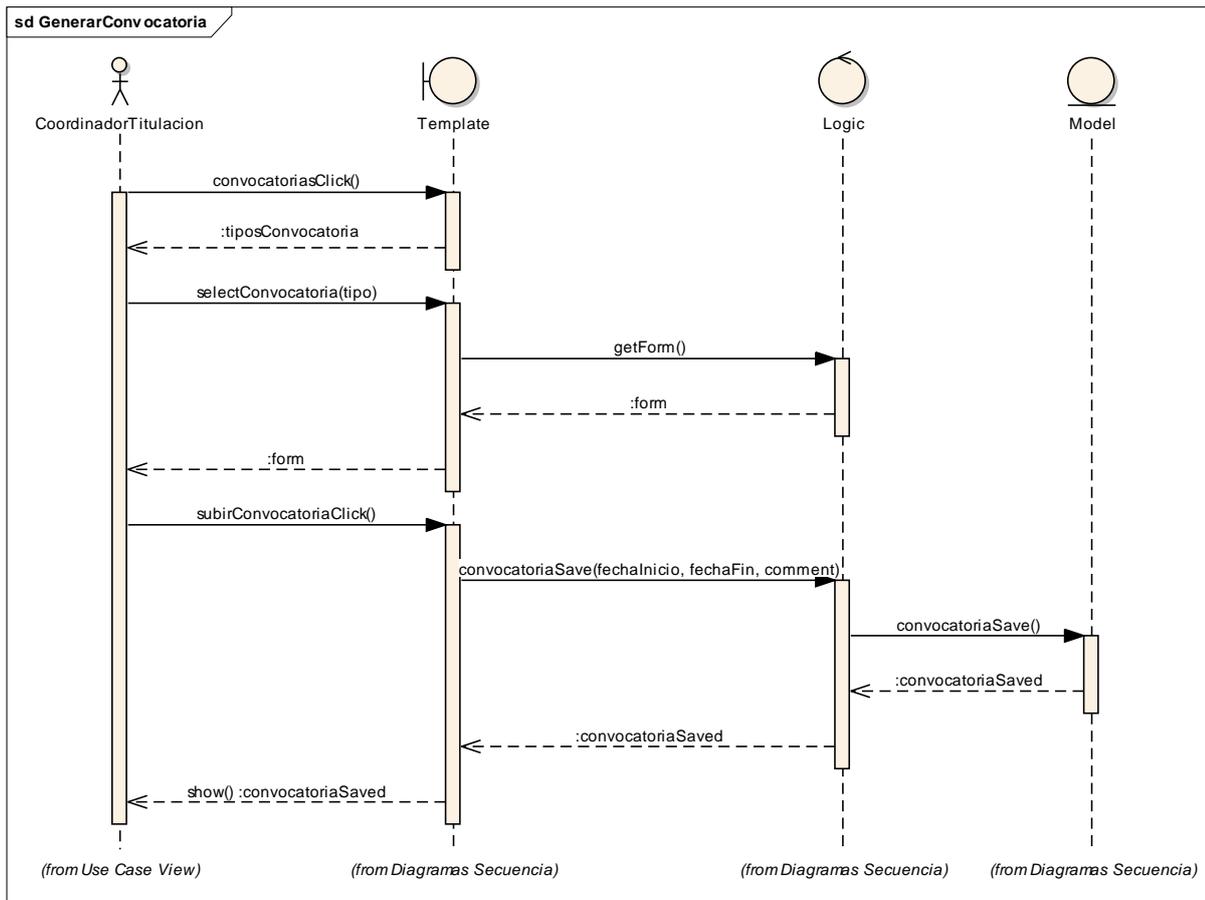


Figura 65. Diagrama de secuencia – Ofertar Convocatoria

Fuente: Diagramación UML.

Elaboración: El autor.

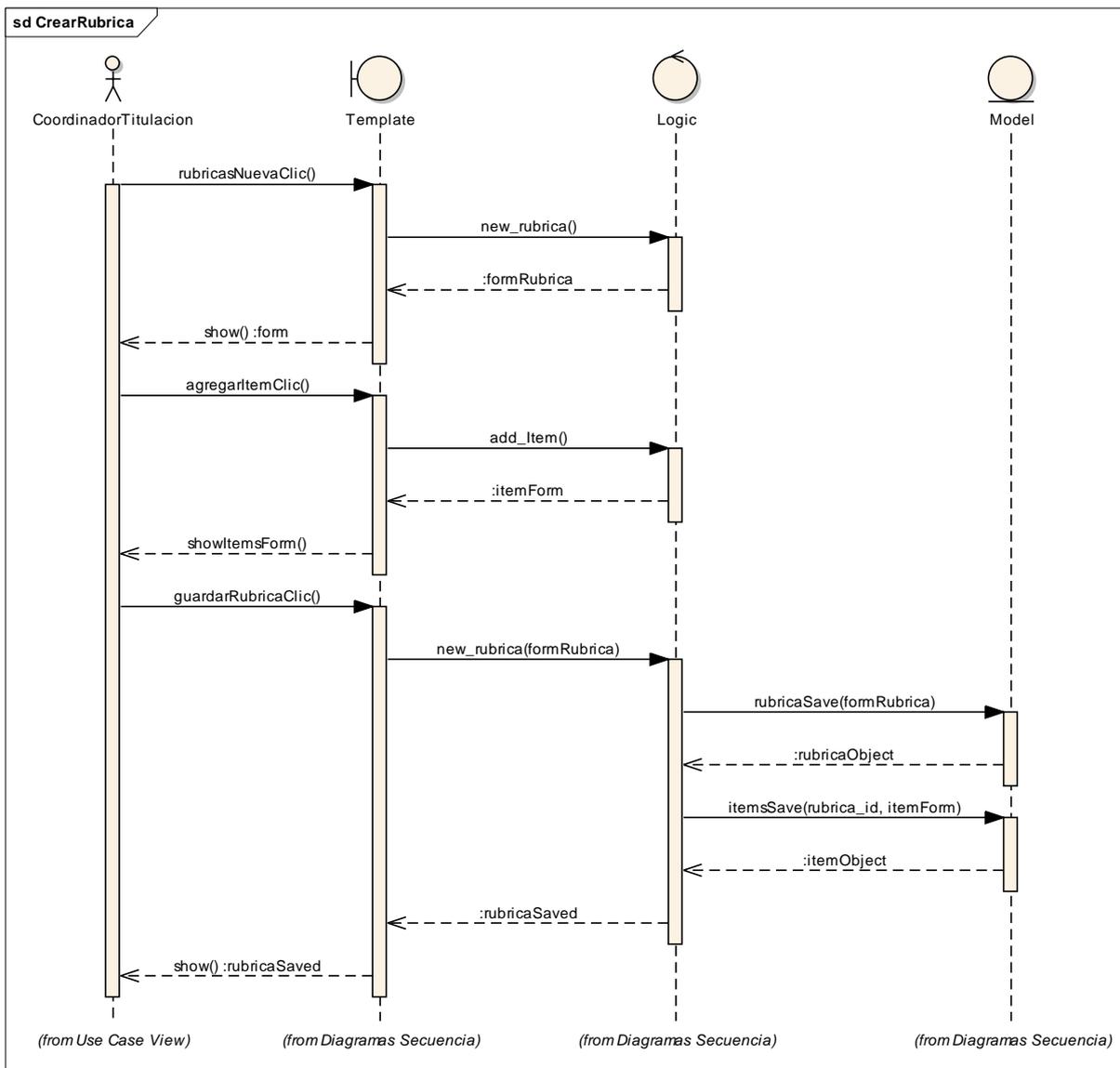


Figura 66. Diagrama de secuencia – Crear Rúbrica (Esquema Evaluación)
 Fuente: Diagramación UML.
 Elaboración: El autor.

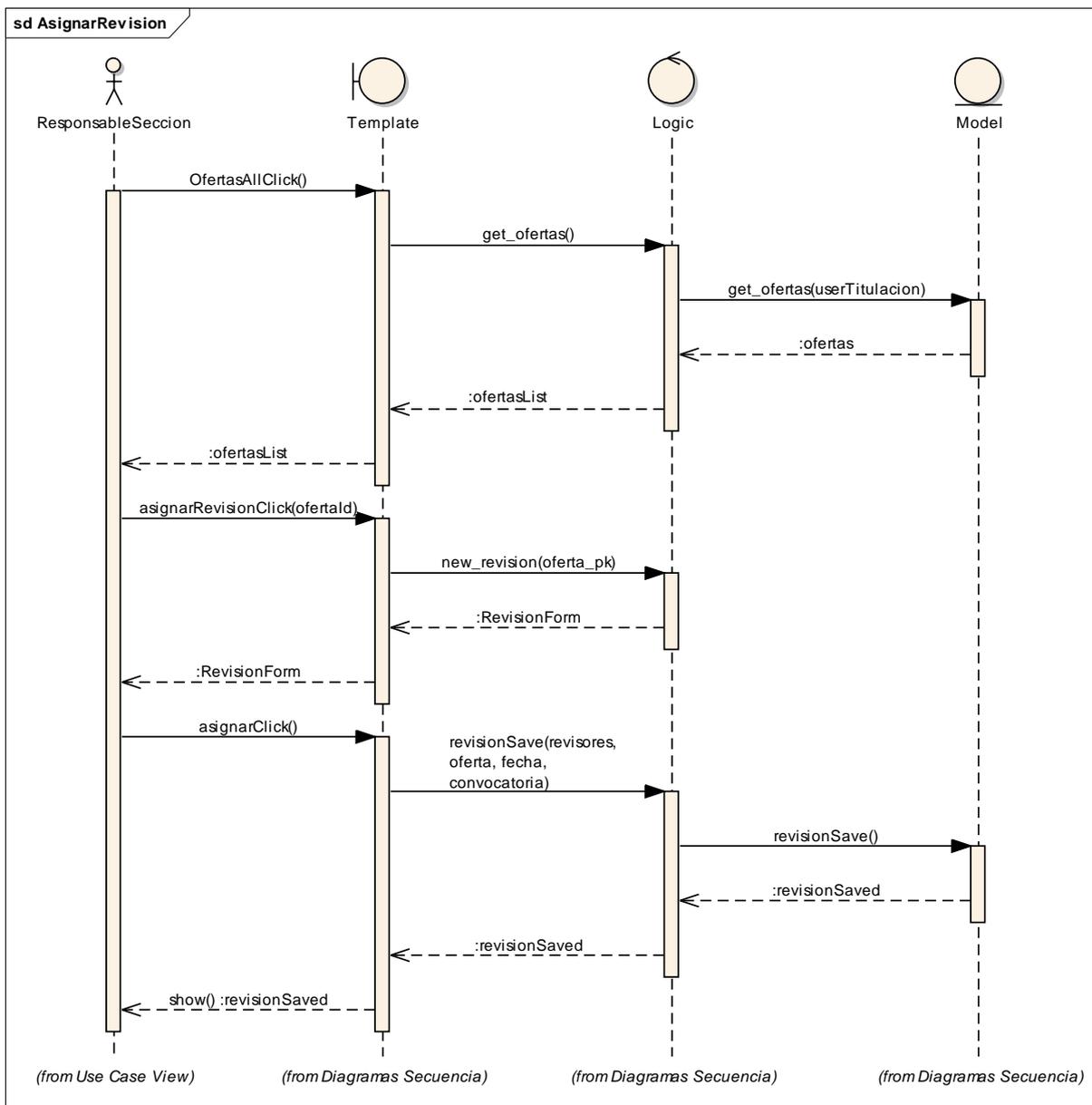


Figura 67. Diagrama de secuencia – Asignar Revisión
 Fuente: Diagramación UML.
 Elaboración: El autor.

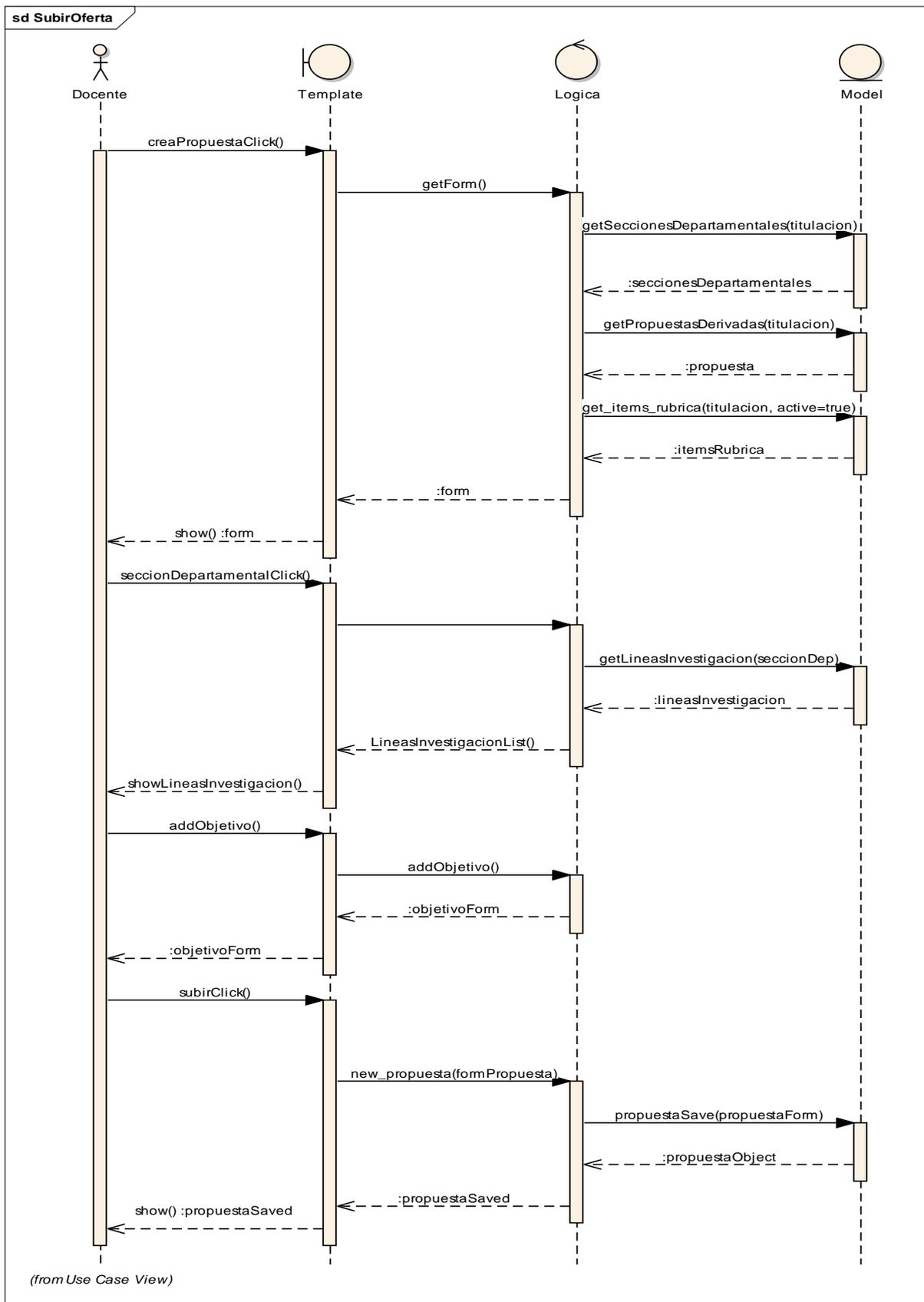


Figura 68. Diagrama de secuencia – Subir Propuesta
 Fuente: Diagramación UML.
 Elaboración: El autor.

