



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

ÁREA TÉCNICA

**TÍTULO DE INGENIERO EN SISTEMAS INFORMATICOS Y
COMPUTACIÓN**

**Marco de trabajo para control de calidad en el desarrollo de aplicaciones
móviles bajo metodologías ágiles**

TRABAJO DE TITULACIÓN.

AUTORA: Puglla Remache, Gabriela Noemí

DIRECTORA: Romero Peláez, Audrey Elizabeth, Mgs

**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 DE LA DIRECTORA DEL TRABAJO DE TITULACIÓN

Ingeniera.

Audrey Elizabeth Romero Peláez.

DOCENTE DE LA TITULACIÓN

De mi consideración:

El presente trabajo de titulación: **Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles bajo metodologías ágiles"** realizado por **Puglla Remache Gabriela Noemí**; ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo.

Loja, Marzo de 2017

f)

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

“Yo **Puglla Remache Gabriela Noemí** declaro ser autora del presente de titulación: Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles bajo metodologías ágiles, de la Titulación Sistemas Informáticos y Computación, siendo Ing. Audrey Elizabeth Romero Peláez directora 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 y resultados vertidos en el presente trabajo investigativo, 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 o trabajos de titulación que se realicen con el apoyo financiero, académico o institucional (operativo) de la Universidad”.

f.
Autor Gabriela Noemí Puglla Remache
Cédula **1103761191**

DEDICATORIA

Con inmenso amor y gratitud a mis queridos padres, Manuel y Rosario; por su paciencia y sacrificio, quienes hicieron todo en la vida para que yo pudiera llegar a este momento tan importante en mi vida y cumplir con esta meta tan anhelada.

A mis queridos hermanos Mayra, Diego y Thalía por su apoyo incondicional, su comprensión y cariño; y a mis sobrinos adorados Damiancito y Alejandrino por su inocente y hermosa compañía, gracias por existir y estar ahí conmigo siempre.

A mi familia, mis amigos y compañeros que me han acompañado y compartido los buenos y malos momentos.

Con mucho cariño.

Gabriela.

AGRADECIMIENTO

Mi sincero agradecimiento a la Universidad Técnica Particular de Loja, de manera muy especial a la Ing. Audrey Romero quien con su apoyo incondicional, sus palabras de aliento, sus consejos y confianza me ha sabido guiar desde el inicio de este trabajo y así poder llegar a la culminación de mi carrera.

Así mismo, extender mi agradecimiento a la empresa AQA Solutions, por darme la oportunidad de realizar las pasantías que me sirvieron de gran ayuda en el desarrollo de este trabajo, de manera muy especial a la Ing. Carlina Rueda y la Ing. Diana Cuenca por sus consejos, sugerencias y pautas que me sirvieron de guía en la culminación de la tesis, por brindarme sus conocimientos, confianza y apoyo. Mil gracias.

Con gratitud.

Gabriela

INDICE DE CONTENIDOS

APROBACIÓN DE LA DIRECTORA DEL TRABAJO DE TITULACIÓN	ii
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	iii
DEDICATORIA.....	iv
AGRADECIMIENTO.....	v
INDICE DE CONTENIDOS.....	vi
INDICE DE FIGURAS.....	ix
INDICE DE TABLAS	x
RESUMEN.....	1
ABSTRACT	2
INTRODUCCIÓN.....	3
CAPÍTULO 1	5
ESTADO DEL ARTE.....	5
1.1 Metodologías tradicionales de desarrollo de software	6
1.2 Desarrollo ágil de software.....	7
1.3 Manifiesto Ágil.....	7
1.4 Metodologías ágiles	8
1.5 Metodologías ágiles frente a metodologías tradicionales	9
1.6 Los sistemas transaccionales y las aplicaciones móviles	11
1.6.1 Aplicaciones móviles	12
1.6.2 Características de las aplicaciones móviles en el entorno de desarrollo	12
1.6.3 Clasificación de las aplicaciones móviles según el entorno de desarrollo	14
1.7 Sistemas operativos en los dispositivos móviles	16
1.7.1 Comparativa de sistemas operativos en tecnología móvil.....	17
1.8 Metodologías ágiles aplicadas en el desarrollo de aplicaciones móviles	19
1.8.1 Selección de palabras claves para la búsqueda	20
1.8.2 Criterios en la selección de los estudios	20
1.8.3 Selección de los estudios de las metodologías ágiles en el desarrollo de aplicaciones móviles.....	21
1.8.4 Resultados de la revisión de los estudios	22
1.8.5 Metodología ágil “Mobile-D”.....	23
1.9 Calidad en el desarrollo de las aplicaciones móviles	27
1.9.1 Gestión de calidad.....	27

1.9.2	Control de calidad.....	29
1.9.3	Pruebas de calidad del software	31
1.9.4	Pruebas de calidad aplicadas en el desarrollo de las aplicaciones móviles	32
1.9.5	Factores, modelos y estándares de calidad.....	35
1.9.6	Métricas y herramientas para evaluar las aplicaciones móviles	38
1.9.7	Herramientas para automatización de pruebas de aplicaciones móviles	39
1.9.8	Emuladores para aplicaciones móviles	40
1.10	Discusión final	40
CAPÍTULO 2.	41
DEFINICIÓN DEL PROBLEMA.....	41
2.1	Introducción.....	42
2.2	Descripción de la definición del problema	42
CAPÍTULO 3.	49
DISEÑO DE LA SOLUCIÓN. MARCO DE TRABAJO	49
3.1	Introducción.....	50
3.2	Diseño de la solución.....	50
3.2.1	Componentes del marco de trabajo	50
3.2.2	Diseño del MTCC	51
3.3	Despliegue del marco de trabajo.....	54
3.3.1	Análisis y validación del proyecto – Fase de Exploración	54
3.3.2	Estimación y planificación del control de calidad – Fase de Exploración	56
3.3.3	Análisis y diseño de pruebas – Fase de inicialización.....	58
3.3.4	Ejecución de las pruebas – Fase de pruebas del sistema	62
3.3.5	Evaluación de los criterios de salida e informes y cierre de las pruebas – Fase de pruebas del sistema.....	64
3.3.6	Análisis de los indicadores de calidad – Fase de Pruebas del sistema.....	66
3.3.7	Clientes	66
3.3.8	Cierre del proceso de control de calidad – Fase de pruebas del sistema.....	67
CAPÍTULO 4.	69
VALIDACIÓN DEL MARCO DE TRABAJO	69
4.1	Validación del marco de trabajo	70
4.2	Caso práctico – “Gestión de tutorías UTPL”	70
4.3	Implementación y validación del MTCC.....	70
4.3.1	Situación inicial del proyecto seleccionado	71
4.3.2	Identificación de roles	71

4.4	Análisis de resultados del MTCC	86
	CONCLUSIONES.....	88
	RECOMENDACIONES.....	89
	BIBLIOGRAFÍA.....	90
	ANEXOS	94
	Anexo 1. Estudios de las metodologías ágiles en el entorno de desarrollo de aplicaciones móviles	95
	Anexo 2. Plan del proyecto.....	108
	Anexo 3. Documento de especificación de requerimientos (ERS)	109
	Anexo 4. Checklist del proyecto	110
	Anexo 5. Plan de calidad	111
	Anexo 6. Plan de medición	112
	Anexo 7. Plan de pruebas.....	113
	Anexo 8. Casos de pruebas.....	114
	Anexo 9. Matriz de reportes de bugs e issues	115
	Anexo 10. Pruebas de usabilidad/Encuestas Usuarios finales.....	116
	Anexo 11. Informe de ejecución de pruebas	117
	Anexo 12. Informe de cierre de proceso.....	118

INDICE DE FIGURAS

Figura 1.1. Metodologías ágiles en el desarrollo de software.....	9
Figura 1.2. Metodología seleccionada	22
Figura 1.3. Modelo de verificación y validación.....	31
Figura 1.4. Pruebas aplicadas en el desarrollo de aplicaciones móviles	32
Figura 1.5. Modelo de calidad ISO/EC 9126-1	37
Figura 1.6. Modelo de calidad interna y externa y sus métricas	37
Figura 2.1. Diagrama Causa-Efecto. Deficiencia del control de calidad en el desarrollo de aplicaciones móviles.....	43
Figura 3.1. Diagrama de procesos SIPOC.....	51
Figura 3.2. Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles	53
Figura 3.3. Proceso de análisis y validación del proyecto	54
Figura 3.4. Proceso de estimación y planificación del control de calidad	56
Figura 3.5. Proceso de análisis y diseño de pruebas	58
Figura 3.5.1. Plantilla de casos de prueba.....	60
Figura 3.6. Proceso de ejecución de las pruebas.....	62
Figura 3.7. Proceso de evaluación de los criterios de salida y cierre de las pruebas	64
Figura 3.8. Proceso de análisis estadístico de los indicadores de calidad.....	66
Figura 3.9. Proceso de cierre del proceso de control de calidad	67
Figura 4.1. Proceso de análisis y validación del proyecto “Gestión de tutorías UTPL”	72
Figura 4.2. Proceso de estimación y planificación del control de calidad del proyecto “Gestión de tutorías UTPL	74
Figura 4.3. Proceso de análisis y diseño de pruebas “Gestión de tutorías UTPL”	76
Figura 4.4. Proceso de ejecución de las pruebas “Gestión de tutorías UTPL”	78
Figura 4.5. Proceso de Evaluación de los criterios de salida e informes y cierre de pruebas “Gestión de tutorías UTPL”	80
Figura 4.6. Proceso de Evaluación de los criterios de salida e informes y cierre de pruebas “Gestión de tutorías UTPL”	82
Figura 4.7. Cierre del proceso del control de calidad. “Gestión de tutorías UTPL”	84

INDICE DE TABLAS

Tabla 1.1. Comparativa entre las metodologías ágiles frente a las tradicionales.....	10
Tabla 1.2.Tabla comparativa de sistemas operativos móviles.....	18
Tabla 1.3. Metodologías más utilizadas en el entorno móvil	22
Tabla 1.4. Metodología ágil Mobile-D.....	23
Tabla 1.5. Matriz de evaluación de las fases de Mobile-D en el desarrollo de aplicaciones móviles con respecto al ciclo de vida de desarrollo de software (CVDS).	25
Tabla 1.6. Control de calidad del proyecto	29
Tabla 1.7. Métricas para evaluar las aplicaciones móviles	38
Tabla 2.1.Detalle de las causas encontradas en la deficiencia del control de calidad en el desarrollo de las aplicaciones móviles.	45
Tabla 4.1. Artefactos del MTCC	85

RESUMEN

En el ámbito del desarrollo de las aplicaciones móviles el uso de las metodologías ágiles y un proceso de calidad adecuado permiten obtener un producto exitoso y de calidad. La elaboración del marco de trabajo para el control de calidad en el desarrollo de las aplicaciones móviles permite establecer principalmente la interacción entre las fases de desarrollo de una metodología ágil seleccionada y las fases del proceso de calidad definidos en el desarrollo de este trabajo, con el propósito de facilitar y agilizar desde la concepción de la idea, el diseño, la construcción y validación del software a través del uso de artefactos, métricas de calidad, técnicas y la ejecución de pruebas adecuadas a este tipo de aplicaciones y obtener un producto de calidad acorde a las necesidades y exigencias del entorno y de los usuarios.

PALABRAS CLAVES: metodologías ágiles, dispositivos móviles, aplicaciones móviles, marco de trabajo, control de calidad.

ABSTRACT

In the field of mobile application development, the use of agile methodologies and a process of adequate quality make it possible to obtain a successful and quality product. The elaboration of the framework for quality control in the development of mobile applications allows to establish mainly the interaction between the development phases of a selected agile methodology and the phases of the quality process defined in the development of this work, with the Purpose of facilitating and streamlining from the conception of the idea, design, construction and validation of the software through the use of artifacts, quality metrics, techniques and the execution of appropriate tests to this type of applications and obtain a quality product According to the needs and demands of the environment and the users.

KEY WORDS: agile methodologies, mobile devices, mobile applications, framework, quality control.

INTRODUCCIÓN

En este proyecto se plantea realizar el estudio de las metodologías ágiles en el desarrollo de aplicaciones móviles; permitiendo así, desarrollar un marco de trabajo principalmente para el proceso de Calidad (Validación), que estará constituido de modelos, estándares, técnicas, procesos y herramientas de prueba para aplicaciones móviles que garantice y ofrezca un producto de calidad a los usuarios finales; además, haciendo uso de este marco de trabajo en aplicaciones desarrolladas en las plataformas Android e iOS, por estar designadas como los entornos de desarrollo y usabilidad de mayor demanda en nuestro medio. En la actualidad el uso de dispositivos móviles se ha convertido en la demanda más grande de consumo de la sociedad, así mismo la oferta de aplicaciones móviles es muy amplia y ofrece una gran diversidad de categorías de acuerdo a las necesidades del consumidor final.

En el desarrollo de aplicaciones móviles la intervención de las metodologías ágiles juega un papel importante estableciendo cada una de las fases que facilitan el diseño, la construcción y validación del software, ya que se acogen de manera rápida y fácil a los equipos de trabajo. Además de la necesidad y la demanda de incluir un proceso de calidad eficiente y acorde a este ambiente de desarrollo.

El primer capítulo consta de una investigación detallada de los temas relacionados con el desarrollo del proyecto, en los que se hace referencia a las metodologías ágiles aplicadas en el entorno de desarrollo de aplicaciones móviles, los sistemas transaccionales, las características y la clasificación de las aplicaciones móviles en el entorno de desarrollo, los sistemas operativos en los dispositivos móviles, la calidad en el desarrollo de las aplicaciones que abarcan subtemas en relación a las pruebas, modelos, métricas, herramientas de automatización y emuladores en el entorno de desarrollo de aplicaciones móviles.

El segundo capítulo detalla la descripción del problema a través del diagrama causa-efecto que permite identificar la deficiencia de la calidad en el desarrollo de las aplicaciones móviles.

El tercer capítulo presenta el diseño de la solución, la creación del Marco de Trabajo para el control de calidad en el desarrollo de aplicaciones móviles,

El cuarto capítulo valida el Marco de trabajo a través de un caso práctico y presenta los resultados de validación.

OBJETIVOS

Objetivos generales

- Definir un marco de trabajo para control de calidad en el desarrollo de aplicaciones móviles bajo metodologías ágiles.

Objetivos específicos

- Realizar estado del arte del control de calidad de aplicaciones móviles bajo metodologías ágiles.
- Diseñar una propuesta de un marco de trabajo para realizar control de calidad en el desarrollo de aplicaciones móviles.
- Desplegar el marco de trabajo definiendo cada una de las fases de desarrollo.
- Validar el funcionamiento del marco de trabajo desarrollado.

La metodología que se propone para este proyecto comprende los siguientes pasos:

- Búsqueda bibliográfica de información
- Selección de información
- Estudio analítico de las metodologías ágiles en las aplicaciones móviles.
- Diseño de la estructura del marco de trabajo para aplicaciones móviles.
- Desarrollo de los componentes del marco de trabajo.
- Pruebas de Validación del marco de trabajo en un entorno real.
- Evaluación y discusión de resultados
- Elaboración de conclusiones y recomendaciones

CAPÍTULO 1
ESTADO DEL ARTE

1.1 Metodologías tradicionales de desarrollo de software

(Amaya Balaguera, 2013) *Utiliza la definición general. “Una metodología de desarrollo de software es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una metodología está formada por fases, cada una se puede dividir en sub-fases que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto a planificarlo, gestionarlo, controlarlo y evaluarlo.”*

“Las metodologías tradicionales también llamadas convencionales tienden a acumular riesgos y dificultades que surgen en el desarrollo del producto al final del proyecto; es decir, repercutiendo en retrasos en la entrega de funciones o influyendo en la incorrecta ejecución de las últimas fases del ciclo de vida.” (Rodríguez González, 2008). Las metodologías tradicionales de desarrollo de software se basan en una planificación previa; es decir, inician el desarrollo de un proyecto con un proceso de elicitación de requerimientos, previo a la etapa de análisis y diseño (Cadavid, Daniel, Martínez, & Vélez, 2013) .

El desarrollo de software a través del uso de las metodologías tradicionales se tornaba un trabajo muy riguroso y cada vez se convertía en un proceso no apto en cuanto a la variabilidad y evolución de los constantes cambios para obtener un producto de calidad. La definición del proyecto se considera como uno solo; es decir, de grandes dimensiones y estructura establecida, se sigue un proceso secuencial en una sola dirección y sin marcha atrás, el proceso de desarrollo es rígido y no cambia, la definición de los requerimientos son acordados de una vez para todo el proyecto, y sobre todo la comunicación con el cliente se establece una vez terminado el proyecto (Cadavid et al., 2013).

Entre las principales metodologías de desarrollo tradicionales más conocidas están RUP y MSF, que se centran en presentar una exhaustiva documentación, planificación y procesos de todo el proyecto.

Las metodologías tradicionales o convencionales se caracterizan por depender del uso excesivo de la documentación, que muchas de las veces se la considera como un recurso no productivo ya que no representa lo que en realidad se está desarrollando y más bien al elaborar consistía en una pérdida valiosa de recursos. Entonces; es así que como alternativa relevante, trascendental y de fácil aceptación surgieron y evolucionaron a través del desarrollo ágil las denominadas metodologías ágiles que se adaptan de mejor manera al desarrollo del software, definiéndolas como apropiadas cuando los requisitos son emergentes y cambian rápidamente. (Rodríguez González, 2008), (Flora & Wang, 2014).

En este caso, en el ambiente del desarrollo de aplicaciones móviles, tomando como base el análisis y estudio de Gartner¹, considera que las prácticas del desarrollo tradicional no funcionan satisfactoriamente en el desarrollo de las aplicaciones móviles debido a las características que poseen, como la diversidad de dispositivos existentes, las plataformas de desarrollo, la conectividad de la red y de otras consideraciones específicas en las que se incluyen longitudes de sesión que se hacen más cortas y la presentación restringida, debido a las limitaciones de tamaño de la pantalla en la que funciona una aplicación móvil; de tal manera, establece que el desarrollo ágil es esencial para el desarrollo de aplicaciones móviles, debido a los constante cambios que se presenta en el mejoramiento de la aplicación móvil en relación a satisfacer las expectativas de los usuarios finales.

1.2 Desarrollo ágil de software.

(Pressman, 2010) Establece que el desarrollo ágil es una combinación de una filosofía y un conjunto de lineamientos de desarrollo. La filosofía pone énfasis en: la satisfacción del cliente y en la entrega rápida de software incremental y los lineamientos de desarrollo enfatizan la entrega sobre el análisis y diseño y la comunicación activa y continua entre desarrolladores y clientes.

El desarrollo ágil de software es un modelo de desarrollo basado en el **desarrollo iterativo e incremental**; es decir, que el desarrollo de software se divide en unidades de tiempo establecidas llamadas iteraciones, en la que se incluye: análisis de requisitos, diseño, codificación, revisión y documentación, en un periodo de tiempo rápido de una a cuatro semanas de duración. El desarrollo ágil se basa en los cuatro valores éticos y los doce principios del manifiesto ágil.

1.3 Manifiesto Ágil.

“El Manifiesto Ágil² es el documento que sirven de guía al desarrollo ágil de software.” Mediante la reunión que se llevó a cabo en 2001 entre Kent Beck y los otros 16 impulsores y creadores de las metodologías, fue discutir sobre los cuatro valores y los doce principios que facilitarían el desarrollo de software de manera rápida y eficaz; así mismo respondiendo a los cambios que surgen a lo largo del desarrollo de un proyecto. La idea era ofrecer una alternativa a los procesos de desarrollo tradicionales, caracterizados por su rigidez y dirigidos por la documentación que se genera en cada etapa (Beck, 2011).

¹Gartner Says Traditional Development Practices Will Fail for Mobile Apps. Disponible en: <http://www.gartner.com/newsroom/id/2823619>

² Manifiesto ágil. Disponible en: <http://www.agilemanifesto.org/iso/es/>

Basándose en los “cuatro valores del manifiesto ágil” se constituyeron los “doce principios” que permitieron establecer las diferencias entre un desarrollo de manera ágil y uno convencional.

La definición de estos principios marca el ciclo de vida de un desarrollo ágil, así como las prácticas y procesos a utilizar. A partir de este manifiesto muchos impulsores y desarrolladores de software, tomaron en cuenta y analizaron cada uno de los valores y principios e implementaron lo que hoy en día se conoce y se utiliza como las metodologías ágiles. En la actualidad existen muchos métodos/metodologías ágiles para desarrollar software de manera ágil.

1.4 Metodologías ágiles

En (Gimson, L. 2012), mediante el análisis de varios autores, consideran que: Un modelo es ágil o liviano cuando se emplea para su construcción una herramienta o técnica sencilla, que apunta a desarrollar un modelo aceptable, bueno y suficiente en lugar de un modelo perfecto y complejo. Un modelo es suficientemente bueno cuando cumple con los objetivos para los que fue creado, esto es, lograr principalmente el propósito de la comunicación, que sea entendible por la audiencia a la que está destinada, que posea un grado de detalle adecuado; suma valor al proyecto; y es suficientemente simple de construir; es decir, las metodologías ágiles son flexibles, modificables a la realidad de cada equipo y proyecto a desarrollar.

El objetivo de las metodologías ágiles es proponer una aplicación conjunta, equilibrada y efectiva de forma que se complementen con ideas desde la perspectiva del negocio, los valores humanos y el trabajo.

El desarrollo de un proyecto se vuelve ágil ya que se subdividen en proyectos más pequeños mediante una lista ordenada de características. Se desarrollan de manera independiente en periodos de tiempo cortos llamadas iteraciones, la comunicación con el cliente es constante, se adaptan a los cambios de manera rápida y tanto el producto como el proceso son mejorados frecuentemente sin mayores inconvenientes (Cadavid et al., 2013).

AGILE METHODS AND PRACTICES

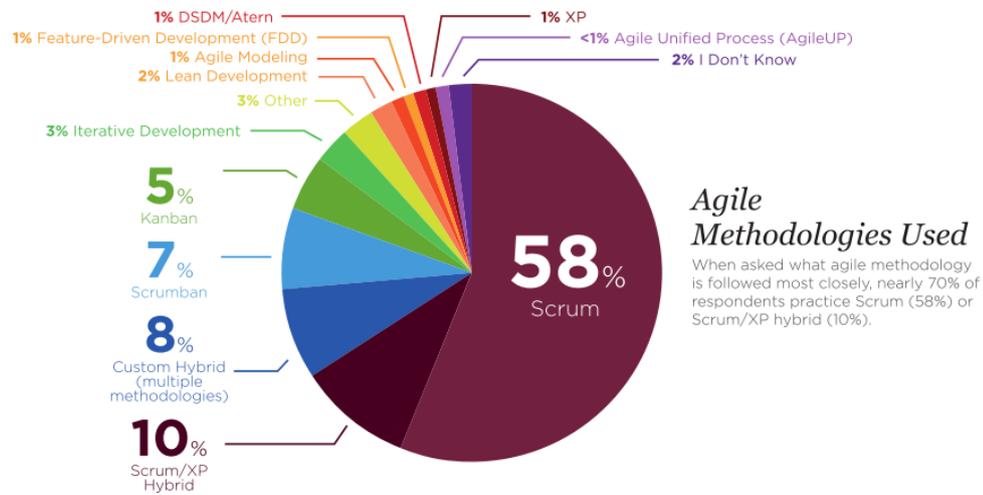


Figura 1.1. Metodologías ágiles en el desarrollo de software

Fuente: (Version One, 2016)

A través del manifiesto ágil, a lo largo de este tiempo se han diseñado metodologías ágiles de gran relevancia e interés, en las que se destacan las más importantes y de mayor aceptación en la figura 1.1. que se presenta en el décimo informe anual de (Version One, 2016), dentro de la lista de metodologías ágiles consideradas las más populares y utilizadas en la actualidad para el desarrollo de software ágil y que son utilizadas por los equipos de desarrollo de software de las empresas que ofrecen ese tipo de productos.

1.5 Metodologías ágiles frente a metodologías tradicionales

Las metodologías ágiles se establecieron bajo las características más importantes y relevantes de las metodologías tradicionales y del manifiesto ágil. En la Tabla 1.1 se presentan las principales diferencias/comparativas entre las metodologías ágiles y las metodologías tradicionales³, que se deben tener en cuenta al momento de incluir y adaptar en el desarrollar un proyecto.

Además, se incorporan cuatro características o criterios a la comparativa entre las metodologías, estos son: tiempo, ciclo de desarrollo, proceso de calidad, complejidad del software; que se definen respectivamente, como: a planificación y estimación de los periodos de tiempo en el desarrollo, el proceso de desarrollo de los requerimientos del proyecto, el proceso de control de calidad que se aplica a un proyecto en todas las fases de desarrollo y la complejidad del software con el objetivo de obtener un producto de calidad.

³ También llamadas convencionales

Estos parámetros han sido establecidos como resultado del estudio de las metodologías, y que se pueden tomar en cuenta para fortalecer la comparativa entre las metodologías ágiles y las tradicionales.

Tabla 1.1. Comparativa entre las metodologías ágiles frente a las tradicionales

Criterio	Metodologías ágiles	Metodologías tradicionales
Normas, políticas y estándares.	Se basan en heurísticas que provienen de prácticas de producción de código	Se basan en normas provenientes de estándares seguidos por el entorno de desarrollo.
Requerimientos	Están preparadas para cambios durante el proyecto	Existe cierta resistencia a los cambios
Metodologías	Impuestas internamente. (Por el equipo del proyecto)	Impuestas externamente. (Por los proveedores)
Procesos	Los procesos son menos controlados, con pocos principios.	Los procesos son mucho más controlados, con numerosas políticas/normas
	No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado
Recursos	El cliente es parte del equipo de desarrollo (Activo).	El cliente interactúa con el equipo de desarrollo mediante reuniones (Pasivo).
	<ul style="list-style-type: none"> • Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio. • Equipos pre-estructurados. 	<ul style="list-style-type: none"> • Grupos grandes y posiblemente distribuidos. • Equipo auto-organizados
	Pocos artefactos	Más artefactos
	Pocos roles	Más roles
Arquitectura	Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos.
*Tiempo	Se establecen periodos de tiempo cortos para desarrollar cada iteración.	Los periodos de tiempo para cada fase de desarrollo se establecen al inicio del proyecto.
*Ciclo de desarrollo	Es iterativo e incremental	En cascada, espiral, secuencial.
*Proceso de Calidad	Es proactivo y se ejecuta de forma continua en cada una de las fases de desarrollo del software.	Es reactivo y se ejecuta al finalizar el desarrollo del software.
*Complejidad del software	<ul style="list-style-type: none"> • Se divide en partes pequeñas (tareas específicas/iteraciones). • Se minimizan los riesgos del desarrollo. • Se aplica de manera eficiente en el desarrollo de proyectos pequeños, de requerimientos inestables y a los cambios de versiones constantes. 	<ul style="list-style-type: none"> • Se desarrolla de manera secuencial cada una de las fases definidas. • Se aplica en el desarrollo de grandes proyectos.

Fuente: (García Rodríguez, 2015),(*) Elaborada por el autor.

Una vez realizada la comparativa de las características relevantes entre las metodologías ágiles y tradicionales se determina y establece que la mejor opción en el desarrollo de un proyecto de software en la actualidad son las metodologías ágiles ya que son adaptativas y orientadas a las personas. Estas metodologías permiten definir mejores prácticas y parámetros apropiados en la refactorización de actividades.

Tomando como referencia el análisis y estudio de Gartner en su artículo “Gartner Says Traditional Development Practices Will Fail for Mobile Apps” establece que el desarrollo ágil es esencial para el desarrollo de aplicaciones móviles, debido a las características y consideraciones específicas que estas poseen, de los dispositivos móviles y los sistemas operativos en los que va a funcionar este tipo de aplicaciones.

1.6 Los sistemas transaccionales y las aplicaciones móviles

Dentro de una empresa los sistemas transaccionales o los sistemas de información se desarrollan con el propósito de lograr la automatización de sus principales procesos operativos. Los sistemas transaccionales:

- Agilizan y automatizan los procesos (tareas) operativos dentro de una empresa con el propósito de lograr ahorros significativos de mano de obra.
- Almacenan grandes cantidades de información.
- Gestionan bases de datos.
- Transforman tareas complejas para los seres humanos en actividades más sencillas.
- Son fácilmente adaptables a paquetes de aplicación que se encuentran en el mercado.
- Sus beneficios son palpables en la organización por el ahorro de tiempo y costos.

El desarrollo de los sistemas de información no resulta una tarea sencilla, a pesar de adquirir tecnologías de información, si no se toman en cuenta las necesidades existentes de la empresa estas podrían fracasar poniendo en peligro la supervivencia de la misma. Para ello resulta fundamental seguir los pasos en el desarrollo de los sistemas de información basándose en el CVDS y cada una de sus fases:

- Definición del proyecto
- Análisis de sistemas
- Diseño de sistemas
- Programación
- Fase de pruebas
- Conversión
- Producción y mantenimiento

En los sistemas transaccionales se identifican a las **aplicaciones transaccionales**, que ejecutan tareas puntuales; es decir, se definen como un conjunto de operaciones específicas y se desarrollan de acuerdo a las necesidades que se presentan o le hacen a

falta a un sistema transaccional⁴. Además este tipo de aplicaciones se las puede incluir dentro de la tecnología OLTP⁵ que es utilizada en innumerables aplicaciones cuyo beneficio se basa en la simplicidad y la eficiencia en la gestión del procesamiento de transacciones en línea que están siendo implementadas en varias industrias.

En el contexto de este trabajo se pueden considerar a las aplicaciones móviles como aplicaciones transaccionales ya que estas ejecutan tareas específicas y pueden formar parte de un sistema transaccional y cumplir con los principales objetivos antes mencionados.

1.6.1 Aplicaciones móviles

Las aplicaciones móviles son pequeños programas diseñados para funcionar de manera específica en un dispositivo móvil. Se generan en un entorno dinámico e incierto debido a que surgen de acuerdo a las necesidades del entorno y están destinadas a un número masivo de usuarios finales. Además, las aplicaciones móviles son liberadas en versiones rápidas para poder satisfacer las demandas del mercado (Delía, 2014).

Este tipo de aplicaciones son muy diferentes a las aplicaciones web o de escritorio; es decir, que se debe tener en cuenta que poseen características y condiciones adicionales para su correcto funcionamiento, como:

- La capacidad para comunicarse desde cualquier lugar.
- La interfaz de usuario para una pantalla y teclados pequeños difiere de forma significativa de la interfaz de una aplicación diseñada para un ordenador de sobremesa o un portátil.
- Los tipos de comunicación son diferentes, los dispositivos móviles incorporan capacidades de voz, mensajería, información de geo localización y vídeo conferencia (en algunos teléfonos), etc.
- El correcto funcionamiento de una aplicación móvil, depende de la calidad de la señal y de la disponibilidad de conexión de las redes inalámbricas cuando la requiere.

1.6.2 Características de las aplicaciones móviles en el entorno de desarrollo

Según (Flora & Wang, 2014), (Amaya Balaguera, 2013), (Blanco & ál, 2009) y , (Spataru, 2010), en el entorno de desarrollo de las aplicaciones móviles se debe tener en cuenta que existen características y condiciones especiales que son parte del entorno de la tecnología móvil.