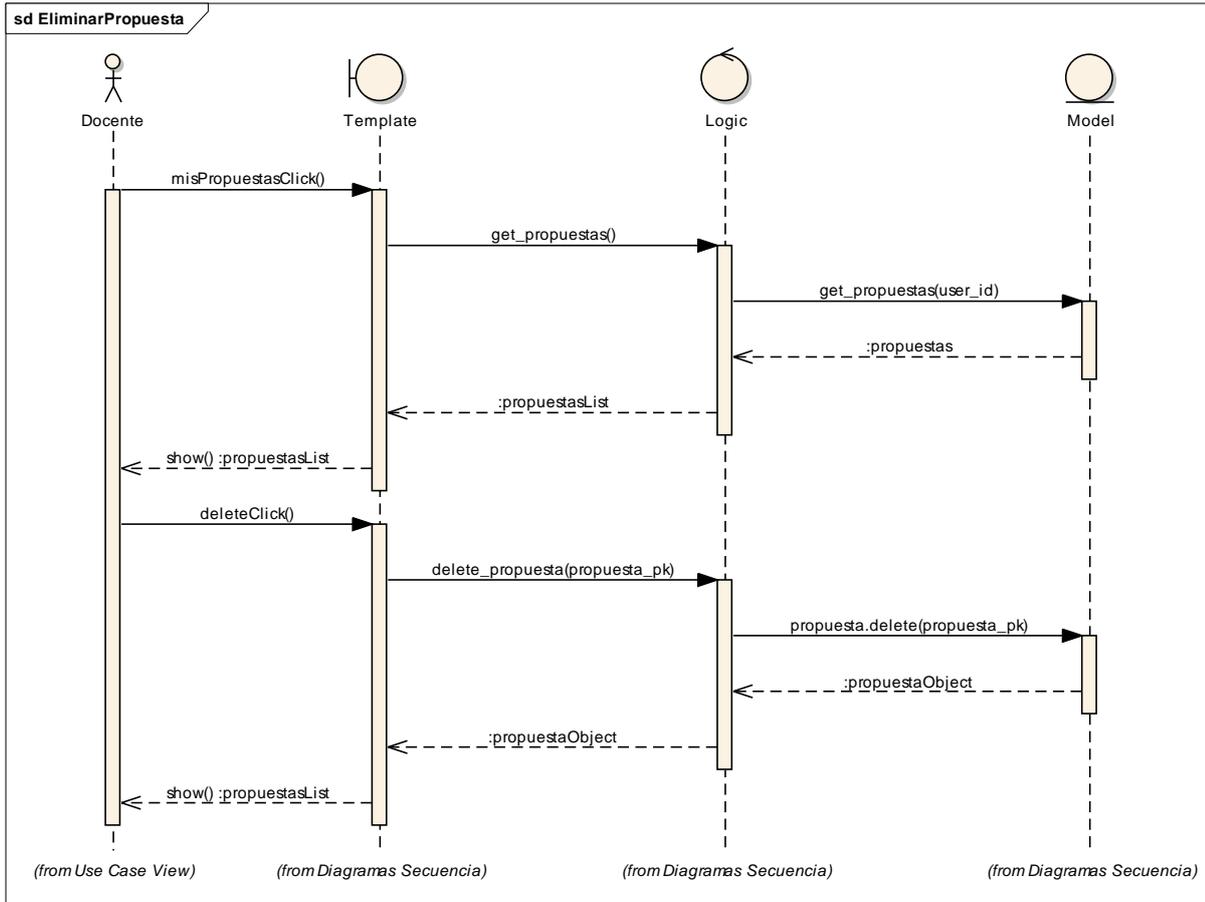


Figura 69. Diagrama de secuencia – Eliminar Propuesta
 Fuente: Diagramación UML.
 Elaboración: El autor.

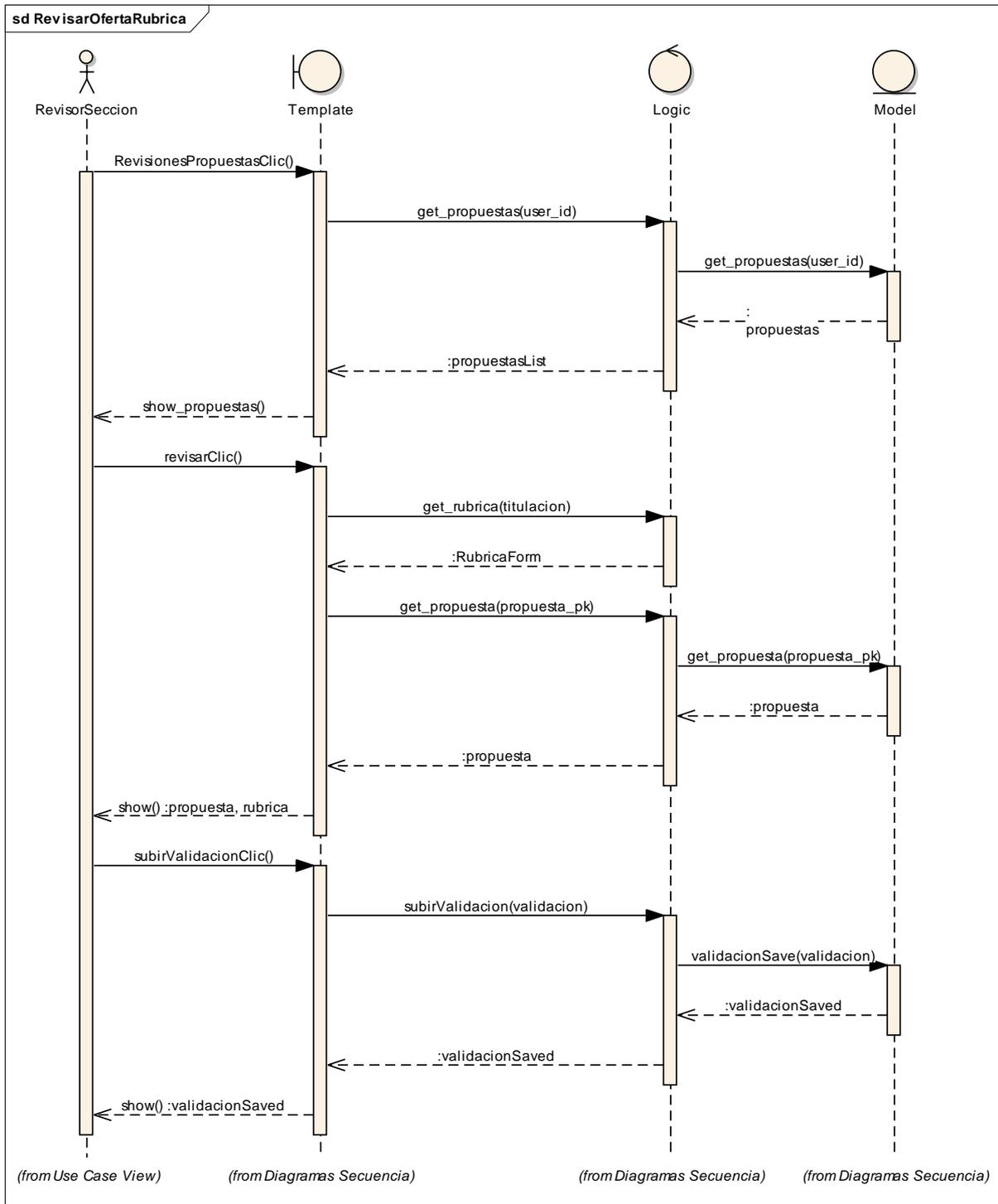


Figura 70. Diagrama de secuencia – Calificar Propuesta

Fuente: Diagramación UML.

Elaboración: El autor.

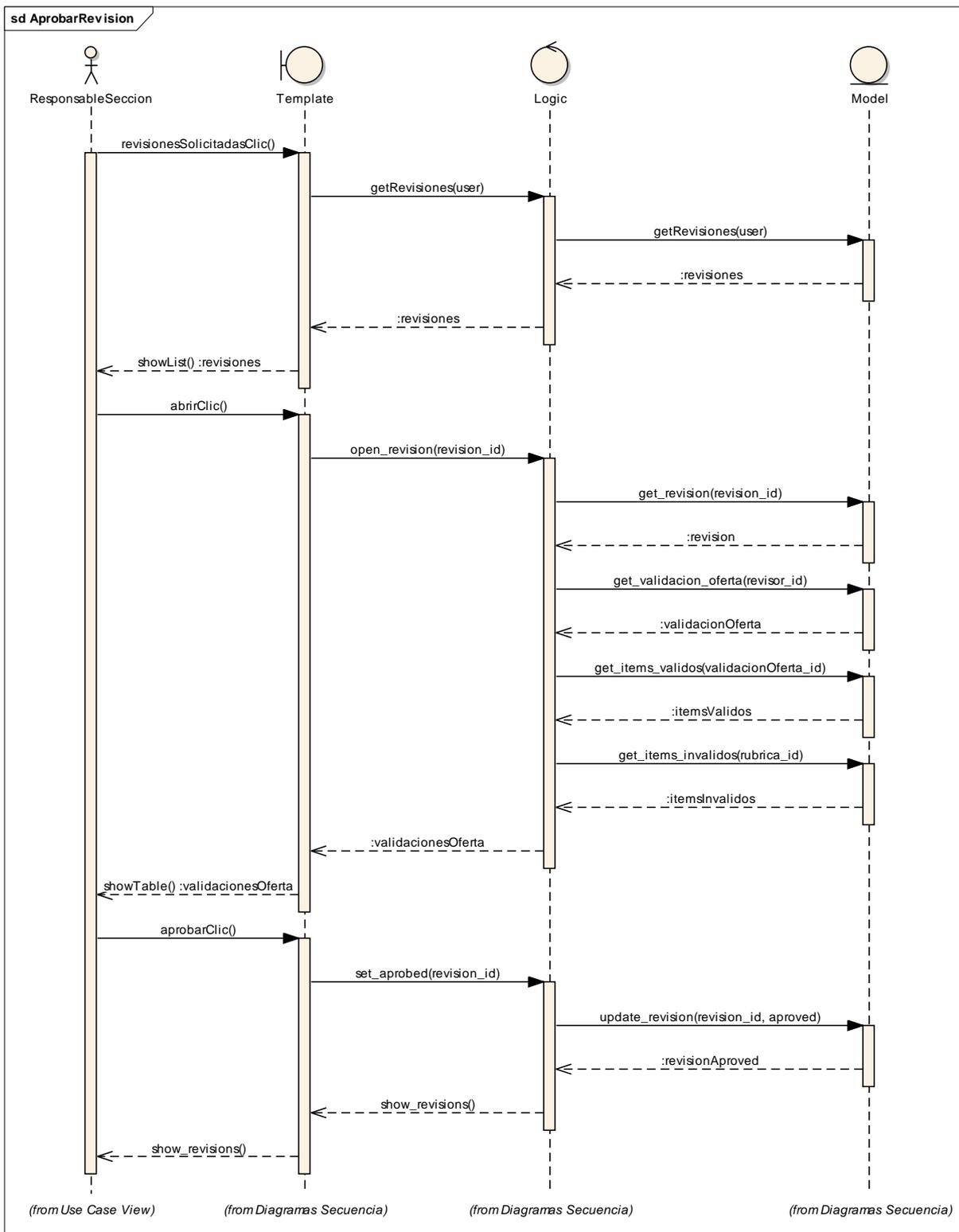


Figura 71. Diagrama de secuencia – Aprobar Revisión
 Fuente: Diagramación UML.
 Elaboración: El autor.

ANEXO 7. Diagramas de Datos

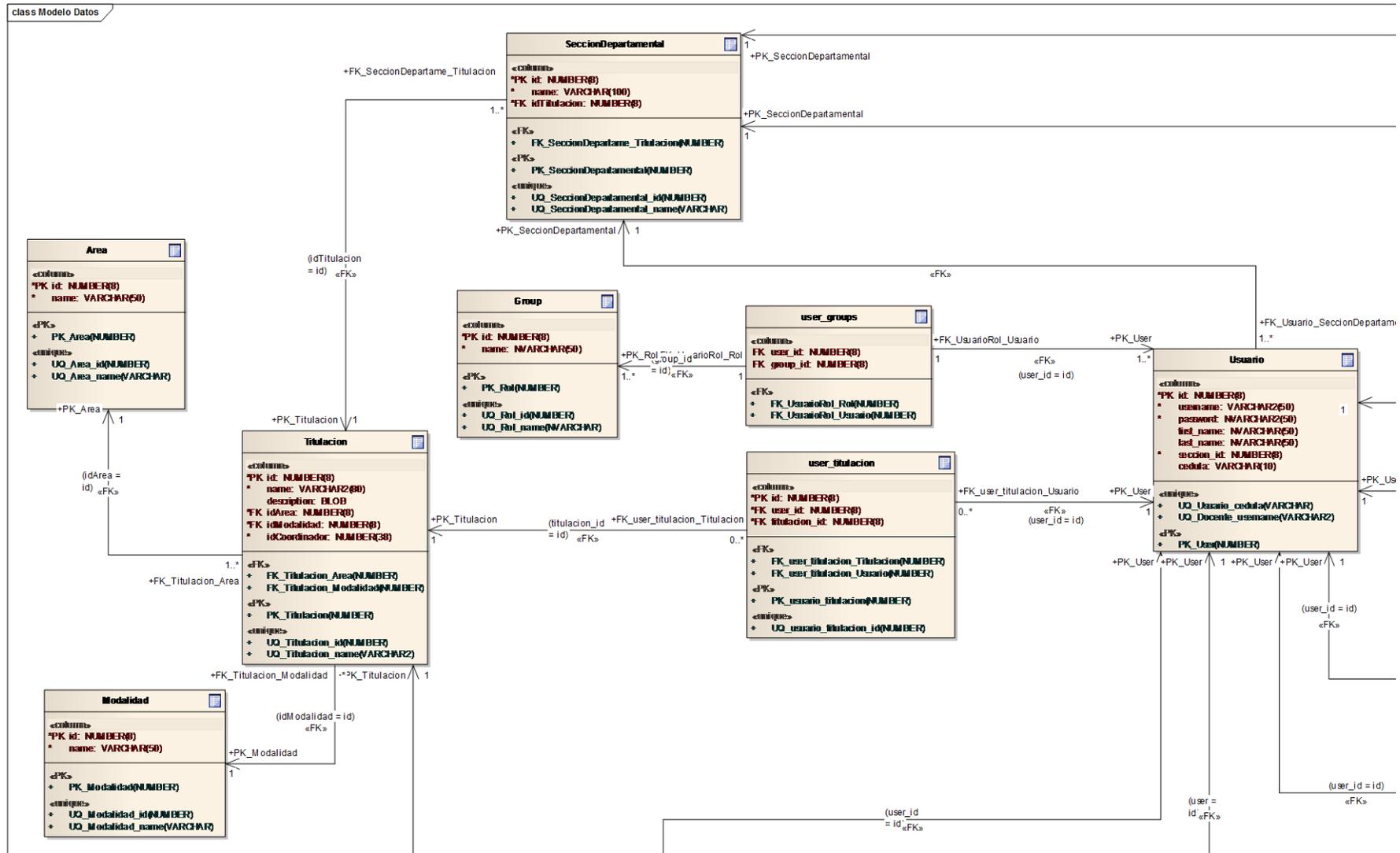


Figura 72. Modelo de datos Usuario Titulación

Fuente: Diagramación UML.

Elaboración: El autor.

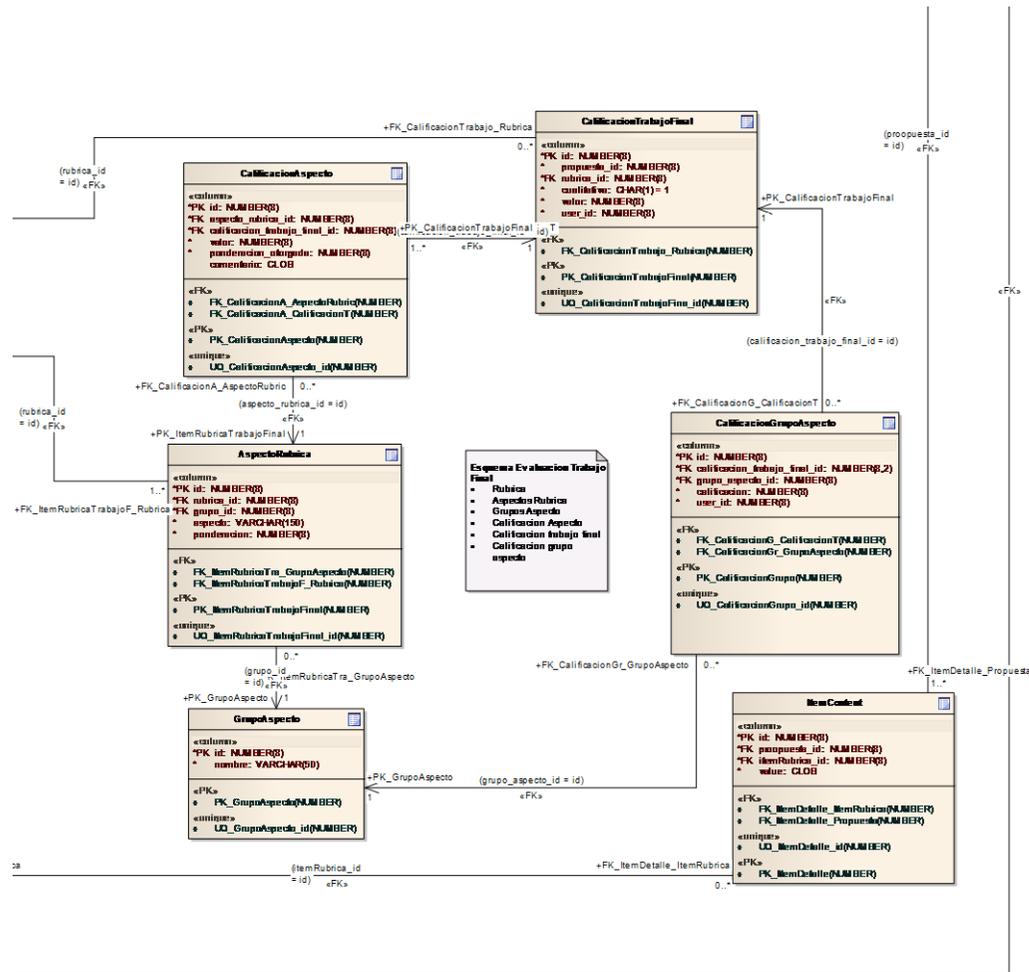


Figura 75. Modelo de datos: Rúbrica - Revisión
 Fuente: Diagramación UML.
 Elaboración: El autor.

ANEXO 8. Product BackLog

Tabla 59. Product BackLog.

ID	Prioridad	Rol (Yo como)	Característica (Deseo)	Razón (Para)	Criterios de Aceptación	
					Nombre	Escenario
HU-001	Media	Docente	Poder ingresar al sistema con mis credenciales	Que el sistema me muestre las características concernientes a mis tareas únicamente.	Autenticación de usuarios	EN CASO QUE desee cumplir alguna tarea asignada a mi en el proceso de postulación de trabajos de titulación CUANDO ingrese al sistema ENTONCES debe mostrarme las funcionalidades de la aplicación que me corresponda.
HU-002	Media	Coordinador de Titulación	Asignar roles de responsables de sección y consejo de departamento a los usuarios	Poder controlar los permisos y los accesos que tiene cada usuario	Asignación de roles	EN CASO QUE existan nuevos usuarios registrados al sistema CUANDO seleccione el usuario y le asigne su respectivo rol o roles ENTONCES el sistema debe habilitar los permisos correspondientes para el usuario o usuarios
HU-003	Media	Docente	Subir una propuesta para trabajo de titulación	Que esta pueda ser revisada y aprobada por sección departamental	Oferta mediante formulario	EN CASO QUE quiera postular mi propuesta de trabajo de titulación CUANDO tenga definidos el tema, descripción, objetivos y el perfil que necesito ENTONCES debe permitirme subir estos parámetros mediante un formulario, para que sea revisado

					Oferta mediante documento Word	EN CASO QUE quiera postular mi propuesta de trabajo de titulación CUANDO tenga la estructura en un documento Word ENTONCES el sistema debe permitirme subir mi archivo para que sea revisado
HU-004	Alta	Docente	Poder conocer el estado de mi propuesta	Hacer correcciones a tiempo en mi propuesta de ser necesario	Estado Propuesta aceptada	EN CASO QUE mi propuesta haya sido aceptada por el responsable de sección CUANDO el consejo departamental haya aprobado la propuesta ENTONCES necesito que se me notifique que está lista para que los estudiantes postulen y poder empezar a entrevistar
					Propuesta requiere cambios	EN CASO QUE mi propuesta requiera una modificación en alguno de sus parámetros CUANDO el responsable de sección emita un comentario de corrección ENTONCES necesito que se me notifique de inmediato la corrección necesaria vía correo electrónico
HU-005	Media	Revisor Sección	Calificar los parámetros que contiene la propuesta de trabajo de titulación mediante un esquema de evaluación (rúbrica)	Validar mediante un esquema de evaluación, si la propuesta es aplicable para postulación como trabajo de titulación	Calificación mediante esquema de evaluación (rúbrica)	EN CASO QUE tenga una propuesta pendiente a revisar CUANDO califique cada ítem de la propuesta ENTONCES el sistema debe mostrarme la calificación de la propuesta, la opción para poner comentarios y ésta ser notificada al responsable de sección y coordinador de titulación.

HU-006	Alta	Revisor Sección	Poder verificar que una propuesta tenga afinidad con las líneas de investigación y con las competencias del estudiante	Mantener la correspondencia de la titulación con las líneas de investigación de la titulación	Propuesta vinculada con proyecto de investigación	EN CASO QUE una propuesta llegue vinculada con un proyecto de investigación CUANDO esté calificando la propuesta ENTONCES el sistema debe permitirme visualizar los requerimientos del proyecto de investigación para validar su correspondencia con la propuesta
HU-008	Media	Responsable de sección	Asignar revisores	Validar la propuesta promediándola entre varias rubricas para tener un avalúo más acertado de la propuesta	Asignación de revisores	EN CASO QUE una propuesta requiera una validación de más de un experto en la materia CUANDO seleccione otro usuario de la sección departamental para asignarlo a la revisión de la propuesta ENTONCES el sistema debe notificar al nuevo usuario asignado que debe revisar con la rúbrica y en el plazo definido
HU-009	Alta	Docente	Que el coordinador de titulación revise y valide las propuestas	Que la propuesta tenga un avalúo del coordinador de titulación, que es quien está más en contacto con los estudiantes y conoce las competencias desarrolladas en la titulación	Validación de coordinador de titulación	EN CASO QUE la propuesta ya haya sido aprobada por sección departamental CUANDO se haya notificado la aprobación ENTONCES el coordinador de titulación debe hacer una validación de la rúbrica y un comentario final antes de aprobarse
HU-011	Media	Coordinador de titulación	Generar convocatoria para ofertar trabajos de titulación	Controlar un cronograma según las actividades académicas designadas por la titulación.	Creación de convocatoria	EN CASO QUE se inicie un periodo académico que contemple las G.P Prácticum CUANDO defina el plazo de oferta máxima ENTONCES debe notificarse a los

						docentes, y no permitirse ofertas luego de la fecha establecida
HU-012	Alta	Coordinador de titulación Responsable sección	Asignar fechas de revisión para actividades de postulación de una propuesta de trabajo de titulación	Tener un control de plazos para actividades de una convocatoria.	Crear convocatoria revisión por cada oferta	EN CASO QUE se haya subido una nueva propuesta de trabajo de titulación CUANDO el responsable de sección/coordinador Titulacion vea una nueva propuesta a revisar ENTONCES el sistema debe asignar una fecha máxima de plazo para efectuar esta revisión al responsable de sección
					Nueva convocatoria revisión	EN CASO QUE se haya subido una nueva propuesta de trabajo de titulación CUANDO seleccione una fecha plazo para revisar la propuesta ENTONCES el sistema debe notificar este plazo al responsable de sección.
HU-013	Alta	Responsable Sección Coordinador titulación	Revisar calificaciones de sección departamental	Aprobar, rechazar, solicitar recalificación o solicitar modificaciones en las propuestas de trabajos de titulación	Acciones a ejecutar en una revisión de una propuesta de trabajo de titulación.	EN CASO QUE una propuesta haya sido validada y aprobada por sección departamental CUANDO revise las calificaciones de los revisores para la propuesta ENTONCES debo poder aprobar, rechazar, solicitar recalificación o solicitar modificaciones y emitir un comentario y que este sea notificado al autor de la propuesta o al revisor cuando se solicite recalificación.
HU-014	Media	Coordinador de titulación	Modificar los plazos de oferta y de revisión asignados	Poder cambiar plazos cuando la titulación o eventos externos lo requieran	Cambio de plazos	EN CASO QUE un evento ocasione un cambio en los plazos de las actividades de una convocatoria CUANDO seleccione la

						convocatoria a la que deseo modificar ENTONCES el sistema debe permitirme cambiar el plazo de la actividad y habilitar la actividad al usuario.
HU-015	Media	Coordinador de titulación	Crear, modificar o eliminar secciones departamentales	Que estos sean seleccionables al subir nuevas ofertas	CRUD secciones departamentales	
HU-016	Media	Coordinador de titulación	Crear, modificar o eliminar periodos académicos	Que estos sean seleccionables al subir nuevas ofertas	CRUD periodos académicos	
HU-017	Alta	Coordinador de titulación	Poder crear un esquema de evaluación (rúbrica) con los ítems que considere necesarios y poder seleccionarlos para que se muestren al subir una propuesta	Calificar las propuestas subidas por un revisor de la sección mediante un esquema de evaluación.	Crear Esquema de Evaluación Propuesta	EN CASO QUE se requiera un nuevo esquema de evaluación para calificar propuestas de trabajo de titulación CUANDO se haya creado una nueva convocatoria ENTONCES el sistema debe permitir llenar un formulario para generar un nuevo esquema de evaluación pudiendo seleccionar los campos base de una propuesta, asignarles un criterio de aceptación para cada calificación cualitativa y poder agregar nuevos ítems.
					Copiar Esquema de Evaluación de otra titulación.	EN CASO QUE se requiera un nuevo esquema de evaluación para calificar propuestas de trabajo de titulación CUANDO se haya creado una nueva convocatoria ENTONCES el sistema me debe permitir buscar un esquema de evaluación de una titulación, visualizarlo y poder copiarlo a mi

						titulación.
HU-018	Alta	Coordinador de titulación	Crear un esquema de evaluación con los aspectos necesarios de mi titulación agrupados en dos grupos (Teórico-técnico y Metodológico)	Calificar un trabajo de titulación finalizado por un estudiante.	Crear Esquema de Evaluación Trabajo Final.	EN CASO QUE se requiera un nuevo esquema de evaluación para calificar propuestas de trabajo de titulación CUANDO se haya creado una nueva convocatoria ENTONCES el sistema debe permitir llenar un formulario para generar un nuevo esquema de evaluación definir los aspectos a evaluar y la ponderación en porcentaje que le corresponde.
HU-019	Media	Director de Departamento	Visualizar las propuestas de trabajos de titulación que han sido revisadas y aprobadas.	Seleccionar las propuestas y autorizar su ejecución para que puedan mostrarse a los estudiantes postule a las mismas.	Autorizar proyectos	EN CASO DE QUE se solicite una reunión de consejo CUANDO se hayan revisado las propuestas que han sido calificadas y aprobadas ENTONCES debo poder seleccionar las propuestas para autorizarlas y que estas pasen al proceso de postulación de los estudiantes en el Sistema del Vicerrectorado Académico.
HU-020	Alta	Sistema de Vicerrectorado Académico	Consumir las propuestas validadas por una titulación, con los campos: id, estado, autor, periodo, sección, tema, objetivo general, objetivos específicos, número de estudiantes, plazo, perfil del estudiante, fecha de registro y fecha de validación.	Poder mostrarlas a los estudiantes y que estos puedan solicitar la asignación a un trabajo de titulación.	Consumo REST de propuestas autorizadas.	EN CASO DE QUE el consejo de departamento haya dispuesto autorizar la ejecución de propuestas de trabajos de titulación calificadas y aprobadas CUANDO el director de consejo autorice las propuestas ENTONCES estas propuestas deben aparecer en una dirección url para ser consumidas en formato de datos JSON para poder ser guardadas en mi sistema.