⁴ Sistema transaccional. Disponible en: <http://www.jfsalazar.com/sistemas1/doc5.pdf>

⁵ OLTP. Disponible en: <https://es.wikipedia.org/wiki/OLTP>

Entre las características más importantes que se debe tener en cuenta en el desarrollo de las aplicaciones móviles están las siguientes:

- **La diversidad de los Sistemas operativos:** se relaciona a la diversidad de los sistemas operativos y las versiones del software que están incluidos en los teléfonos móviles, en relación a los requerimientos y exigencias que se presentan al momento de descargar una aplicación,
- **El diseño y usabilidad de la aplicación:** cada uno de los fabricantes y sistemas operativos tienen sus propios estándares de diseño de interfaz, programación, usabilidad, etc.
- **El tiempo de salida al mercado “Time-to-Market”:** se considera el dinamismo de la industria tecnológica; tomando en cuenta el factor más valioso que es el tiempo, que se utiliza desde convertir una idea o una necesidad, pasando por el desarrollo y el periodo de lanzamiento de una aplicación al mercado. La competencia entre miles de usuarios y cientos de miles de productos.
- **Prestaciones en el uso de las aplicaciones móviles:** se presentan en relación a diferentes causas como: el acceso o características que ofrecen las aplicaciones gratuitas en comparación a las aplicaciones que son de pago y las restricciones de ciertas opciones de uso, la capacidad de procesamiento de la aplicación, el espacio de memoria que necesita para almacenar cierta aplicación en el dispositivo, la adaptabilidad, compatibilidad y la correcta funcionalidad de las aplicaciones en los dispositivos móviles.
- **Las redes y comunicaciones:** se presentan en las comunicaciones móviles, como la disponibilidad de las redes y la calidad de la señal en el área a utilizar, la variabilidad del ancho de banda, los riesgos de seguridad que se presentan en el entorno; es decir, el tipo de conexión que se utiliza para hacer uso de los servicios que se presentan en los dispositivos móviles; como 3G, 4G, Redes WI-FI, desde el hogar, la oficina, etc.
- **La movilidad del usuario:** se toma en cuenta aspectos como la migración de direcciones, la gestión de la información, de la localización, del servicio al momento de utilizar una aplicación, asociada a la información que se presenta.
- **La portabilidad:** relacionado directamente con la múltiple variedad de hardware/dispositivos móviles existentes; ya que en la actualidad la gran demanda de uso de móviles con pantalla táctil ha evolucionado el mercado de los más grandes proveedores, con características como: el tamaño o la resolución de la pantalla, dimensiones, configuraciones, métodos de entrada, etc.

- **Las capacidades de los terminales (dispositivos móviles):** se determinan por factores como: la integridad de los datos, la capacidad de almacenamiento, la potencia de cálculo, el consumo de la batería al momento de utilizar una o algunas aplicaciones al mismo tiempo.

1.6.3 Clasificación de las aplicaciones móviles según el entorno de desarrollo

El desarrollo de las aplicaciones móviles es, actualmente, un gran desafío, debido a las demandas específicas, las restricciones y características del entorno de las aplicaciones móviles y los sistemas operativos que se presentan en el apartado 1.6.1, todas estas características tienen que ser cuidadosamente consideradas para desarrollar este tipo de aplicaciones.

En el entorno de desarrollo de las aplicaciones móviles existen tres tipos de enfoques, estos son:

1.6.3.1 Aplicaciones Nativas

Las aplicaciones nativas son aplicaciones diseñadas y desarrolladas de forma específica para el sistema operativo de cada dispositivo móvil. Por ejemplo:

- Las aplicaciones móviles para Android utilizan el SDK para Java
- Las aplicaciones móviles para iOS utilizan el SDK para Objective-C.

Las principales características de las aplicaciones móviles nativas que se deben tener en cuenta para el desarrollo, se detallan a continuación:

- Interactúan utilizando todas las funciones y características del dispositivo como: cámara, GPS, calendario, etc.
- Estas aplicaciones requieren de una compilación previa
- El costo en el desarrollo e implementación tiende a incrementarse
- El código desarrollado no es reutilizable entre diferentes plataformas/sistemas operativos.

Para desarrollar una aplicación nativa se tiene que tener en cuenta que su precio es muy elevado y sobretodo tener la certeza y la confianza de que lo propuesto será rentable y específico para cada plataforma.

1.6.3.2 Aplicaciones web

Las aplicaciones web o web App son desarrolladas con el apoyo de HTML, JavaScript y CSS y se ejecutan en el propio navegador del dispositivo a través de una URL. Por ejemplo en el sistema operativo iOS se utiliza el navegador safari. Estas aplicaciones se puede instalar de manera directa en el dispositivo, y la promoción y comercialización se realizan de manera independiente.

Las principales características de las aplicaciones móviles web que se deben tener en cuenta para el desarrollo, se detallan a continuación:

- El código desarrollado se puede utilizar en múltiples plataformas
- El proceso de desarrollo es más sencillo y económico
- El usuario puede disponer de la última actualización.
- Las aplicaciones web requiere de conexión a internet
- Requieren mayor esfuerzo en realizar una promoción y visibilidad de la aplicación

Las aplicaciones web son consideradas como una excelente opción a la hora de adaptar a un dispositivo móvil.

1.6.3.3 Aplicaciones híbridas

Las aplicaciones híbridas son una combinación de las aplicaciones nativas y web: estas se desarrollan con lenguajes propios de las web App y acceden a gran parte de las características del hardware del dispositivo.

Aplicación híbrida: Web App: En este caso, la programación consiste en diseñar la aplicación como si fuera una web app y será el propio navegador el que la ejecute. Sin embargo, ésta es precisamente su gran virtud pero también su principal inconveniente. Nos encontramos con que el código es válido para todos los dispositivos pero la experiencia del usuario y la apariencia no es tan buena como en la app nativa. Phonegap⁶ es el framework más popular en esta tecnología.

Aplicación híbrida: interpretadas: es programada utilizando diversas tecnologías y lenguajes y luego cada terminal la traduce a su propio lenguaje de programación. Facilita el desarrollo de aplicaciones y reduce el esfuerzo considerablemente. Aunque el resultado no es idéntico a la app nativa, la apariencia es bastante buena, y en muchas ocasiones puede ser la solución al problema del desarrollo de aplicaciones interpretadas o multiplataforma.

⁶ Phonegap: disponible en: <http://phonegap.com/>

Las principales características de las aplicaciones móviles híbridas que se deben tener en cuenta para el desarrollo, se detallan a continuación:

- Se puede distribuir en las App Stores de Android, iOS, etc.
- El código es utilizable para múltiples plataformas.
- El acceso al hardware de los dispositivos es muy amplia.
- Se puede limitar la publicación de la aplicación, ya que los distribuidores que en la mayoría son propietarios de las plataformas no pueden ingresar de manera total a los datos e información.

Tomando en cuenta las predicciones y estadísticas de Gartner⁷, el desarrollo de aplicaciones móviles que se utilizarán en tres años, más del 50% serán híbridas. La consultora observó que los desarrollos cada vez irán más orientados a este modelo híbrido que aproveche lo mejor del mundo nativo y lo mejor de HTML.

El avance de las tecnologías presenta cada vez un incremento notable de mejoras en los dispositivos móviles, y de la mano el uso de las aplicaciones móviles, es por esa razón que la categorización y la línea de desarrollo se han ido desglosando en nuevos servicios.

Estos servicios se presentan en estas tecnologías con el propósito de solucionar problemas en los ámbitos con más importancia e interés de la sociedad y además de considerar *las expectativas del mercado de las aplicaciones móviles*⁸.

1.7 Sistemas operativos en los dispositivos móviles

Los sistemas operativos en los dispositivos móviles están diseñados para cumplir con actividades específicas orientadas a la conectividad inalámbrica. A pesar de la gran variedad de sistemas operativos existentes en la actualidad, en el desarrollo de este proyecto se ha establecido realizar la investigación de los sistemas operativos que tienen mayor influencia en el mercado.

Apoyándose en la información estadística que ofrece el sitio web StatCounter⁹ y el análisis de Gartner¹⁰ sobre los sistemas operativos, se consolida que la mayor demanda de usabilidad de los sistemas operativos a nivel mundial y en Ecuador son Android¹¹ e iOS¹²; ya

⁷Gartner: Disponible en: <http://www.infochannel.com.mx/mas-del-50-de-las-aplicaciones-moviles-seran-hibridas-para-2016-gartner>

⁸Expectativas del mercado de apps. Disponible en: <http://www.datacenterconsultores.com/expectativas-del-mercado-de-apps>

⁹ StatCounter. Disponible en: <http://gs.statcounter.com/>

¹⁰Gartner. Android sigue dominando el mercado de los smartphones. Disponible en: <https://www.wayerless.com/2015/08/android-sigue-dominando-el-mercado-de-smartphones/>

¹¹ Plataforma iOS. Disponible en : <http://www.apple.com/ios/>

¹² Plataforma Android. Disponible en: <https://www.android.com/>

que cuentan con el mayor número de desarrolladores interesados en participar de las “App Stores”, además de superar y dominar el mercado en los dispositivos móviles.

1.7.1 Comparativa de sistemas operativos en tecnología móvil

Una vez identificados a Android e iOS como los sistemas operativos más utilizados, en la tabla 1.6 se presentan una comparativa en relación a las principales características que estos poseen.

Tabla 1.2.Tabla comparativa de sistemas operativos móviles

Característica		Sistemas operativos	
		iOS	Android
Entorno de desarrollo	Tipo de programación	Orientado a objetos	Orientado a objetos
	Lenguaje de programación	Objective-C, C++	Java C++
	Plataforma de desarrollo	MAC	Windows, Mac, Linux
	Núcleo del sistema operativo	Mac OS X	Linux, WebKit, SQLite
	Herramientas de pruebas	Incluye un simulador, no emulador e Instruments para recabar datos de iPhones reales, XCode	Incluye un emulador de dispositivos, herramientas para depuración de memoria y análisis del rendimiento del software, Android SDK
	Licencia del software	Propietario (único fabricante)	Código Libre y abierto(multiusuario)
Entorno de distribución	Variedad de dispositivos	Modelo Único	Múltiples dispositivos, se adapta a la gran variedad de terminales que existen en el mercado
	Distribución de aplicaciones	La tienda de distribución es única	Múltiples tiendas de distribución
	Comercialización de aplicaciones	Para ejecutar la publicación es necesario tener una suscripción de Apple Developer Connection que tiene un costo	Gratuito para el desarrollador, solo requiere una cuenta de google
	Tienda de aplicaciones	App Store (Apple iTunes)	Google Play (Android Market)
	Archivo ejecutable	.app	.apk

Fuente Tomado de: (Alvarado Ruiz, Guamán Eras, & Sigcho Armijos, 2012), (Blanco, Camarero, Fumero, Werterski, & Rodriguz, 2009), Comparativa de plataformas en 2013¹³, Las plataformas para móviles en 2013¹⁴

¹³ Comparativa de plataformas móviles. Disponible en: <http://www.androidcurso.com/index.php/tutoriales-android/31-unidad-1-vision-general-y-entorno-de-desarrollo/98-comparativa-con-otras-plataformas>

¹⁴ Las plataformas para móviles 2013. Disponible en: <https://www.youtube.com/watch?v=N6NlbpN1zVk>

Basándose en la tabla 1.2, se identifican a las principales características que poseen cada uno de los sistemas operativos, además (Alvarado Ruiz et al., 2012) resalta que se debe tener en cuenta en el entorno de desarrollo de aplicaciones móviles, las dos características siguientes:

- **Familiaridad en lenguaje:** el conocimiento sobre el lenguaje de programación, permite ahorrar el tiempo en el aprendizaje. Además de considerar la disponibilidad de acceder a información y documentación que facilite el desarrollo.
- **Acceso a dispositivos:** la accesibilidad y la facilidad de disponer de una múltiple variedad de dispositivos móviles que permitan ejecutar las pruebas con mayor facilidad y de esa manera obtener un producto de mayor calidad.

1.8 Metodologías ágiles aplicadas en el desarrollo de aplicaciones móviles

El uso de las metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles se ajusta y se acomodan a las propiedades y características que estas poseen presentadas en el apartado 1.6.2. Así también en este mismo sentido, (Blanco & ál, 2009), (Cáceres, Genoff, Ayala, & Zachman), (Amaya Balaguera, 2013) y (Education, Science, Khalid, Zahra, & Khan, 2014) afirman que el desarrollo ágil se constituye como la mejor solución para aquellos proyectos, donde los requisitos están cambiando constantemente y que se presentan principalmente en este entorno.

Analizar y seleccionar una metodología ágil que se adapte, que sea eficaz y viable de manejar en el desarrollo de aplicaciones para dispositivos móviles no es una tarea fácil, debido a que está directamente relacionado con el propósito y el tamaño del proyecto; sin embargo, la inclusión de una metodología ágil juega un papel muy importante ya que a través de sus fases permiten desarrollar software de manera organizada y asequible para la adaptación en los equipos de trabajo.

Para observar la tendencia del uso de metodologías ágiles en el desarrollo de aplicaciones móviles se establece un proceso metodológico donde se incluye: la selección de palabras claves para la búsqueda de la literatura, las consideraciones que se toman en cuenta para la selección de los trabajos, un análisis minucioso de los estudios seleccionados y sus principales criterios, además se presenta un análisis estadístico de las metodologías ágiles más utilizadas, con el objetivo de facilitar la selección de la metodología y finalmente, se detalla la metodología seleccionada para el desarrollo de este trabajo.

1.8.1 Selección de palabras claves para la búsqueda

La selección de búsqueda se enfatiza en el tema que se propone que es el uso de las metodologías ágiles en el desarrollo de aplicaciones móviles; utilizando como principales motores de búsqueda a Google Scholar, IEEE y otros artículos relacionados de mayor relevancia, con base a los siguientes parámetros:

- Metodologías ágiles + aplicaciones móviles
- Desarrollo ágil + aplicaciones móviles
- Agile methodologies + mobile applications
- Agile Development + mobile applications

Además, se especifican las siguientes palabras claves para la búsqueda ya que en la mayor parte de los estudios seleccionados y analizados incluyen a Mobile-D como la metodología utilizada en el desarrollo de aplicaciones para el entorno móvil.

- Mobile-D + aplicaciones móviles
- Mobile-D + mobile applicattions

1.8.2 Criterios en la selección de los estudios

A través de la revisión de literatura de los diferentes estudios que evalúan, seleccionan y utilizan a las metodologías ágiles en el desarrollo de aplicaciones móviles ha permitido facilitar el proceso de selección de la metodología ágil con el fin de incluirla en el desarrollo del marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles bajo metodologías ágiles. Este proceso de revisión se basa en los siguientes criterios:

- Todos los trabajos incluyen al desarrollo ágil en el desarrollo de las aplicaciones móviles.
- Todos los estudios emiten su criterio y explicación en el proceso de selección de la metodología ágil para el desarrollo de aplicaciones móviles.
- La mayoría de los estudios presentan un análisis y comparación de las metodologías ágiles más populares y utilizadas y que se pueden adaptar en el entorno de desarrollo de las aplicaciones móviles.
- Todos los estudios utilizan a las metodologías ágiles más utilizadas y populares para crear y diseñar nuevas propuestas de metodologías que se adaptan al desarrollo de las aplicaciones móviles.

1.8.3 Selección de los estudios de las metodologías ágiles en el desarrollo de aplicaciones móviles

Se ha seleccionado 26 estudios de diferentes autores, con base a los criterios de selección identificados, en la que cada uno de los documentos se analiza en el siguiente orden: el título del estudio, el resumen, la introducción y las conclusiones.

Los resultados de este análisis se fueron registrando de acuerdo a los siguientes parámetros:

- **Tipo de trabajo:** se establece el tipo de trabajo seleccionado en base a dos categorías “Tesis” y “Artículo”.
- **Nombre del estudio:** el título formal del estudio.
- **Autores/Organización:** los nombres de los autores y/o la organización que ha desarrollado el estudio.
- **Año:** el año de publicación del estudio.
- **País:** el nombre del país donde se desarrolló el estudio.
- **Nº de metodologías ágiles:** el número de metodologías ágiles que están incluidas en cada uno de los estudios.
- **Metodologías ágiles analizadas e incluidas:** el nombre de las metodologías incluidas en cada uno de los estudios.
- **Uso**
 - **Análisis(A):** este parámetro se define para presentar el análisis y definición de las metodologías ágiles más óptimas en el desarrollo de las aplicaciones móviles.
 - **Diseño (D):** representa a los estudios que seleccionan las metodologías ágiles más utilizadas en este entorno y que permiten crear y diseñar nuevas propuestas de metodologías ágiles para el desarrollo de aplicaciones móviles.
 - **Implementación (I):** representa el análisis y selección de las metodologías ágiles para implementar y utilizar en el desarrollo de aplicaciones móviles específicas, considerando el criterio de cada uno de los estudios.
- **Metodología seleccionada:** el nombre de la metodología seleccionada y utilizada de cada uno de los estudios.
- **Argumentación:** se menciona y se toman en cuenta el análisis y la justificación de cada uno de los estudios para seleccionar las metodologías ágiles que se adaptan de mejor manera en el entorno de desarrollo de aplicaciones móviles.

En el **anexo 1**, se presenta a la tabla 1 que contiene el resultado de los 26 estudios seleccionados en el entorno del desarrollo de las aplicaciones móviles que cumplen con los criterios de selección especificados en el apartado 1.8.2.

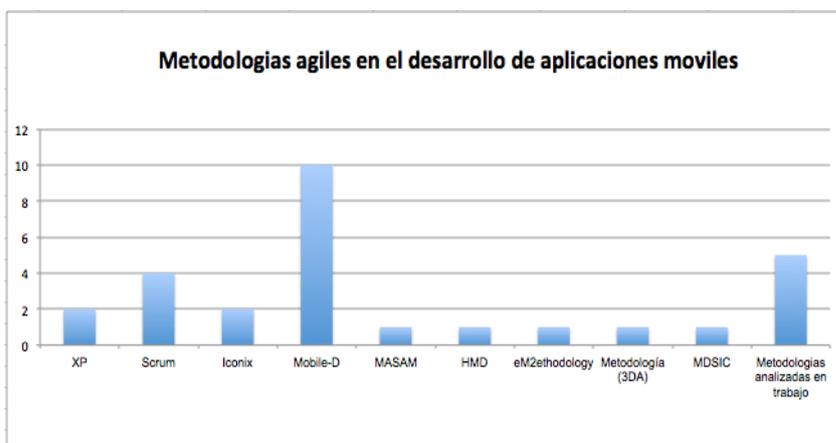
1.8.4 Resultados de la revisión de los estudios

El anexo 1 que presenta la tabla 1 contiene los parámetros más relevantes que se consideraron para analizar las metodologías ágiles en el ciclo de desarrollo de aplicaciones móviles.

Las siguientes estadísticas están basadas en la información obtenida de la tabla 1.3, que lista a las metodologías ágiles más utilizadas en el entorno de desarrollo de las aplicaciones móviles en cada uno de los estudios.

Tabla 1.3. Metodologías más utilizadas en el entorno móvil

Metodologías ágiles seleccionadas	Frecuencia
XP	2
Scrum	4
Iconix	2
Mobile-D	11
MASAM	1
HMD	1
eM ² ethodology	1
Metodología (3DA)	1
MDSIC	1
Metodologías analizadas en trabajo ¹⁵	4
Total	28



Fuente: el autor.

Figura 1.2. Metodología seleccionada

Fuente: el autor

Con base en la información de la tabla 1.3, de todas las metodologías seleccionadas la metodología ágil Mobile-D es la más mencionada, de los 26 estudios analizados, 20 estudios la incluyen y 11 la seleccionan y hacen uso de la misma.

Estos datos permiten seleccionar a la metodología ágil Mobile-D como parte del marco de trabajo.

¹⁵ Cualquier metodología analizada. En este grupo los autores de los trabajos no seleccionan ninguna metodología, por tanto con base a la conclusión y resultados de cada uno de los estudios la selección de la metodología depende del tipo de proyecto que se va a desarrollar.

1.8.5 Metodología ágil “Mobile-D”

La metodología Mobile-D¹⁶ desarrollada en Finlandia en 2005, por (VTT), para el proceso de desarrollo de aplicaciones móviles, es una combinación de metodologías ágiles y tiene como objetivo principal conseguir ciclos de desarrollo muy rápidos en equipos pequeños (menor a 10 personas) que laboran en un mismo espacio físico e interactúa de manera permanente con los clientes. Es una metodología ágil híbrida que combina: XP para el proceso de implementación, Crystal en términos de escalabilidad de los métodos y RUP que se integra en la definición del ciclo de vida de desarrollo y el uso de los principales artefactos.

En la tabla 1.4, se presenta de manera precisa la definición de la metodología Mobile-D.

Tabla 1.4. Metodología ágil Mobile-D

Metodología ágil Mobile-D	
Características	<ul style="list-style-type: none"> • Metodología híbrida conformada de los principales componentes de XP, Crystal y RUP. • Construida principalmente para el entorno de desarrollo de aplicaciones móviles. • Se basa en 9 características elementales que se incluyen en cada una de las fases de ciclo de desarrollo (Ajuste y enfoque en fases, definición de la línea de arquitectura, desarrollo basado en pruebas, integración continua de funcionalidades, programación en pares, definición de métricas, mejora en el proceso, intervención del cliente externo, enfoque centrado en el usuario). • Por cada función a desarrollar se realiza un ciclo iterativo e incremental de 3 días para planificar, trabajar y liberar. • Se puede utilizar en otros contextos, por ejemplo, la seguridad, sistemas financieros, logística y aplicaciones de simulación de productos.
Fases de desarrollo	<p>Está constituida de cinco fases; y en cada una de ellas asociado fases, tareas y prácticas, artefactos y roles</p> <ul style="list-style-type: none"> • Exploración • Inicialización • Producción • Estabilización • Pruebas del sistema
Ventajas	<ul style="list-style-type: none"> • Los costos son bajos cuando se necesita realizar algún cambio en el proyecto. • La entrega de los resultados es rápida. • Se enfoca a trabajar con grupos de trabajo pequeños. • Permite mejorar el producto mediante iteraciones pequeñas. • Se centra en la satisfacción del usuario final. • En la liberación del producto la densidad de los defectos es baja. • La comunicación entre los miembros del equipo es constante. • Mobile-D cuenta con certificación CMMI de nivel 2.
Desventajas	<ul style="list-style-type: none"> • No se adapta a equipos demasiado grandes o segmentados.
Ambiente de desarrollo de software	<ul style="list-style-type: none"> • Software para aplicaciones móviles

Fuente: el autor.

¹⁶ Mobile-D. Disponible en: <http://agile.vtt.fi/mobiled.html>

De la misma manera se presenta en la tabla 1.5, un análisis mediante una matriz de evaluación de las fases de desarrollo de la metodología ágil Mobile-D con respecto a las fases del Ciclo de Vida de Desarrollo de Software (CVDS), en las que se identifican principalmente las tareas, artefactos y roles.

Tabla 1.5. Matriz de evaluación de las fases de Mobile-D en el desarrollo de aplicaciones móviles con respecto al ciclo de vida de desarrollo de software (CVDS).

CVDS	MOBILE-D	Fases	Tareas	Artefactos	Roles
Análisis	Exploración	<ul style="list-style-type: none"> Identificación de los interesados Definición del alcance Establecimiento del proyecto 	<p>Las tareas asociadas a esta fase son:</p> <ul style="list-style-type: none"> Establecimiento de los interesados, establece los grupos de actores necesarios en la planificación y el seguimiento del proyecto. La planificación inicial del proyecto, define y establece los objetivos y el alcance del desarrollo del proyecto Recogida de los requerimientos y La definición de los procesos 	<ul style="list-style-type: none"> Plan del proyecto Plan de arquitectura de software Plan de medición Plan de formación Plantilla de checklist del proyecto 	<ul style="list-style-type: none"> Equipo de dirección Equipo de exploración Equipo de clientes
Planificación					
Diseño	Inicialización	<ul style="list-style-type: none"> Configuración del proyecto Día de planificación en repetición 0 Día de trabajo en repetición 0 Día de liberación en repetición 0 	<ul style="list-style-type: none"> Se establece el entorno dinámico técnico en relación a los recursos físicos, tecnológicos y de comunicaciones para iniciar con el desarrollo Se desarrolla un plan de formación/capacitación del equipo, y La primera versión de la cartera de los productos a desarrollar. 	<ul style="list-style-type: none"> <i>Plan del proyecto actualizado</i> <i>Plantilla de especificación de requerimientos actualizado.</i> Pruebas de aceptación para cada requerimiento. 	<ul style="list-style-type: none"> Equipo de proyecto
Desarrollo	Producción	<ul style="list-style-type: none"> Día de planificación Día de trabajo Empaquetado de documentación Día de liberación 	<ul style="list-style-type: none"> Se repite iterativamente la programación de tres días de planificación. Trabajo y liberación, hasta implementar todas las funcionalidades, desarrollando e integrando el código, seguido de las pruebas de aceptación 	<ul style="list-style-type: none"> <i>Plan del proyecto actualizado</i> <i>Documento de especificación de requerimientos actualizado.</i> <i>Plan de arquitectura de software</i> Plantilla de lista de puntos de acción - Post-iteración Historias de usuario Tarjetas de tareas Plantilla de issues y bugs Plantilla de documentación de código Documento de diseño e interfaz de usuario. 	<ul style="list-style-type: none"> Equipo de proyecto

Pruebas	Estabilización	<ul style="list-style-type: none"> • Día de planificación • Día de trabajo • Día de liberación 	<ul style="list-style-type: none"> • Se llevan a cabo las pruebas de integración para asegurar que el producto completo funcione correctamente. • Se incluye la producción de la documentación 	<ul style="list-style-type: none"> • <i>Plan del proyecto finalizado</i> • <i>Plantilla de especificación de requerimientos finalizado.</i> 	<ul style="list-style-type: none"> • Equipo de proyecto
Mantenimiento	Pruebas del Sistema	<ul style="list-style-type: none"> • Pruebas del sistema • Día de planificación • Día de trabajo • Día de publicación 	<ul style="list-style-type: none"> • El producto finalizado se comprueba con los requisitos del cliente y • Se eliminan todos defectos encontrados • Presenta una versión estable y funcional del producto 	<ul style="list-style-type: none"> • Registro de pruebas del sistema • Plan de pruebas del sistema • Informe de prueba del sistema 	<ul style="list-style-type: none"> • Equipo de clientes • Equipo de pruebas del sistema

Fuente: Elaborado por el autor

Mobile-D está organizada en base a los principales procesos de planificar, diseñar, implementar, probar y liberar del ciclo de vida de software, con los procesos de apoyo de la gestión de proyectos, gestión de configuración del software y en la mejora de procesos de software; tomando en cuenta que el desarrollo de aplicaciones móviles son dependientes del tiempo y principalmente las competencia entre proveedores de hardware y software de este entorno.

1.9 Calidad en el desarrollo de las aplicaciones móviles

(Pressman, 2010) Afirma que: “La calidad del software se debe implementar en todo el ciclo de vida de desarrollo de un producto”.

Garantizar la calidad en el software inicia realmente con la aplicación de una metodología formal de desarrollo para enfrentar las etapas de análisis y diseño del producto a construir; a través de la especificación del producto (o prototipo), que se va a implementar.

Desde el punto de vista de los desarrolladores se define a la calidad como el grado en el que un producto de software satisface la especificación de los requerimientos, y desde el punto de vista del cliente la calidad se define como el grado en que un cliente y/o usuario percibe que el producto de software satisface sus necesidades.

En el entorno del desarrollo de las aplicaciones móviles, el interés por obtener un producto de calidad crece de forma continua, los clientes cada vez son más selectivos; es decir, que al momento de seleccionar y utilizar un producto, en este caso una aplicación móvil, analizan de manera rápida su funcionamiento y si cumple con sus necesidades e interés para hacer uso de la misma.

1.9.1 Gestión de calidad

“La gestión de la calidad es aplicable a todos los proyectos, independientemente de la naturaleza de su producto, y las medidas y técnicas de calidad del producto son específicas al tipo del producto generado por el proyecto”(PMI, n.d.)

(PMI, n.d.), define que “La gestión de la calidad del proyecto incluye los procesos y actividades de la organización ejecutante que determinan responsabilidades, objetivos y políticas de calidad a fin de que el proyecto satisfaga las necesidades por las cuales se emprendió. La gestión de la calidad del proyecto comprende los procesos:

- 1. Planificación de Calidad:** es el proceso por el cual se identifican los requisitos de calidad y/o normas para el proyecto y el producto, documentando la manera en que el proyecto demostrará el cumplimiento de los mismos. La planificación de la calidad debe realizarse en forma paralela a los demás procesos de planificación del proyecto.
- 2. Aseguramiento de calidad:** es el proceso que consiste en auditar los requisitos de calidad y los resultados de las medidas de control de calidad, para asegurar que se utilicen las normas de calidad apropiadas y las definiciones operacionales.
- 3. Control de calidad:** es el proceso por el cual se monitorean y se registran los resultados de la ejecución de actividades de control de calidad, a fin de evaluar el desempeño y recomendar cambios necesarios.

Además se debe tener en cuenta que en el proceso de calidad de un producto software, contar con el apoyo de un equipo de calidad se considera como una de las mejores oportunidades para obtener un producto competitivo que garantice lo que se está ofreciendo al usuario.

La definición de los roles de un **equipo de calidad** en este proceso permiten asignar a los miembros un conjunto de responsabilidades, actividades y autorizaciones cuyo objetivo principal es el prevenir conflictos y establecer un grado de control respecto al trabajo que están desarrollando; entre los roles de calidad más importantes se consideran:

- Gerente de calidad
- Gerente de pruebas
- Analista de pruebas
- Evaluadores

Para cada uno de estos roles, se definen objetivos, actividades e interacción con otros roles, herramientas a utilizar, perfil de las personas y un plan de trabajo.

El desarrollo de este trabajo se enfoca en aplicar las principales actividades, procesos, y tareas que son parte del **control de calidad** en el desarrollo de software, específicamente en las aplicaciones móviles. A continuación se presentan algunas definiciones del control de calidad.

1.9.2 Control de calidad

El control de calidad¹⁷ es el proceso de regulación a través del cual podemos medir la calidad real, compararla con las normas de calidad y actuar sobre la diferencia. Es el conjunto de técnicas y procedimientos para orientar, supervisar y controlar todas las etapas.

La norma ISO 8402 define el control de calidad como el conjunto de técnicas y actividades de carácter operativo, utilizadas para verificar los requerimientos relativos a la calidad del producto o servicio. *El control de calidad se aplica tanto al proyecto como al producto del proyecto, y se puede definir de manera independiente.*

1.9.2.1 Control de calidad del proyecto

El Control de calidad del proyecto (QC) es el proceso por el cual se monitorean y registran los resultados de la ejecución de actividades de calidad, a fin de evaluar el desempeño y recomendar cambios necesarios. El control de calidad se lleva a cabo durante todo el proyecto.

Las principales actividades del control de calidad permiten identificar las causas de una calidad deficiente del proceso y/o del producto, y recomiendan y/o implementan acciones para eliminarlas. Para realizar el control de calidad, PMBOK presenta entradas, herramientas y técnicas y salidas:

Tabla 1.6. Control de calidad del proyecto

Entradas	Herramientas y Técnicas	Salidas
<ul style="list-style-type: none">• Plan de gestión de calidad• Métricas de calidad• Listas de control de calidad• Datos de desempeño del trabajo• Solicitudes de cambio aprobadas• Entregables• Documentos del proyecto• Activos de los procesos de la organización	<ul style="list-style-type: none">• 7 herramientas básicas de calidad• Muestreo estadístico• Inspección• Revisión de solicitudes de cambios requeridos	<ul style="list-style-type: none">• Mediciones de control de calidad• Cambios validados• Entregables validados• Información del desempeño del trabajo• Actualización a los activos de los procesos de la organización• Solicitudes de cambio• Actualización al plan para la dirección del proyecto• Actualización a los documentos del proyecto

Fuente: Gestión de la calidad. Control de Calidad. Tomado de PMBOK.

¹⁷ Control de calidad. Disponible en: <http://www.gestiopolis.com/que-son-calidad-aseguramiento-de-la-calidad-y-control-de-calidad/>

1.9.2.2 Control de calidad del software

(Pressman, 2010), define que: El control de calidad del software incluye un conjunto de acciones de ingeniería de software que ayudan a asegurar que todo producto del trabajo cumpla con sus metas de calidad. Es decir, que:

- Los modelos se revisan para garantizar que están completos y que son consistentes.
- El código se inspecciona con objeto de descubrir y corregir errores antes de que comiencen las pruebas
- Se aplica una serie de etapas de pruebas para detectar los errores en procesamiento lógico, manipulación de datos y comunicación con la interfaz
- La combinación de mediciones con retroalimentación permite que el equipo de software sintonice el proceso cuando cualquiera de estos productos de trabajo falla en el cumplimiento de las metas de calidad.

Las principales actividades que debe cumplir toda implementación de control de calidad en el desarrollo de software son:

- Aplicación de metodologías y técnicas de desarrollo para conseguir una especificación y un diseño de alta calidad.
- Realización de procesos de revisiones técnicas formales
- Pruebas de software
- Ajustes a los estándares de desarrollo
- Control de cambios
- Mediciones
- Recopilación de información y
- Gestión de informes sobre el control de calidad.

Mientras que (Bahamon L, n.d.) Detalla que el **control de calidad** se fundamenta en el principio de que la calidad se construye a través de un proceso continuo de desarrollo, verificación, revisión y optimización en diferentes etapas. Se basa en las siguientes actividades:

1. Uso de métodos y herramientas de análisis, diseño, codificación y pruebas, que ayudan al analista a conseguir una especificación y un diseño de alta calidad del proyecto.
2. Revisiones técnicas formales, que se aplican durante cada paso de la ingeniería de software, con el propósito de descubrir problemas de calidad. Se consideran como una prueba para descubrir defectos de software.

3. Estrategias de pruebas multiescala para la garantía de calidad del producto software.
4. Control de cambios realizados, donde cada cambio realizado sobre el software en potencia puede introducir errores o crear efectos laterales que propaguen errores.
5. Procedimientos que aseguren un ajuste a los estándares de desarrollo.
6. Mecanismos de medida de calidad a través del uso de métricas.
7. Control de la documentación del software

1.9.3 Pruebas de calidad del software

Las pruebas del software se definen como el proceso de ejecutar un programa con la intención de encontrar defectos. Es un proceso que determina el diseño de los casos de prueba y la asignación de responsabilidades. Se considera que una prueba es exitosa, si por lo menos detecta un defecto.

“Una de las actividades más importantes a considerar en el control de la calidad del software son las Pruebas del software”

En un concepto más amplio, las pruebas conforman la verificación y validación; donde, la verificación se refiere al conjunto de actividades que aseguran que el software implementa de manera correcta una función específica; y la validación se refiere a un conjunto diferente de actividades que aseguran que el software construido se ajusta a los requisitos del cliente (Scalone, 2006), como se muestra en la figura 1.3.

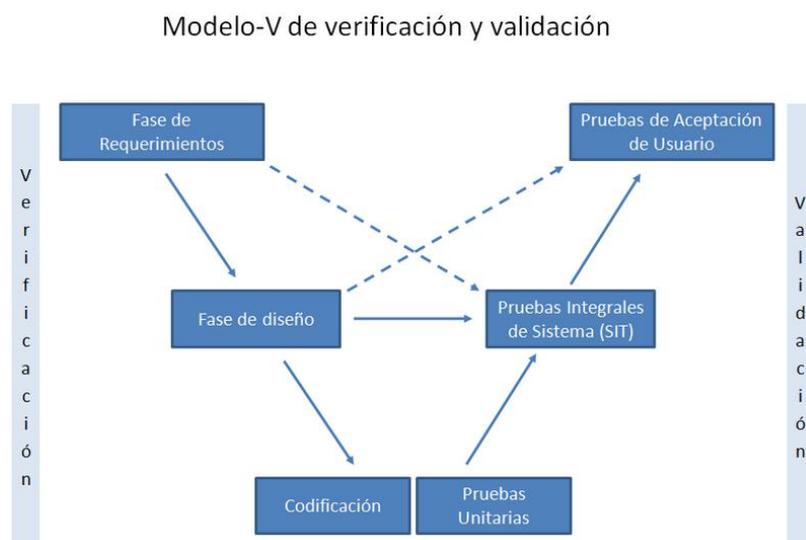


Figura 1.3. Modelo de verificación y validación.
Fuente: (Scalone, 2006)

Las pruebas se ejecutan en cada una de los niveles del ciclo de vida de desarrollo y permiten comprobar que el sistema o la aplicación cumplan con el propósito para la que fue creada.

Una prueba diseñada correctamente reduce el nivel general del riesgo de un sistema, entonces un conjunto de pruebas integradas adecuadamente se definen como una actividad más del proceso de garantía de calidad.

De acuerdo a los términos de calidad de software basados en ISTQB¹⁸ las pruebas sirven para cumplir con los requisitos contractuales o legales, o estándares específicos del ambiente. Gracias a las pruebas se puede medir la calidad de un software en términos de los defectos detectados por lo que respecta a requisitos y características funcionales y no funcionales. Entonces se puede decir que las pruebas aportan fiabilidad a la calidad del software en caso de detectar algún defecto.

1.9.4 Pruebas de calidad aplicadas en el desarrollo de las aplicaciones móviles

Con base al análisis de los trabajos relacionados de (Silva del Rosario, 2012), (Alvarado Ruiz, Guamán Eras, & Sigcho Armijos, 2012) y (Saraguro Bravo, 2012), de la metodología de desarrollo Mobile-d y por otro lado de la información que ofrece la PMO¹⁹, se han establecido en la ejecución de las pruebas y validación en el desarrollo de las aplicaciones móviles los siguientes tipos de pruebas que se pueden tomar en cuenta e implementar en este ambiente.

La identificación y selección de las pruebas para las aplicaciones móviles permiten clasificar y asignar cada una de las pruebas a los principales grupos que intervienen en el proceso de desarrollo. Esta clasificación se presenta en la figura 1.4.

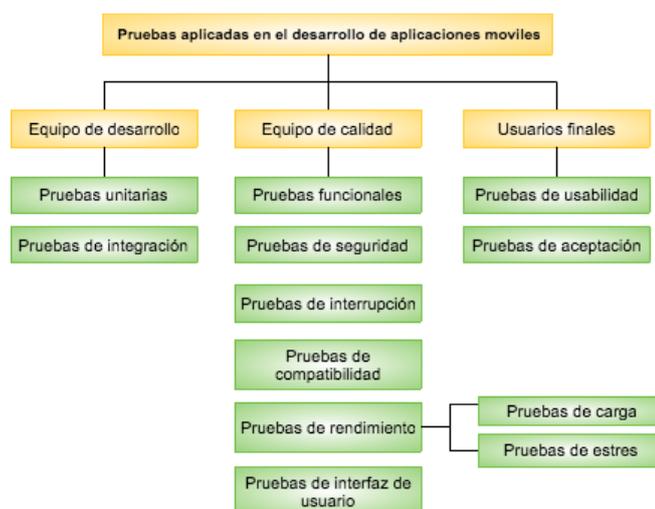


Figura 1.4. Pruebas aplicadas en el desarrollo de aplicaciones móviles
Fuente: Elaborada por el autor.

¹⁸ ISTQB: Disponible en: http://www.sstqb.es/ficheros/sstqb_file95-a69acf.pdf

¹⁹ PMO La oficina de proyectos de informática. Disponible en: http://www.pmoinformatica.com/2013/08/guia-desarrollo-aplicaciones-moviles_25.html

Cada una de las pruebas que se presentan en la figura 1.4, se detallan a continuación:

- **Pruebas unitarias:** o también llamadas pruebas de caja blanca se ejecutan para comprobar el correcto funcionamiento de un módulo de código que realiza el programador mientras está desarrollando.
- **Pruebas de integración:** Se ocupan de probar las interfaces entre los componentes, las interacciones con las distintas partes de un mismo sistema, es decir con el sistema operativo, sistema de archivos, el hardware, y las interfaces entre varios sistemas.
- **Pruebas funcionales:** Este tipo de prueba en aplicaciones para dispositivos móviles es una prueba de caja negra, la cual permite, asegurar que las funciones de la aplicación cumplan con las especificaciones comerciales y los requerimientos que darán solución a la necesidad del cliente. Estas pruebas se pueden ejecutar manualmente o apoyándose con una herramienta que permita automatizar este tipo de pruebas.
- **Pruebas de seguridad:** se llevan a cabo con el propósito de comprobar y evaluar la capacidad que tiene la aplicación móvil en este caso para proteger y controlar la confidencialidad, integridad, autenticación, autorización, y disponibilidad de los datos de la aplicación que manipula el usuario validando el acceso que tiene a las diferentes funcionalidades.

Las pruebas de seguridad se rigen por la norma introducida por OWASP²⁰ Mobile Security Project.

- **Pruebas de interrupción:** en las aplicaciones móviles las pruebas de interrupción se incluyen y funcionan ejecutando otra actividad o suceso; por ejemplo: considerando escenarios como; si el usuario recibe una llamada o mensaje en el momento exacto que se está ejecutando un proceso de la aplicación, o cuando se pierde cobertura de red de telecomunicaciones (conectividad) en medio de un proceso de una aplicación que utiliza geo localización.
- **Pruebas de compatibilidad:** se ejecutan con el propósito de asegurar que las aplicaciones móviles funcionan como se pretende; en los dispositivos seleccionados en relación a los tamaños de la pantalla, resolución y versión del sistema operativo. Este tipo de pruebas se pueden realizar de manera manual o automática mediante la captura de pantallas que son revisadas por los evaluadores.
- **Pruebas de desempeño:** se realizan para determinar la rapidez y la capacidad para ejecutar una tarea de la aplicación móvil, en condiciones particulares de trabajo. En el lado del servidor, se analizan las variaciones en los tiempos de respuesta, los retrasos

²⁰OWASP. Disponible en: <http://www.seguridadparatodos.es/2013/02/OWASP-Parte1MetodologiaAppMobile.html>

en la entrega de mensajes, si la aplicaciones se bloquea, etc. En el lado del usuario implica analizar el comportamiento de la aplicación en diversas plataformas y dispositivos, el consumo de memoria, la velocidad y el consumo de batería.

- **Pruebas de carga:** este tipo de prueba permite mostrar los tiempos de respuestas de todas las transacciones y funciones de la aplicación. En el ambiente de las aplicaciones móviles se basa específicamente en realizar múltiples consultas u operaciones en la aplicación y verificar el tiempo promedio de respuesta de cada una de las consultas que realiza el usuario.
- **Pruebas de estrés:** se utiliza generalmente para romper la aplicación. Mientras se incrementa el número de usuarios al uso continuo de la aplicación y se ejecuta una prueba de carga hasta que se rompe. Este tipo de prueba se realiza para determinar la solidez de la aplicación en los momentos de carga extrema y permite a los administradores determinar si la aplicación rendirá lo suficiente en caso de que la carga real supere a la carga esperada.
- **Pruebas de Usabilidad:** considerada como una de las más destacadas e irremplazables en el desarrollo de software dado que la información es obtenida directamente de los usuarios reales que utilizan la aplicación; las pruebas de usabilidad se enfocan en medir la capacidad de un producto, con el propósito para el que fue diseñada. Las pruebas de usabilidad miden la facilidad de uso, de un objeto o un conjunto de objetos; seleccionando un grupo de usuarios de una aplicación y solicitarles a cabo las tareas y acciones para las que fue diseñada, permitiendo así al grupo de desarrollo tomar nota de los errores y dificultades que encuentran los usuarios, con el objetivo de obtener un producto de excelente calidad.
- **Pruebas de aceptación:** se establecen a este tipo de pruebas que son realizadas por los usuarios finales del producto desarrollado. Las pruebas de aceptación, son básicamente pruebas funcionales sobre el sistema completo (aplicación móvil), y buscan comprobar que se satisfacen los requisitos establecidos. En las aplicaciones móviles la ejecución de las pruebas de aceptación directamente en una versión final del entorno de producción. (Valdez Huaraca, 2013) Las pruebas de aceptación son definidas por el usuario del sistema, preparadas por el equipo de desarrollo y ejecutadas y aprobadas por el usuario final. Cuando dan el visto bueno se proporciona la certificación final del correcto funcionamiento del producto.
- **Pruebas de interfaz de usuario (GUI):** las pruebas de interfaz de usuario permiten verificar que la navegación de cada una de las ventanas cumpla con lo establecido de cada una de las funcionalidades implementadas. Además, las pruebas de interfaz de usuario permiten asegurar que los objetos se encuentren dentro de los

estándares establecidos por las tiendas de distribución de las aplicaciones móviles. “Para el usuario final las pantallas son la aplicación”.

- **Validación del diseño de interfaz en las aplicaciones móviles** permite al desarrollador ajustarse a los lineamientos, condiciones y características de diseño exigido por las App Stores de Android²¹ e iOS²² y que deben tener la aplicación. En las que se debe tener en cuenta características como: ventanas, botones, iconos, imágenes, fondos de pantalla, diseño de la aplicación, colores, ubicación de los componentes, resolución, las dimensiones, etc. este tipo de pruebas se agregan al ámbito del desarrollo de las aplicaciones móviles tomando en cuenta que sería el único medio de distribución para los clientes.

En la ejecución de las pruebas de las aplicaciones móviles se ha especificado que los principales ambientes de validación son el **uso de emuladores** como una alternativa para ahorrar recursos como: tiempo y dinero; y de las **pruebas en dispositivos móviles** que se considera como la mejor opción al momento de realizar las pruebas de instalación, funcionamiento de cámara, sensores, multimedia, etc. La ejecución de las pruebas con dispositivos móviles y la adquisición de los mismos se deben incluir en el presupuesto del proyecto, e incluso existe la posibilidad de buscar el apoyo de futuros clientes o personas adeptas a la última tecnología en cuanto a dispositivos móviles y que se prestan a probar y adquirir las nuevas aplicaciones. Este entorno de validación de las aplicaciones es una de las mejores opciones en cuanto al control de calidad ya que se pueden obtener un volumen alto y confiable de sugerencias, por parte de los usuarios que la utilizan.

1.9.5 Factores, modelos y estándares de calidad

La calidad representa un factor determinante en la competitividad de la empresa. La implementación de Modelos o Estándares de Calidad tiene como objetivo principal que las empresas desarrollen sistemáticamente, productos, bienes y servicios de mejor calidad y cumplan con las necesidades y exigencias de los clientes. Para esto, se requiere de un Modelo / Estándar que: permita: (1) unir la misión de la empresa y el esfuerzo de cada área en una sinergia de resultados hacia la competitividad y la calidad de clase mundial; y (2) tener procesos y procedimientos ágiles; y comprensibles para todos los involucrados, pasando por las etapas de desarrollo, prueba, producción y satisfacción del cliente. (Scalone, 2006).

²¹ Guía de diseño de Android. <http://developer.android.com/design/patterns/navigation-drawer.html>

²² iOS Human Interface Guidelines.

<https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/index.html#>

La importancia de utilizar un modelo de calidad es la especificación de los requisitos de calidad para el producto final y a la vez que pueda ser utilizado para realizar una evaluación cualitativa y cuantitativa de todos los productos elaborados durante el proceso de desarrollo.

Los modelos de calidad son aquellos documentos que integran la mayor parte de las mejores prácticas, proponen temas de administración en los que cada organización debe hacer énfasis, integran diferentes prácticas dirigidas a los procesos clave y permiten medir los avances en calidad.²³

Los estándares de calidad son aquellos que permiten definir un conjunto de criterios de desarrollo que guían la forma en que se aplica la ingeniería de software, los estándares de calidad suministran los medios para que todos los procesos se realicen de la misma forma y son una guía para lograr la productividad y la calidad.²⁴

El uso de modelos y estándares de calidad del software ayuda a lograr una mejor gestión de la calidad que a su vez está centrada en niveles de trabajo, establecidos en nivel de organización, de proyecto y de producto software.

Siguiendo la filosofía de los modelos clásicos de calidad de un producto software, la norma ISO 9126 descompone la calidad jerárquicamente en una serie de características y sub-características que pueden usarse como una lista de comprobación de aspectos relacionados con la calidad del producto software.

Con base del análisis de investigación de los artículos relacionados con la calidad en el ambiente de tecnología móvil, (Canelón , Losavio, Matteo, & Chirinos, Modelo conceptual para modelación de aplicaciones móviles sensibles al contexto, 2009), (Pretel García & Lago Vilariño, 2010) y (Herrera, Najar, Rocobado, Fennema, & Cianferoni, 2013), (Rosado, 2015),(Silvia et al., 2016), (Yanquén Ramírez & Otálora Luna, 2016), se ha establecido y considerado como modelo de calidad con mayor demanda de usabilidad y de confianza a los estándares y las respectivas características de la norma ISO/IEC 9126, ya que se ajustan de forma eficiente a las necesidades de calidad y evaluación para el desarrollo de este proyecto.

La **ISO/IEC 9126** es un estándar internacional utilizado para ejecutar la evaluación de la calidad del software; está dividido en 3 aspectos que se relacionan y se muestra en la figura 1.5.

²³ Piattini, García, "Calidad en el desarrollo y mantenimiento del software", RA-MA Editorial, Madrid, 2006.

²⁴ Piattini, García, "Calidad en el desarrollo y mantenimiento del software", RA-MA Editorial, Madrid, 2006.

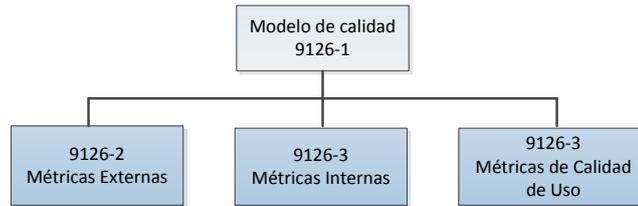


Figura 1.5. Modelo de calidad ISO/EC 9126-1
Fuente: (Piattini, 2011)

En la figura 1.5 el estándar define el modelo de términos de calidad en tres aspectos: Calidad interna, calidad externa y calidad de uso.

La **calidad interna** es medida y evaluada en base a atributos internos del software a través de la métricas internas (métricas estáticas) que pueden ser mejorados durante la implementación del código, la revisión y las pruebas.

La **calidad externa** es la calidad del software expresada a través de su comportamiento utilizando las métricas externas; cuando éste se encuentra en ejecución, que normalmente es medida y evaluada durante las pruebas del software en un entorno simulado.

La **calidad de uso** constituye un concepto nuevo y de suma importancia que ha sido introducido en este nuevo marco de calidad. El uso y validación de las métricas de calidad de uso representa la calidad percibida por el usuario en el software cuando éste es usado en un entorno y un contexto específico real.

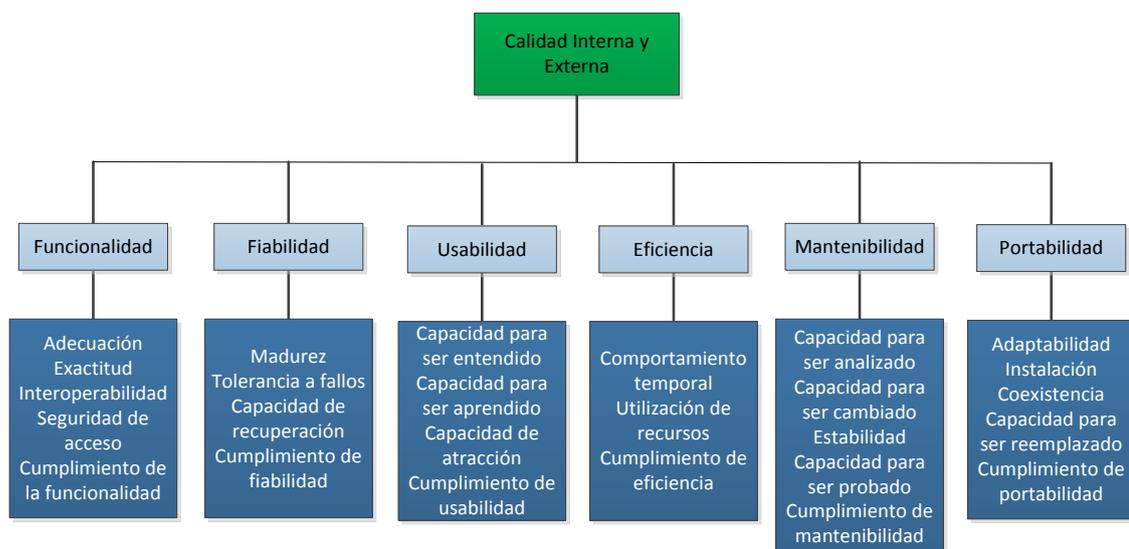


Figura 1.6. Modelo de calidad interna y externa y sus métricas

Fuente: (Piattini, 2011)



Figura 1.7. Modelo de calidad de uso y sus métricas.

Fuente: (Piattini, 2011)

(Piattini, García, & Caballero, 2006) define: “La medición de producto software está centrada en evaluar la calidad de los entregables. Los productos del software son las salidas del proceso de producción del software durante el ciclo de vida del desarrollo.”

1.9.6 Métricas y herramientas para evaluar las aplicaciones móviles

En el entorno de las aplicaciones móviles se pueden considerar otros criterios adicionales para evaluar el rendimiento y la demanda de usabilidad de las aplicaciones móviles, con el único objetivo de atraer nuevos usuarios que hagan uso de los servicios prestados y evaluar si vale la pena invertir en este entorno de tecnología móvil.

Mediante la especificación de palabras clave para la búsqueda, en este caso “**métricas de las aplicaciones móviles**” se han seleccionado algunos sitios web confiables como: Marketing Digital²⁵, Ignacio Santiago²⁶, ProfitWell²⁷; que permiten identificar las principales métricas para evaluar la calidad de las aplicaciones móviles.

A través de la selección de las métricas más utilizadas para mejorar el rendimiento de las aplicaciones móviles en las tiendas de distribución, los desarrolladores buscan comprender mejor a los usuarios y mejorar la experiencia del usuario con la aplicación. Entre las métricas más utilizadas para validar aplicaciones móviles se mencionan:

Tabla 1.7. Métricas para evaluar las aplicaciones móviles

Usuarios activos	<ul style="list-style-type: none"> • Número de usuarios activos que descargan la aplicación • Número de usuarios activos que utilizan la aplicación
Duración de la sesión	<ul style="list-style-type: none"> • Tiempo de usabilidad de los usuarios en la sesión
Intervalo de sesiones	<ul style="list-style-type: none"> • Tiempo entre la primera sesión de usuario con relación a la siguiente. • Valor inmediato obtenido al descargar y ejecutar la aplicación
Tiempo consumido dentro de la aplicación	<ul style="list-style-type: none"> • Tiempo (periodo de tiempo) de uso de la aplicación
Adquisición	<ul style="list-style-type: none"> • Número de usuarios que descargan e instalan la aplicación

²⁵ Glosario de métricas para apps: <http://comunidad.iebschool.com/iebs/marketing-digital/metricas-para-app/>

²⁶ 8 métricas claves para tu estrategia en apps móviles: <http://ignaciosantiago.com/blog/metricas-aplicaciones-moviles/>

²⁷ How to calculate LTV for SaaS A the right way: <http://blog.profitwell.com/how-to-calculate-ltv-for-saas-the-right-way>

Flujos de pantalla	<ul style="list-style-type: none"> • Numero de pantallas y que pantallas son las más utilizadas en la aplicación
Retención	<ul style="list-style-type: none"> • Porcentaje de usuarios que utilizan la aplicación en relación a la fecha de la primera descarga
Valor del ciclo de vida (LTV)	<ul style="list-style-type: none"> • Ingreso económico que genera cada usuario durante la vida útil de la aplicación
Experiencia de usuario	<ul style="list-style-type: none"> • Número de visitas
Datos del usuario	<ul style="list-style-type: none"> • Demográficos: Edad, genero, localización, • Preferencias de usuario: ajustes

Fuente: el autor.

Este tipo de métricas se pueden evaluar con herramientas disponibles²⁸, sencillas y gratuitas que se adaptan fácilmente en el proceso de desarrollo y que permiten sacar el mayor rendimiento posible a la aplicación móvil. Entre las herramientas de evaluación se pueden mencionar: Google Analytics²⁹, Adobe Analytics, etc.

1.9.7 Herramientas para automatización de pruebas de aplicaciones móviles

A través de la especificación de palabras clave para la búsqueda, en este caso “**Herramientas para automatización de pruebas de aplicaciones móviles**” y basándose en la información de (Díaz, 2013), permiten identificar las principales herramientas para la automatización de pruebas de las aplicaciones móviles en el proceso de desarrollo.

Entre los simuladores para aplicaciones móviles más utilizadas en el ambiente de pruebas como Software Open Source están los siguientes:

- Robotium³⁰, Calabash³¹, Appium³², JUnit³³
- UIAutomation³⁴
- KIF³⁵
- MonkeyRunner³⁶, Monkop³⁷
- Testdroid Cloud – bitbar.³⁸
- Test object.³⁹

²⁸ 10 herramientas de analítica para apps: <http://lametrica.com/10-herramientas-de-analitica-para-apps/>

²⁹ Analytics para Android. Disponible en: <https://developers.google.com/analytics/devguides/collection/android/v4/?hl=es#configurar-el-proyecto>

³⁰ Robotium. : <https://github.com/RobotiumTech/robotium>, <http://robotium.com/>, <http://robotium.com/pages/installation-android-studio>

³¹ Calabash: <http://calaba.sh/>

³² Appium: <http://appium.io/>

³³ JUnit: http://www.tutorialspoint.com/android/android_testing.htm

³⁴ UIAutomation

<https://developer.apple.com/library/ios/documentation/DeveloperTools/Conceptual/InstrumentsUserGuide/UIAutomation.html>

³⁵ KIF: <https://github.com/kif-framework/KIF>

³⁶ Monkey runner <https://developer.android.com/studio/test/monkeyrunner/index.html>

³⁷ Herramienta para pruebas de desempeño, funcionalidad y seguridad. Monkop. <https://www.monkop.com/>

³⁸ Testdroid Cloud. Bitbar. <https://cloud.testdroid.com/#service/projects/113219231>

1.9.8 Emuladores para aplicaciones móviles

A través del uso de la tecnología, esto ha facilitado a los usuarios disfrutar de las aplicaciones en ordenadores personales; así como se siguen desarrollando aplicaciones móviles también se están desarrollando emuladores para utilizar en los ordenadores. Uno de los principales objetivos para utilizar un emulador es aprovechar los recursos que tiene el ordenador.

Básicamente este tipo de herramientas son utilizadas por el equipo de desarrolladores de aplicaciones móviles para probar la aplicación desarrollada antes de ser distribuida a los usuarios finales. A continuación, se presenta una lista de los mejores emuladores gratuitos ⁴⁰ que se pueden utilizar:

- BlueStacks Android Emulator ⁴¹
- GenyMotion Android Emulator ⁴²
- Andy Android Emulator⁴³
- Jelly Bean Android Emulator⁴⁴
- YouWave⁴⁵

1.10 Discusión final

En las aplicaciones móviles el ambiente desarrollo difiere de las aplicaciones de escritorio, ya que estas poseen características y condiciones adicionales. Cada una de las secciones detalladas en este capítulo está asociada a este entorno de trabajo.

³⁹ Test Object. <https://testobject.com/>

⁴⁰ Emuladores: <https://www.wondershare.es/mirror-emulador/mejores-emuladores-de-android-para-pc-mac.html#part3>

⁴¹ BlueStacks: <http://www.bluestacks.com/>

⁴² GennyMotion: <https://www.genymotion.com/>

⁴³ Andy: <http://www.andyroid.net/>

⁴⁴ Jelly Bean Emulador: <https://software.intel.com/en-us/android/articles/android-43-jelly-bean-x86-emulator-system-image>

⁴⁵ YouWave: <https://youwave.com/download/>

CAPÍTULO 2.
DEFINICIÓN DEL PROBLEMA

2.1 Introducción

En la actualidad, la tecnología móvil a través del uso de dispositivos móviles se ha considerado como una nueva línea de negocio beneficiosa y de gran interés tanto para las empresas como para los usuarios que hacen uso de los mismos. Aprovechando la disponibilidad de estos nuevos y mejores dispositivos, se ha ampliado el mercado existente en el desarrollo de las aplicaciones móviles. Sin embargo, el ciclo de desarrollo de este tipo de aplicaciones se considera complejo de acuerdo a las características adicionales que se presentan, y obtener un producto de calidad y competitivo se convierte en un trabajo arduo.

2.2 Descripción de la definición del problema

Para identificar la principal problemática referente al desarrollo de este proyecto se utiliza el Diagrama Causa-Efecto o de Ishikawa; *esta herramienta de calidad que es parte del control de la calidad en la Gestión de la Calidad del Proyecto y de Six-Sigma, que sirve para identificar, explorar y mostrar todas las posibles causas de un problema específico (efecto)* (Piattini, 2006). El diagrama Causa-Efecto se presenta en la figura 2.1, y utiliza las 5M⁴⁶ personalizadas para este trabajo, desglosa las principales causas en 3 niveles que permiten identificar el principal problema que es la deficiencia de la calidad en el desarrollo de las aplicaciones móviles.