HU-021	Baja	Estudiante de modalidad abierta	Visualizar las propuestas de trabajos de titulación que están ofertándose en mi titulación	Poder postular a una propuesta e iniciar el trabajo de titulación.	Visualización de propuestas modalidad abierta.	EN CASO DE QUE se haya llamado a una nueva convocatoria para postulación a trabajos de titulación CUANDO las propuestas hayan sido validadas para mi titulación ENTONCES estas propuestas deben mostrarse y permitirme postular a una de ellas.
HU-022	Media	Responsable de Sección	Visualizar las propuestas que ya han sido autorizadas y puestas para postulación de estudiantes.	Poder asignar un equipo de acompañamiento a la propuesta postulada.	Asignación de equipo de acompañamiento	EN CASO DE QUE la titulación determine que se requiere asignar equipo de acompañamiento a las propuestas CUANDO las propuestas hayan sido autorizadas y se visualice el listado de las mismas ENTONCES estas propuestas aprobadas deben permitirme asignar el equipo de acompañamiento a la misma, que lo componen un director y uno o más docentes de acompañamiento.
HU-023	Media	Secretaria	Poder consultar del S.V.A. las propuestas que ya tienen un estudiante asignado.	Poder notificar al estudiante y al equipo de acompañamiento la asignación a un proyecto.	Extracción de propuestas con estudiante de S.V.A.	EN CASO DE QUE una convocatoria determine que el plazo de postulación se aproxima a su finalización. CUANDO los estudiantes hayan postulado y hayan sido asignados a un trabajo de titulación ENTONCES el sistema debe permitirme consultar el estudiante de cada propuesta del S.V.A.
HU-024	Alta	Coordinador de titulación	Poder consultar del S.V.A. las propuestas que ya han sido completadas por el estudiante	Poder asignar tribunal al trabajo terminado.	Extracción de trabajos de titulación completados	EN CASO DE QUE la titulación solicite la asignación de tribunales CUANDO un trabajo de titulación haya sido completado en su

						totalidad por un estudiante. ENTONCES el sistema debe permitirme consultar los trabajos de titulación completados del S.V.A.
HU-025	Media	Coordinador de titulación	Asignar tribunal a un trabajo de titulación completado.	Que el presidente del tribunal pueda emitir una calificación del trabajo finalizado, con el esquema de evaluación definido por la titulación	Asignación de tribunal	EN CASO DE QUE la titulación solicite la asignación de tribunales CUANDO se hayan extraído los trabajos de titulación finalizados del S.V.A. ENTONCES el sistema debe permitirme asignar Presidente, Vocal 1 y Vocal 2 al trabajo de titulación.
HU-026	Media	Secretaria	Notificar la asignación del tribunal a un trabajo de titulación.	Que el presidente del tribunal sea notificado y pueda revisar el trabajo de titulación para emitir una calificación final.	Notificación a tribunal	EN CASO DE QUE la titulación solicite la asignación de tribunales CUANDO se hayan extraído los trabajos de titulación finalizados del S.V.A. ENTONCES el sistema debe permitirme notificar
HU-027	Alta	Docente	Visualizar el listado de las propuestas en las que he sido asignado como presidente	Poder emitir una calificación final al trabajo de titulación completado por un estudiante.	Calificación trabajo de titulación completado.	EN CASO DE QUE la titulación solicite la calificación de un trabajo de titulación completado CUANDO ingrese al listado de los trabajos de titulación en los que soy presidente (Mis Proyectos) ENTONCES el sistema debe permitirme visualizar el proyecto terminado y emitir una calificación según el esquema de evaluación definido por la titulación.
HU-028	Baja	Coordinador de titulación	Visualizar un reporte del estado de las propuestas de trabajo de titulación	Conocer el estado, asignaciones, equipo, tribunal, fechas de aprobación, estudiante, etc.	Reporte de trabajos de titulación.	EN CASO DE QUE necesite saber el estado de los trabajos de titulación CUANDO ingrese al listado de los trabajos de titulación de las

						titulaciones a las que pertenezco ENTONCES el sistema debe permitirme visualizar un reporte de los trabajos con los campos: Periodo, titulo/tema, fecha aprobación, plazo, estudiante, director, equipo, estado.
--	--	--	--	--	--	--

Fuente: Scrum.

Elaboración: El autor.

ANEXO 9. Diagrama de componentes - Subsistemas

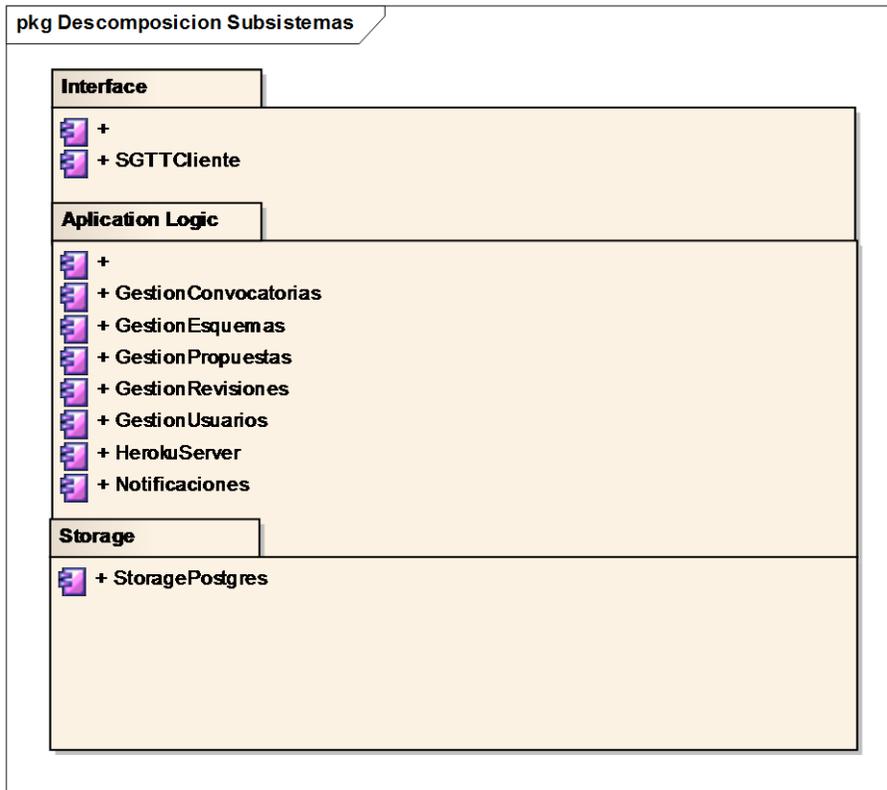


Figura 76. Descomposición de sistema - Package

Fuente: Diagramación UML.

Elaboración: El autor.

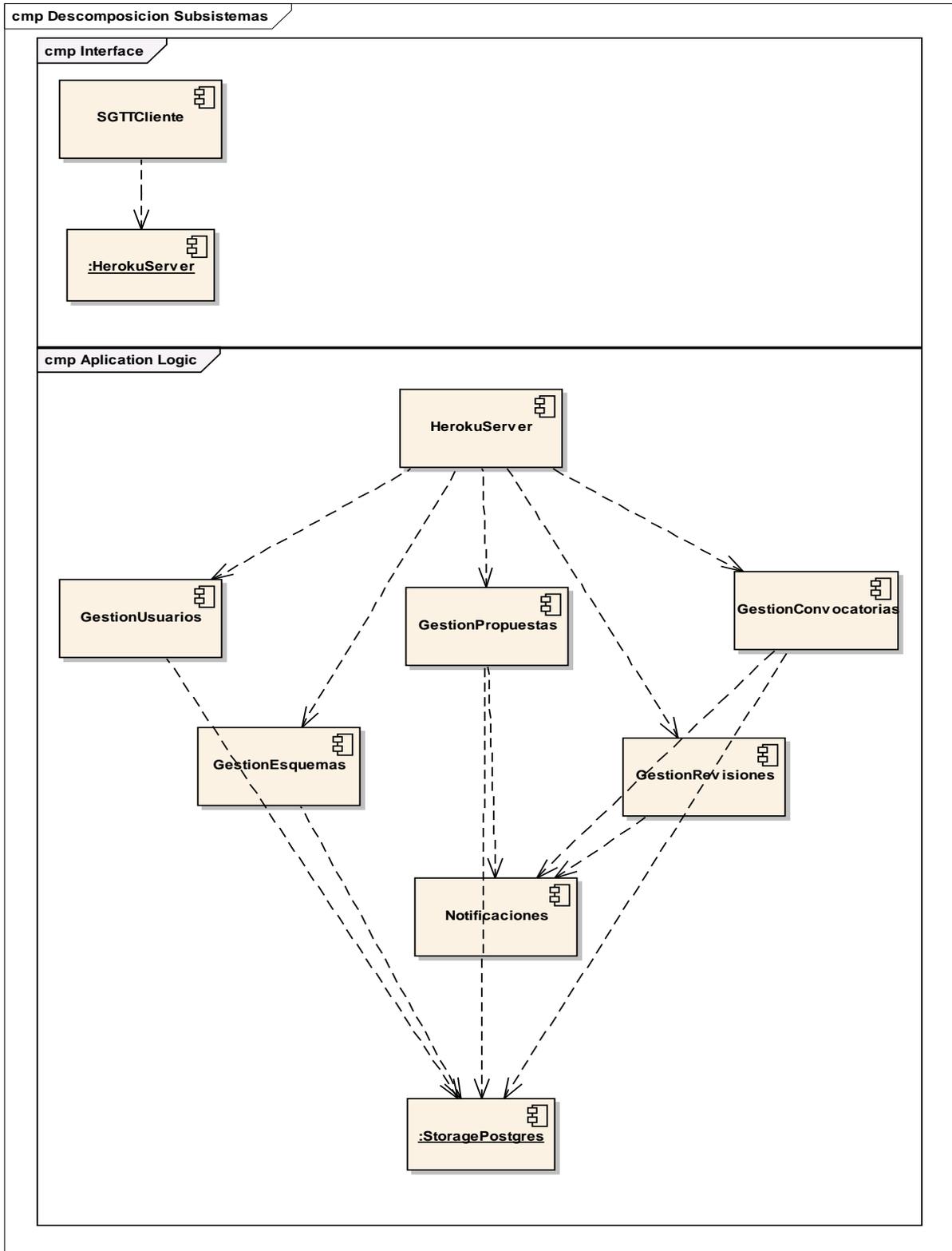


Figura 77. Descomposición subsistemas
 Fuente: Libro Prentice Object Oriented Software.
 Elaboración: El autor.

ANEXO 10. Modelado de Componentes Django

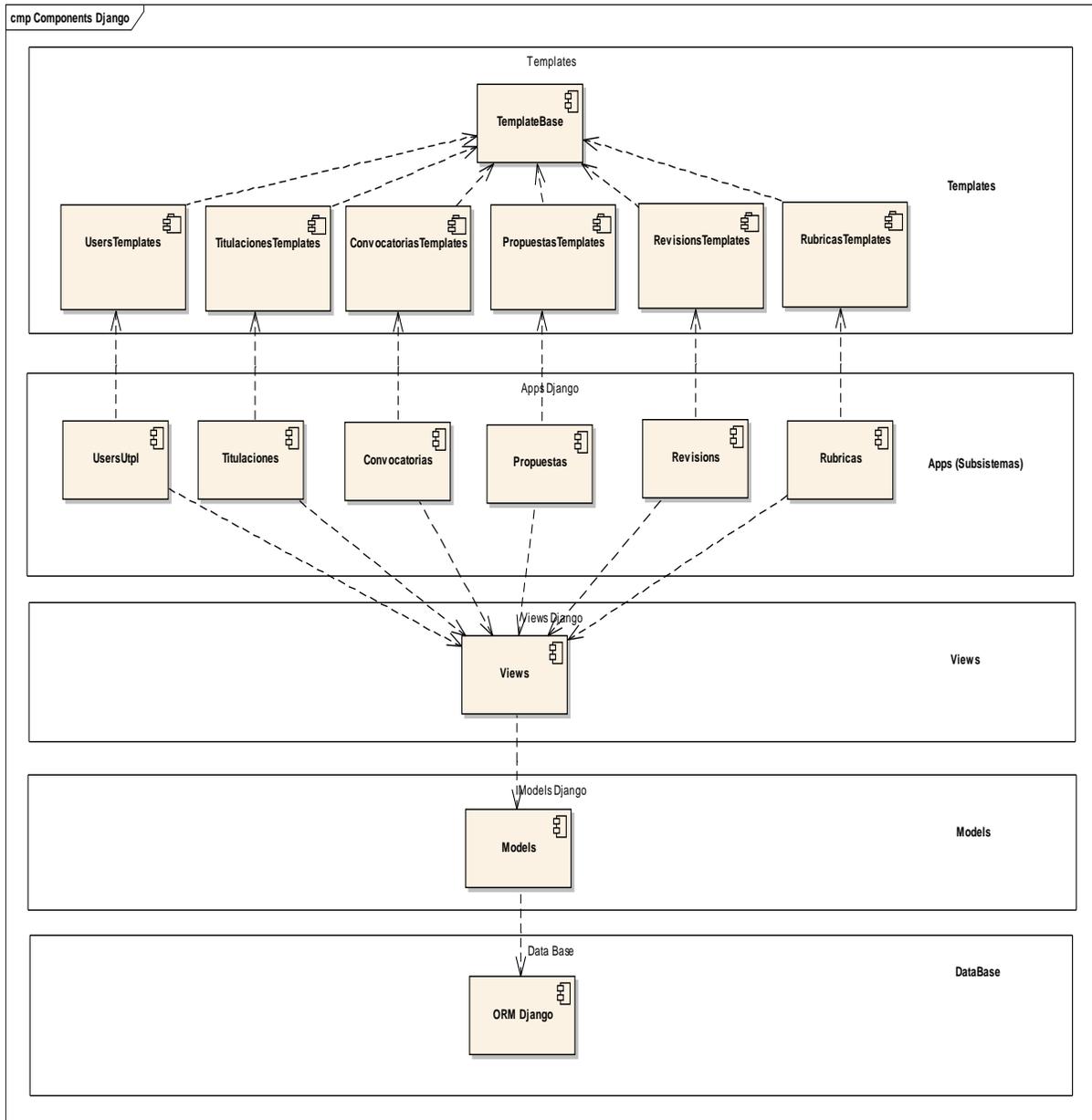


Figura 78. Modelo de componentes Django

Fuente: El autor.

Elaboración: El autor.

ANEXO 11. Diagrama de estados

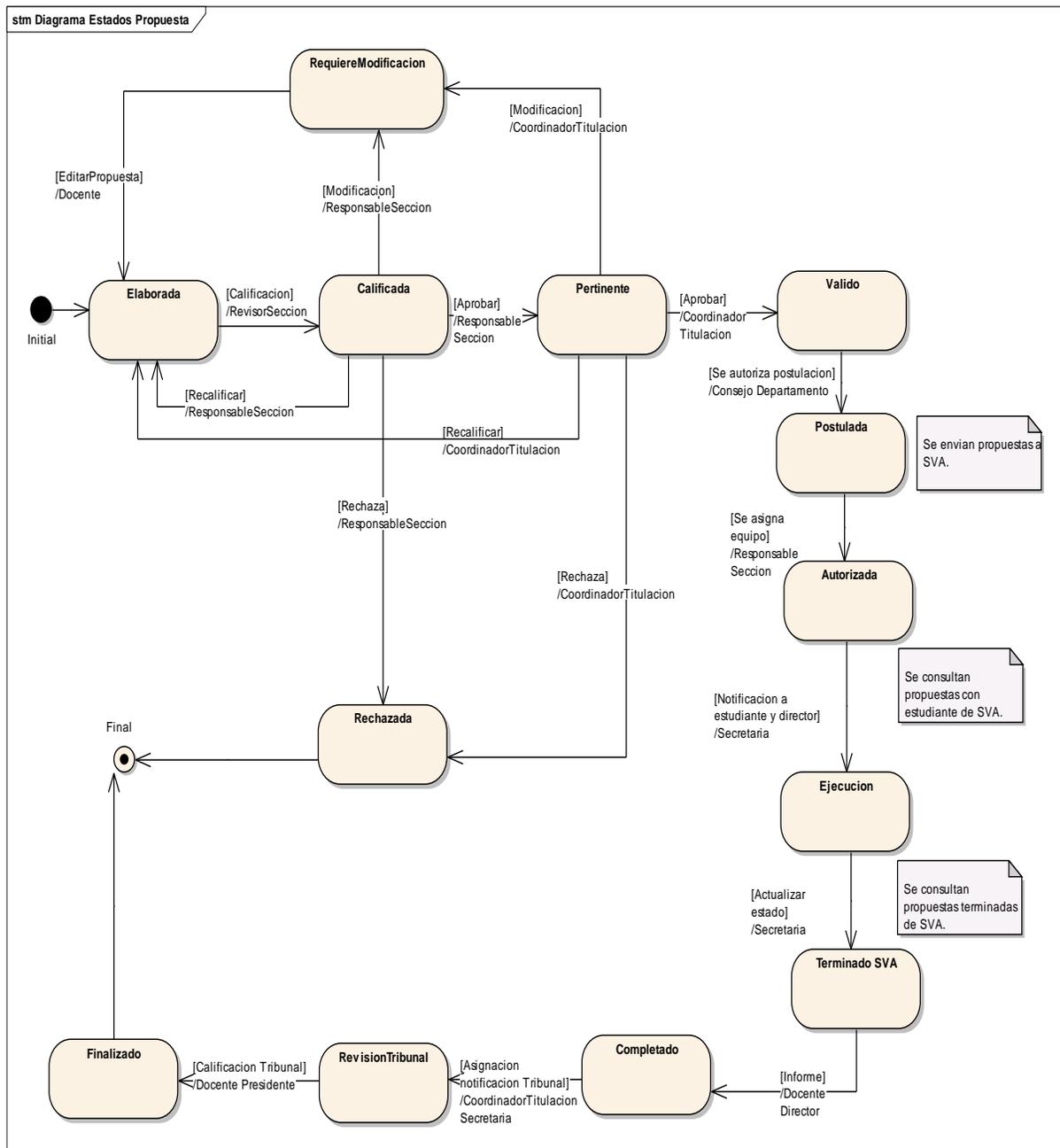


Figura 79. Diagrama de estados “Propuesta T.T”

Fuente: Modelado UML.

Elaboración: El autor.

ANEXO 12. Prototipos exploratorios

SGT
Sistema de Gestión de Trabajos de Titulación

Home About Contact
Twitter Facebook Google plus

Subir Propuesta Validar Propuesta Título 3 Título 4 Título 5

Ofertar Trabajo de Titulación

En esta sección el docente podrá validar cada componente de la oferta, y emitir un comentario de la misma.

Formulario

DENOMINACIÓN DEL PROYECTO DE INVESTIGACIÓN
(Estudio, aplicación, desarrollo)

LÍNEA ESTRATÉGICA DE LA UTPL (Seleccione una)
Desarrollar una universidad como alma máter para el siglo XXI

SECCIÓN DEPARTAMENTAL (Seleccione una)
ELECTRÓNICA Y TELECOMUN

LÍNEA DE INVESTIGACIÓN DE LA SECCIÓN DEPARTAMENTAL (Seleccione una)

JUSTIFICACIÓN

OBJETIVOS
General

Específicos

FINANCIAMIENTO
Monto aproximado

LISTADO DE TEMAS DE TRABAJO DE FIN DE TITULACIÓN DERIVADOS DEL PROYECTO

- ESTUDIO Y DISEÑO DEFINITIVO DEL SISTEMA DE _____ (1 estudiante Ing)
- EVALUACIÓN Y DIAGNÓSTICO DEL _____ (1 estudiante de Geología y Minas)
- CARACTERIZACIÓN ESPACIAL _____ (1 estudiante de)

GUARDAR

© Copyright UTPL

Figura 80. Prototipo Subir Propuesta para T.T
Fuente: El autor.
Elaboración: El autor.



Figura 81. Prototipo Calificación de propuesta para T.T
Fuente: El autor.

Elaboración: El autor.

ANEXO 13. Lineamientos para la presentación, aprobación y seguimiento de los Trabajos de Titulación.



**ÁREA TÉCNICA
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**TITULACIÓN DE INGENIERÍA EN SISTEMAS INFORMÁTICOS
Y COMPUTACIÓN E INGENIERÍA EN INFORMÁTICA**

**Lineamientos para la presentación, aprobación y seguimiento de los
Trabajos de Titulación**

**Patricio Abad Espinoza
Coordinador de Titulación**

**Equipo de Calidad María
del Carmen Cabrera
Fernanda Soto**

Pablo Torres Carrión

Abril 2015

TABLA DE CONTENIDOS

Tabla de contenido

1. PROCESO DE APROBACIÓN DE PROPUESTAS DE TT	4
2. EL PROCESO GENERAL DE APROBACIÓN DE PROPUESTAS DE TT.....	1
3. SEGUIMIENTO DEL TRABAJO.....	3
4. EXTERNALIZACIÓN DEL PROYECTO.....	4
5. ANEXOS.....	5

El Departamento de Ciencias de la Computación y Electrónica, por intermedio del Equipo de Calidad Académica de las titulaciones de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática.

Considerando:

- ! Que el artículo 153 de la Constitución de la República del Ecuador, dispone: “El sistema de Educación Superior se regirá por 1: Un organismo público de planificación, regulación y coordinación interna del sistema y de la relación entre actores de la Función Ejecutiva (...)”
- ! Que el Consejo de Educación Superior (CES) elaboró un instructivo al reglamento de presentación y aprobación de proyectos y programas de grado, donde expone de forma general los lineamientos de carrera.
- ! Que el CES expide el Reglamento de Régimen Académico (RRA).
- ! Que el Art. 107 de la LOES expone: “Principio de pertinencia.- El principio de pertinencia consiste en que la educación superior responda a las expectativas y necesidades de la sociedad, a la planificación nacional, y al régimen de desarrollo, a la prospectiva de desarrollo científico, humanístico y tecnológico mundial, y a la diversidad cultural. Para ello las instituciones de educación superior articulará su oferta docente, de investigación y actividades de vinculación con la sociedad, a la demanda académica, a las necesidades de desarrollo local, regional y nacional, a la innovación y diversificación de profesiones y grados académicos, a las tendencias del mercado ocupacional local, regional y nacional, a las tendencias demográficas locales, provinciales y regionales: a la vinculación con la estructura productiva actual y potencial de la provincia y la región, y a las políticas nacionales de ciencia y tecnología”.
- ! Que el CES mediante resolución PRES-CES-No. 132-2013 expide el instructivo del indicador del mérito académico de graduación.
- ! Que en el estatuto orgánico de la Universidad Técnica Particular de Loja, Art.9, lit. d) expone como uno de sus objetivos: “Contribuir con el cumplimiento de los objetivos planteados en el Plan Nacional de Desarrollo del Ecuador, respetando los principios de autonomía responsable, cogobierno, igualdad de oportunidades, calidad, pertinencia, integridad y autodeterminación establecidos en la Ley”.
- ! Que la UTPL solicitó a las titulaciones que de acuerdo a las normas establecidas en el RRA definan las opciones de titulación con las cuales los estudiantes tanto de modalidad presencial como a distancia, podrán aplicar para titularse.
 - Que, es necesario expedir un conjunto de lineamientos que operativicen el procedimiento de presentación de solicitudes de proyectos de grado de las carreras ofertadas en el ámbito de Ciencias de la Computación.

Expede los siguientes **lineamientos** para presentación de propuestas para trabajos de fin de titulación, los cuales son aplicables tanto a modalidad presencial como a modalidad a distancia.

1. PROCESO DE APROBACIÓN DE PROPUESTAS DE TT

Conforme lo establecido en artículo 21 del Reglamento de Régimen Académico, se determina que para poder titularse, los estudiantes pueden optar por una de las opciones de Trabajo de Titulación (TT), establecidas por la Universidad, el cual reemplaza a Tesis de Grado, o trabajo de tesis que se venía utilizando.

Las propuestas de TT pueden ser presentados por docentes investigadores de las diferentes secciones departamentales o por estudiantes tanto de la modalidad presencial como de modalidad a distancia, en cuyo caso podrán recibir asesoría de un docente investigador afín al tema propuesto.

Los estudiantes matriculados en el Practicum 4, recibirán orientaciones de parte de los tutores a cargo del componente.

Se consideran aptos para postular a un TT los estudiantes en los siguientes casos:

Situación del estudiante	Modalidad
Con matrícula aceptada en la GP 4	Presencial, distancia
Estudiantes que tengan terminada su malla académica, y aún no hayan desarrollado su TT.	Presencial, Distancia
Estudiantes que habiendo aplicado al examen complejo no lograron titularse y solicitan aplicar al	Presencial, Distancia

Nota: Para las propuestas de TT ofertadas por los docentes investigadores, se dará prioridad a los estudiantes que aplican por primera vez.

Según los documentos: *Opciones de titulación para Ingeniería en Sistemas Informáticos y Computación* (Anexo 2) y *Opciones de titulación para Ingeniería en Informática* (Anexo 3), se establece que el TT de las carreras objeto de estos lineamientos, corresponde a Proyecto Técnico (Estudio Técnico), cuyo lineamientos se definen en el art. 21 del RRA, el cual sostiene

que:

“Todo trabajo de titulación deberá consistir en una propuesta innovadora que contenga, como mínimo, una investigación exploratoria y diagnóstica, base conceptual, conclusiones y fuentes de consulta. Para garantizar su rigor académico, el trabajo de titulación deberá guardar correspondencia con los aprendizajes adquiridos en la carrera y utilizar un nivel de argumentación, coherente con las convenciones del campo del conocimiento.

Cada carrera deberá considerar en su planificación e implementación curricular, al menos dos opciones para la titulación.”

Los trabajos de titulación, deberán contener al menos las siguientes fases (Unidad Curricular de Titulación-CES, 2013):

- ! Determinación del problema, dilema o tensión del que se deriva la pregunta o hipótesis planteada para el proceso de investigación-acción.
- ! Objetivos o propósitos que definen la finalidad de la propuesta en términos del conocimiento, la profesión y la experiencia de aprendizaje.
- ! Contextualización y caracterización teórica y profesional del problema, que integra el diagnóstico de la experiencia de investigación-acción y la fundamentación conceptual.
- ! Metodología de proceso de investigación-acción, definiendo los modos de recolección, procesamiento e interpretación de los datos, obtenidos en la experiencia de aprendizaje práctico y de indagación realizada por el estudiante.
- ! Conclusiones y propuestas de alternativas de solución al problema.