⁴⁶ Diagrama causa-efecto. (Piattini, 2006) pág.19

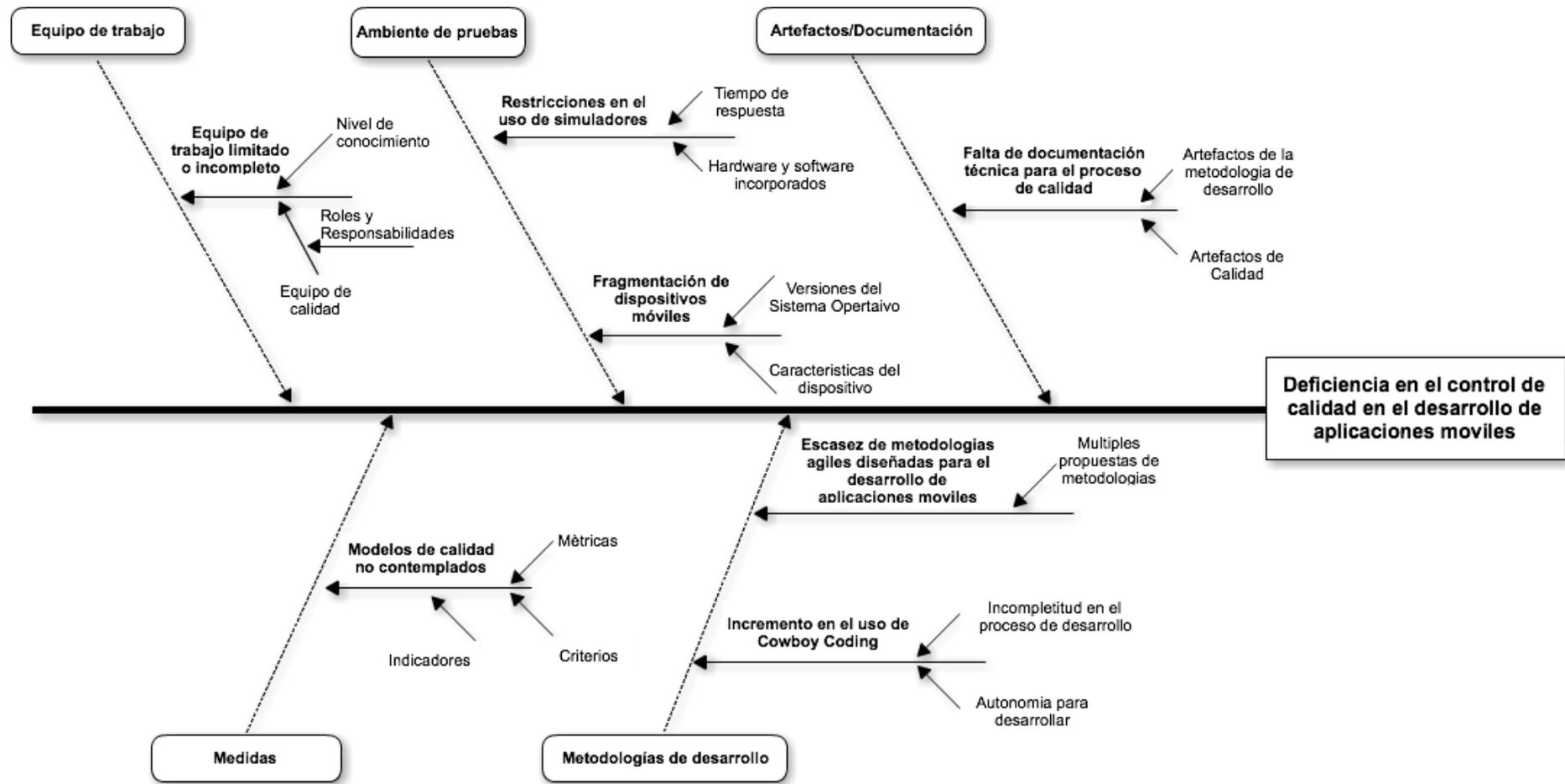


Figura 2.1. Diagrama Causa-Efecto. Deficiencia del control de calidad en el desarrollo de aplicaciones móviles.

Fuente: El autor.

En la tabla 2.1 se explica de manera detallada cada una de las causas que permitieron identificar el problema establecido como “Deficiencia del control de calidad en el desarrollo de las aplicaciones móviles” que se presenta en la figura 2.1, en la que se incluyen las principales referencias de las investigaciones y de los casos de estudio analizados sobre las aplicaciones móviles y los principales campos de investigación como el uso de metodologías para su desarrollo, las principales características y su funcionamiento, y la inclusión de la calidad en las aplicaciones móviles.

Tabla 2.1. Detalle de las causas encontradas en la deficiencia del control de calidad en el desarrollo de las aplicaciones móviles.

5M'S	CAUSAS	DESCRIPCIÓN	REFERENCIAS Y JUSTIFICACIÓN
Equipo de trabajo	Equipo de trabajo incompleto o limitado	<p>En la definición de las aplicaciones móviles que se presentan en (1), (2), (3), (4) y (5), se considera que en este ambiente de desarrollo, el equipo de trabajo es limitado o incompleto; tomando en cuenta que para el desarrollo de estas aplicaciones se presenta la necesidad de adquirir un conocimiento de nuevas tecnologías, entonces se puede decir que:</p> <ul style="list-style-type: none"> • Asignar, crear o contratar un equipo de trabajo específicamente para desarrollar aplicaciones móviles implicaría adicionar y consumir más recursos como tiempo y dinero. • Y en este caso contar con el apoyo de un equipo de calidad externo implicaría igualmente el consumo adicional de recursos o en su defecto no se considera, ya que en la mayoría de los casos los mismos miembros del equipo asumen estos roles y responsabilidades para ejecutar la validación del producto desarrollado. 	<ol style="list-style-type: none"> 1. (Flora & Wang, 2014) 2. (Daniel & Balaguera, 2013) 3. (RA-MA, 2012) 4. (Spataru, 2010) 5. (Blanco et al., 2009)
Ambiente de pruebas	Limitaciones en el uso de simuladores	<p>El uso de simuladores se considera como una experiencia muy ajustada y limitada en cuanto a la calidad de los resultados, a pesar de ser considerado como uno de las principales herramientas de validación, como una alternativa para ahorrar recursos; el uso de los simuladores no siempre producen una experiencia satisfactoria del usuario en el mundo real. Basándose en el desarrollo de las aplicaciones de (1), (2) y la definición de (3) y (4), consideran que en la mayoría de los casos se presentan inconvenientes, como:</p> <ul style="list-style-type: none"> • El tiempo de respuesta en la ejecución de las pruebas establecidas para la aplicación móvil difiere entre el entorno del simulador y un dispositivo móvil, ya que la capacidad de procesamiento del ordenador en relación del dispositivo móvil es superior. • Se presentan fallas o imperfecciones en el funcionamiento correcto de los sensores dentro del simulador, así como también en el hardware y software incorporados en el dispositivo, como: la cámara, servicios multimedia que se requieren para el correcto funcionamiento de la aplicación móvil. • La inconsistencia en la ergonomía del diseño de la aplicación en cuanto a la interfaz del usuario varían entre el simulador y el dispositivo móvil. 	<ol style="list-style-type: none"> 1. (Alvarado Ruiz, Guamán Eras, & Sigcho Armijos, 2012) 2. (Saraguro Bravo, 2012) 3. ("Desarrollo de software para móviles en 5 fases - La Oficina de Proyectos de Informática," n.d.) 4. (Fennema et al., 2016)

	Diversidad de dispositivos móviles	<p>Con base en el análisis de (1), (2), (3), (4) y (5), estos estudios consideran que al igual que las causas anteriores la fragmentación de los dispositivos móviles es una de las principales características que se debe tener en cuenta en el desarrollo de aplicaciones móviles.</p> <ul style="list-style-type: none"> • El principal problema con la fragmentación se encuentra en el momento de presentar los dispositivos. Debido a la gran diversidad existente de tecnología móvil en la actualidad, el equipo de proyecto, especialmente los desarrolladores de este tipo de aplicaciones, consideran que es complicado desarrollar funcionalidades específicas para cada versión existente de plataforma/software de la tecnología móvil; es decir, con las características y especificaciones para su correcto funcionamiento. • Además, adquirir la mayor cantidad de dispositivos móviles para ejecutar y validar el correcto funcionamiento de la aplicación resulta muy costoso. 	<ol style="list-style-type: none"> 1. (Flora & Wang, 2014) 2. (Amaya Balaguera, 2013) 3. (RA-MA, 2012) 4. (Blanco & ál, 2009) 5. (Fennema et al., 2016)
Artefactos /Documentación	Falta de documentación técnica para el proceso de calidad	<p>La falta de elaboración de documentación sobre lo desarrollado puede ocasionar el desconocimiento del funcionamiento del producto y la ausencia de pruebas para comprobar que lo desarrollado funciona correctamente. En (1), (2) y (3) consideran a la documentación/artefactos como un recurso indispensable para presentar formalmente al producto desarrollado ante los clientes.</p> <p>En el desarrollo de las aplicaciones móviles, la falta de documentación técnica se presenta cuando existe la necesidad de incluir un proceso de calidad detallado, en el que se requiere que la información establecida para cada una de las fases de la metodología ágil utilizada esté debidamente documentada, validada y aprobada.</p>	<ol style="list-style-type: none"> 1. (Wang, Jiang, & Wei, 2012) 2. (Garcia & Lago, 2010) 3. (Hernandez Trasobares, 1996)
Métricas de Calidad	Modelos de calidad no contemplados	<p>En este contexto los limitados estudios e investigaciones que se presentan como en (1), (2), (3), y (4) se han tomado como guía para establecer esta causa.</p> <p>En el proceso de desarrollo de las aplicaciones móviles, incluir un proceso de calidad a través del uso de criterios, indicadores, modelos, estándares y métricas, en el proceso de evaluación de la calidad del proyecto, del proceso y del producto es escaso, se limita a criterios específicos de evaluación o en su defecto no se contempla, esto con el objetivo de realizar una valoración cualitativa y cuantitativa que permita mejorar la calidad de la aplicación móvil a nivel de competencia en las tiendas de distribución.</p>	<ol style="list-style-type: none"> 1. (Flora & Wang, 2014) 2. (Enriquez & Casas, 2013) 3. (Silva del Rosario, 2012) 4. (Fennema et al., 2016)(Ejecutivo, Acerca, & Bjetivos, n.d.)

Métodos/ Metodologías de desarrollo	Escasez de metodologías ágiles diseñadas para el desarrollo de aplicaciones móviles	<p>A pesar de la múltiple variedad de metodologías ágiles existentes y utilizadas en el desarrollo de software convencional, en el entorno de desarrollo de las aplicaciones móviles existe una lista muy limitada de metodologías ágiles diseñadas específicamente para este entorno que se presentan en (1) o en su defecto se presentan como propuestas de metodologías híbridas adaptadas para el desarrollo de aplicaciones móviles en proceso de validación (2).</p> <p>Para utilizar y ejecutar cada una de las fases de una metodología ágil seleccionada debe tomarse en cuenta las principales características que tienen las aplicaciones móviles para su desarrollo y se detallan en el apartado 1.6.2.</p>	<ol style="list-style-type: none"> 1. Tabla 1.2 de la sección 1.9. 2. (Hidalgo Toctaguano & Iza Quizhpe, 2016) 3. (Fennema et al., 2016)
	Incremento en el uso de Cowboy Coding	<p>El uso de Cowboy Coding que se detalla en (1) y se menciona en (2) permite a los desarrolladores sobresalir con su autonomía para construir sus propias aplicaciones existiendo una escasa participación de los miembros del equipo. Esto incluye el control de la programación, lenguajes, algoritmos, herramientas, marcos del proyecto, estilo de codificación e incluso un proceso de calidad básico. Entre las principales desventajas de utilizar Cowboy Coding están:</p> <ul style="list-style-type: none"> • La falta de estructura de liberación • Desarrolladores sin experiencia • Incertidumbre en el diseño de los requisitos • Incompletitud <p>Además se considera que en el ambiente de desarrollo de aplicaciones móviles, existe una gran cantidad de sitios web que ofrecen al usuario (desarrolladores sin experiencia) la posibilidad de crear su propia aplicación; incluso sin la necesidad de poseer conocimientos de programación, del uso de metodologías y demás procesos, esto ocasiona que exista una excesiva proliferación de aplicaciones móviles en el mercado sin ningún control con los mínimos estándares de calidad y la falta de valoración por parte de los demás usuarios y de equipos de expertos técnicos.</p>	<ol style="list-style-type: none"> 1. Cowboy Coding⁴⁷ 2. (Carvajal Riola, 2008)

Fuente: EL autor.

⁴⁷ Cowboy Coding: https://en.wikipedia.org/wiki/Cowboy_coding

Cada una de las causas identificadas y detalladas en la tabla 2.1, permiten definir la deficiencia del control de calidad en el entorno de desarrollo de las aplicaciones móviles, este aspecto es determinante, tanto en el entorno de desarrollo como en el uso de las aplicaciones móviles, cuyo objetivo es garantizar la calidad de un producto que se va a ofrecer al usuario. Por lo tanto, la calidad está directamente asociada a los requisitos del software. En este contexto a través de la construcción de un marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles facilita el proceso de obtener un producto competitivo cuyo beneficio se mide en función de la calidad de los servicios que ofrece. El cual se detalla en el capítulo 3 como la solución al problema.

CAPÍTULO 3.

DISEÑO DE LA SOLUCIÓN. MARCO DE TRABAJO

3.1 Introducción

En este capítulo se presenta la solución propuesta como objetivo principal de desarrollo de este trabajo, dirigido en las causas detalladas en la definición de problema.

El objetivo de construir este Marco de Trabajo para el Control de Calidad en el Desarrollo de Aplicaciones Móviles (en adelante **MTCC**⁴⁸), es implementar de manera factible un proceso detallado de las principales actividades del control de calidad, que se despliega con las fases de Mobile-D, que es la metodología ágil seleccionada como idónea en este ámbito.

3.2 Diseño de la solución

El diseño de la solución se materializa en el MTCC; para desarrollar se utiliza como estrategia el diagrama **SIPOC**⁴⁹ que permite presentar a través de una estructura básica la interrelación y delimitación de cada uno los procesos que se van a ejecutar.

A continuación se detallan los componentes del marco de trabajo según la estrategia SIPOC, el diseño y finalmente el despliegue del MTCC.

3.2.1 Componentes del marco de trabajo

Los componentes del MTCC de acuerdo a SIPOC se detallan a continuación:

- **Proveedores (Supplier):** son las entidades, quienes proporcionan las entradas al proceso.
- **Entradas (Inputs):** son los recursos que se requieren para iniciar el proceso, todos los artefactos e información (tangible o intangible) que se necesita para ejecutar el proceso.
- **Procesos (Process):** es el conjunto de actividades relacionadas que interactúan para transformar los elementos de entrada en salidas.
- **Salidas (Outputs):** son los productos o servicios obtenidos cuando termina de ejecutarse un proceso, el cual debe tener medida o ser medible, son los resultados tangibles de un proceso.
- **Clientes (Customer):** son las entidades, quienes reciben el resultado del proceso, para quien la salida es desarrollada. El objetivo es obtener la satisfacción de este cliente.

⁴⁸ (MTCC) siglas que se utiliza en adelante para definir al "Marco de Trabajo para el Control de Calidad en el Desarrollo de Aplicaciones Móviles".

⁴⁹ Diagrama SIPOC. Disponible en: <https://genesishwh.files.wordpress.com/2011/06/05-sipoc.pdf>, homepages.stmartin.edu/fac_staff/dstout/Ba320/Lecture%20PPTs/SIPOC.ppt

3.2.2 Diseño del MTCC

A través del uso de SIPOC, que provee una vista macro del flujo del proceso y/o producto y sus interrelaciones dentro del negocio, se define los límites del proceso; es decir, los puntos de inicio y fin del proceso que se contemplan.

La estrategia del trabajo permitirá:

- Establecer e identificar de manera precisa el propósito y alcance del MTCC
- Establecer los pasos que se tiene que dar para realizar
- Definir los recursos que se necesitan
- Identificar que roles están involucrados e
- Identificar cuáles son las actividades que aportan valor para ejecutar el proceso y garantizar resultados de calidad

En la figura 3.1, se presenta el esquema principal y la secuencia de cada uno de los pasos que se deben seguir para establecer el proceso del MTCC.



Figura 3.1. Diagrama de procesos SIPOC.

Fuente: SIPOC. <https://www.isixsigma.com/tools-templates/sipoc-copis/sipoc-diagram/>.

Elaborado por: El autor

Para comprender la estructura de SIPOC se detalla la secuencia a seguir:

- **Paso1:** Identificar y definir el **proceso** de control de calidad que se va a ejecutar estableciendo los límites de inicio y fin, en función de cada una de las fases de Mobile-D.
- **Paso2:** Identificar las **salidas** que va a generar cada uno de los procesos. Los recursos (artefactos, producto e información tangible o intangible) medibles.
- **Paso3:** Identificar a los **clientes** que van a recibir las salidas del proceso, que se especifican en la ejecución de cada uno de los procesos. En este MTCC los clientes

son reconocidos como el equipo de calidad quienes van a ejecutar el control de calidad. Entre los principales roles se establecen:

- **Gerente de calidad** y líder del equipo de calidad que gestiona el proceso del control de calidad, define las estrategias y políticas del proceso de calidad que se van a llevar a cabo en el desarrollo de aplicaciones móviles.
- **Analista de pruebas:** se encarga de la planificación y dirección de las pruebas a través de la elaboración del plan de pruebas, informa de seguimiento y cierre de las pruebas, además de analizar y revisar los requisitos del usuario, diseñar y ejecutar casos de prueba, preparar el entorno de pruebas, preparar y adquirir todos los recursos necesarios para ejecutar las pruebas y revisa las pruebas desarrolladas por otros.
- **Evaluadores:** ejecuta las pruebas a todos niveles y registra los resultados, utiliza herramientas de administración o gestión de las pruebas, automatiza las pruebas.
- **Paso4:** Identificar las **entradas** que se necesitan para realizar el proceso correctamente, se consideran principalmente a todos los recursos que se generan en cada una de las fases de Mobile-D.
- **Paso5:** Identificar a los **proveedores** que proporcionan las entradas necesarias para ejecutar el proceso; en este caso se establecen a todo los miembros del equipo definidos en Mobile-D. Entre los principales roles se identifican:
 - **Equipo de dirección:** tiene la autoridad para tomar decisiones y controlar la planificación y progreso del proyecto.
 - **Equipo de proyecto:** está conformado por el líder y los desarrolladores que llevan a cabo las actividades de construir y desarrollar el software.
 - **Grupo de interesados:** proporcionan la descripción principal de la funcionalidad inicial del proyecto y participan en el desarrollo del producto.

En la figura 3.2, se presenta el diseño general del MTCC, este marco proporciona las pautas para ejecutar las principales actividades de control de calidad en el desarrollo de aplicaciones móviles con el propósito de obtener un producto de alta calidad.

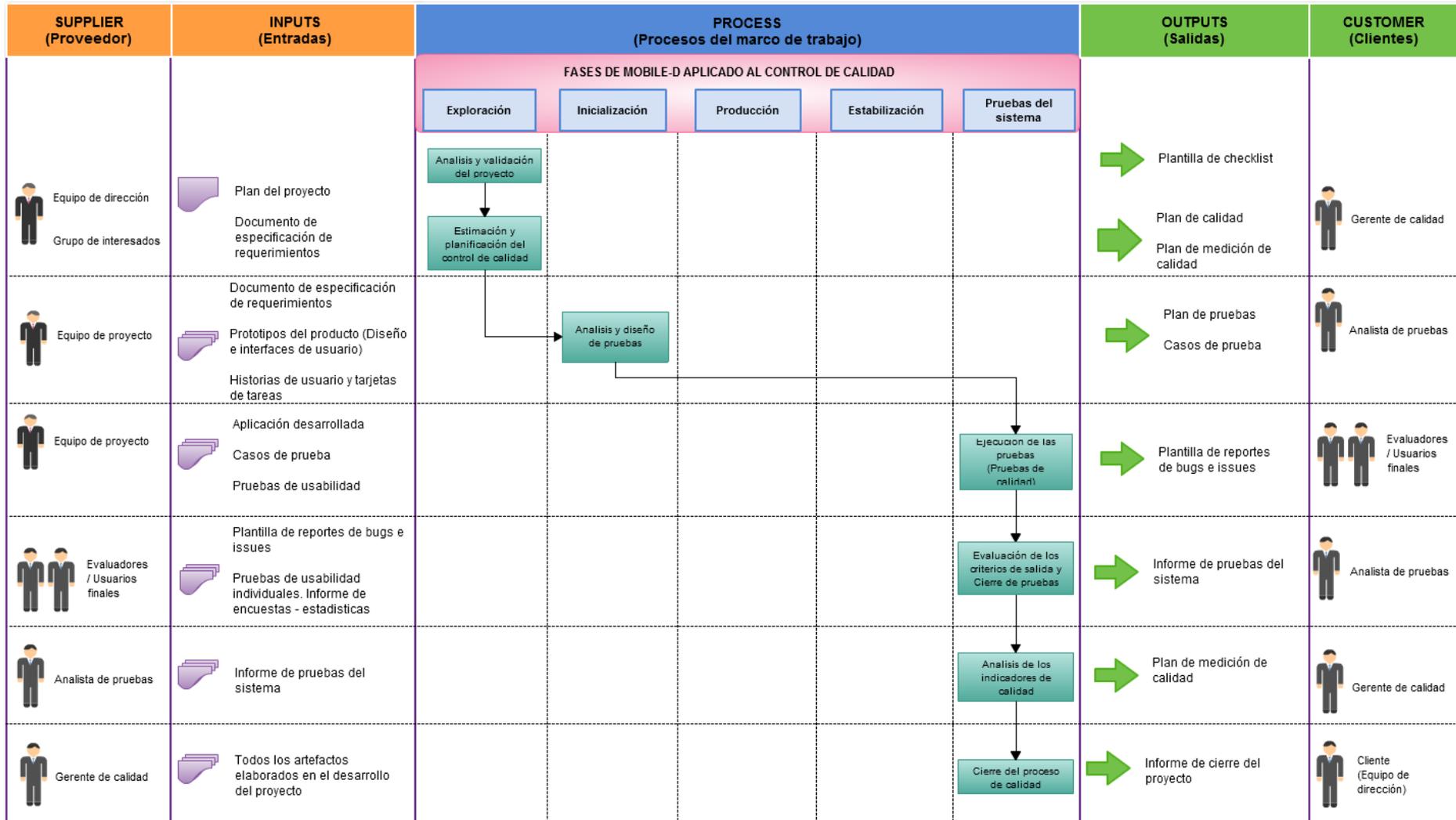


Figura 3.2. Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles

Fuente: El autor

3.3 Despliegue del marco de trabajo

Luego de presentar el diseño general del marco de trabajo, se procede a explicar cada uno de los procesos que lo conforman:

3.3.1 Análisis y validación del proyecto – Fase de Exploración

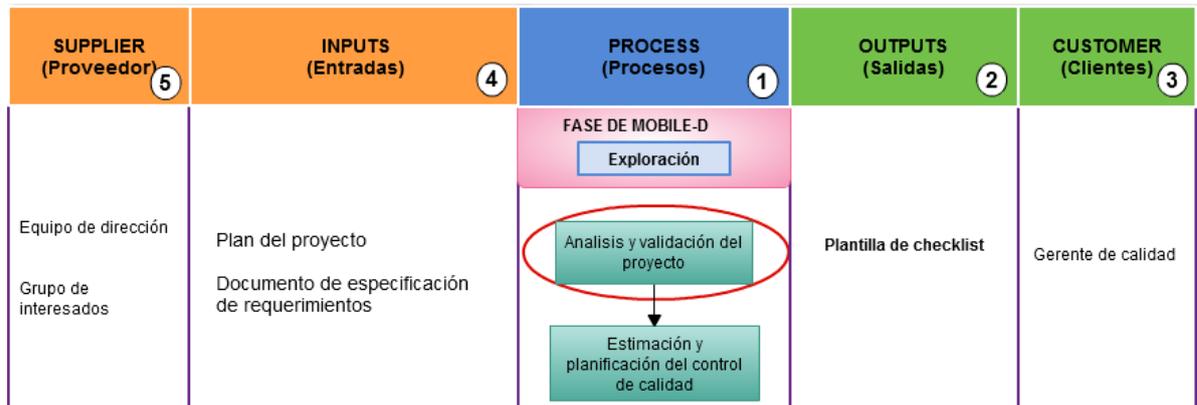


Figura 3.3. Proceso de análisis y validación del proyecto

Fuente: el autor.

3.3.1.1 Definición del proceso

El proceso de **análisis y validación del proyecto** permite controlar que la planificación y definición del proyecto cumple con lo establecido a través de los principales artefactos generados en la fase de exploración.

El objetivo de este proceso es comprobar que los recursos y requerimientos del proyecto estén debidamente registrados y para ello se utiliza una técnica específica a través de la herramienta *checklist* de validación o lista de control de calidad.

3.3.1.2 Salidas

La **salida** principal de este proceso es la plantilla de checklist o lista de control de calidad

- **Plantilla de checklist (Anexo 4):** o lista de control de calidad; es una herramienta estructurada, específica para controlar y verificar que se hayan incluido todos los recursos necesarios definidos para desarrollar el proyecto y se utiliza en cada una de las fases de desarrollo.

3.3.1.3 Clientes

Se considera como **cliente** al gerente de calidad quien va recibir las entradas para ejecutar este proceso.

3.3.1.4 Entradas

Las entradas de este proceso son elaborados en la fase de exploración y proporcionan la idea principal del producto a desarrollar.

- **Plan del proyecto (Anexo 2):** El plan del proyecto es el documento que describe el propósito y la definición del proyecto y establece la escala de recursos y tiempo para el proyecto; además de incluir la descripción del diseño y arquitectura del software, las principales actividades de formación y capacitación del equipo del proyecto si se considera necesario para el proyecto.

El plan del proyecto se actualiza de forma iterativa durante el desarrollo del proyecto y se elabora estrictamente en la fase de exploración.

- **Documento de especificación de requerimientos (ERS) (Anexo 3):** este documento permite describir la aplicación de software a ser desarrollada, incluyendo beneficios relevantes, objetivos y metas. Está dirigido al equipo de dirección y al grupo de interesados y dar a conocer el desarrollo de la propuesta del producto.

3.3.1.5 Proveedores

Se consideran como proveedores al equipo de dirección y el grupo de interesados quienes proporcionan las principales entradas para ejecutar el proceso.

3.3.2 Estimación y planificación del control de calidad – Fase de Exploración

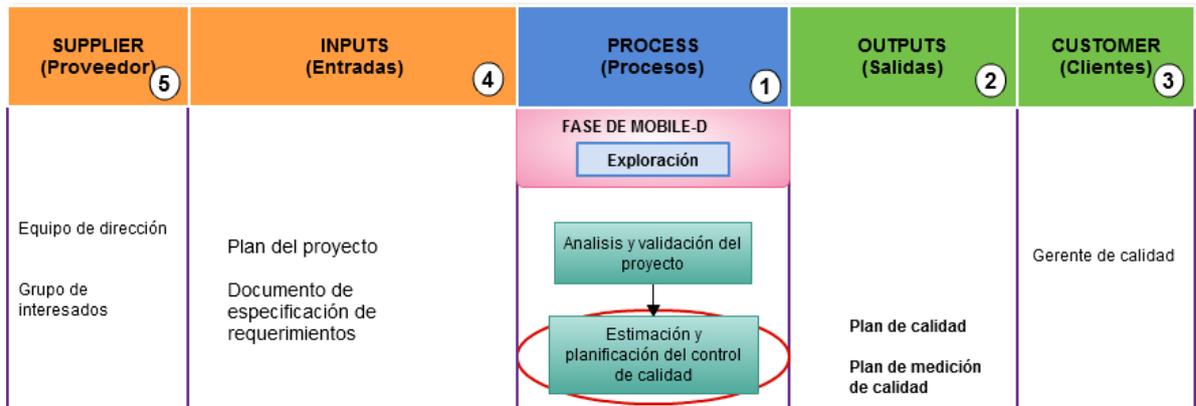


Figura 3.4. Proceso de estimación y planificación del control de calidad

Fuente: el autor.

3.3.2.1 Definición del proceso

Una vez finalizado el proceso de análisis y validación del proyecto, se procede a ejecutar el **proceso de estimación y planificación del control de calidad** que es parte de la fase de exploración.

La estimación y planificación del control de calidad permite determinar los aspectos de calidad que se deben llevar a cabo en todo el desarrollo del proyecto y se logra a través de la definición de un plan de calidad y de un plan de medición.

3.3.2.2 Salidas

Las **salidas** de este proceso son:

- **Plan de calidad (Anexo 5):** este documento especifica que procesos, procedimientos y recursos asociados se aplicaran, por quien y cuando, para cumplir los requisitos de un proyecto, producto o contrato específico.

Este plan de calidad se desarrolla en función del plan del proyecto; es decir, proporciona las principales entradas para la dirección del proyecto, y aborda los procesos de control de calidad, aseguramiento de calidad y mejora continua de los procesos en relación al desarrollo del software.

La inclusión del plan de calidad en la fase de exploración en este MTCC tiene como propósito principal reducir el reproceso de actividades.

- **Plan de medición de calidad (Anexo 6):** el propósito de este documento es especificar las métricas de calidad que permitirán realizar la evaluación y control de calidad del producto en las diferentes etapas del ciclo de vida del desarrollo del software.

3.3.2.3 Clientes

Se considera como cliente al **gerente de calidad** quien va recibir las entradas para ejecutar este proceso.

3.3.2.4 Entradas

Las entradas de este proceso son elaborados en la fase de exploración y proporcionan la idea principal del producto a desarrollar, en este caso las entradas se encuentran actualizadas.

- El plan del proyecto (Anexo 2)
- Documento de especificación de requerimientos (ERS) (Anexo 3)

3.3.2.5 Proveedores

Se consideran como proveedores al **equipo de dirección** y **el grupo de interesados** quienes proporcionan las principales entradas para ejecutar este proceso.

3.3.3 Análisis y diseño de pruebas – Fase de inicialización

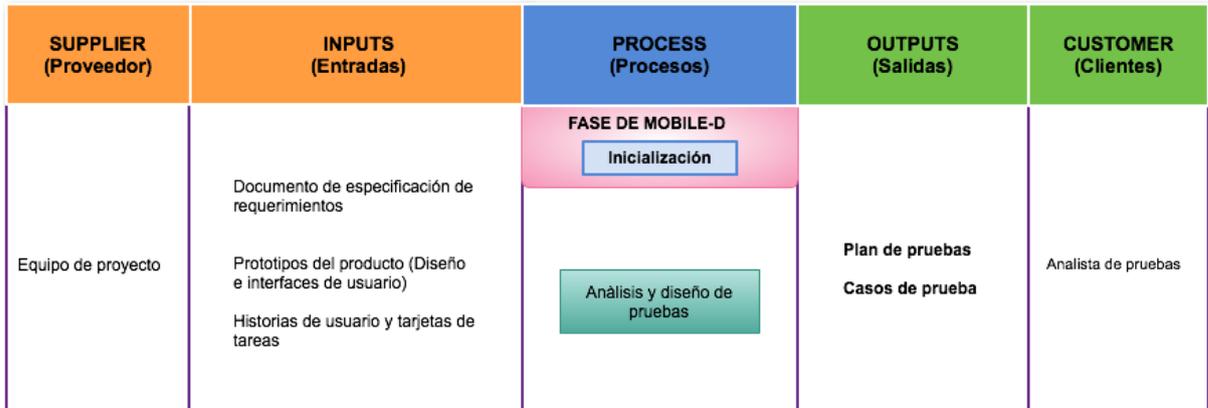


Figura 3.5. Proceso de análisis y diseño de pruebas

Fuente: el autor.

El éxito de este y de los siguientes procesos del MTCC se basa en haber abordado una comprensión íntegra del proyecto y del producto.

3.3.3.1 Definición del proceso

El proceso de **análisis y diseño de las pruebas** es parte de la fase de inicialización, que consiste en la preparación de los recursos físicos, técnicos y humanos establecidos por el grupo de clientes.

Específicamente, el control de calidad se define el entorno de pruebas y las herramientas para la ejecución de las pruebas en las aplicaciones móviles.

Este proceso del MTCC se ejecuta a partir de los documentos de especificación de requerimientos, historias de usuario, tarjetas de tareas; con el propósito de desarrollar el plan de pruebas para las aplicaciones móviles. Este plan de pruebas se desarrolla con base a la estructura que define el ISTQB, específicamente en este proceso se cumplen las actividades siguientes:

- **Planificación y control de pruebas:** la planificación es la actividad de definir los objetivos de las pruebas y la especificación de actividades con el propósito de cumplir con los objetivos y la misión establecidos. El control de pruebas es la actividad constante de comparar el proceso real con el plan previsto, e informar sobre el estado de las pruebas.
- **Análisis y diseño de pruebas:** es la actividad donde los objetivos de las pruebas se transforman en condiciones de prueba y casos de prueba tangibles y sus respectivos

datos de entrada, los cuales serán utilizados para evaluar la calidad del software y determinar si cumplen con los requisitos del software solicitados por el usuario.

3.3.3.2 Salidas

Las salidas en este proceso adquieren características y condiciones especiales debido a que son aplicados al entorno de pruebas de las aplicaciones móviles.

Las principales **salidas** de este proceso son:

- Plan de pruebas
- Plantilla de casos de prueba
- **Plan de pruebas (Anexo 7):** en este documento se desarrolla el enfoque para asegurar los objetivos de calidad en el ciclo de desarrollo de las aplicaciones móviles, y los aspectos clave de las pruebas del software. Este plan de pruebas se presenta a los principales miembros del equipo del proyecto (equipo de dirección, equipo de exploración y desarrollo) y el equipo de calidad.

El plan de pruebas para las aplicaciones móviles difiere en aspectos referentes a la de una aplicación de escritorio en relación a la definición del entorno de pruebas, las herramientas de pruebas (dispositivos móviles, emuladores), riesgos y mitigaciones que se deben tener en cuenta para ejecutar las pruebas.

- **Plantilla de casos de prueba (Anexo 8):** la plantilla de casos de prueba con sus respectivos datos de entrada es uno de los principales artefactos de salida del proceso de análisis y diseño de las pruebas, tiene el propósito de evaluar la calidad del software y determinar si cumple con los requerimientos establecidos por el grupo de interesados.

Tabla 3.5.1. Plantilla de casos de prueba.

Id Caso de prueba	Identificador (código) único.		
Nombre del Caso de Prueba	Título descriptivo del caso de prueba.		
Descripción	Descripción del caso de prueba, indicando sus elementos, funcionalidades y acciones a ser ejercidas en el caso de prueba.		
Funcionalidad a probar:	Nombre de la funcionalidad a ejecutar		
Pre-requisitos	Lista de los pre-requisitos		
Post-condiciones	Lista de las post-condiciones		
Escenario:			
Paso a seguir			
Resultados esperados:		Resultados obtenidos	
Errores asociados:			
Estado	Ejecutado ()	Exitoso () Fallido () Parado ()	
	Pendiente en ejecución ()		
	En construcción ()		
Prioridad	Alta () Media () Baja ()		

Fuente: PMO. <http://www.pmoinformatica.com/2014/06/plantilla-de-casos-de-prueba.html>

Elaborado: El autor

- **Definición de escenarios de prueba en el desarrollo de aplicaciones móviles**

Al diseñar los casos de prueba se identifican todos los posibles escenarios que se pueden presentar para ejecutar las pruebas en el desarrollo de las aplicaciones móviles. En el **Anexo 7** se detallan los principales tipos de pruebas que se pueden ejecutar, con base a PMO e ISTQB.

3.3.3.3 Clientes

Se considera al **analista de pruebas** como el cliente quien va a recibir las entradas y ejecutar el análisis y diseño de las pruebas y construir los principales artefactos en relación al ambiente de las aplicaciones móviles.

3.3.3.4 Entradas

Como entradas se presentan los principales artefactos definidos en la fase de inicialización, como son:

- *Documento de especificación de requerimientos (Anexo 3)*
- *Prototipos del producto (Diseño de interfaces de usuario)*
- *Historias de usuario y/o tarjetas de tareas*

Estos artefactos permiten analizar la construcción del plan de pruebas que se aplicará en el entorno de pruebas de las aplicaciones móviles.

Además en este proceso se validan los primeros prototipos del diseño de la aplicación, siguiendo los principales estándares de diseño de las tiendas de distribución de Android⁵⁰ e iOS⁵¹ de acuerdo a las proporciones, densidades y resoluciones que exigen.

Cada uno de estos artefactos se actualiza de forma iterativa durante el desarrollo del proyecto.

3.3.3.5 Proveedores

Se consideran al **equipo de proyecto (desarrollo)** como el principal proveedor quienes proporcionan las principales entradas para ejecutar este proceso.

⁵⁰ Android design. <https://developer.android.com/design/index.html>

⁵¹ iOS Human Interface Guidelines. <https://developer.apple.com/ios/human-interface-guidelines/>

3.3.4 Ejecución de las pruebas – Fase de pruebas del sistema

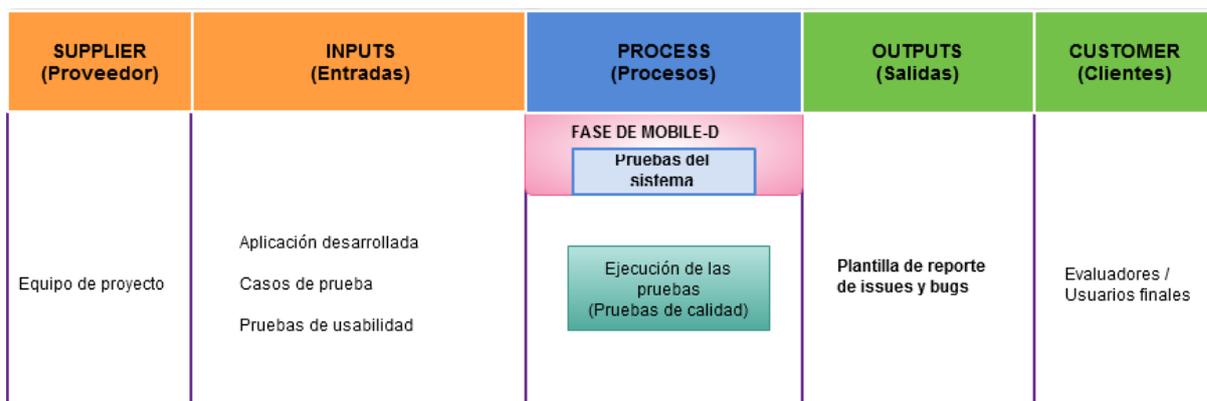


Figura 3.6. Proceso de ejecución de las pruebas

Fuente: el autor.

Este proceso depende de que se hayan ejecutado y finalizado las fases de producción y de estabilización de Mobile-D.

- La **fase de producción** implementa las funcionalidades mediante el proceso iterativo e incremental a través de las etapas de planificación-trabajo-liberación y la validación de lo desarrollado.

Esta fase se basa en la ejecución de las pruebas unitarias y la programación en pares que permite a los programadores validar el código desarrollado.

- La **fase de estabilización** ejecuta las pruebas de integración de todo el sistema e incluye la producción de documentación, considerando que son ejecutadas directamente por los desarrolladores, para así entregar una versión estable del producto final al equipo de calidad con el propósito de ejecutar el control de calidad.

3.3.4.1 Definición del proceso

El proceso de **ejecución de las pruebas** que es parte de la fase de pruebas del sistema de Mobile-D, se especifican los procedimientos de prueba mediante la definición de los casos de prueba en un orden determinado para ejecutar las pruebas. Entre las principales actividades de este proceso se incluyen:

- La finalización, priorización e implementación de los casos de prueba.
- La verificación del entorno de pruebas correctamente configurado y listo.
- La ejecución de las pruebas conforme a la secuencia prevista.

- La iteración continua de las actividades de ejecución de las pruebas que permite eliminar los errores y defectos encontrados, con el objetivo de garantizar que el software funciona correctamente.
- La comparación de los resultados reales con los resultados esperados.

3.3.4.2 Salidas

La principal **salida** de este proceso es:

Matriz de reporte de bugs e issues (Anexo 9): La matriz de reporte de errores (issues y bugs) permite describir de manera detallada cada uno de los inconvenientes que presenta el equipo de calidad al realizar la validación de la aplicación móvil a través de la ejecución de las pruebas establecidas.

Esta plantilla detalla los issues y bugs que permite presentar al equipo de desarrollo para asegurar su corrección.

3.3.4.3 Clientes

Se considera a los **evaluadores** como los clientes quienes van a recibir las entradas y ejecutar las pruebas.

3.3.4.4 Entradas

La entrada principal en este proceso corresponde al producto desarrollado, la aplicación móvil liberada en una versión estable, de acuerdo a las necesidades definidas por el cliente y los principales artefactos actualizados hasta esta fase de desarrollo.

- Aplicación móvil
- Plan de pruebas
- Casos de prueba

Cada uno de estos artefactos se actualiza de forma iterativa durante el desarrollo del proyecto.

3.3.4.5 Proveedores

Se considera al equipo de proyecto como el principal **proveedor** quien proporciona las principales entradas para ejecutar este proceso.

3.3.5 Evaluación de los criterios de salida e informes y cierre de las pruebas – Fase de pruebas del sistema

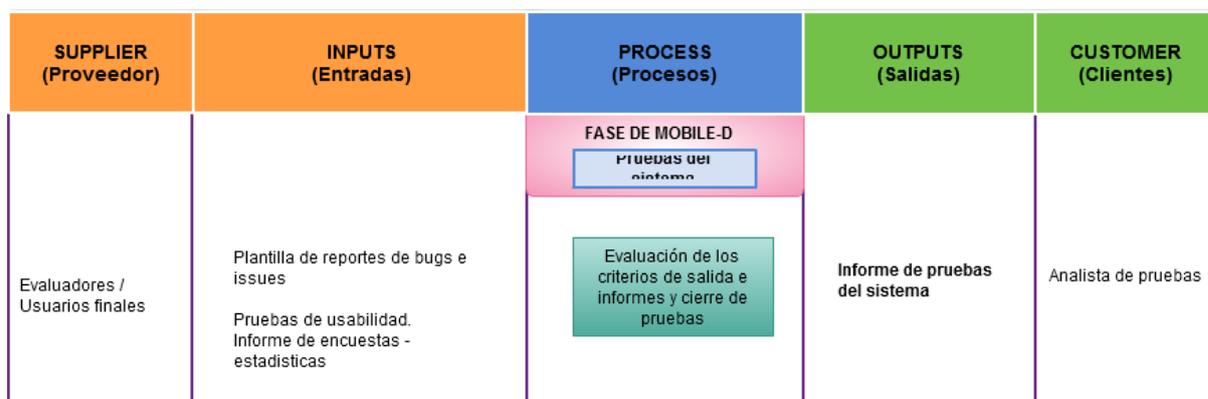


Figura 3.7. Proceso de evaluación de los criterios de salida y cierre de las pruebas

Fuente: el autor.

3.3.5.1 Definición del proceso

El proceso de **evaluación de los criterios de salida e informes** permite validar la ejecución de pruebas frente a los requerimientos definidos. Se procede a revisar la matriz de reportes de issues y bugs para verificar el estado de los casos de prueba y comprobar que los resultados de las pruebas ejecutadas cumplan con los criterios (condiciones) de salida definidos en el plan de pruebas. Se analiza y evalúa si es necesario ejecutar más pruebas para validar el producto o si se deben modificar los criterios (condiciones) de salida definidos.

El proceso de **cierre de las pruebas** tiene como propósito la finalización del proceso de pruebas y realizar el lanzamiento del producto software. Recopila la información de las pruebas ejecutadas y finalizadas y se cierra la matriz de reportes issues y bugs, con el objetivo de comprobar que el o los productos han sido entregados efectivamente.

3.3.5.2 Salidas

La salida de este proceso genera el **informe de ejecución de pruebas** revisado y aprobado (**Anexo 11**) una vez terminada la ejecución de las pruebas.

3.3.5.3 Clientes

Se consideran como clientes a los **evaluadores** quienes finalizan el proceso de ejecución de pruebas y generan los informes de pruebas del sistema.

3.3.5.4 Entradas

La principal entrada corresponde al producto desarrollado, que es la aplicación móvil en una versión estable. Además, también se considera como entradas a este proceso a los principales artefactos actualizados, que son los que se enlistan a continuación:

- Plantilla de reportes de bugs e issues (Anexo 9)
- Pruebas de usabilidad individuales. Informe de encuestas – estadísticas.(Anexo 10)

3.3.5.5 Proveedores

Se considera como proveedores al **equipo de proyecto** (líder de desarrollo) y al **equipo de calidad** (analista de calidad) que proporciona los artefactos de entrada para ejecutar la evaluación de los criterios de salida e informes y cierre de pruebas.

3.3.6 Análisis de los indicadores de calidad – Fase de Pruebas del sistema

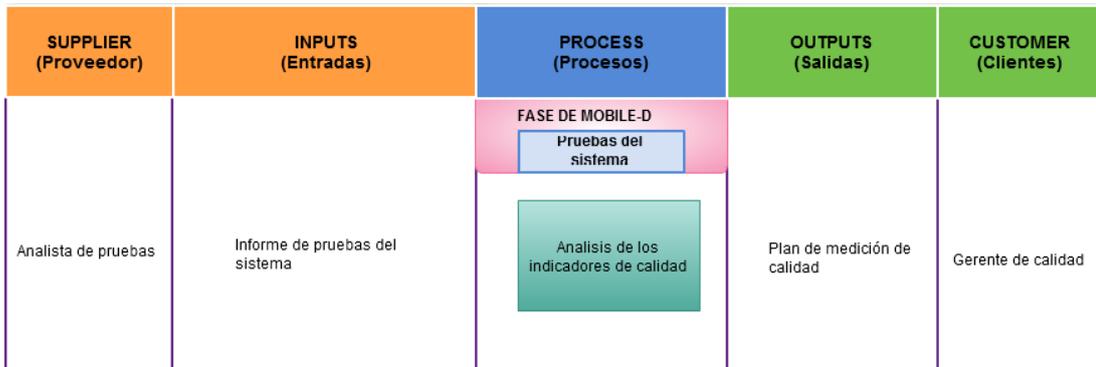


Figura 3.8. Proceso de análisis estadístico de los indicadores de calidad

Fuente: el autor.

3.3.6.1 Definición del proceso

El proceso de **análisis de los indicadores de calidad** permite analizar los datos de las pruebas ejecutadas para obtener e interpretar las métricas y evaluar la satisfacción del cliente. Se realiza a través de la medición del producto que utiliza las métricas asociadas en las diferentes etapas del ciclo de vida del desarrollo del software, se basa en las tres características de calidad del Modelo ISO/EC 9126 que son: calidad interna, calidad externa y calidad de uso.

3.3.6.2 Salidas

Como salida principal para este proceso se presenta el **plan de medición revisado y aprobado**.

3.3.7 Clientes

Se considera como cliente al **Gerente de calidad** que es el responsable de ejecutar este proceso.

3.3.7.1 Entradas

Las principales entradas para la ejecución de este proceso son:

- Informe de ejecución de pruebas (**Anexo 11**)

3.3.7.2 Proveedores

En este proceso se considera como proveedor al **analista de pruebas** quien proporciona las principales entradas para ejecutar este proceso.

3.3.8 Cierre del proceso de control de calidad – Fase de pruebas del sistema

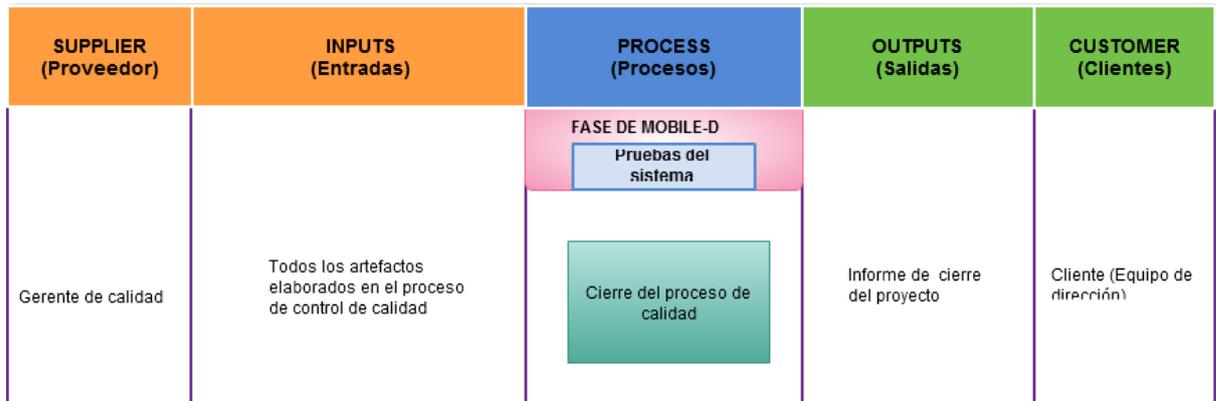


Figura 3.9. Proceso de cierre del proceso de control de calidad

Fuente: el autor.

3.3.8.1 Definición del proceso

El proceso de cierre de calidad consiste en la entrega formal del producto desarrollado y el cierre del proyecto a través de la entrega de los documentos validados y aprobados son todos los artefactos elaborados en todo el proceso de control de calidad definidos como entradas y salidas del marco de trabajo.

3.3.8.2 Salidas

El principal entregable de este proceso es:

- Informe final de entrega del producto aprobado (**Anexo 12**).

3.3.8.3 Clientes

Se considera como clientes al **equipo de dirección** a quien se entrega las salidas que son los responsables de aprobar cada uno de los entregables.

3.3.8.4 Entradas

Las entradas en este proceso son todos los artefactos elaborados en el proceso de control de calidad aprobados en sus últimas versiones revisadas y aprobadas.

- Plan del proyecto aprobado
- Documento de especificación de requerimientos aprobado
- Prototipos del producto (Diseño e interfaces de usuario)
- Historias de usuario y tarjetas de tareas
- Plan de calidad aprobado.

- Plan de medición aprobado.
- Plantilla de Checklist del proyecto aprobado.
- Plan de pruebas aprobado.
- Casos de prueba aprobados
- Pruebas de usabilidad
- Plantilla de matriz de reportes de bugs e issues cerrado y aprobado
- Informe de pruebas del sistema aprobado.
- Plan de medición. Informe de métricas de calidad aprobado.
- Informe de cierre del proyecto aprobado

3.3.8.5 Proveedores

El **gerente de calidad** del equipo de control de calidad asume el papel de proveedor quien entrega todas las entradas actualizadas en su última versión para ejecutar este último proceso.

CAPÍTULO 4.

VALIDACIÓN DEL MARCO DE TRABAJO

4.1 Validación del marco de trabajo

El uso masivo y la constante evolución y mejora de la tecnología móvil en la actualidad ha permitido que el desarrollo de las aplicaciones móviles se incremente en este entorno, facilitando el uso y los beneficios prácticos que ofrecen este tipo de aplicaciones a los usuarios. Es así, que las aplicaciones móviles se están convirtiendo en elementos clave para facilitar la comunicación e interacción con los usuarios a través de actividades y funciones muy concretas, con el objetivo de mejorar la productividad de los servicios que ofrece una empresa.

Por tal razón, se propone utilizar el MTCC, en el desarrollo de las aplicaciones móviles con el fin de facilitar el seguimiento y evaluación de la calidad desde una filosofía de desarrollo ágil sin perder la documentación elemental en el desarrollo de software.

El desarrollo de este capítulo comprende la implementación y validación del marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles bajo el uso de metodologías ágiles.

4.2 Caso práctico – “Gestión de tutorías UTPL”

El proyecto *“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”* y su principal producto, la aplicación móvil *“Gestión de tutorías UTPL”* que actualmente se encuentra en proceso de desarrollo como TFT (Trabajo de Fin de Titulación).

La información del desarrollo de este proyecto se ha contrastado con el diagrama causa/efecto que se presenta en la figura 2.1 del capítulo 2. Definición del problema, donde se ha comprobado que existe una deficiencia del control de calidad en el desarrollo de la aplicación móvil. La documentación del proyecto seleccionado se encuentra almacenada en el repositorio GIT-TAW⁵². Es por esto, que el proyecto mencionado es adecuado para ejecutar el MTCC desde el primer proceso.

4.3 Implementación y validación del MTCC

La implementación y validación del MTCC, permite ejecutar cada uno de los procesos que se detallan en la figura 3.2 del capítulo 3. Diseño de la solución en el proyecto seleccionado; donde se incluye: el análisis y validación del proyecto, la estimación y planificación del proyecto, el análisis y diseño de pruebas, la ejecución de las pruebas, la evaluación de los criterios de salida y cierre de las pruebas, el análisis de los indicadores de calidad y el cierre

⁵² Repositorio GIT-Tecnologías Avanzadas en la Web de UTPL.
<https://git.taw.utpl.edu.ec/dimoreno/Tesis/tree/master>

del proceso de calidad con el objetivo de presentar una aplicación móvil de mayor calidad y demanda.

4.3.1 Situación inicial del proyecto seleccionado

En el desarrollo de la aplicación móvil “Gestión de tutorías UTPL” la definición de las fases de desarrollo, la identificación de roles y los artefactos utilizados se establecen a través de la metodología ágil SCRUM que ha sido seleccionada específicamente para el proceso de desarrollo de software.

4.3.2 Identificación de roles

En este caso práctico, el MTCC establece a los proveedores y los clientes con los roles específicos en cada uno de los procesos:

Proveedores

- **Equipo de dirección:** Gerente del proyecto, Directora del TFT de aplicación móvil, quien controla la planificación y el progreso del desarrollo del proyecto.
- **Grupo de interesados:** Vicerrectorado de UTPL. Proporcionan las necesidades y objetivos para el desarrollo del proyecto.
- **Equipo de desarrollo:** (Desarrolladores de aplicación móvil-Tesistas). Construyen y desarrollan lo que se ha solicitado y aprobado en el proyecto.

Clientes

- **Gerente de calidad:** Directora del TFT - MTCC, gestiona el proceso de control de calidad del MTCC en el proyecto seleccionado.
- **Analista de pruebas:** (Experto en pruebas-Tesista del MTCC). Se encarga de la planificación, seguimiento y cierre de las pruebas de calidad.
- **Evaluadores de pruebas:** (Tesista del MTCC) ejecutan la aplicación en los dispositivos móviles, además de revisar, detallar y reportar los errores encontrados.
- **Usuarios finales:** Estudiantes de la UTPL. Ejecutan la aplicación Gestión de tutorías UTPL en sus dispositivos móviles y las pruebas de usabilidad.

A continuación se presenta detalladamente la implementación y validación de cada uno de los procesos del MTCC en la aplicación móvil “Gestión de tutorías UTPL”.

4.3.2.1 Análisis y validación del proyecto - fase de inicialización

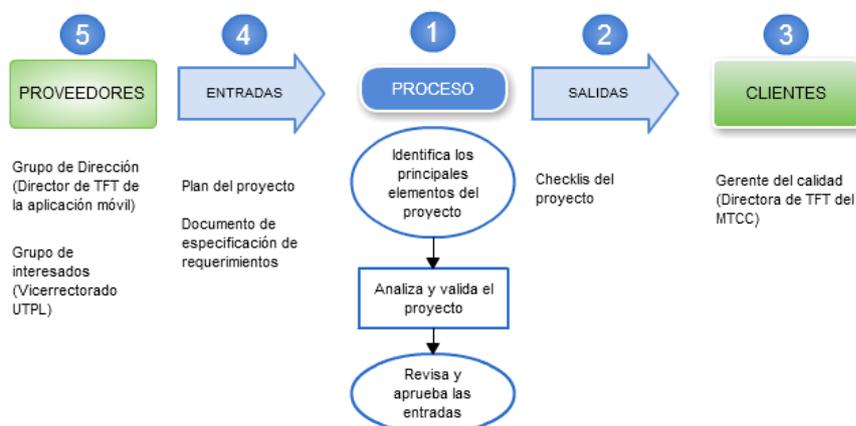


Figura 4.1. Proceso de análisis y validación del proyecto “Gestión de tutorías UTP”

Fuente: el autor.

Definición del proceso

En este proceso se realiza una entrevista formal con el equipo del proyecto, que tiene como objetivo principal, aceptar e incluir el uso del MTCC con el proyecto seleccionado, cuyo propósito es formalizar el proceso de desarrollo del proyecto sin perder la simplicidad y la agilidad que ofrece la metodología ágil seleccionada específicamente en el entorno de desarrollo de software (aplicación móvil “Gestión de tutorías UTP”).

Este proceso permite analizar y validar que la planificación y la definición del proyecto se han establecido, y los principales recursos se encuentran debidamente considerados y registrados.

A pesar de no contar con los artefactos que permitan presentar de manera formal la documentación del proyecto al grupo de interesados, se ha identificado que la información del proyecto se presenta de manera general. Esto se debe a que en el desarrollo de la aplicación móvil se ha utilizado la metodología ágil SCRUM. En este caso, se incluye la elaboración de los principales artefactos solicitados como entradas en el MTCC.

Salidas

Como salida de este proceso contamos con el *Checklist del proyecto (Anexo 4)* que contiene la verificación que el proyecto cumple con los principales elementos y características para continuar con la implementación de los siguientes procesos.

Clientes

El **gerente de calidad** verifica lo solicitado a través de las entradas de este proceso.

Entradas

En este caso el documento disponible donde se encuentra la información de la aplicación móvil “Gestión de tutorías UTPL” es la memoria de TFT.

De acuerdo al MTCC se solicita como entradas principales el Plan del proyecto (Anexo 2) y el Documento de Especificación de Requerimientos de Software (Anexo 3) que se han elaborado y presentado como principales artefactos del proyecto, tomando como referencia la memoria de TFT del proyecto seleccionado.

Proveedores

El **equipo de dirección y el grupo de interesados** colaboran con la elaboración de los principales artefactos solicitados como entradas en el MTCC y a su vez son revisados y aprobados.

4.3.2.2 Estimación y planificación del control de calidad – Fase de exploración



Figura 4.2. Proceso de estimación y planificación del control de calidad del proyecto “Gestión de tutorías UTPL”
Fuente: el autor.

Definición del proceso

Este proceso identifica los aspectos de calidad de todo el proyecto con el propósito de determinar los principales aspectos de calidad que se van a llevar a cabo conjuntamente en el desarrollo del proyecto y del producto, y especificar las métricas que permitirán la evaluación y control de calidad del producto (aplicación móvil “Gestión de tutorías UTPL”) en las diferentes etapas del ciclo de vida del desarrollo del software.

Salidas

Los artefactos que se obtienen y se presentan en este proceso son:

- Plan de calidad (Anexo 5) revisado y aprobado.
- Plan de medición de calidad (Anexo 6) revisado y aprobado.

Cientes

El gerente de calidad presenta las salidas para dar a conocer el proceso de control de calidad que se va llevar a cabo durante todo el desarrollo del proyecto.

Entradas

Los principales artefactos de entrada han sido revisados y aprobados, con el propósito de realizar una presentación formal del proyecto y continuar con el desarrollo.

- Plan del proyecto revisado y aprobado

- Documento de especificación de requerimientos revisado y aprobado

Proveedores

El **equipo de dirección** y **grupo de interesados** entregan las entradas con la finalidad de dar a conocer el objetivo del desarrollo del proyecto.

4.3.2.3 *Análisis y diseño de pruebas – Fase de inicialización*

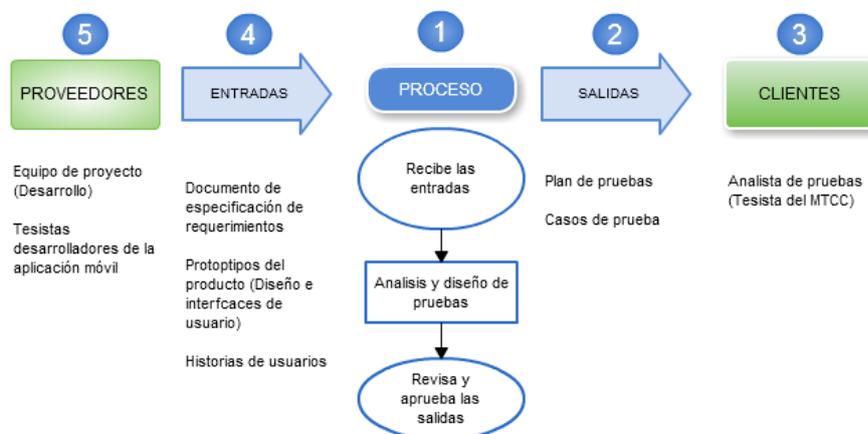


Figura 4.3. Proceso de análisis y diseño de pruebas “Gestión de tutorías UTPL”

Fuente: el autor.

Definición del proceso

En este proceso se realiza la preparación de las pruebas de calidad. Se elaboran los casos de pruebas. Se establece el entorno de pruebas, las herramientas para la ejecución de las pruebas, a través de la elaboración del plan de pruebas en donde se especifican y detallan las pruebas que se van a ejecutar en la aplicación móvil tomando como referencia la información de las entradas de este proceso.

Salidas

- Plan de pruebas (Anexo 7) revisado y aprobado
- Documento de Casos de prueba revisados y aprobados (Anexo 8).

Clientes

El cliente de este proceso es el **Analista de pruebas** que presenta las salidas al equipo de proyecto (Desarrollo).

Entradas

- Documento de especificación de requerimientos.
- Prototipos del producto (Diseño e interfaces de usuario)
- Historias de usuario

Proveedores

El **equipo de proyecto** (desarrollo) presenta las entradas revisadas y aprobadas a los clientes para ejecutar este proceso.

4.3.2.4 Ejecución de las pruebas – Fase de pruebas del sistema

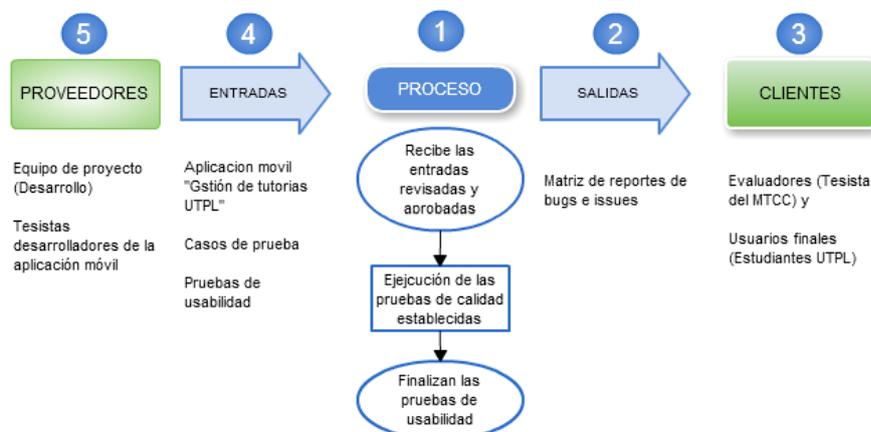


Figura 4.4. Proceso de ejecución de las pruebas "Gestión de tutorías UTPL"

Fuente: el autor.