Respecto del **nivel de investigación** de los trabajos de titulación o de grado, esta debe ser de tipo “exploratorio y diagnóstico”, siendo sus características las siguientes:

- ! Se basa en procesos de exploración e indagación de la realidad para la determinación y comprensión de los problemas.
- ! La metodología es flexible y de carácter expansiva, pues se realiza en función de los contextos y actores que intervienen en el proceso de investigación-acción y posibilita la identificación de variables, dimensiones e indicadores para la configuración de los problemas que son objeto de estudio.
- ! Se orienta a la identificación de relaciones y circuitos de interacción entre los contextos, los sujetos y las prácticas de aprendizaje, definiendo un sistema de indicadores para el desarrollo del análisis e interpretación de los problemas.

- ! Sistematiza las prácticas de investigación-acción a partir de redes y sistemas conceptuales que favorecen las propuestas de innovación y desarrollo para la solución de los problemas.
- ! Implica a los estudiantes en una práctica de aprendizaje por descubrimiento orientada a la construcción de significados relacionados con las características, rasgos, dimensiones, contextos, componentes, etc., del problema seleccionado para su comprensión y transformación.

2. EL PROCESO GENERAL DE APROBACIÓN DE PROPUESTAS DE TT.

Con estas características definidas para los proyectos, se establece el proceso general de aprobación de propuestas que se muestra en la figura 1.

PROCESO DE PRESENTACIÓN DE PROPUESTAS DE TRABAJO DE TITULACIÓN

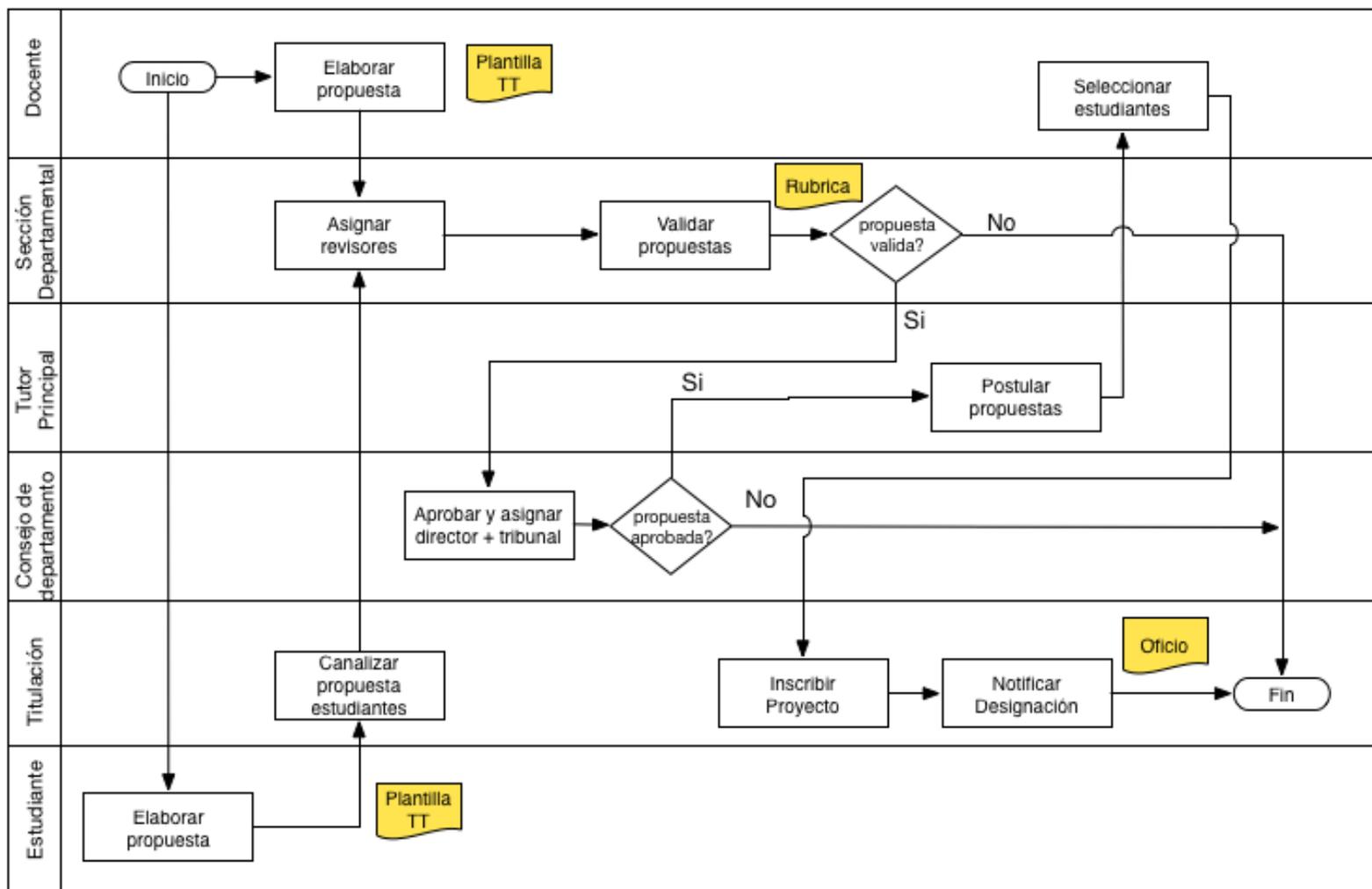


Figura 1. Proceso de presentación de propuestas

Paso 1: Elaborar propuesta:

Tanto docentes como estudiantes pueden elaborar una propuesta de proyecto utilizando la plantilla establecida en el Anexo 1, y Anexo 2, en la cual constan los elementos necesarios, para la presentación de las propuestas tanto para modalidad presencial como para modalidad a distancia.

Cabe recalcar que de acuerdo a lo que establece el reglamento, las propuestas deben incluir los elementos siguientes conforme constan en las opciones de titulación aprobadas por la junta de área.

Los trabajos de titulación se enmarcarán en la línea de **Proyecto técnico** el cual se define como: “trabajos que tienen como objeto la realización de estudios a equipos, sistemas, servicios, etc., relacionados con los campos propios de la titulación, referidos a aspectos de diseño, planificación, producción, gestión, explotación y cualquier otro campo de la ingeniería, relacionando con alternativas técnicas, evaluaciones económicas y valoración de los resultados” (CES)

En función de lo cual se establece que las propuestas presentadas para las carreras de Ingeniería en Sistemas Informáticos y Computación e Ingeniería en Informática, deben incluir tres componentes:

- ! Un componente técnico, con un porcentaje entre el 50 y 60% del total del TT.
- ! Un componente de innovación, con un porcentaje máximo del 30% del total del TT.
- ! Un componente de investigación, con un porcentaje máximo del 30% del total del TT.

Las propuestas deben presentarse en el formato establecido en las plantillas correspondiente (Anexo 3 y Anexo 4) considerando lo siguiente:

- ! Deben estar enmarcadas en una de áreas de conocimiento de las Ciencias de la Computación, conforme consta en las opciones de titulación (anexos 1 y 2).

! Las propuestas de TT pueden ser desarrolladas hasta por dos estudiantes de la misma titulación y hasta un máximo de 3 estudiantes en el caso de que el proyecto aplique para más de una titulación o involucre a otras IES.

! Es necesario especificar el perfil de los estudiantes requeridos para identificar sus competencias, en función de sus aptitudes de manera que se favorezca la selección de los postulantes.

En el caso de recibir propuestas elaboradas por estudiantes no vinculados a Prácticum 4, estas serán remitidas a la Titulación, y el coordinador o alguna persona delegada por el coordinador evaluará la propuesta y la **canalizará hacia la sección departamental** que tenga relación con el tema para que sea evaluada en las mismas condiciones que las otras propuestas.

Paso 2: Asignar revisores

Este proceso tiene como fin principal, la validación de la pertinencia y de los aspectos metodológicos, exigiendo que la propuesta esté enfocada en los lineamientos de la Titulación y esté acorde a las líneas de investigación del departamento.

Todas las propuestas deben pasar por un proceso de por parte de las secciones departamentales, para lo cual el responsable de sección designara revisores (recomendable 2) por cada propuesta.

Paso 3: Validar propuestas

Para la validación de las propuestas, se diseñando una rúbrica, la cual consta en el anexo 5 del presente documento, el propósito es asegurar que se cumplen con las características mínimas establecidas por la titulación para justificar el grado de ingeniero.

La revisión con la rúbrica permitirá que la propuesta obtenga una nota valorada en escala de 1 a 5 cuya interpretación será la indicada en la tabla 1.

Tabla 1. Escalas de valoración de propuestas de TT

Puntuación)	Valoración)	Acción)a)seguir)
5"	Excelente"	Aprobado. Seguir proceso regular."
4 y 5"	Aceptable	La propuesta es aceptable, pero se pueden sugerir algunas mejoras
> 3 y < 4"	No aceptable	El proyecto puede calificar siempre y cuando se realicen los ajustes necesarios
<"3"	Rechazado"	La propuesta no califica

Las secciones departamentales pueden solicitar los ajustes necesarios para la presentación de propuestas válidas, en el caso de no calificar finalizan el proceso y si el proponente considera necesario puede modificar sus propuestas y volverla a presentar.

Paso 4: Aprobar y asignar tribunal

Conforme lo establece instructivo de la Unidad de Titulación Especial emitido por el Vicerrectorado Académico y el Vicerrectorado de Modalidad Abierta y a Distancia, vigente desde

13-Abril-2015, establece que "una vez aprobado el tema del trabajo de titulación, se designará un director de trabajo de titulación y tribunal evaluador, quienes serán nombrados, de acuerdo a su afinidad o conocimiento del tema, por el Consejo de Departamento correspondiente"

Por lo tanto todas las propuestas validadas por las secciones departamentales se presentarán al Consejo de Departamento con la rúbrica correspondiente, y en sesión aprobarán los proyectos y designarán director y tribunal para el mismo.

En el caso de que alguna propuesta no sea aprobada, se someterá a revisiones e iniciará como nueva propuesta.

Las secciones departamentales, deberán llevar una propuesta de director de trabajo de titulación por cada tema válido incluyendo a las propuestas presentadas directamente por estudiantes.

Se debe contemplar las políticas del departamento respecto del número de trabajos de titulación por docente.

Paso 5: Postular propuestas

Todas las propuestas aprobadas, se pasarán al tutor de Prácticum 4, para que sean puestas disposición de los estudiantes que esperan al asignación de un tema. En el caso de estudiantes que no tengan matrícula en este componente, igualmente deberán acercarse con el tutor principal para optar por un tema de tesis.

Los estudiantes podrán postular para cualquiera las propuestas aprobadas siempre y cuando cumplan con las condiciones académicas de la titulación: créditos, asignaturas, no impedimentos.

Tendrán prioridad los estudiantes que postulan por primera vez.

Una vez que los estudiantes hayan postulado, los tutores de GP 4, validarán los requerimientos de cada postulante

Los proyectos de TT que no hayan tenido acogida por parte de los estudiantes, así como el listado de estudiantes que no tengan un proyecto asignado, pasarán a consideración de los responsables de sección para su respectiva solución esperando los siguientes resultados:

- Todos los estudiantes tanto de modalidad presencial como de modalidad a distancia, deberán tener un tema de tesis asignado.
- El resultado de estas resoluciones se notificará a los tutores de GP 4, siguiendo las especificaciones de postulación de TFT.

**Paso 6:
Seleccionar
estudiantes**

Con los resultados de la postulación, cada docente recibirá la propuesta de trabajo, y la lista de personas que han postulado para cada tema, con ello realizará una entrevista donde explicará al postulante la necesidad y características del trabajo y se calificará su idoneidad para el desarrollo del proyecto en función de: actitudes, aptitudes, disponibilidad de tiempo y otros aspectos que el docente considere válidos.

El resultado de las entrevistas será comunicado al tutor principal por parte del docente quienes notificarán los proyectos aprobados a la titulación.

**Paso 7:
Inscribir
proyectos**

Los proyectos enviados a la titulación serán inscritos oficialmente y se registrarán los datos del proyecto, considerando como línea base la siguiente información:

- Título
- Objetivos
- Alcance
- Plazo

Los cuales se usarán para las revisiones intermedias y presentación del trabajo terminado ante el tribunal examinador, en caso de cambios a lo largo del desarrollo y si estos afectan a algunos de los elementos anteriores, deberá notificarse a la titulación para su aprobación y registro.

**Paso 8:
Notificar
designación**

La titulación emitirá mediante oficio tanto a directores como a estudiantes sobre la inscripción del proyecto y solicitará que en el plazo de 15 días contados a partir de la notificación, los estudiantes entregarán a la titulación un plan de ejecución del proyecto.

3. SEGUIMIENTO DEL TRABAJO.

- ! Todos los TT se rigen por la línea base establecida.
- ! El avance de este plan debe reportarse regularmente (bimestral) por parte del director de tesis.
- ! El tutor de GP, es el encargado de asegurarse de que la información respecto del avance se haga con regularidad, para ello los estudiantes deben reportar los avances con el visto bueno del director del TT.
- ! Una vez por bimestre los estudiantes presentarán los avances ante un equipo de revisores designado por el Consejo de Departamento, quienes tenderán como parámetros de revisión: el plan definido y los entregables del TT. Esta presentación se calificará como parte de la nota de la GP.

- ! En caso de requerir cambios a alguno de los aspectos identificados como línea base, deberá presentarse una solicitud de cambio especificando los motivos correspondientes, la cual pasará a revisión del comité y se ejecutarán siempre y cuando la misma sea aprobada.

4. EXTERNALIZACIÓN DEL PROYECTO.

- ! Todos los proyectos aprobados deberán de ser enviados para su valoración en los concursos y eventos relacionados que promueve la UTPL de forma interna.
- ! Todos los proyectos postularán a un proceso de selección previa al concurso Galardones (o similar) que organiza la Secretaría Nacional de Ciencia y Tecnología. De este proceso se seleccionarán al menos cinco a participar en cada una de las categorías que competen a la carrera y departamento.
- ! Las categorías de participación al concurso Galardones serán seleccionadas por el

Equipo de Calidad de la Carrera, en reunión con el/la directora/a de departamento y coordinadores de las secciones departamentales.

- ! De considerarlo conveniente, y previa aprobación del Comité de Calidad en primera instancia, se podrá postular un proyecto a eventos internacionales (o publicaciones en revistas científicas o de divulgación).