Definición del proceso

El equipo de desarrollo ha ejecutado las pruebas unitarias del código desarrollado cumpliendo en la fase de producción lo planificado. Ejecuta y finaliza la fase de estabilización a través de la integración de lo desarrollado; cumpliendo en su totalidad con los requerimientos establecidos, las historias de usuarios y los casos de pruebas, es decir, el equipo de desarrollo entrega el producto completo, la aplicación móvil "Gestión de tutorías UTPL", para proceder a ejecutar las pruebas.

En este proceso los clientes utilizan y validan la aplicación móvil "Gestión de tutorías UTPL" en sus dispositivos móviles.

Salidas

En la plantilla de reporte de bugs e issues (Anexo 9) se detallan los errores encontrados en la ejecución de las pruebas de la aplicación móvil "Gestión de tutorías UTPL" y se reportan al equipo de desarrollo a través del repositorio GIT-TAW para que se resuelvan los inconvenientes reportados.

Cientes

El equipo de pruebas se componen de dos tipos de roles: **evaluadores y usuarios finales** que reciben las entradas para validar la aplicación móvil "Gestión de tutorías UTPL", considerando como actividad adicional para los evaluadores (rol del equipo de calidad) quienes revisan, reportan y detallan los errores presentados en las pruebas de usabilidad en la salida correspondiente.

Entradas

Las entradas solicitadas en este proceso, son las siguientes:

- Aplicación móvil “Gestión de tutorías UTPL”
- Pruebas de usabilidad (Anexo 10) que se establecen como encuestas generadas en la herramienta www.e-encuestas.com enviadas a los usuarios finales que a su vez se clasifican en rol de Docente y Estudiante.

Proveedores

El **equipo de desarrollo** del proyecto entrega las entradas para ejecutar este proceso y validar la calidad del producto.

4.3.2.5 Evaluación de los criterios de salida e informes y cierre de pruebas – Fase de pruebas del sistema

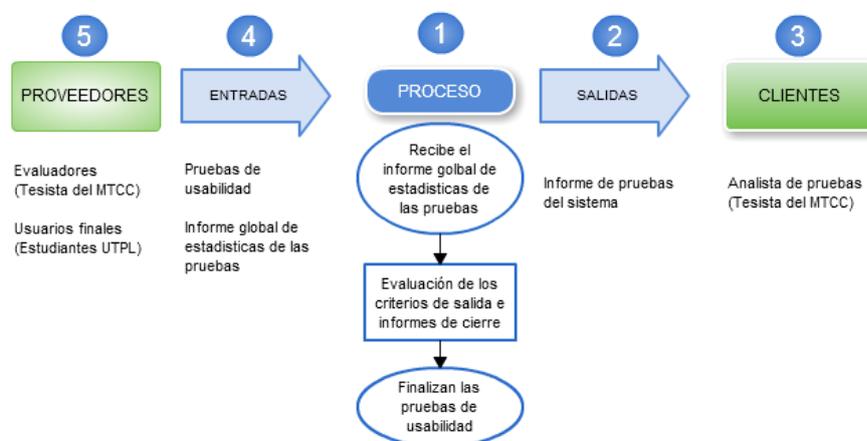


Figura 4.5. Proceso de Evaluación de los criterios de salida e informes y cierre de pruebas “Gestión de tutorías UTPL”
Fuente: el autor.

Definición del proceso

En este proceso se constatan los resultados de la ejecución de las pruebas y los requerimientos aprobados. Se ha revisado y corregido los issues y errores que se detallan en la matriz de reportes y el estado de los casos de pruebas se encuentran estado ejecutado y exitoso y el cierre de las pruebas de usabilidad aplicados en la herramienta seleccionada.

Se concluye que los resultados de todas las pruebas ejecutadas satisfacen los criterios de calidad establecidos en el plan de pruebas para determinar que se ha finalizado el ciclo de pruebas de la aplicación móvil “Gestión de tutorías UTPL”.

Salidas

A través de la entrega del **informe de pruebas del sistema** se concluye el proceso de ejecución de pruebas (Anexo 11) y se procede a entregar la aplicación móvil.

Cientes

El **analista de pruebas** entrega la salida principal que detalla los resultados de las pruebas que se ejecutaron en la aplicación móvil.

Entradas

Como entradas en este proceso se requirió de:

- Las **pruebas de usabilidad** a través de las encuestas llenadas individuales.
- El informe global de estadísticas que genera la herramienta utilizada.

Proveedores

Los **evaluadores y usuarios finales** son los encargados de entregar la entrada principal

4.3.2.6 Análisis de los indicadores de calidad – Fase de pruebas del sistema

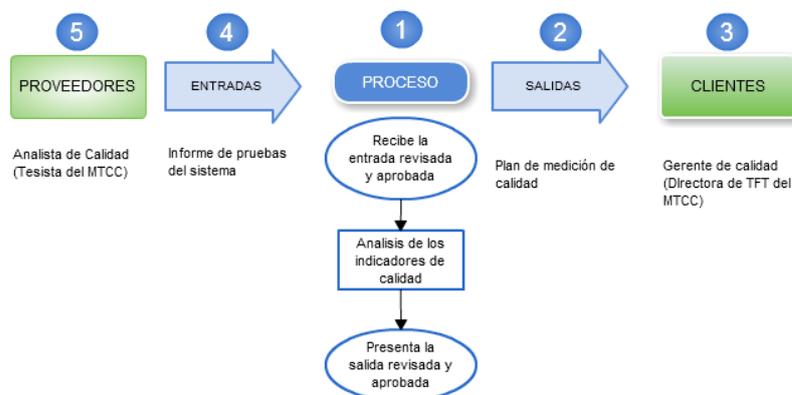


Figura 4.6. Proceso de Evaluación de los criterios de salida e informes y cierre de pruebas “Gestión de tutorías UTPL”
Fuente: el autor.

Definición del proceso

Este proceso permite generar los indicadores de calidad de la aplicación móvil. Se procede a realizar el análisis de los datos obtenidos de las pruebas ejecutadas e interpretar los resultados que se utilizan para medir y evaluar la calidad de la aplicación móvil “Gestión de tutorías UTPL” a través de la satisfacción del usuario.

Este proceso permite simplificar la tarea de implementar acciones preventivas y el control de los resultados alcanzados.

Salidas

El plan de medición de calidad del proyecto (Anexo 6), se establece como el artefacto principal para el cierre de este proceso y del MTCC, que permite concluir que se puede ejecutar el siguiente proceso; es decir, cierre del proceso de control de calidad.

Clientes

El **gerente de calidad** como responsable de la dirección del proceso de control de calidad en este MTCC es el encargado de entregar la salida principal de este proceso, que lo recibe el **gerente del proyecto**

Entradas

Los artefactos solicitados como entradas en este proceso son:

- Informe de cierre del proceso (Anexo 12)

Proveedores

En este proceso el **analista de calidad** proporciona el informe de pruebas de sistema revisado y aprobado para ejecutar este proceso.

4.3.2.7 Cierre del proceso del control de calidad – Fase de pruebas del sistema



Figura 4.7. Cierre del proceso del control de calidad. “Gestión de tutorías UTPL”
Fuente: el autor.

Definición del proceso

El cierre del proceso de control de calidad de este MTCC, se ejecuta con el propósito de realizar el cierre formal del desarrollo del proyecto; es decir, se acepta el resultado (aplicación móvil Gestión de tutorías UTPL) y se llega a la terminación del proyecto.

El cierre del proceso de control de calidad incluye organizar y archivar la documentación del proyecto, tanto de los artefactos definidos como entradas y salidas del MTCC.

Salidas

Como salida principal de este proceso se presenta el **informe de cierre del proyecto** (Anexo 12) revisado y aprobado.

Clientes

Se define como cliente en este proceso de cierre del proyecto al **Grupo de interesados** y el **Equipo de dirección** definidos en el MTCC quienes reciben todos los artefactos elaborados en el MTCC que representa la entrega formal del proceso de control de calidad en el desarrollo del proyecto seleccionado y su producto principal la aplicación móvil “Gestión de tutorías UTPL”.

Entradas

Como entradas de este proceso se definen a todos los artefactos elaborados, establecidos como entradas y salidas del MTCC revisados y aprobados y presentados en su versión final.

Tabla 4.1. Artefactos del MTCC

Artefactos de la Metodología Mobile-D	Artefactos de MTCC
Plan del proyecto	Plantilla de checklist
Documento de especificación de requerimientos	Plan de calidad
Historias de usuario, Diagramas de clases, casos de uso, Prototipos del producto (Diseño e interfaces de usuario)	Plan de medición
Documento de estándares de código desarrollado/Manual de usuario del desarrollador	Plan de pruebas
Casos de pruebas	Informe de pruebas del sistema
Matriz de reporte de bugs e issues	Informe de cierre del proceso de control de calidad

Fuente: el autor

Proveedores

El **gerente de calidad** entrega todos los artefactos del MTCC, estén debidamente revisados y aprobados para asegurarse de que todo el trabajo está completo y que el proyecto ha cumplido con los objetivos definidos.

4.4 Análisis de resultados del MTCC

El objetivo del MTCC es poderse incluir en el desarrollo de aplicaciones móviles independientemente del entorno y las necesidades, que permite gestionar el control de calidad a través de procesos que se ejecutan de forma paralela con las fases de desarrollo de la metodología ágil seleccionada Mobile-D; estos procesos se han validado en el caso práctico “Gestión de tutorías UTPL”.

En el caso práctico “Gestión de tutorías UTPL”, el desarrollo de este proyecto se ha basado en la metodología ágil SCRUM aplicada específicamente a la etapa de desarrollo, que ajusta el uso de los principales elementos; es decir, la identificación de roles, el uso de artefactos y un proceso de desarrollo detallado a través de la definición de procesos, sin perder la filosofía de la agilidad.

La validación del Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles, MTCC, en el caso práctico “Gestión de tutorías UTPL” se detalla para cada uno de los procesos:

Análisis y validación del proyecto) este proceso se ha ejecutado de manera completa, incluyendo la elaboración de los artefactos de entrada, y la verificación y validación de los principales elementos/recursos y características del proyecto a través del checklist, y se continúa con el siguiente proceso.

Estimación y planificación del control de calidad) ha permitido identificar los principales aspectos de control de calidad a través del uso del Plan de Calidad y el Plan de medición para evaluar en las diferentes etapas del desarrollo del proyecto y de producto”.

Análisis y diseño de pruebas) a través de las historias de usuario elaborados por el equipo de desarrollo, se han construido los casos de prueba y el plan de pruebas que contienen las principales características y consideraciones que se han especificado para ejecutar la aplicación móvil “Gestión de tutorías UTPL”.

Ejecución de las pruebas) en este proceso los evaluadores y usuarios finales (Docentes y Estudiantes de UTPL) han utilizado la aplicación móvil, y han completado la encuesta para validar la aplicación móvil “Gestión de tutorías UTPL”.

El analista de calidad interviene en este proceso, ha obtenido la información de los usuarios finales a través de la encuesta y ha verificado los problemas/inconvenientes presentados en el uso de la aplicación móvil, y se han reportado a través de la matriz de reporte de errores y bugs y han sido solucionados en su totalidad por el equipo de desarrollo.

Evaluación de los criterios de salida y cierre de las pruebas) el analista de pruebas ha evaluado los criterios de aceptación de salida de las pruebas a través del plan de pruebas y ha presentado el informe de cierre de pruebas revisado y aprobado.

Análisis de los indicadores de calidad) a través del plan de medición el analista de calidad ha evaluado cada una de las métricas del proyecto y producto en las fases de desarrollo y el análisis de las métricas de calidad a través del modelo establecido, y concluye que la aplicación móvil “Gestión de tutorías UTPL” cumple con las necesidades y requerimientos establecidos por el grupo de interesados y la satisfacción de los usuarios finales.

Cierre del proceso de control de calidad) se ha revisado y entregado cada uno de los artefactos establecidos en el MTCC al Equipo de dirección, con el propósito de formalizar y cerrar el proceso el desarrollo del proyecto y el proceso de control de calidad.

El MTCC optimiza el uso de recursos compartidos desde el inicio, en el proceso de desarrollo del proyecto y del producto, independientemente de la metodología ágil de desarrollo seleccionada, con el objetivo de presentar aplicaciones móviles exitosas.

CONCLUSIONES

Al finalizar el presente trabajo, se concluye que:

- Las aplicaciones móviles están siendo desarrolladas mediante el uso de metodologías ágiles de desarrollo de software convencional (escritorio/Web), a pesar de que este tipo de aplicaciones posee condiciones y características diferentes.
- Para la eficiencia del MTCC se requiere implicación por parte de los gerentes del proyecto, desarrolladores y equipo de calidad para ejecutar exitosamente cada uno de los procesos.
- La inclusión de todo un proceso de control de calidad en cada una de las fases de desarrollo no siempre es evidente, considerando que este debería iniciar en la fase de levantamiento de requerimientos y no al final cuando la aplicación ya ha sido desarrollada.
- La complejidad del control de calidad en el desarrollo de aplicaciones móviles radica desde el inicio de su construcción/desarrollo, es necesario validar teniendo en cuenta el contexto del desarrollo de las aplicaciones móviles y las características de funcionalidad, usabilidad seguridad y rendimiento que poseen las mismas.
- A pesar de la existencia de modelos y atributos de calidad para medir la calidad del producto, difícilmente el equipo del proyecto aplica en su producto. El MTCC facilita e incluye un proceso de calidad para garantizar a los usuarios la calidad del servicio que exigen.

RECOMENDACIONES

Al culminar el presente proyecto y para proyectos futuros se recomienda:

- En la actualidad existen muchos proyectos de desarrollo cuyo principal objetivo es presentar una aplicación móvil, es por esto que se sugiere conveniente incluir el MTCC en el desarrollo del proyecto, con el fin de asegurar un proceso de control de calidad completo y eficiente.
- Para optimizar la definición de procesos con sus respectivos artefactos roles y actividades en el desarrollo de un proyecto, se sugiere utilizar la herramienta SIPOC que es lo suficientemente básico para su adaptación.
- En el proceso de análisis de los indicadores de calidad del MTCC a través de los resultados obtenidos en la ejecución de las pruebas, para validar la calidad del software en función de las características y sub-características, se recomienda basarse en el modelo de calidad ISO/EC 9126 que permite evaluar de forma cualitativa y cuantitativa una aplicación móvil a través de indicadores y valores establecidos que formaliza la especificación de los requisitos de calidad y obtener un producto de calidad exitoso.
- El MTCC se puede implementar desde cualquier fase de desarrollo del proyecto ya que es adaptable y ágil para aplicarlo y continuar con su validación y usabilidad, para determinar las mejoras, ajustes y el grado de adecuación a los proyectos de desarrollo de aplicaciones móviles.

BIBLIOGRAFÍA

- Al-khanjari, Z., Kraiem, N., Al-shanfari, A., & Baomar, O. (2014). A Software Development Methodology to Extend E-Services to M-Services (eM 2 ethodology).
- Alvarado Ruiz, P., Guamán Eras, D., & Sigcho Armijos, J. (2012). *Aplicación de tecnologías móviles para la búsqueda de recursos educativos abiertos*. Loja.
- Andres, R., Genoff, A., & Paola, P. (2012). Apps móviles como herramientas de apoyo al aprendizaje matemático informal en Educación Superior Resumen Introducción Contexto Universitario.
- Bahamon L, J. H. (n.d.). Control de calidad en el software.
- Blanco, P., Camarero, J., Fumero, A., Werterski, A., & Rodriguz, P. (2009). Metodología de desarrollo ágil para sistemas móviles. Introducción al desarrollo con Android y el iPhone. *Universidad Politécnica de Madrid*, 1–30.
- Bolaños Lima, B. A. (2013). Diseño de la investigación del proyecto dante sistema móvil para la notificación, ubicación y alerta de incidentes y áreas de peligro en la ciudad de Guatemala.
- Bonilla, E., Santiago, G., Isarael, A., & Armadas-espe, F. (2014). Diseño e implementación de una aplicación móvil que cumpla la función de estación en tierra para el monitoreo de UAV' s en el centro de investigación y desarrollo de la fuerza aérea.
- Bravo, R. A. S. (2012). Implementación de una Aplicación Android basada en Realidad Aumentada aplicada a Puntos de Interés de la UTP.
- Cáceres, R. A., Genoff, R. A., Ayala, L., & Zachman, P. P. (2013). Diseño de una aplicación de aprendizaje matemático basada en tecnología Android. *XVIII Congreso Argentino de Ciencias de La Computación*, 556–565. Retrieved from <http://hdl.handle.net/10915/31962>
- Cadavid, A. N., Daniel, J., Martínez, F., & Vélez, J. M. (2013). Revisión de metodologías ágiles para el desarrollo de software A review of agile methodologies for software development, 30–39.
- Carvajal Riola, J. C. (2008). Metodologías. Agiles: Herramientas y modelo de desarrollo para aplicaciones java EE como metodología empresarial, 215. Retrieved from <https://upcommons.upc.edu/bitstream/handle/2099.1/5608/50015.pdf?sequence=1>
- Corral, L., Sillitti, A., & Succi, G. (2013). Software development processes for mobile systems: Is agile really taking over the business? *2013 1st International Workshop on the Engineering of Mobile-Enabled Systems, MOBS 2013 - Proceedings*, 19–24. <https://doi.org/10.1109/MOBS.2013.6614218>
- Daniel, Y., & Balaguera, A. (2013). Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles . Estado actual.
- Desarrollo de software para móviles en 5 fases - La Oficina de Proyectos de Informática. (n.d.). Retrieved May 8, 2015, from <http://www.pmoinformatica.com/2014/03/desarrollo->

de-software-moviles-5-fases.html

- Díaz, S. (2013). Mejores prácticas en las pruebas de aplicaciones móviles Tests en dispositivos o en simuladores Mejores prácticas en pruebas de aplicaciones móviles Tipo de pruebas.
- Education, I. J. M., Science, C., Khalid, A., Zahra, S., & Khan, M. F. (2014). Suitability and Contribution of Agile Methods in Mobile Software Development, (February), 56–62. <https://doi.org/10.5815/ijmeecs.2014.02.08>
- Ejecutivo, I. R., Acerca, I. I., & Bjetivos, a O. (n.d.). FOMENTAR EL DESARROLLO Y USO DE APLICACIONES MÓVILES Índice.
- Enriquez, J. G., & Casas, S. I. (2013). Usabilidad En Aplicaciones Móviles. *Ict.Unpa.Edu.Ar*, 1–23. <https://doi.org/1852 - 4516>
- Eom, H.-E., & Lee, S.-W. (2013). Human-centered software development methodology in mobile computing environment: agent-supported agile approach. *EURASIP Journal on Wireless Communications and Networking*, 2013(1), 1–16. <https://doi.org/10.1186/1687-1499-2013-111>
- Fennema, M. C., Herrera, S. I., Palavecino, R. A., Ruiz, P. J. N., Budán, D., Suárez, G. I., & Córdoba, M. (2016). Aumentada , Herramientas De Medición , Desarrollo Híbrido, 539–543.
- Flora, H. K., & Chande, S. V. (2013). A Review and Anaysis on Mobile Application Development Processes Using Agile, 3(4), 9–18.
- Flora, H. K., & Wang, X. (2014). Adopting an Agile Approach for the Development of Mobile Applications, 94(17), 43–50.
- Gamboa Safia, D. L. (2015). Aplicación móvil para el control de notas de los estudiantes de la Universidad Técnica de Ambato utilizando la plataforma Android, 5(54), 453544.
- Garcia, I., & Lago, A. (2010). Framework para la evaluación de la calidad en uso de servicios móviles.
- Garcia Rodriguez, M. J. (2015). Estudio Comparativo entre las metodologías ágiles y las tradicionales para la gestión de proyectos software.
- Grimaldo Botero, G. (2013). Desarrollo de aplicación móvil de apoyo a la plataforma web del observatorio “Monitoreo de variables físicas y fisiológicas en niños y adolescentes en edad escolar de Risaralda.”
- Heberto, F., Arturo, V. C., María, M., Guadalupe, R., Erick, D. B., & Luis, C. J. (2014). Diseño de una metodología para el desarrollo de aplicaciones móviles en empresas PyMES de la región centro occidente de Universidad Popular Autónoma del Estado de Puebla , Posgrados Instituto Tecnológico de Morelia, 1–13.
- Hernandez Trasobares, A. (1996). Los sistemas de información: Evolución y desarrollo.
- Hidalgo Toctaguano, L. R., & Iza Quizhpe, M. G. (2016). Análisis Comparativo De Las Metodologías De Desarrollo Móvil: Hybrid Methodology Desing Y Mobile- D, Caso Práctico Implementación De Una Aplicación De Visita Virtual a La Casa De Los

- Marqueses De Miraflores En La Ciudad De Latacunga Durante El Periodo 201, 147. Retrieved from <http://www.utmachala.edu.ec/portal/public/general/informacion/hl/es/item/12-24-3>
- Jeong, Y. J., Lee, J. H., & Shin, G. S. (2008). Development process of mobile application SW based on agile methodology. *International Conference on Advanced Communication Technology, ICACT*, 1, 362–366. <https://doi.org/10.1109/ICACT.2008.4493779>
- Kaleel, S., & Harishankar, S. (2013). Applying Agile Methodology in Mobile Software Engineering: Android Application Development and its Challenges. *Computer Science Technical Reports*, 1–11. Retrieved from http://digitalcommons.ryerson.ca/compsci_techrpts/4/?utm_source=digitalcommons.ryerson.ca%2Fcompsci_techrpts%2F4&utm_medium=PDF&utm_campaign=PDFCoverPages
- Piattini, M. (2011). *Calidad de sistemas informaticos*.
- Piattini, M., García, F., & Caballero, I. (2006). Calidad de Sistema Informáticos. In RA-MA (Ed.), *Calidad de Sistema Informáticos* (p. 12,58,151).
- PMI. (n.d.). PMBOK. In *Guía de los fundamentos para la dirección del proyectos (Guía del PMBOK)*.
- Ponce, U., Mcs, M., Víctor, I. T. S., Moreno, Y., Rafael, M., & Soto, A. (2014). Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para Dispositivos Android, (3), 1429–1434.
- Prieto, M. D. (2011). Tecnología y desarrollo en dispositivos móviles.
- Rahimian, V., & Ramsin, R. (2008). Designing an agile methodology for mobile software development: a hybrid method engineering approach. *Research Challenges in Information Science*, 337–342. <https://doi.org/10.1109/RCIS.2008.4632123>
- Rosado, A. A. (2015). Comparación de dos tecnologías de desarrollo de aplicaciones móviles desde la perspectiva del rendimiento como atributo de calidad perspective as Quality attribute, *20(1)*, 81–87.
- Sánchez Cacique, L. J. (2014). Desarrollo de una aplicación para dispositivos Android y diseño de servicio web, para el manejo de las operaciones por parte de clientes y personal de empresas aseguradoras, 16–35. Retrieved from <http://biblioteca.unet.edu.ve/index.php?module=fast>
- Sardasht Mahmood, J. L. (2013). An Investigation into Mobile Based Approach for Healthcare Activites Occupational Therapy System.
- Silvia, L., Pérez, V., María, A., Hernández, S., Escobar, J. P., Francisco, A., ... Pérez, V. (2016). GENERAR SISTEMAS DE INFORMACIÓN, (120), 1461–1478.
- Spataru, A. C. (2010). Agile development methods for mobile applications. *The University of Edinburgh, Edinburgh*.
- Supan, D., Tekovic, K., Skalec, J., & Zlatko, S. (2013). Using mobile-d methodology in development of mobile applications: challenges and issues. Retrieved from

http://www.researchgate.net/publication/270904005_Aspectos_relevantes_desarrollo_digital_rigido_por_modelos_para_aplicaciones_mviles

Version One. (2016). EXECUTIVE SUMMARY. *The 10th Annual State of Agile Report*, 10, 16.

Wang, Y., Jiang, M., & Wei, Y. (2012). A Software Quality Framework for Mobile Application Testing. *The Fourth International Conference on Advances in System Testing and Validation LifeCycle*, (c), 68–72.

Yanquén Ramírez, C. A., & Otálora Luna, J. E. (2016). Medición de la usabilidad en el desarrollo de aplicaciones educativas móviles. *Revista Virtual Universidad Católica Del Norte*, (47), 128–140.

ANEXOS

Anexo 1. Estudios de las metodologías ágiles en el entorno de desarrollo de aplicaciones móviles

Tabla 1.2. Estudios de las metodologías ágiles en el entorno de desarrollo de aplicaciones móviles

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
1	Artículo	Diseño de una metodología para el desarrollo de aplicaciones móviles en empresa PYMES de la región centro occidente de México	(Heberto et al., 2014)	2014	México	7	XP Scrum AUP MASAM Mobile-D DIMP	D	MDSIC (Modelo de Desarrollo de Software Integra y Colaborativo)	<ul style="list-style-type: none"> Analiza e incluye a las metodologías ágiles más utilizadas en el mercado de desarrollo de las aplicaciones móviles El objetivo de esta metodología es que sea un referente de desarrollo y permita competitividad para las empresas de software. Detalla el diseño de la metodología MDSIC. MDSIC está orientada al desarrollo rápido de aplicaciones y a las necesidades de las PYMES que desarrollan software para dispositivos móviles.
2	Artículo	Diseño e implementación de una aplicación móvil que cumpla la función de estación en tierra para el monitoreo de UAV' s en el centro de investigación y desarrollo de la fuerza aérea	(Bonilla, Santiago, Isarael, & Armadas-espe, 2014)	2014	Ecuador	1	Mobile-D	I	Mobile-D	<ul style="list-style-type: none"> Establece a Mobile-D como la metodología de desarrollo de la aplicación, cuyo objetivo es conseguir ciclos de desarrollo rápidos en periodos de tiempo cortos a través de cada una de sus fases. A través del uso de Mobile-D seleccionan y establecen los requisitos iniciales con los cuales se trabaja, cuyo propósito es presentar pequeños y completos demos de la aplicación. Basándose en cada una de la fase de Mobile-D se diseña e implementa la aplicación cumpliendo con todas las especificaciones necesarias para presentar al usuario final.
3	Artículo	A Software Development Methodology to Extend E-Services to M-Services (eM ² ethodology)	(Al-khanjari, Kraiem, Al-shanfari, & Baomar, 2014)	2014	Muscat, Asia	4	FDD ⁵³ Mobile-D	D	eM ² ethodology (Electronic to Mobile Methodology to	<ul style="list-style-type: none"> eM²ethodology es una combinación de las metodologías ágiles existentes y utilizadas en el entorno móvil y web diseñado para desarrollar aplicaciones

⁵³ FDD (Feature Driven Development) <http://www.agilemodeling.com/essays/fdd.htm>

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
									extend E-Services to M-Services)	<p>web para dispositivos móviles.</p> <ul style="list-style-type: none"> • La metodología está diseñada con el propósito de reducir los costos de las pruebas, la construcción de productos de calidad y acelerar el proceso de entrega. • Se definen 4 fases: <ul style="list-style-type: none"> • Factibilidad de estudio • Análisis • Generación de servicios (Inicialización, Proceso de iteración de: diseño, implementación, testing, integración y validación) • Liberación de la aplicación • La metodología se ha validado en un caso de estudio, el producto es una aplicación web móvil eficiente.
4	Artículo	Suitability and Contribution of Agile Methods in Mobile Software Development	(Education et al., 2014)	2014	Taxila, Pakistan	4	Mobile-D Rapid-7 MASAM (Mobile Application Software Agile Methodology) Sless Lean HMD(Híbrido Methodology Design)	A	Cualquiera de las metodologías analizadas	<ul style="list-style-type: none"> • Presenta las características principales de las aplicaciones móviles que se deben tener en cuenta en el entorno de desarrollo. • Las metodologías ágiles son reconocidas como la mejor práctica para el desarrollo de aplicaciones móviles. • Realiza un análisis de las metodologías ágiles que tiene más demanda y se han implementado en el entorno de las aplicaciones móviles. • Las metodologías ágiles para el desarrollo de software para móviles aún se están introduciendo, ya que es un proceso continuo, para elegir alguna metodología entre las detalladas hay que ir a través de los pros y los contras del uso de estas metodologías, al desarrollar alguna aplicación específica, y luego elegir la que tiene las máximas ventajas. • La selección depende del desarrollador que se adhiere al enfoque, dependiendo de la complejidad del producto y el tamaño del equipo de proyecto.

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
5	Artículo	Adopting an Agile Approach for the Development of Mobile Applications	(Flora & Wang, 2014)	2014		8	Mobile-D HMD MASAM Sless Scrum XP Lean Kanban	A	XP Scrum Lean	<ul style="list-style-type: none"> Realiza un análisis de las principales características de las metodologías ágiles diseñadas para el desarrollo de aplicaciones móviles. Presenta como parte del trabajo el resultado de varias encuestas realizadas a expertos de la comunidad de desarrollo de software para móviles que recomiendan que la integración de enfoques ágiles han demostrado ser beneficioso en el desarrollo de software. Con base en el estudio de la literatura y los resultados de la investigación, el estudio evalúa el uso del enfoque ágil para el éxito en el desarrollo de aplicaciones móviles se da mediante la determinación de la importancia de los paradigmas de ingeniería agile más utilizados: XP, Scrum, y Lean basándose en sus principales características.
6	Artículo	Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para Dispositivos Android	(Ponce et al., 2014)	2014	México	4	Scrum Kanban Desarrollo por prototipos MVC	D	Metodología (D3A)	<ul style="list-style-type: none"> La metodología propuesta deriva de un proceso iterativo e incremental con las principales características de Scrum, Kanban, Desarrollo por prototipos y Modelo Vista-Controlador. Establece roles: Cliente, Asesor del proyecto y equipo de desarrollo. El ciclo de desarrollo está formado de las fases: Recolección de requerimientos, diseño rápido, definición de tareas, construcción del prototipo (interfaces y código), evaluación del prototipo por el representante del proyecto, evaluación del prototipo por el cliente y lanzamiento. Se presenta el prototipo funcional de la metodología listo para presentarlo al cliente, especialmente en aquellas PYMES que deseen adentrarse a la programación de tecnología móvil.

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
7	Artículo	Diseño de una aplicación de aprendizaje matemático basada en tecnología Android	(Cáceres, Genoff, Ayala, & Zachman, 2013)	2013	Argentina	1	Mobile-D	I	Mobile-D	<ul style="list-style-type: none"> • Selecciona la metodología Mobile-D como una aproximación ideal para el desarrollo ágil de aplicaciones móviles. • Cada una de las fases de Mobile-D se aplican directamente en el desarrollo del proyecto. • La aplicación se presentó y se entregó a los docentes para la que fue desarrollada y fue aceptada de manera positiva.
8	Artículo	Software development processes for mobile systems: Is agile really taking over the business?	(Corral, Sillitti, & Succi, 2013)	2013	Italia	5	Mobile-D MASAM Hybrid Scrum Sless	A	Cualquiera de las metodologías analizadas	<ul style="list-style-type: none"> • Realiza un análisis sobre como contribuye y se adapta el desarrollo ágil y las metodologías ágiles en un entorno real. • El uso de las metodologías ágiles aportan el valor de ayudar a mantenerse al día con las tendencias de negocios móviles y de comercialización de un producto. • La selección de la metodología a implementar va depender del tipo de negocio y la aceptación de la empresa y de los equipos de desarrollo.
9	Artículo	Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles. Estado actual.	(Daniel & Balaguera, 2013)	2013	Colombia	5	XP Scrum TDD Mobile-D HMD Mobile-D process spiral	A	Mobile-D	<ul style="list-style-type: none"> • Se exponen las principales características de las metodologías ágiles más referenciadas, con mayor presencia de documentación en internet, orientadas a desarrollo de tamaño reducido como las aplicaciones móviles y con la mayor aceptación dentro de los equipos de desarrollo. • Detalla las generalidades que se deben tener en cuenta en el entorno de desarrollo de aplicaciones para dispositivos móviles. • Presenta un estudio detallado de las

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										<p>metodologías ágiles diseñadas específicamente para el desarrollo de aplicaciones móviles.</p> <ul style="list-style-type: none"> El uso de la Mobile-D es una excelente alternativa para guiar proyectos de desarrollo de software, como las aplicaciones móviles, gracias a la facilidad de adaptación que poseen, y que pueden ser adaptadas a las características especiales de los dispositivos móviles con el fin de obtener productos de calidad.
10	Artículo	Human-centered software development methodology in mobile computing environment: agent-supported agile approach	(Eom & Lee, 2013)	2013	República de Corea	3	XP Scrum Mobile-D	D	Scrum Mobile-D	<ul style="list-style-type: none"> En este trabajo se propone una metodología de desarrollo de software para móviles para apoyar y mejorar los cambios en los requisitos de adaptación del producto. Incluye el proceso de desarrollo iterativo e incremental de Scrum, XP y Mobile-D. La metodología propuesta MCE (Mobile Computing Environment) define procesos y artefactos que se puede aplicar y utilizar en el proceso de desarrollo de software para móviles. Se establece un caso de estudio (desarrollo de una aplicación) para validar la metodología.
11	Artículo	A Review and Analysis on Mobile Application Development Processes Using Agile	(Flora & Chande, 2013)	2013	India	6	Mobile-D Rapid/ HMD MASAM Sless	A	Cualquiera de las metodologías analizadas	<ul style="list-style-type: none"> Presenta una revisión de proceso de desarrollo de aplicaciones móviles utilizando el enfoque ágil, selecciona y detalla las metodologías ágiles que se han utilizado en varios trabajos. Un método ágil apropiado es seleccionado para un proyecto

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										determinado que se puede adaptar a un requisito específico sobre la base de la complejidad y el tamaño del equipo de proyecto.
12	Artículo	Applying Agile Methodology in Mobile Software Engineering: Android Application Development and its Challenges	(Kaleel & Harishankar, 2013)	2013	Canadá	1	Scrum	A	Scrum	<ul style="list-style-type: none"> En la ingeniería de software para móviles, el uso de las metodologías ágiles es de gran importancia, ya que las aplicaciones de software están siempre cambiando y evolucionando sobre la base de necesidades de los usuarios inmediatos. En este caso el uso de la metodología de desarrollo de Agile-SCRUM no sólo proporciona la capacidad de adaptación y flexibilidad a las necesidades cambiantes, sino que también ofrece un modelo de trabajo estricto y programado para el equipo. Presenta una mejora del framework SCRUM aplicado al desarrollo de aplicaciones para móviles.
13	Artículo	An Investigation into Mobile Based Approach for Healthcare Activities Occupational Therapy System	(Sardasht Mahmood, 2013)	2013	United Kingdom	1	Mobile-D	I	Mobile-D	<ul style="list-style-type: none"> En este trabajo se presenta de manera detallada las principales características que se deben tomar en cuenta para desarrollar aplicaciones para dispositivos móviles; en relación, al HW, SW, el tipo de aplicación y las principales características de este tipo de aplicaciones. Selecciona Mobile-D como la metodología ágil de desarrollo ya que presenta mayor visibilidad y un constante progreso en el desarrollo y baja densidad de defectos en el producto final.

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										<ul style="list-style-type: none"> Se ejecutan las fases de Mobile-D para el desarrollo del proyecto. Se entrega la aplicación aprobada y validada por los usuarios que hacen uso de la misma.
14	Artículo	USING MOBILE-D METHODOLOGY IN DEVELOPMENT OF MOBILE APPLICATIONS: CHALLENGES AND ISSUES	(Supan, Tekovic, Skalec, & Zlatko, 2013)	2013	Croacia	1	Mobile-D	A	Mobile-D	<ul style="list-style-type: none"> Presenta un estudio detallado de la metodología Mobile-D en relación a cada una de sus fases. Detalla las principales actividades, entradas y salidas (artefactos) que se utilizan en cada una de las fases. Mobile-D se presenta como una disciplina viable en desarrollo de aplicaciones móviles que trae sus propias ventajas y desventajas. Los equipos que están familiarizados con la metodología y están preparados para todo tipo de inconvenientes pueden encontrar muchos beneficios en su uso.
15	Artículo	Apps móviles como herramientas de apoyo al aprendizaje matemático informal en Educación Superior Resumen Introducción Contexto Universitario	(Andres, Genoff, & Paola, 2012)	2012	Argentina	1	Mobile-D	I	Mobile-D	<ul style="list-style-type: none"> Presenta al desarrollo ágil y las metodologías ágiles como un nuevo enfoque en el desarrollo de software debido a la simplicidad de sus reglas y prácticas. Define a Mobile-D como una aproximación ideal para el desarrollo ágil de apps, que se enfoca en las

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										<p>particularidades del desarrollo de software para dispositivos móviles.</p> <ul style="list-style-type: none"> • Se ejecutan cada una de las fases de Mobile-D al desarrollo de la aplicación. • La aplicación es presentada y utilizada por los docentes y estudiantes como una herramienta de apoyo de total aceptación.
16	Artículo	Tecnología y desarrollo en dispositivos móviles	(Prieto, 2011)	2011	España	4	Waterfall RAD Desarrollo ágil Mobile-D	A	Cualquiera de las metodologías analizadas	<ul style="list-style-type: none"> • Es parte del material docente de la UOC (Universitat Oberta de Catalunya) orientado a iniciar a los ingenieros informáticos en el desarrollo de aplicaciones sobre dispositivos móviles. • Incluye un módulo específico de los métodos para el desarrollo de aplicaciones móviles y detalla al modelo waterfall, el desarrollo rápido de aplicaciones, desarrollo ágil y Mobile-D y las fases de los proyectos de desarrollo de aplicaciones móviles.
17	Artículo	Agile development methods for mobile applications	(Spataru, 2010)	2010	Reino Unido	1	Mobile-D	A	Mobile-D	<ul style="list-style-type: none"> • Selecciona Mobile-D como la metodología ágil ideal en el desarrollo de aplicaciones móviles e incluye algunas sugerencias que se deben tomar en cuenta como, la conciencia de mejora en el mercado, soporte de la línea de productos de software y soporte de reutilización. • Incluye la línea de arquitectura en la que se utiliza para capturar el conocimiento de una organización y se puede adaptar al proyecto que se esté desarrollando. • Considera a Mobile-D como una propuesta de metodología madura para aplicarse a este entorno de desarrollo,

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										ya que se presentan casos de estudios reales exitosos.
18	Artículo	Metodología de desarrollo ágil para sistemas móviles. Introducción al desarrollo con Android y el iPhone	(Blanco et al., 2009)	2009	España	2	Mobile-D HMD	A	Mobile-D	<ul style="list-style-type: none"> • Presenta un análisis sobre el desarrollo ágil las metodologías ágiles en el ciclo de vida de desarrollo de un proyecto. • Presenta a las metodologías ágiles más aproximadas y adaptadas para el desarrollo de aplicaciones móviles. • Mobile-D proporciona mayor satisfacción y seguridad al momento de seleccionar una metodología ágil ya que esta presenta algunos casos de uso en el mundo real a través de numerosos proyectos en el entorno móvil.
19	Artículo	Development process of mobile application SW based on agile methodology	(Jeong, Lee, & Shin, 2008)	2008		1	MASAM	D	MASAM	<ul style="list-style-type: none"> • El objeto de la metodología MASAM es proporcionar el proceso para el desarrollo de SW que opera en plataformas móviles. • MASAM se basa en el enfoque ágil. • MASAM establece 4 fases: <ul style="list-style-type: none"> ○ Fase de preparación ○ Fase de realización ○ Fase de desarrollo ○ Fase de comercialización • Aunque no se presentan casos de estudio reales, MASAM es una nueva propuesta de metodología para aplicarse en el entorno de desarrollo de aplicaciones móviles
20	Artículo	Designing an agile methodology for mobile software development: a	(Rahimian & Ramsin, 2008)	2008		3	HMD Mobile-D ASD	D	HMD	<ul style="list-style-type: none"> • HMD es una propuesta de metodología para el desarrollo de aplicaciones

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
		hybrid method engineering approach					NPD			<p>móviles.</p> <ul style="list-style-type: none"> • Es una metodología híbrida que se apoya en la combinación de ASD, NPD, RUP e incluye una fase de comercialización. • Su estructura está conformada en base a 4 iteraciones. • A pesar de ser nombrada entre las metodologías de desarrollo para aplicaciones móviles, no se presenta casos de estudio.
21	Tesis Pregrado	Aplicación móvil para el control de notas de los estudiantes de la Universidad Técnica de Ambato utilizando la plataforma Android	(Gamboa Safia, 2015)	2015	Ecuador	4	XP Scrum Mobile-D Crystal Clear	I	XP	<ul style="list-style-type: none"> • Realiza una comparación de las metodologías ágiles seleccionadas en base a características, ventajas, desventajas y fases. • Selecciona y detalla XP como la metodología a utilizar en este proyecto. • La aplicación móvil se desarrolla a través de cada una de las fases de XP; la exploración, planificación del proyecto, iteraciones, producción y prueba. • Utiliza principalmente las historias de usuario, prototipos y diagramas como sus principales artefactos. • La aplicación se desarrolló y se presentó con éxito a través de cada una de las fases de XP.
22	Tesis Pregrado	Desarrollo de una aplicación para dispositivos Android y diseño de servicio web, para el manejo de las operaciones por parte de clientes y personal de empresas aseguradoras	(Sánchez Cacique, 2014)	2014	Venezuela	1	Scrum	I	Scrum	<ul style="list-style-type: none"> • El desarrollo del proyecto se apoya en SCRUM, utilizando el conjunto de prácticas y roles que se define en este marco de trabajo. • Los principales roles que se toman en cuenta son: ScrumMaster, ProductOwner y el Team. • Los elementos del Product Backlog que forman parte del sprint (de 30 días) se

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										<p>determinan durante la reunión del Sprint Planing.</p> <ul style="list-style-type: none"> • Complementa métricas para el control de costos. • Se apoya en elementos de diseño basado en UML.
23	Tesis Pregrado	Diseño de la investigación del proyecto dante sistema móvil para la notificación, ubicación y alerta de incidentes y áreas de peligro en la ciudad de Guatemala	(Bolaños Lima, 2013)	2013	Guatemala	1	Mobile-D	I	Mobile-D	<ul style="list-style-type: none"> • Selecciona Mobile-D como le metodología ágil para el desarrollo de aplicaciones móviles. • Mobile-D aprovecha las buenas prácticas y recomendaciones ampliamente reconocidas de otras metodologías que forman parte de su estructura. • Detalla cada una de las fases de la metodología con base al desarrollo del proyecto. • Se presenta la aplicación desarrollada que es aprobada por los usuarios a la que está dirigida.
24	Tesis Pregrado	Desarrollo de aplicación móvil de apoyo a la plataforma web del observatorio "Monitoreo de variables físicas y fisiológicas en niños y adolescentes en edad escolar de Risaralda"	(Grimaldo Botero, 2013)	2013	Colombia	1	Mobile-D	I	Mobile-D	<ul style="list-style-type: none"> • El uso de Mobile-D en este proyecto permite agilizar y organizar el trabajo de manera que para futuras modificaciones se tiene una excelente bases de documentación. • Utiliza prototipos de interfaz de usuario que facilita la recolección de los requerimientos no funcionales de la aplicación móvil. • Utiliza diagramas de casos de uso que permiten realizar una codificación de forma organizada y simple. • Utiliza story cards para realizar la valoración de tiempos y complejidad de las tareas a desarrollar durante la

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										<p>codificación.</p> <ul style="list-style-type: none"> Ejecuta pruebas de aceptación y usabilidad y entrega la aplicación para que se puedan utilizar.
25	Tesis Pregrado	Aplicación de tecnologías móviles para la búsqueda de recursos educativos abiertos	(Alvarado Ruiz et al., 2012)	2012	Ecuador	1	Iconix	I	Iconix	<ul style="list-style-type: none"> Iconix permite identificar de manera rápida un sistema y de forma versátil en función de casos de uso y componentes permitiendo la retroalimentación iterativa. Establece 4 fases de desarrollo con tareas específicas que generan artefactos, como se detallan a continuación : <ul style="list-style-type: none"> Análisis de requisitos (Modelo de dominio, Prototipación rápida, Modelos de casos de uso) Análisis y diseño preliminar (Especificación de casos de uso) Diseño (Diagramas de secuencia) Implementación (Código generado y Pruebas)
26	Tesis Pregrado	Implementación de una Aplicación Android basada en Realidad Aumentada aplicada a Puntos de Interés de la UTPL	(Bravo, 2012)	2012	Ecuador	1	Iconix	I	Iconix	<ul style="list-style-type: none"> La metodología utilizada en todo el proceso de desarrollo del sistema es ICONIX, misma que maneja un conjunto de artefactos, actividades y procesos requeridos en el desarrollo de aplicaciones que manejan tiempos reducidos y pocos recursos. Sus características principales son: Iterativo-incremental, permite trazabilidad, y maneja una UML dinámica.

	Tipo de trabajo	Nombre del estudio	Autores/ Organización	Año	País	# de metodologías ágiles	Metodologías ágiles analizadas e incluidas	Uso Análisis(A) Implementación(I) Diseño(D)	Metodología Seleccionada	Argumentación
										<ul style="list-style-type: none"> Para llevar a cabo las fases de desarrollo se han empleado distintos artefactos, entre ellos especificación de requerimientos, diagrama y especificación de casos de uso, arquitectura de la aplicación, diagramas de clases y otros.

Fuente: el autor

Nota: Estos anexos establecidos como las entradas y salidas del MTCC se han utilizado directamente en el proceso de validación en el caso práctico seleccionado “Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL” y su producto la aplicación móvil “Gestión de tutorías UTPL” en la cual se especifica que se deben llenar ciertos campos y parámetros de acuerdo a las necesidades y requerimientos del desarrollo del proyecto. La versión de los artefactos se actualizará según la iteración y la fase del proyecto.

Anexo 2. Plan del proyecto

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”

Plan del proyecto

Versión 1.0

- *[Nota: La siguiente plantilla se ha desarrollado para su uso con Rational Unified Process. El texto que se encuentra entre corchetes y presentado en estilo itálico azul se ha incluido para proporcionar una guía para el autor y se debería borrar antes de la entrega del documento.]*
- *Hay que sustituir el texto resaltado con marcador amarillo por su equivalente en el proyecto de desarrollo y eliminar el resaltado.*
- *La versión del documento se actualizará según la iteración y la fase del proyecto]*

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
14/10/2016	1.0	Versión preliminar como propuesta de desarrollo.	Equipo de dirección - Diego Moreno - Israel Rey Grupo de interesados - Ing. Fernanda Soto - DGA UTPL Gerente de calidad - Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

Tabla de Contenidos

1. Introducción.....	4
1.1 <i>Propósito</i>	4
1.2 <i>Alcance</i>	4
2. Vista General del Proyecto	4
2.1 <i>Propósito, Alcance y Objetivos</i>	4
2.2 <i>Suposiciones y Restricciones</i>	5
2.3 <i>Entregables del proyecto</i>	5
2.4 <i>Evolución del Plan de Desarrollo del Software</i>	5
3. Organización del Proyecto	5
3.1 <i>Participantes en el Proyecto</i>	6
3.2 <i>Roles y Responsabilidades</i>	6
4. Gestión del Proceso.....	6
4.1 <i>Estimaciones del Proyecto</i>	6
4.2 <i>Plan del Proyecto</i>	6
4.2.1 <i>Plan de las Fases – (Propuesta del proyecto)</i>	7
4.2.2 <i>Calendario del Proyecto</i>	8
5. Diseño de arquitectura de software.....	8
5.1 <i>Objetivos y limitaciones de la arquitectura</i>	8
5.2 <i>Interacción de los componentes</i>	9
5.3 <i>Vista de alto nivel</i>	9
5.3.1 <i>Vista lógica</i>	9
5.3.2 <i>Vista de despliegue</i>	9
5.3.3 <i>Vista de implementación</i>	10
6. Seguimiento y Control del Proyecto.....	10
7. Referencias.....	11

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

Plan del Proyecto

1. Introducción

Este plan del proyecto es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al proyecto. Este documento provee una visión global del enfoque de desarrollo propuesto, constituye una configuración de la metodología Mobile-D de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos (entregables) que serán generados. Este documento es a su vez uno de los artefactos de RUP.

1.1 Propósito

El propósito de este proyecto es implementar un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL

1.2 Alcance

El desarrollo del proyecto incluye la elaboración de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL.

El proyecto incluye:

- Elaborar propuesta de un modelo de tutorías apto para la modalidad presencial de la UTPL.
- Analizar, diseñar, desarrollar y/o adaptar aplicaciones software (web y móvil) que soporten el Framework propuesto.
- Implementar el Framework en un caso de estudio Modalidad Presencial UTPL: algunos componentes de la Titulación de Sistemas Informáticos y Computación, componentes con altos índices de reprobación como Matemáticas, Cálculo y Física, componente del Área Biológica, componente del Área Administrativa y como componente de Formación Básica, Expresión Oral y Escrita.
- Divulgar los resultados del proyecto

El proyecto no incluye:

- La implementación de nuevos requerimientos ya que se ha considerado en el desarrollo de la aplicación móvil como producto final del proyecto como primera versión a obtener.

2. Vista General del Proyecto

2.1 Propósito, Alcance y Objetivos

La información que a continuación se incluye ha sido extraída de las diferentes reuniones que se han celebrado con el interesado de la empresa desde el inicio del proyecto.

- Obtener un marco teórico sobre Innovación Educativa en instituciones de Educación Superior, enfocado en la acción tutorial docente.
- Elaborar propuesta de un modelo de tutorías apto para la modalidad presencial de la UTPL.
- Analizar, diseñar, desarrollar y/o adaptar una aplicación software (web y móvil) que soporte el Framework propuesto.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

- Implementar el Framework en un caso de estudio Modalidad Presencial UTPL: algunos componentes de la Titulación de Sistemas Informáticos y Computación, componentes con altos índices de reprobación como Matemáticas, Cálculo y Física, componente del Área Biológica, componente del Área Administrativa y como componente de Formación Básica, Expresión Oral y Escrita.
- Divulgar los resultados del proyecto y caso de estudio.

2.2 Suposiciones y Restricciones

Se consideran como suposiciones y restricciones temporales:

- Se ha seleccionado un grupo específicos de usuarios finales para realizar la ejecución y validación del framework propuesto a través de la aplicación móvil, en su primera versión.
- En el proceso instalación de la aplicación móvil en su primera versión a los dispositivos móviles se realiza de manera manual; es decir a través de un enlace que permite seguir el proceso de instalación detallado (infografía) para cada plataforma.

2.3 Entregables del proyecto

Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para este proyecto. Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

Que entregable	Propósito
Plan del proyecto	Documenta los requisitos principales del proyecto.
Documento de especificación de requerimientos	Describe detalladamente las necesidades del proyecto.
Diagrama de casos de uso	Identifica las actividades y actores que intervienen para llevar a cabo un proceso
Diagrama de clases	Presenta la estructura del sistema a través de las clases, atributos, operaciones (o métodos), y las relaciones entre los objetos.
Historias de usuarios	Presentan de manera específica los requisitos definidos por los interesados

2.4 Evolución del Plan de Desarrollo del Software

- El Plan de Desarrollo del Software se revisará semanalmente y se refinará antes del comienzo de cada iteración.
- El proceso de control de cambios y el repositorio de los documentos esta creado y listo para que accedan los involucrados.

3. Organización del Proyecto

Los recursos humanos involucrados en el desarrollo, indicando para cada uno, su función o rol, así como sus responsabilidades.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

3.1 Participantes en el Proyecto

Equipo de dirección.

- Ing. Fernanda Soto. Líder del Proyecto.

Grupo de interesados

- Vicerrectorado de UTPL.

Equipo del proyecto.

- Diego Moreno. Desarrollador
- Israel Rey. Desarrollador

Equipo de control de calidad.

Gabriela Puglla. Control de calidad

3.2 Roles y Responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Exploración e Inicialización, de acuerdo con los roles que desempeñan en Mobile-D.

Rol	Responsabilidad	Asumido por:
Equipo de dirección	El equipo de dirección asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El equipo de dirección también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el equipo de dirección se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.	Ing. Fernanda Soto
Equipo de exploración	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.	Vicerrectorado UTPL
Equipo de proyecto	Construcción de prototipos. Desarrollo de software. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario. Elaborar modelos de implementación y despliegue.	Diego Moreno. Israel Rey
Equipo de calidad	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación.	Gabriela Puglla

4. Gestión del Proceso

4.1 Estimaciones del Proyecto

El presupuesto del proyecto y los recursos involucrados se adjuntan.

(En este caso el proyecto se considera como Trabajo de Fin de Titulación y no se ha considerado un presupuesto para el desarrollo especificando que el producto se entrega como una primera versión para su posterior mejora).

4.2 Plan del Proyecto

Presenta la organización en fases e iteraciones y el calendario del proyecto.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

4.2.1 Plan de las Fases – (Propuesta del proyecto)

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase.

Fase	Nro. Iteraciones	Tiempo establecido
Inicialización	2	5 semanas
Exploración	2	4 semanas
Producción	7	11 semanas
Estabilización	7	11 semanas
Pruebas del sistema	2	5 semanas

Los hitos que marcan el final de cada fase se describen en la siguiente tabla.

Descripción	Hito
Fase de inicialización	En esta fase desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Plan del proyecto. Los principales casos de uso serán identificados y se hará un refinamiento del Plan del Proyecto. La aceptación del cliente /usuario del artefacto el final de esta fase.
Fase de Exploración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera release de la fase de Producción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. La primera iteración tendrá como objetivo la identificación y especificación de los principales casos de uso, así como su realización preliminar en el Modelo de Análisis / Diseño, también permitirá hacer una revisión general del estado de los artefactos hasta este punto y ajustar si es necesario la planificación para asegurar el cumplimiento de los objetivos.
Fase de Producción	Durante la fase de producción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 3 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión de la release 2.0, con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.
Fase de Estabilización	En esta fase se prepararán los releases para distribución, asegurando una implantación y cambio del sistema previo de manera adecuada, incluyendo el

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

	entrenamiento de los usuarios.
Fase de Pruebas y correcciones del sistema	El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y la presentación del producto final.

4.2.2 Calendario del Proyecto

Para este proyecto se ha establecido el siguiente calendario. La fecha de aprobación indica cuándo el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no quita la posibilidad de su posterior refinamiento y cambios.

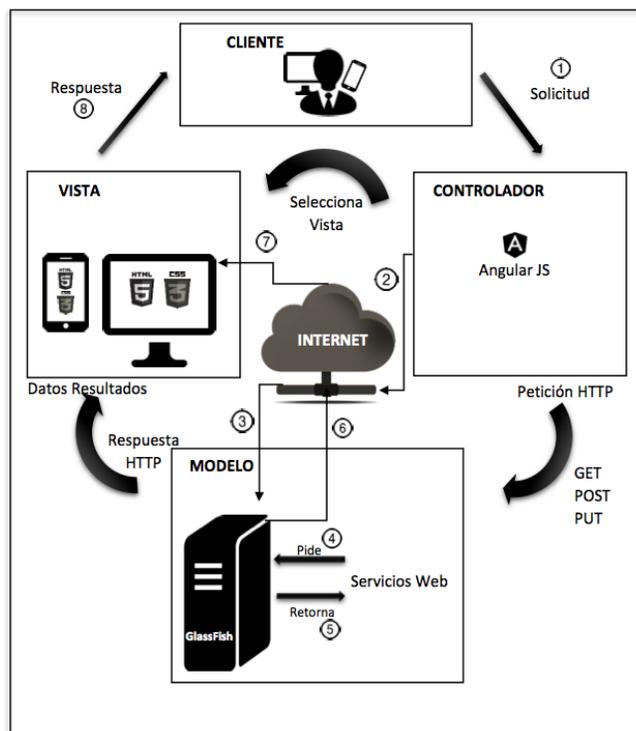
Disciplinas / Artefactos generados	Comienzo	Aprobación
Modelado del Negocio		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	Exploración	Aprobado
Requisitos		
Modelo de Casos de Uso	Exploración	Aprobado
Especificación de Casos de Uso	Exploración	Aprobado
Especificaciones Adicionales	Exploración	Aprobado
Análisis/Diseño		
Modelo de Análisis/Diseño	Exploración	Aprobado
Modelo de Datos	Exploración	Aprobado
Implementación		
Prototipos de Interfaces de Usuario	Exploración	Aprobado
Modelo de Implementación	Exploración	Aprobado
Pruebas		
Casos de Pruebas Funcionales	Durante las fases de producción y estabilización	
Gestión de Cambios y Configuración	Durante todo el proyecto	
Gestión del proyecto		
Plan del Proyecto en su versión 1.0 y planes de las Iteraciones	Inicialización y Exploración	
Ambiente	Durante todo el proyecto	

5. Diseño de arquitectura de software

5.1 Objetivos y limitaciones de la arquitectura

Para llevar a cabo el desarrollo del proyecto se utiliza una arquitectura orientada a servicios (SOA), esta arquitectura permite construir la funcionalidad de la aplicación, combinando la información de múltiples servicios, mediante la utilización del protocolo HTTP, ya que los servicios son distribuibles. Además se utilizará el patrón de arquitectura MVC (Modelo-Vista-Controlador), que permitirá separar la lógica de negocio de la interfaz de usuario.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	



5.2 Interacción de los componentes

El cliente o usuario interactúa con la interfaz de la aplicación, enviando una petición al controlador en base a los parámetros y opciones de consulta de la aplicación.

- Controlador: Recibe la petición por parte del usuario y gestiona el evento, realiza peticiones (Get, Post o Put) al Modelo, mediante el protocolo HTTP. El controlador selecciona una vista.
- Modelo: Recibe la petición, se conecta al servidor web GlassFish, gestiona el acceso a la información para recibir los datos y envía resultados a la vista.
- Vista: Recibe los datos del Modelo para generar la interfaz y presenta los resultados al usuario, mediante el navegador o la interfaz de la aplicación.

5.3 Vista de alto nivel

El cliente es una aplicación ionic (aplicación híbrida) que se ejecuta en web y en teléfono móvil con sistema operativo IOS o Android.

5.3.1 Vista lógica

La lógica del negocio se implementó a través de Angular JS y un servicio web en un servidor de aplicaciones GlassFish Server Open Source Edition 3.1.2.2 que corre en el servidor de la Universidad Técnica Particular de Loja.

5.3.2 Vista de despliegue

Las peticiones que se invocan al Modelo, cuando se ejecuta una solicitud, se realizan dentro de los controladores de la aplicación, donde se utiliza los métodos del protocolo HTTP: GET, POST y PUT.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

5.3.3 Vista de implementación

La información se presenta mediante una interfaz de usuario, creado con HTML5, CSS (generado por SASS).

6. Seguimiento y Control del Proyecto

Gestión de Requisitos

Se listan los requisitos a los que se irá haciendo un seguimiento a lo largo de todo el desarrollo del proyecto.

- **Ingresar al QR:** permite ingresar al sistema de gestión de tutorías especificando el rol del usuario (Docente, Estudiante, DGA) respectivamente para gestionar, acceder y generar reportes de las tutorías.
- **Ver componentes:** presenta los componentes educativos para crear, ver y generar reportes de las tutorías.
- **Crear tutoría:** permite al usuario con rol específico (Docente) crear una tutoría.
- **Editar tutoría:** permite al usuario con rol específico (Docente) modificar los datos de la tutoría creada.
- **Ver notificaciones:** permite al usuario con rol (Docente / Estudiante) informar a través del correo, sobre los comentarios y participantes de una tutoría.
- **Ver participantes:** permite al usuario con rol específico (Docente) ver a los estudiantes que asistirán a la tutoría.
- **Ver comentarios:** permite al usuario con rol específico (Docente) ver los comentarios de cada estudiante sobre el tema de la tutoría creada.
- **Validar asistencia:** permite al usuario con rol específico (Docente) validar la asistencia del estudiante a la tutoría.
- **Ver tutorías creadas:** permite al usuario con rol (Docente / Estudiante) ver las tutorías creadas.
- **Participar en tutoría:** permite al usuario con rol específico (Estudiante) participar en una tutoría.
- **Comentar tutoría:** permite al usuario con rol específico (Estudiante) opinar/comentar sobre un tema de la tutoría.
- **Calificar tutoría:** permite al usuario con rol específico (Estudiante) calificar una tutoría asistida.

Control de Plazos

Los plazos de entrega de cada una de las fases planificadas en base al apartado 4.2.1.

Control de Calidad

El proceso de control de calidad es un proceso iterativo e incremental que se aplica en cada una de las fases de desarrollo del proyecto.

Gestión de Riesgos

Definidos por el cliente en relación a la variación del cronograma y presupuesto para el desarrollo del proyecto.

Gestión de Configuración

Resumen de los requisitos de configuración del producto generado en el proyecto

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Plan del proyecto	

7. Referencias

- Trabajo de fin de titulación (TFT) “Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”.

Anexo 3. Documento de especificación de requerimientos (ERS)

**“Análisis y desarrollo de un framework metodológico
para la innovación educativa enfocada a la acción
tutorial docente de la Modalidad Presencial de la
UTPL”**

**Especificación de requerimientos de software (ERS)
Versión 1.0**

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Especificación de requerimientos	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
14/10/2016	1.0	Versión preliminar como propuesta de desarrollo.	Equipo de Dirección Ing. Fernanda Soto Grupo de interesados Vicerrectorado de UTPL Equipo de Desarrollo Diego Moreno Israel Rey Equipo de calidad Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Especificación de requerimientos	

Tabla de Contenidos

1.	Introducción.....	4
1.1	Propósito.....	4
1.2	Alcance	4
1.3	Referencias	4
2.	Descripción general.....	4
2.1	Perspectiva del producto.....	4
2.2	Características del producto.....	4
2.3	Funcionalidades del producto	4
2.4	Características de los usuarios	4
2.5	Restricciones de diseño	5
2.6	Supuestos y dependencias	5
3.	Requerimientos funcionales	5
4.	Otros requerimientos	7
5.	Definiciones, siglas y abreviaturas.	7

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Especificación de requerimientos	

Especificación de requerimientos de software (ERS)

1. Introducción

El documento de especificación de requerimientos de software (ERS) presenta una descripción completa del sistema que se va a desarrollar.

1.1 Propósito

Detalla cada uno de los requisitos definidos por el cliente (Grupo de interesados, Vicerrectorado de UTPL) con el propósito de compartir al equipo del proyecto para su desarrollo.

1.2 Alcance

- Analizar, diseñar, desarrollar y/o adaptar una aplicación software web/móvil que sistematice el modelo de tutorías propuesto.