5. ANEXOS

Titulación: Ingeniería en Sistemas Informáticos y Computación

Modalidad: Presencial

Área: Técnica

Modalidades de titulación:

1. Examen complejo o de grado:

El artículo 21 del Reglamento de Régimen Académico, establece como una de las opciones de titulación al examen complejo el cual “deberá ser de carácter complejo, con el mismo nivel de complejidad, tiempo de preparación y demostración de competencias, habilidades, destrezas y desempeños, que el exigido en las diversas formas del trabajo de titulación teórico-práctico que evalúa resultados de aprendizaje definidos en el perfil de egreso de la titulación con niveles de complejidad y equivalentes a un trabajo de titulación”

Por tanto el examen está diseñado para que el estudiante demuestre el logro de las competencias necesarias para resolver problemas de manera creativa y haciendo uso de las habilidades y conocimientos adquiridos durante su formación profesional.

Para el desarrollo del examen, la titulación ha definido los núcleos fundamentales de la profesión y que se constituyen en la base del desempeño profesional de los graduados

Por ser de naturaleza teórico-práctica el examen contemplará dos componentes: una parte escrita que recogerá los aprendizajes y fundamentación teórica en los núcleos fundamentales de la profesión y un segundo elemento será el desarrollo de un caso práctico donde el estudiante demostrará su capacidad para resolver los problemas en su área de conocimiento.

2. Trabajo de titulación:

El Reglamento de Régimen Académico, establece que “Todo trabajo de titulación deberá consistir en una propuesta innovadora que contenga, como mínimo, una investigación exploratoria y diagnóstica, base conceptual, conclusiones y fuentes de consulta. Para garantizar su rigor académico, el trabajo de titulación deberá guardar correspondencia con los aprendizajes adquiridos en la carrera y utilizar un nivel de argumentación, coherente con las convenciones del campo del conocimiento”.

a. Opciones de trabajo de titulación

ANEXO 1: OPCIONES DE TITULACIÓN, MODALIDAD PRESENCIAL

De entre las opciones dadas la Titulación de Ingeniería en Sistemas Informáticos y Computación, considerando la naturaleza y la formación establecida en el perfil profesional establece las siguientes opciones de titulación.

Opción 1: Proyecto Técnico

El un proyecto técnico tiene como propósito el desarrollo de estudios y aplicación de los conocimientos propios de la titulación en el diseño, desarrollo e implementación de una solución basada en los principios, herramientas y técnicas del campo de las Ciencias de la Computación, los cuales pueden ser aplicados no solo en el ámbito de la carrera como sino en otros campos del conocimiento. Todos las propuestas de trabajo se alinearán al perfil profesional de la titulación.

Los trabajos de titulación deben combinar una investigación de tipo exploratoria, la innovación como base para las soluciones y un elemento técnico centrado en las competencias de la titulación.

Opción 2: Proyecto de Investigación

Son proyectos cuyo objetivo es la investigación de principios de casos, recolección de evidencia empírica y evaluación de modelos y validación de resultados, aplicable a diferentes ámbitos de las Ciencias de la Computación.

Esta alternativa será válida para todas propuestas correspondientes al período octubre/2014-febrero/2015 o anteriores, autorizadas para su desarrollo hasta el 31 de marzo de 2015, y quedarán habilitadas para completarse de acuerdo a su enfoque y planificación originales.

En relación a plazos de culminación, se someterán a las prórrogas establecidas en el Reglamento de Régimen Académico.

A partir del período académico Abril-Agosto 2015, serán válidas únicamente la propuestas de proyecto técnico.

b. Relación con el perfil profesional:

El perfil profesional contempla las siguientes funciones que desarrollará el estudiante como profesional graduado en sistemas informáticos y computación:

1. Aplicar soluciones de ingeniería a problemas relacionados con el uso de TI.
2. Gestionar proyectos de tecnologías de la información y comunicación.
3. Operar, auditar y fiscalizar proyectos de TI.
4. Consultoría y transferencia de conocimientos en el área de TI.
5. Investigar en el área de ciencias de la computación.

Para cumplir con estas funciones el perfil profesional establece las siguientes competencias:

1. Analizar, diseñar e implementar soluciones de ingeniería basadas en TI.
2. Gestionar y fiscalizar proyectos de TI
3. Colaborar con proyectos de investigación en el área de Ciencias de la Computación

4. Transferir conocimientos en el área de TI y generar emprendimientos de base tecnológica

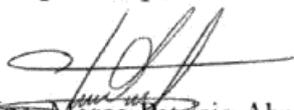
Las actividades propuestas como trabajo de titulación se ajustan a las competencias del perfil profesional establecidas y aseguran un alto nivel de desarrollo del profesional y propician el desempeño en la funciones establecidas en el perfil.

c. Cuestiones generales:

- Los proyectos de tesis pueden ser planteados tanto por estudiantes como por los docentes de la titulación, para el caso de las propuestas de los estudiantes, estos deberán contar con el aval o la tutela de un docente.
- Además la Universidad maneja convenios con empresas e instituciones, que pueden requerir el desarrollo de una solución en una de las áreas establecidas, en este caso y acogiéndose al reglamento, los proyectos pueden ser multi disciplinarios y multi profesionales pudiendo conformar equipos de 2 personas si son de la misma carrera y hasta 3 si son de más de una carrera o pertenecen a distintas universidades.
- Los resultados pueden publicarse en eventos nacionales e internacionales con la participación del director o tutores del trabajo, los derechos son propiedad de la Universidad.
- Los estudiantes podrán utilizar los laboratorios académicos o de investigación en coordinación el tutor del trabajo de tesis y la sección departamental correspondiente.
- El proyecto asume una duración máxima de 400 horas y se contempla un tiempo de desarrollo en meses de entre 6 y 8 meses, en función del alcance planteado en la propuesta y de máximo un año en el caso de proyectos multi disciplinarios y multi profesionales.

d. Observaciones adicionales

Propuesto por:

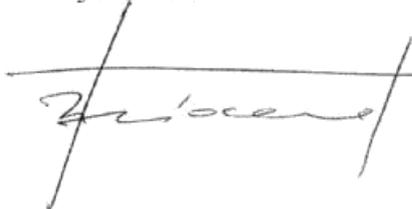


Ing. Marco Patricio Abad Espinoza
COORDINADOR DE TITULACIÓN DE
CIENCIAS DE LA COMPUTACIÓN



Aprobado por:

Junta de área



ANEXO 2: OPCIONES DE TITULACIÓN MODALIDAD ABIERTA Y A DISTANCIA

Dirección General Académica

Dirección General de Modalidad Abierta y a Distancia



UTPL
UNIVERSIDAD TECNOLÓGICA DE PANAMÁ

Titulación: Ingeniería en Informática

Modalidad: Abierta y a distancia

Área: Técnica

Modalidades de titulación:

1. Examen complejo o de grado:

El artículo 21 del Reglamento de Régimen Académico, establece como una de las opciones de titulación al examen complejo el cual “deberá ser de carácter complejo, con el mismo nivel de complejidad, tiempo de preparación y demostración de competencias, habilidades, destrezas y desempeños, que el exigido en las diversas formas del trabajo de titulación teórico-práctico que evalúa resultados de aprendizaje definidos en el perfil de egreso de la titulación con niveles de complejidad y equivalentes a un trabajo de titulación”.

Por tanto el examen está diseñado para que el estudiante demuestre el logro de las competencias necesarias para resolver problemas de manera creativa y haciendo uso de las habilidades y conocimientos adquiridos durante su formación profesional.

Para el desarrollo del examen, la titulación ha definido los núcleos fundamentales de la profesión y que se constituyen en la base del desempeño profesional de los graduados.

Por ser de naturaleza teórico-práctica el examen contemplará dos componentes: una parte escrita que recogerá los aprendizajes y fundamentación teórica en los núcleos fundamentales de la profesión y un segundo elemento será el desarrollo de un caso práctico donde el estudiante demostrará su capacidad para resolver los problemas en su área de conocimiento.

2. Trabajo de titulación:

El Reglamento de Régimen Académico, establece que “Todo trabajo de titulación deberá consistir en una propuesta innovadora que contenga, como mínimo, una investigación exploratoria y diagnóstica, base conceptual, conclusiones y fuentes de consulta. Para garantizar su rigor académico, el trabajo de titulación deberá guardar correspondencia con los aprendizajes adquiridos en la carrera y utilizar un nivel de argumentación, coherente con las convenciones del campo del conocimiento”.

a. Opciones de trabajo de titulación

De entre las opciones dadas la Titulación de Ingeniería en Informática, considerando la naturaleza y la formación establecida en el perfil profesional establece las siguientes opciones de titulación.

Opción 1: Proyecto Técnico

Un proyecto técnico tiene como propósito el desarrollo de estudios y aplicación de los conocimientos propios de la titulación en el diseño, desarrollo e implementación de una solución basada en los principios, herramientas y técnicas del campo de las Ciencias de la Computación, los cuales pueden ser aplicados no solo en el ámbito de

- Los estudiantes podrán utilizar los laboratorios académicos o de investigación en coordinación con el tutor del trabajo de tesis y la sección departamental correspondiente.
- El proyecto asume una duración máxima de 400 horas y se contempla un tiempo de desarrollo entre 6 y 12 meses, en función del alcance planteado en la propuesta y de máximo un año en el caso de proyectos multidisciplinarios y multiprofesionales.

c. **Observaciones adicionales**

- Los proyectos de modalidad a distancia requieren tiempo adicional debido a las condiciones de trabajo de los estudiantes.

Propuesto por:

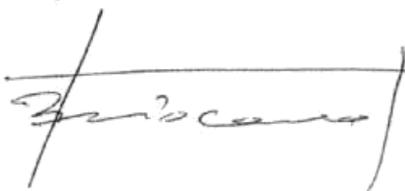


Ing. Marco Patricio Abad Espinoza
COORDINADOR DE TITULACIÓN DE
CIENCIAS DE LA COMPUTACIÓN



Aprobado por:

Junta de área:



Universidad Técnica Particular de Loja

TITULACIÓN DE INGENIERÍA EN SISTEMAS

INFORMÁTICOS Y COMPUTACIÓN

Propuesta de Trabajo de Titulación

Modalidad Presencial

A. Información General

Título del proyecto:

Nombre:

Propuesto por

Docente [] Estudiante []

Departamento/Sección:

Duración:

Especifique el tiempo en meses!

Área de Conocimiento:

Especifique el área de conocimiento del ámbito de las ciencias de la computación

Línea de Investigación:

Especifique la línea de investigación del departamento de Ciencias de la Computación y electrónica en la que se enmarca el proyecto.

Otros proyectos relacionados:

Naturaleza del Proyecto

% Técnico : *(mín 60)*

% Innovación : *(max 30)*

% Investigación : *(max 30)*

Nro de Tesistas requeridos:

máximo 3 estudiantes

B. Perfil Requerido de los Postulantes

Se puede tener estudiantes de diferentes carreras, con un número máximo de 3 estudiantes por proyecto.

	Postulante 1	Postulante 1	Postulante 3	
Áreas de afinidad requeridas en el postulante	<i>Carrera</i>	<i>Carrera</i>	<i>Carrera</i>	Observaciones
	<i>Universidad</i>	<i>Universidad</i>	<i>Universidad</i>	
Programación				
Análisis de sistemas/negocio				
Base de datos				
Gestión proyectos				
Inteligencia artificial				
Redes y telecomunicaciones				
Otras	<i>Especifique cuáles</i>	<i>Especifique cuáles</i>	<i>Especifique cuáles</i>	

C. Descripción del TT

Breve descripción del TT que incluya: problemática y justificación para la elaboración del mismo.

D. Objetivos

Describe los objetivos a alcanzar con el desarrollo del Trabajo de Titulación.

General	G	<objetivo general>
----------------	---	--------------------

Específicos	E1	<específico 1 ...>
--------------------	----	--------------------

	E2	
--	----	--

	E3	
--	----	--

	E4	
--	----	--

E. Estrategia o Metodología de desarrollo

Describir la metodología o estrategia a seguir para el desarrollo del TT

F. Componentes:

Describir los componentes que se van a desarrollar en el TT por cada uno de los niveles de GP.

Nro.	Componentes	Plazo(Meses)	Nivel del GP
1	Componente 1		
2	...		GP 4.1
3			
4	Componente 4		
5	...		GP 4.2
6			

G. Resultados esperados

Enumerar los resultados esperados con el desarrollo del TT. (Los resultados no son el esquema del documento final), puede ser aplicaciones, documentación, implementaciones, etc

 Resultado 1

 Resultado 2

 ...



H. Bibliografía / Recursos

Especifique los recursos bibliográficos que ayudarán a una mayor comprensión del tema propuesto.

 Recurso Bibliográfico 1

 Recurso Bibliográfico 1

 ...



.....

Firma Proponente

Universidad Técnica Particular de Loja

TITULACIÓN DE INGENIERÍA EN INFORMÁTICA

Propuesta de Trabajo de Titulación

Modalidad Abierta y a Distancia

A. Información General

Título del proyecto:		
Propuesto por	Nombre:	
	Docente [] Estudiante []	
	Departamento-Sección:	
Duración:	<i>Especifique el tiempo en meses!</i>	
Área de Conocimiento:	<i>Especifique el área de conocimiento del ámbito de las ciencias de la computación</i>	
Línea de Investigación:	<i>Especifique la línea de investigación del departamento de Ciencias de la Computación y electrónica en la que se enmarca el proyecto.</i>	

Otros proyectos relacionados:

Naturaleza del Proyecto	% Técnico :	<i>(min 60)</i>
	% Innovación :	<i>(max 30)</i>
	% Investigación :	<i>(max 30)</i>

Nro de Tesistas requeridos: *máximo 3 estudiantes*

B. Perfil Requerido de los Postulantes

Se puede tener estudiantes de diferentes carrera, con un número máximo de 3 estudiantes por proyecto.

	Postulante 1	Postulante 1	Postulante 3	
				Observaciones
Afinidad del postulante	<i>Carrera</i>	<i>Carrera</i>	<i>Carrera</i>	
	<i>Universidad</i>	<i>Universidad</i>	<i>Universidad</i>	
Programación				
Análisis de sistemas/negocio				
Base de datos				

Gestión proyectos

Inteligencia artificial

Redes y telecomunicaciones

Otras

C. Descripción del TT

Breve descripción del TT que incluya la problemática y justificación para la elaboración del mismo.

D. Objetivos

Describa los objetivos a alcanzar en el TT.

General	G	<objetivo general>
----------------	---	--------------------

Específicos	E1	<específico 1 ...>
--------------------	----	--------------------

E2

E3

E4

E. Estrategia o Metodología de desarrollo

Describir la metodología o estrategia a seguir para el desarrollo del TT

F. Componentes:

Describir los componentes que se van a desarrollar en el TT.

Nro.	Componentes	Plazo(meses)
1	Componente 1	
2	...	
3		
4	Componente 4	
5	...	
6		

G. Resultados esperados

Enumerar los resultados esperados con el desarrollo del TT. (Los resultados no son el esquema del documento final), puede ser aplicaciones, documentación, implementaciones, etc

 Resultado 1

 Resultado 2

 ...



H. Bibliografía / Recursos

Especifique los recursos bibliográficos que ayudarán a una mayor comprensión del tema propuesto.

 Recurso Bibliográfico 1

 Recurso Bibliográfico 1

 ...



.....

Firma Proponente

ANEXO 5: RÚBRICA PARA CALIFICACIÓN DE PROPUESTAS DE TT

Ítem	Deficiente	Regular	Bueno	Excelente	Peso %
Tema	No hay claridad en el tema, no demuestra el ámbito al cual se orienta.	El tema es poco claro, y el ámbito de aplicación no corresponde a la carrera o no está definido.	El tema es claro y específico pero tiene poca relación con las competencias del perfil profesional.	El tema es claro, específico y muy relacionado con las competencias del perfil profesional.	5
Plazo	No hay claridad en el plazo y este tampoco tiene relación con los objetivos del proyecto.	El plazo planteado no alcanza el tiempo mínimo de duración del proyecto y tampoco se evidencia el uso de al menos 400 horas.	El plazo está establecido en los tiempos previstos, sin embargo este no concuerda con el alcance establecido. Las horas de trabajo aproximado, superan en más del 10% las 400 horas establecidas.	El plazo está claramente definido y se evidencia que se completará en un tiempo de dos ciclos y asumiendo un tiempo de dedicación de aproximadamente 400 horas.	5
Componente Técnico	Inferior a 20%	Entre 21% y 49%	Más del 60%	Entre 50% y 60%	5
Componente Investigación	Inferior a 10%	Entre 11% y 20%	Más del 30%	Entre 21% y 30%	5
Componente Innovación	Inferior a 10%	Entre 11% y 20%	Más del 30%	Entre 21% y 30%	5
Integrantes	No hay claridad para establecer si el número y carrera de los participantes es el	El número de participantes es el adecuado, sin embargo las carreras establecidas no	El número de participantes es inferior al que se necesita de acuerdo al alcance del proyecto, el perfil de los	El número de integrantes es el necesario en relación al alcance, las profesiones y el plazo establecido son concordantes.	5

	adecuado.	guardan relación con el tema.	integrantes no cubre lo que necesita el proyecto.		
Descripción del trabajo	No hay descripción del trabajo, o no se entiende	La descripción del trabajo es ambigua, y no se centra en el problema.	La descripción del trabajo es incompleta.	La descripción del trabajo es específica, el alcance se evidencia con claridad.	5
Descripción del problema	La descripción del problema es insuficiente y no se evidencia necesidad de solución en ninguno de los ámbitos de la titulación.	Hay una descripción poco clara y las posibilidades de solución no tienen relación con las áreas de conocimiento de la titulación. No sustenta con literatura relevante.	Hay una descripción del problema, y las posibilidades de solución superan el nivel de ingeniería. Incluye literatura relevante.	El problema está descrito con precisión y demuestra posibilidades de solución en alguno de los ámbitos de la titulación y se sustenta en literatura de relevancia.	10
Justificación	No hay justificación del trabajo.	La justificación es ambigua.	La justificación es incompleta o no está relacionada con el problema.	La justificación es completa y detallada, se justifica con claridad el desarrollo del trabajo.	5
Objetivos	Los objetivos no definen lo que se pretende desarrollar en el proyecto y tampoco se encuentran bien detallados.	Se colocan objetivos tanto generales como específicos, sin embargo no están bien redactados y tampoco son medibles.	Objetivos bien redactados, sin embargo la relación con el tema y el problema no es clara.	Objetivos claros y medibles, muy relacionados con el tema y problema. Se vislumbra posibilidades de solución/implementación.	10
Estrategia/Metodología de Trabajo.	No hay una estrategia o	Describe vagamente una	La estrategia descrita es	La estrategia es clara y demuestra posibilidades de	10

	metodología definidas.	metodología y no se le ve la utilidad.	incompleta.	aplicación	
Componentes	Los componentes no se entienden o no están definidos.	Se definen algunos componentes, sin embargo no está clara su relación con el trabajo.	Hay claridad en los componentes, sin embargo hace falta alguna para que se complete el trabajo.	Los componentes evidencian completamente el propósito del proyecto y establecen con precisión el trabajo necesario para completar el T.T.	10
Resultados esperados	No muestra ningún resultado tangible.	Muestra resultados pero no son significativos para obtener el grado.	Los resultados esperados son significativos pero están incompletos.	Los resultados esperados son los necesarios y están claramente definidos.	10
Bibliografía	No incluye bibliografía.	La bibliografía propuesta es de poca relevancia, o es muy escasa.	La bibliografía está poco relacionada con el tema.	La bibliografía es pertinente y suficiente como para sustentar el proyecto.	10

ANEXO 14. Instalación y configuración de Heroku

Para la instalación de Heroku hay que seguir los siguientes pasos:

1. Crear una cuenta en Heroku.
2. Instalar Python localmente.
3. Instalar los gestores de paquetes *pip* y *setuptools*.
4. Instalar el gestor de entornos virtuales *virtualenv*.
5. Instalar postgres localmente
6. Instalar *Heroku CLI*, o el paquete completo *Heroku Toolbelt*

Para realizar el despliegue de la aplicación en Heroku se requiere seguir los siguientes pasos y realizar las siguientes configuraciones:

1. Crear una app en Heroku vía línea de comandos: *Heroku create utpl-sgtt*
2. Instalar el paquete *django-toolbelt*: *pip install django-toolbelt*.
3. Crear un archivo con el nombre *Procfile* en el directorio raíz
 - a. Escribir dentro del archivo: *web: gunicorn sgtt.wsgi*
4. Crear un archivo llamado *.env* puede estar vacío o contener: *TIMES=2*
5. Crear un archivo *runtime.txt* y especificar dentro la versión de python, en este caso: *python-3.5.1*.
6. Modificar el archivo de configuración *settings.py* del directorio del proyecto. El archivo debe tener la siguiente configuración:

```

# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/1.10/howto/static-files/
PROJECT_ROOT = os.path.dirname(os.path.abspath(__file__))

STATIC_ROOT = os.path.join(BASE_DIR, 'static')
STATIC_URL = '/static/'
# https://warehouse.python.org/project/whitenoise/

STATICFILES_STORAGE = 'whitenoise.django.GzipManifestStaticFilesStorage'

AUTH_USER_MODEL = 'usersUtpl.UserUtpl'

MEDIA_ROOT = 'media'
MEDIA_URL = '/media/'

import dj_database_url
db_from_env = dj_database_url.config()
DATABASES['default'].update(db_from_env)

SECURE_PROXY_SSL_HEADER = ('HTTP_X_FORWARDED_PROTO', 'https')

ALLOWED_HOSTS = ['*']

DEBUG = True

import mimetypes
mimetypes.add_type("text/css", ".css", True)

```

Figura 82. Configuración Heroku

Fuente: El autor.

Elaboración: El autor.

7. Instalar el paquete “*whitenoise*”.
8. Modificar el archivo “*wsgi.py*” del proyecto. El archivo debe contener lo siguiente:

```

import os
os.environ.setdefault("DJANGO_SETTINGS_MODULE", "sgtt.settings")

from django.core.wsgi import get_wsgi_application
from dj_static import Cling
from whitenoise.django import DjangoWhiteNoise

application = get_wsgi_application()
# application = Cling(get_wsgi_application())
application = DjangoWhiteNoise(application)
# Local config:
# application = get_wsgi_application()

```

Figura 83. Configuración Heroku

Fuente: El autor.

Elaboración: El autor.

9. Agregar las modificaciones realizadas con la sentencia git: “*git add .*”
10. Desplegar a Heroku con el comando: “*git push heroku master*”.
11. Ejecutar las migraciones en el servidor Heroku con el comando: “*heroku run python manage.py migrate*”.

ANEXO 15. Formularios para pruebas y validación

1. Proceso de negocio, ¿A su criterio, el lanzamiento de convocatoria en el sistema satisface el proceso de oferta de convocatorias en la titulación? 

Muy satisfactorio

Satisfactorio

Poco Satisfactorio

Insatisfactorio

Agrega un comentario y observación (opcional)

2. ¿Qué tan complicado le pareció el lanzamiento de convocatoria? 

Muy Sencillo

Sencillo

Complicado

Muy Complicado

Ingrese un comentario (opcional)

3. ¿Cuál fue el tiempo de espera para que se guarde la convocatoria? 

Muy Rápido

Rápido

Lento

Muy Lento

4. ¿Los datos guardados para la convocatoria son los correctos? 

Si

No

Figura 84. Validación escenario “Lanzar convocatoria”

Fuente: El autor.

Elaboración: El autor.

5. Proceso de Negocio. ¿La creación de esquemas de evaluación satisface el proceso de evaluación objetiva de propuestas en la titulación? 

Muy satisfactorio

Satisfactorio

Poco satisfactorio

Insatisfactorio

Comentario (opcional)

6. ¿Qué tan complicado le pareció la creación de esquema de evaluación de propuestas para T.T.? 

Muy sencillo

Sencillo

Complicado

Muy complicado

Comentario (opcional)

7. ¿Cuál fue el tiempo de espera para que se guarde el esquema de evaluación? 

Muy rápido

Rápido

Lento

Muy lento

8. ¿Los datos guardados para el esquema de evaluación son correctos? 

Sí

No

Figura 85. Validación escenario “Crear esquema evaluación”
Fuente: El autor.
Elaboración: El autor.

9. Proceso de negocio, ¿La subida de propuestas satisface el proceso de oferta de Propuestas para Trabajos de Titulación? 

- Muy satisfactorio
- Satisfactorio
- Poco satisfactorio
- Insatisfactorio

Comentario(opcional)

10. ¿Qué tan complicado le pareció la subida de Propuesta para Trabajo de Titulación? 

- Muy sencillo
- Sencillo
- Complicado
- Muy complicado

Comentario (opcional)

11. ¿Cómo fue la velocidad para que se guarde la Propuesta para Trabajo de Titulación? 

- Muy rápido
- Rápido
- Lento
- Muy lento

12. ¿Los datos guardados para la Propuesta son los correctos? 

- Sí
- No

Figura 86. Validación escenario “Subida de Propuesta”

Fuente: El autor.

Elaboración: El autor.

13. Proceso de negocio, ¿La calificación de propuesta satisface el proceso de Revisión de Propuestas para Trabajos de Titulación? 

- Muy satisfactorio
- Satisfactorio
- Poco satisfactorio
- Insatisfactorio

Comentario (opcional)

14. ¿Qué tan complicado le pareció la calificación de Propuesta para Trabajo de Titulación? 

- Muy sencillo
- Sencillo
- Complicado
- Muy complicado

Comentario (opcional)

15. ¿Cómo fue la velocidad para que se realice y se guarde la Calificación de Propuesta para Trabajo de Titulación? 

- Muy rápido
- Rápido
- Lento
- Muy lento

16. ¿Los datos guardados para la Calificación de Propuesta son los correctos? 

- Sí
- No

Figura 87. Validación escenario “Calificación de Propuesta”
Fuente: El autor.
Elaboración: El autor.

17. Proceso de negocio, ¿Esta validación satisface el proceso de Aprobación de Propuestas para Trabajos de Titulación? 

- Muy satisfactorio
- Satisfactorio
- Poco satisfactorio
- Insatisfactorio

Comentario (opcional)

18. ¿Qué tan complicado le pareció la validación de Propuesta para Trabajo de Titulación? 

- Muy sencillo
- Sencillo
- Complicado
- Muy complicado

Comentario (opcional)

19. ¿Cómo fue la velocidad para que se guarde la Validación de Propuesta para Trabajo de Titulación? 

- Muy rápido
- Rápido
- Lento
- Muy lento

20. ¿Los datos guardados para la Validación de Propuesta son los correctos? 

- Sí
- No

Figura 88. Validación escenario "Validación de revisión"

Fuente: El autor.

Elaboración: El autor.

21. Proceso de negocio, ¿Esta validación satisface el proceso de Autorización y Postulación de estudiantes a Propuestas para Trabajos de Titulación? 

- Muy satisfactorio
- Satisfactorio
- Poco satisfactorio
- Insatisfactorio

Comentario (opcional)

22. ¿Qué tan complicado le pareció la Autorización de Propuestas para Trabajos de Titulación? 

- Muy sencillo
- Sencillo
- Complicado
- Muy complicado

Comentario (opcional)

23. ¿Cómo fue la velocidad para que se guarde la Autorización de Propuestas para Trabajo de Titulación? 

- Muy rápido
- Rápido
- Lento
- Muy lento

24. ¿Los datos guardados para la Autorización de Propuesta son los correctos?



- Sí
- No

Figura 89. Validación escenario “Autorización de propuestas”

Fuente: El autor.

Elaboración: El autor.

25. Proceso de negocio, ¿Esta validación satisface el proceso de Postulación y Notificación de estudiantes a Propuestas para Trabajos de Titulación? 

- Muy satisfactorio
- Satisfactorio
- Poco satisfactorio
- Insatisfactorio

Comentario (opcional)

26. ¿Qué tan complicado le pareció la consulta y notificación de estudiantes de Propuestas para Trabajos de Titulación? 

- Muy sencillo
- Sencillo
- Complicado
- Muy complicado

Comentario (opcional)

27. ¿Cómo fue la velocidad para que se consulte y se notifique la asignación de Propuestas para Trabajo de Titulación? 

- Muy rápido
- Rápido
- Lento
- Muy lento

28. ¿Los datos guardados para el estudiante de la Propuesta son los correctos?



- Si
- No

Figura 90. Validación escenario “Consumo de estudiantes para Propuesta”

Fuente: El autor.

Elaboración: El autor.