1.3 Referencias

- Especificación de requerimientos según el estándar de IEEE 830. IEEE Std. 830-1998.
- Historias de usuario

2. Descripción general

2.1 Perspectiva del producto

EL modelo de tutorías propuesto se automatiza a través del software web/móvil, cuya función principal es presentar la mejora del proceso de acción tutorial de la UTPL, con la finalidad de sistematizar la secuencia repetitiva de actividades de una tutoría.

2.2 Características del producto

El sistema de gestión de tutorías dispone de las siguientes características:

- Gestión de usuarios: permite identificar al usuario que va hacer uso de la aplicación con sus respectivos parámetros de información para posteriormente ejecutar la gestión de tutorías.
- Gestión de tutorías: permite al usuario interactuar con la/s tutoría/s.

2.3 Funcionalidades del producto

- **Ingresar al QR:** permite ingresar al sistema de gestión de tutorías especificando el rol del usuario (Docente, Estudiante, Vicerrectorado de UTPL) respectivamente para gestionar, acceder y generar reportes de las tutorías.
- **Ver componentes:** presenta los componentes educativos para crear, ver y generar reportes de las tutorías.
- **Crear tutoría:** permite al usuario con rol específico (Docente) crear una tutoría.
- **Editar tutoría:** permite al usuario con rol específico (Docente) modificar los datos de la tutoría creada.
- **Ver notificaciones:** permite al usuario con rol (Docente / Estudiante) informar a través del correo, sobre los comentarios y participantes de una tutoría.
- **Ver participantes:** permite al usuario con rol específico (Docente) ver a los estudiantes que asistirán a la tutoría.
- **Ver comentarios:** permite al usuario con rol específico (Docente) ver los comentarios de cada estudiante sobre el tema de la tutoría creada.
- **Validar asistencia:** permite al usuario con rol específico (Docente) validar la asistencia del estudiante a la tutoría.
- **Ver tutorías creadas:** permite al usuario con rol (Docente / Estudiante) ver las tutorías creadas.
- **Participar en tutoría:** permite al usuario con rol específico (Estudiante) participar en una tutoría.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Especificación de requerimientos	

- **Comentar tutoría:** permite al usuario con rol específico (Estudiante) opinar/comentar sobre un tema de la tutoría.
- **Calificar tutoría:** permite al usuario con rol específico (Estudiante) calificar una tutoría asistida.

2.4 Características de los usuarios

- **Usuario registrado:** Educación superior, conocimiento y experiencia en los procesos, favorecido con todas las funcionalidades implementadas dentro del sistema.
- Esta aplicación es de uso exclusivo para personal de la UTPL, los usuarios que hacen uso de la misma adquieren el rol de Estudiante, Docente y Personal (Vicerrectorado) de la UTPL.

2.5 Restricciones de diseño

- El diseño de la aplicación, correcta implementación y su posterior operación dependerá de la correcta aplicación de lineamientos macro institucionales, medidas de seguridad y políticas de la universidad analizadas en las etapas previas a su implementación así como de las limitaciones del hardware empleado.
- La información que proporciona la aplicación es de uso exclusivo para usuarios con roles específicos de la UTPL.

2.6 Supuestos y dependencias

- La información de los usuarios y sus preferencias serán ingresadas y actualizadas constantemente según le corresponda.

3. Requerimientos funcionales

REQ001 Ingresar al QR

Descripción

Su función es permitir a los usuarios ingresar al sistema de gestión de tutorías, a través de su nombre de usuario y contraseña.

Los usuarios pueden ingresar al sistema de gestión de tutorías con el rol de Docente, Estudiante, Vicerrectorado de UTPL.

Entradas

El usuario ingresa los datos a través del teclado.

REQ002 Ver componentes

Descripción

Su función es permitir ver los componentes educativos en los que se van gestionar las tutorías.

Entradas

El usuario puede visualizar los componentes educativos en los que está registrado.

REQ003 Crear tutoría

Descripción

Su función es permitir al usuario con rol de docente acceder al componente educativo para crear tutorías.

Entradas

El usuario con rol de Docente puede acceder a la opción de crear tutorías.

REQ004 Ver tutorías

Descripción

Su función es permitir a los usuarios con el rol de Docente, Estudiante, Vicerrectorado de UTPL ver las tutorías creadas.

Entradas

El usuario puede visualizar las tutorías creadas en cada componente educativo.

REQ005 Editar tutoría

Descripción

Su función es permitir al usuario con rol de docente modificar/editar tutorías que se han creado

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Especificación de requerimientos	

en cada componente educativo.

Entradas

El usuario con rol de Docente puede acceder a la opción de modificar tutorías.

REQ006 Participar en tutoría

Descripción

Su función es permitir al usuario con rol de Estudiante participar en la tutoría creada con el objetivo de solucionar dudas que se presentaron en la clase.

Entradas

El usuario selecciona una fecha, hora de inicio y fin para participar en la tutoría.

REQ007 Comentar tutoría

Descripción

Su función es permitir al usuario con rol de Estudiante comentar en la tutoría.

Entradas

El usuario puede ingresar a la tutoría creada y editar un comentario.

REQ008 Ver notificaciones

Descripción

Su función es permitir a los usuarios con el rol de Docente, Estudiante ver las notificaciones de las tutorías creadas a través de un correo electrónico de confirmación.

Entradas

El usuario puede visualizar las notificaciones de las tutorías

REQ009 Ver participantes

Descripción

Su función es permitir al usuario con rol de Docente ver a los estudiantes que asistirán a la tutoría para controlar su asistencia y calificar su participación.

Entradas

El usuario puede ver la lista de estudiantes la fecha, hora de inicio y fin en la que van a participar en la tutoría.

REQ010 Ver comentarios

Descripción

Su función es permitir al usuario con rol de Docente ver los comentarios de los estudiantes para conocer las dudas sobre el tema creado en la tutoría.

Entradas

Se necesita que los estudiantes hayan comentado sobre un tema en la tutoría creada

REQ011 Validar asistencia

Descripción

Su función es permitir al usuario con rol de Docente verificar la participación del estudiante en la tutoría para validar su respectiva asistencia.

Entradas

Se necesita que los estudiantes hayan asistido al horario de tutoría asignado.

REQ012 Calificar tutoría

Descripción

Su función es permitir al usuario con rol de Estudiante calificar la tutoría a la que asistió con una valoración de Excelente, Buena o Mala.

Entradas

Se necesita que los estudiantes hayan asistido a la tutoría para poder calificar.

REQ013 Ver estado de tutoría

Descripción

Su función es permitir al usuario con rol de Estudiante ver el estado de asistencia en la tutoría.

Entradas

El estudiante necesita haber creado un horario de participación en la tutoría.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 14/10/2016
Especificación de requerimientos	

4. Otros requerimientos

- El sistema debe responder de inmediato (dentro de un tiempo estimado en segundos) a las interacciones del usuario.
- Disponible en cualquier momento.
- Ser amigable al usuario y sencillo de usar.

5. Definiciones, siglas y abreviaturas.

[Esta sección debe proporcionar las definiciones de todos los términos, las siglas, y abreviaciones requeridas para interpretar apropiadamente el documento **Especificación de Requerimientos de Software**. Esta información puede proporcionarse por la referencia al Glosario del proyecto.]

•

Anexo 4. Checklist del proyecto

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”

Checklist del proyecto

Versión 1.0

Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL	Versión: 1.0
Checklist del proyecto	Fecha 14/10/2016
Plan del proyecto	

FASES DEL MOBILE-D			
FASE DE EXPLORACIÓN	SI	NO	OBSERVACIONES
La definición del alcance y los objetivos del proyecto es precisa y concisa	x		
La matriz de roles y responsabilidades esta establecida y distribuida a los responsables	x		
El cronograma del proyecto se ha definido y acordado	x		
El presupuesto inicial esta definido y ligado claramente a cada una de las actividades establecidas	x		
Las necesidades de formación de equipo del proyecto se han incluido en el plan del proyecto	x		
Se ha incluido un periodo de tiempo específico de entrenamiento en el plan del proyecto		x	Durante el desarrollo de la aplicación se adquirió los conocimientos y habilidades necesarias
El diseño y arquitectura de software se ha incluido en el plan del proyecto	x		
Los controles previstos del plan del proyecto se han realizado	x		
FASE DE INICIALIZACIÓN	SI	NO	OBSERVACIONES
El plan del proyecto aprobado se ha actualizado a las disposiciones para la iteración actual		x	No porque se han realizado nuevos cambios en los requisitos.
El documento de especificación de requerimientos se ha aprobado y distribuido a los integrantes del equipo del proyecto	x		
El plan del proyecto aprobado se ha distribuido a todos los involucrados del proyecto	x		
El repositorio de documentos esta creado y listo para que accedan todos los involucrados	x		Se utiliza Git-Taw como repositorio de documentación
El proceso de control de cambios esta definido y comunicado a todos los involucrados	x		
El personal cuenta con todos los recursos (equipos, materiales, humanos) para realizar su trabajo	x		
FASE DE PRODUCCIÓN	SI	NO	OBSERVACIONES
El plan del proyecto y el documento de especificación de requerimientos se han actualizado para desarrollar las iteraciones en la sección de requisitos, cronograma y recursos.	x		
Se han evaluado de manera correcta y detallada los estándares definidos para el desarrollo de código.	x		
Se ha verificado si se han aprobado, validado los diagramas utilizados como: de clases, de casos de uso, de secuencia, de base de datos, etc.	x		
Se ha verificado si se han desarrollado todas las Story Card	x		
Se ha verificado si se han desarrollado todas las Story Task	x		
FASE DE ESTABILIZACIÓN	SI	NO	OBSERVACIONES
Los requisitos funcionales han sido incluidos en sus totalidad	x		
Los requisitos no funcionales han sido incluidos en su totalidad	x		
FASE DE PRUEBAS DEL SISTEMA	SI	NO	OBSERVACIONES
Se ha presentado la versión de auditoria del sistema, el registro de pruebas del sistema, el plan de pruebas del sistema y el informe de prueba del sistema firmado y aprobado en su última versión.		x	No porque aún está en la fase de desarrollo.
Se ha desarrollado el manual de usuario de la aplicación final de manera comprensiva y detallada		x	No porque aún está en la fase de desarrollo.

Anexo 5. Plan de calidad

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”

Plan de Calidad

Versión 1.0

[Nota: Hay que tener en cuenta que esta plantilla está orientada a organizaciones que cuenten con procesos maduros. Para organizaciones que aun no cuenten con dicha madurez en sus procesos esta plantilla puede resultar complejo llevarla a la práctica.]

[La versión del documento se actualizará según la iteración y la fase del proyecto]

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
15/10/2016	1.0	Versión preliminar como propuesta de desarrollo.	Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

Tabla de Contenidos

1. General	4
1.1 <i>Introducción</i>	4
1.2 <i>Alcance del plan de calidad</i>	4
1.3 <i>Elementos de entrada del plan de calidad</i>	4
1.4 <i>Objetivos de la calidad</i>	4
2. Organización y responsabilidades de la garantía de calidad	4
2.1 <i>Organización</i>	4
2.2 <i>Responsabilidades</i>	5
2.3 <i>Recursos de garantía de calidad necesarios</i>	5
3. Control de documentos y datos	6
4. Recursos	6
4.1 <i>Provisión de recursos</i>	6
4.2 <i>Materiales</i>	6
4.3 <i>Recursos humanos</i>	6
4.4 <i>Infraestructura y ambiente de trabajo</i>	6
5. Requisitos	7
6. Comunicación con el cliente	7
7. Diseño y desarrollo	8
7.1 <i>Proceso de diseño y desarrollo</i>	8
7.2 <i>Control de cambios del diseño y desarrollo</i>	8
8. Compras	8
9. Producción y prestación del servicio	9
10. Preservación del producto	9
11. Seguimiento y medición	9

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

Plan de Calidad

1. General

1.1 Introducción

El Plan de Calidad:

- Describe los estándares que se aplicarán para garantizar la calidad, tanto en relación con la solución que se va a crear, como con los procesos del programa utilizados para crear dicha solución.
- Resume las actividades para la garantía de calidad necesarias para evaluar el modo en que se van a cumplir tales requisitos y estándares.
- Documenta los requisitos y responsabilidades de los recursos para la garantía de calidad.
- Proporciona criterios para evaluar la calidad de la solución completa, así como procesos para asegurar el cumplimiento de dicho nivel de calidad.
- Ofrece un método para la evaluación del rendimiento del sistema de calidad del proveedor de la solución.
- Garantiza la incorporación de la calidad a los aspectos organizativos y del negocio del programa, así como a las facetas técnicas.

1.2 Alcance del plan de calidad

- Este documento definirá la calidad de los productos y servicios proporcionados y las acciones que se llevarán a cabo para comprobar este nivel de calidad en el proyecto.
- Especificar tanto las metas como lo que queda fuera del alcance.

1.3 Elementos de entrada del plan de calidad

Las entradas para el plan de calidad proporcionan referencias para todos los documentos que se consideren relevantes para el objetivo y el contenido del Plan de Calidad.

Los siguientes documentos se deben utilizar como referencia al producir y mantener el Plan de Calidad:

- Plan del proyecto
- Documento de especificación de requerimientos, ERS.
- Historias de usuarios

1.4 Objetivos de la calidad

El plan de calidad presenta los objetivos de la calidad para el caso específico y como se van a lograr. Los objetivos de la calidad pueden ser establecidos, por ejemplo, en relación con:

- Las características de la calidad para un caso específico
- Cuestiones importantes para la satisfacción del cliente o de las partes interesadas, y
- Oportunidades para la mejora de las prácticas de trabajo

Estos objetivos de la calidad deben ser expresados en términos medibles.

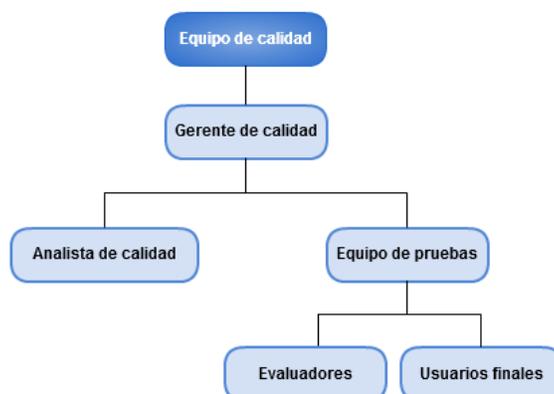
2. Organización y responsabilidades de la garantía de calidad

2.1 Organización

Insertar el organigrama.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

ORGANIGRAMA DEL EQUIPO DE CALIDAD



En programas pequeños, el equipo de dirección es el responsable de la garantía de calidad. Con frecuencia, se puede asignar un asesor de calidad para ayudar al equipo de dirección del proyecto.

Los programas más extensos o complejos pueden tener un director de calidad dedicado y un grupo de garantía de calidad. Tenga en cuenta que se debe incluir también al personal del cliente y de terceros. Las responsabilidades del trabajo afectan al gerente de calidad, analista de pruebas, evaluadores.

2.2 Responsabilidades

Describir las funciones o responsabilidades del personal de garantía de calidad en el proyecto; incluyendo al personal del proyecto que realiza las actividades de garantía de calidad.

- **Gerente de calidad y líder del equipo de calidad** gestiona el proceso del control de calidad, define las estrategias y políticas del proceso de calidad que se van a llevar a cabo en el desarrollo de aplicaciones móviles.
- **Analista de calidad:** se encarga de la planificación y dirección de la calidad a través de la definición y planificación de las pruebas mediante el plan de pruebas, informe de seguimiento y cierre de las pruebas. Además se encarga de analizar y revisar los requisitos del usuario, diseña y ejecuta casos de prueba, prepara el entorno de pruebas, prepara y adquiere todos los recursos necesarios para ejecutar las pruebas y revisa las pruebas desarrolladas por otros.
- **Evaluadores/usuarios finales:** ejecuta las pruebas a todos niveles y registra los resultados, utiliza herramientas de administración o gestión de las pruebas, automatiza las pruebas.

2.3 Recursos de garantía de calidad necesarios

Enumerar la persona o personas asignadas.

Roles	Asignado a:
Gerente de calidad	Gabriela Puglla
Gerente de pruebas	Gabriela Puglla
Analista de pruebas	Gabriela Puglla Diego Moreno Israel Rey
Evaluadores	Gabriela Puglla.
Usuarios finales	Equipo de proyecto, Grupo de interesados, Docentes, Estudiantes de UTPL.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

3. Control de documentos y datos

Para documentos y datos aplicables al caso específico, el plan de calidad debe indicar:

- Como serán identificados los documentos y datos: cada uno de los documentos y datos se identifican con el nombre específico que hacen referencia a su contenido.
- Por quien serán revisados y aprobados los documentos y datos: los roles y responsabilidades asignados hacen referencia al marco de trabajo (Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles bajo el uso de metodologías ágiles. MTCC) como clientes y proveedores.
- A quien se le distribuirán los documentos, o se le notificara su disponibilidad: a los roles asignados en el MTCC como clientes y proveedores en cada uno de los procesos.
- Como se puede obtener acceso a los documentos y datos: a través de repositorio Git-Taw UTPL, almacenado en el archivo “GESTION DE TUTORIAS UTPL”.

4. Recursos

4.1 Provisión de recursos

El plan de calidad debe incluir el tipo y cantidad e recursos necesarios para la ejecución exitosa del plan.

Tipo de recurso	Documento
Artefactos	Plan del proyecto
	Documento de especificación de requerimientos de software (ERS)
	Historias de usuario

4.2 Materiales

Cuando hay características específicas para materiales (en este caso dispositivos móviles, software, etc.), debe declararse o hacer referencia en el plan de calidad a las especificaciones o normas con las cuales los materiales tienen que ser conformes.

- Framework de desarrollo: Ionic Framework
- Simulador/Emulador de pruebas de la aplicación: Android Studio, Xcode
- Smartphone's Android
- Tablet's Android
- Smartphone's iOS
- Tablet's iOS

4.3 Recursos humanos

El plan de calidad permite especificar, donde sea necesario, las competencias particulares requeridas para las funciones ya actividades definidas dentro del caso específico. El plan de calidad define cualquier formación específica u otras acciones requeridas en relación con el personal.

Esto debería incluir:

- La necesidad de nuevo personal y su formación: No se ha solicitado.
- La formación del personal existente en métodos de operación nuevos: no se ha solicitado.

4.4 Infraestructura y ambiente de trabajo

El plan de calidad debe indicar los requisitos particulares del caso específico con respecto a la instalación para el desarrollo de la aplicación, espacio de trabajo, equipo, tecnología de información y comunicación, servicios de apoyo, etc.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

Entorno de desarrollo TIC's	<ul style="list-style-type: none"> • IONIC framework para desarrollo de aplicaciones móviles híbridas multiplataforma. • Córdoba: Es un marco de desarrollo móvil con un conjunto de APIs, que son necesarios para se ejecute la aplicación. • AngularJS: Angular es un framework MVC (Modelo, Vista, Controlador) de JavaScript, que trabaja del lado del cliente, permite utilizar datos en JSON y menos manipulación del DOM, además tiene etiquetas propias como por ejemplo: ng-app, ng-controller, ng-model, ng-view. • HTML5: HTML (Hyper Text Markup Language), es un lenguaje markup, utilizado para dar estructura a una página web, que contiene etiquetas que sirven para presentar el contenido en el navegador. • CSS: CSS (Cascading style sheets), hojas de estilo en cascada es un lenguaje utilizado para dar un aspecto, crear la presentación de un documento estructurado en HTML, mediante el cual se tiene un control sobre el formato y estilo del documento. • Glassfish Server: Glassfish es un servidor de aplicaciones que utiliza una licencia Open Source, además implementa las tecnologías definidas en la plataforma Java EE, este servidor permite recibir peticiones de otros dispositivos y da la respuesta a los mismos
Espacio de trabajo	Laboratorio de Datos e Ingeniería de software de UTPL
Equipo de trabajo	<ul style="list-style-type: none"> • Equipo de proyecto definido de acuerdo a los roles de Mobile-D • Equipo de calidad definido en el MTCC.
Equipos	<ul style="list-style-type: none"> • Ordenadores que permitan facilitar la instalación de los TICS, para el desarrollo del proyecto. • Portátiles personales del equipo del proyecto (Desarrollo) • Ordenador MAC que cumple con los requerimientos para desarrollar y ejecutar la aplicación en versión iOS y Android.
Simuladores	<ul style="list-style-type: none"> • Android Studio: para la aplicación desarrollada para el sistema operativo Android. • Xcode: para la aplicación desarrollada para el sistema operativo iOS.
Dispositivos móviles	<ul style="list-style-type: none"> • Smartphone Android • Tablet Android • Smartphone iOS • Tablet iOS

5. Requisitos

El plan de calidad incluye o hace referencia a los requisitos a ser cumplidos para el caso específico. A través del documento de especificación de requerimientos (ERS) se puede conocer a detalle cada uno de los requisitos establecidos por los clientes y que son parte del desarrollo del proyecto, como uno de los principales artefactos del MTCC.

6. Comunicación con el cliente

El plan de calidad debe indicar lo siguiente:

- Quién es responsable de la comunicación con el cliente en casos particulares: En este caso se asigna al gerente de calidad como responsable de realizar la comunicación con el cliente/interesado.
- Los medios a utilizar para la comunicación con el cliente: a través de correo electrónico y vía telefónica se procede a comunicarse directamente con el cliente/interesado.
- Los registros a conservar de la comunicación con el cliente. Se almacenan y registran los correos

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

electrónicos y comunicación telefónica como constancia de la comunicación establecida. Se establece la frecuencia de las reuniones con el cliente para el control, seguimiento y corregir el avance de lo desarrollado.

7. Diseño y desarrollo

7.1 Proceso de diseño y desarrollo

El plan de la calidad debe incluir o hacer referencia al plan o planes para el diseño y desarrollo. Conforme sea apropiado, el plan de la calidad debe tener en cuenta los códigos aplicables, normas, especificaciones, características de calidad y requisitos reglamentarios. Debe identificar los criterios por los cuales debe aceptarse los elementos de entrada y los resultados del diseño y desarrollo, y cómo, en qué etapa o etapas, y por quién deberían revisarse, verificarse y validarse los resultados.

- El diseño y desarrollo es un proceso complejo y debería buscarse una orientación en fuentes apropiadas, incluyendo los procedimientos de diseño y desarrollo de la organización.
- El proceso de diseño y desarrollo de este proyecto se establece de acuerdo a los requerimientos solicitados por el equipo de DGA de UTPL y la líder y gerente del proyecto, que este caso se rigen específicamente a los definido como proyecto de TFT y primera versión de la aplicación móvil a presentar.
- El diseño del modelo del Framework tiene como propósito mejorar los procesos de la gestión de tutorías de la UTPL, a través del desarrollo de una aplicación web/móvil basado en los estándares de iconos y colores establecidos el manual de imagen de la UTPL.
- El proceso de revisión verificación y validación de los resultados del desarrollo del proyecto se realiza de forma iterativa e incremental en cada una de las fases de desarrollo y será aprobados por el equipo de Mobile-D y de calidad definidos en el MTCC.

7.2 Control de cambios del diseño y desarrollo

El plan de la calidad debería indicar lo siguiente:

- Cómo se controlarán las solicitudes de cambios al diseño y desarrollo: las solicitudes de cambio se realizan previo una reunión formal de trabajo con el equipo de interesados (Cliente).
- Quién está autorizado para iniciar la solicitud de cambio: En este caso, el gerente del proyecto (equipo de dirección) autoriza si se puede iniciar el/los cambios solicitados por el cliente.
- Cómo se revisarán los cambios en términos de su impacto: Se define y asigna un estado (Alto, Medio, Bajo) al impacto que va a producir el cambio en el desarrollo del proyecto.
- Quién está autorizado para aprobar o rechazar cambios: En este caso, el gerente del proyecto (equipo de dirección) autoriza si se puede aprobar el/los cambios solicitados por el cliente.
- Cómo se verificará la implementación de los cambios.: a través de la comunicación directa con el equipo de Mobile-D, y de calidad y la actualización y versionamiento de los artefactos.

En algunos casos puede no haber requisito para el diseño y desarrollo. Sin embargo, aún puede existir una necesidad de gestionar los cambios a los diseños existentes.

8. Compras

El plan de la calidad debería definir lo siguiente:

- Las características críticas de los productos comprados que afecten a la calidad del producto de la organización.: En este caso en el entorno de ejecución de las pruebas directamente con el usuario final el principal problema que se da actualmente es en el proceso de instalación de la aplicación móvil a las diferentes plataformas, ya que no se ha cancelado ni aprobado que las aplicaciones estén subidas a las tiendas online.
- Cómo se van a comunicar esas características a los proveedores, para permitir el control adecuado a lo largo de todo el ciclo de vida del producto o servicio: en este momento se procede

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de calidad	

a indicar el proceso de instalación y adquisición a través de un enlace web que contiene el respectivo manual de instalación y el archivo de la aplicación.

- Los métodos a utilizar para evaluar, seleccionar y controlar a los proveedores: en este caso se considera al entorno de desarrollo **El framework Ionic**. Ionic es una herramienta, gratuita y open source, para el desarrollo de aplicaciones híbridas basadas en HTML5, CSS y JS. Está construido con Sass y optimizado con AngularJS. Ionic es de código abierto publicado bajo una licencia MIT. Esto significa que se puede utilizar en proyectos personales o comerciales, de forma gratuita.
- Donde sea apropiado, los requisitos para los planes de la calidad del proveedor y otros planes, y su referencia: Cada uno de los artefactos de entrada y salida están debidamente requeridos en cada uno de los proceso del MTCC.
- Los métodos a utilizar para satisfacer los requisitos pertinentes de aseguramiento de la calidad, incluyendo los requisitos legales y reglamentarios que apliquen a los productos comprados: a través de la ejecución de cada uno de los proceso del MTCC.
- Cómo pretende verificar la organización la conformidad del producto comprado respecto a los requisitos especificados. Se realiza a través de reportes de usabilidad de los usuarios finales y las pruebas de usabilidad definidos en el MTCC.
- Las instalaciones y servicios requeridos que serán contratados externamente: se necesita únicamente cancelar un valor en las tiendas de Android e iOS, para que la aplicación este disponible y se pueda cancelar automáticamente.

9. Producción y prestación del servicio

El plan de la calidad debería identificar los elementos de entrada, las actividades de realización y los resultados requeridos para llevar a cabo la producción y/o la prestación del servicio.

La producción y prestación del servicio se ejecuta a través del uso del MTCC en el que hace referencia a sus principales componentes que son la definición de los procesos con sus respectivas entradas y salidas, clientes y proveedores que son parte del proceso de control de calidad de las aplicaciones móviles.

10. Preservación del producto

El plan de la calidad debería indicar:

- La solución desarrollada deberá ser guardada dentro del servidor de pruebas en una ruta destinada para ello, que será indicada por el área de Aplicaciones de Integración, los cuales a su vez será los encargados de indicarle al proveedor bajo que especificaciones almacenará el desarrollo realizado.
- La preservación del producto en este caso, todo lo desarrollado y elaborado se almacena en el repositorio GIT-TAW de la UTPL que esta disponible para todos los usuarios.

Y la aplicación móvil como producto final del proyecto tiene como fin ser subido a las tiendas de aplicaciones de las plataformas iOS y Android respectivamente para que pueda ser descargado y utilizado por los usuarios finales.

11. Seguimiento y medición

Los procesos de seguimiento y medición proporcionan los medios por los cuales se obtendrá la evidencia objetiva de la conformidad con los requisitos especificados.

Anexo 6. Plan de medición

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”

Plan de Medición de calidad

Versión 1.0

[La versión del documento se actualizará según la iteración y la fase del proyecto]

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
10/02/2017	1.0	Primera versión del plan de medición “Gestión de tutorías UTPL”	Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Tabla de Contenidos

1. Introducción.....	4
1.1 <i>Propósito</i>	4
1.2 <i>Alcance</i>	4
1.3 <i>Referencias</i>	4
1.4 <i>Vista general</i>	4
2. Metas y submetas	4
3. Métricas.....	5
3.1 <i>Para el Producto</i>	5
3.2 <i>Para el Proceso</i>	6
3.3 <i>Métricas en el desarrollo de las aplicaciones móviles</i>	6
3.4 <i>Métricas de las aplicaciones móviles en el mercado</i>	10

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Plan de medición de calidad

1. Introducción

1.1 Propósito

El propósito de este documento es especificar las métricas que permitirán la evaluación y control de calidad del producto en las diferentes etapas del ciclo de vida del desarrollo del software.

Para monitorear el progreso general del proyecto debe ser especificada una serie de metas y submetas a llevar a cabo durante el desarrollo (Apartado 2).

1.2 Alcance

El presente documento define un conjunto de métricas, incluyendo las metas de medición, las métricas asociadas, y las métricas de calidad interna, externa y de uso a ser evaluadas durante el desarrollo del proyecto, con el fin de monitorear su desempeño global.

La orientación de este documento está basada en los requerimientos del software tal y como fueron especificados en el Documento de especificación de Requerimientos.

1.3 Referencias

Lista de los documentos necesarios que se necesitan de referencia para elaborar de manera satisfactoria este plan.

- Plan del proyecto
- Documento de especificación de requerimientos

1.4 Vista general

Este Plan de Medición del Software contiene la siguiente información:

- **Metas de Medición:** detalles del programa de medición que va a llevarse a cabo en términos de logros, mejoras y calidad.
- **Métricas:** las métricas que serán evaluadas en intervalos regulares del proyecto con el fin de apoyar la medición del logro de las metas.
- **Métricas de calidad interna, externa y de uso:** las métricas que serán utilizadas para el proceso de evaluación del producto software.

2. Metas y submetas

Existen dos aproximaciones esenciales que pueden ser tomados en cuenta para asegurar la calidad final del producto. El primer enfoque es el estudio del Proceso por el que un producto es desarrollado, y el otro, es la evaluación de la calidad del Producto mismo.

- El Proceso: visto como la secuencia de actividades llevadas a cabo para producir el software final, y los artefactos que lo describen.
- El Producto: vista como el conjunto de artefactos del proceso, incluyendo Software desarrollado, Documentos y Modelos.

Las Metas del Plan de Medición del Software, en líneas generales, son definir las herramientas y métricas a utilizar para:

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Evaluar:

- Calidad del producto.
- Calidad de los documentos y modelos.
- El impacto de los cambios en los requerimientos, según el documento de riesgo.
- Las tareas que se derivan de cada requerimiento, según los casos de uso a implementar (sus diagramas y descripciones)
- El nivel de completitud de cada requerimiento, según los diagramas ya hechos y
- La validación del usuario.

Asegurar:

- Un seguimiento documentado del proyecto.
- El progreso de las tareas de acuerdo a los estimados realizados.
- El estado progresivo de las actividades.

Facilitar:

- La planificación y estimación de proyectos similares futuros.
- La importancia del desarrollo de cada actividad dentro de las disciplinas.

3. Métricas

3.1 Para el Producto

Los productos son: el código, los documentos y los modelos. Algunas características y sus métricas asociadas se listan en la siguiente tabla:

Producto	Características	Métricas	Datos	
			Plan del proyecto	Doc. ERS
Documentos	Tamaño	Número de Páginas	11	7
	Esfuerzo	Tiempo invertido en producción, cambios y reparaciones. (Numero de Horas)	16	16
	Volatilidad	Número de páginas cambiadas.	4	2

Producto	Características	Métricas	Datos				
			(n)	C.U. 1	C.U. 2	C.U. 3	C.U.4
Modelos	Tamaño	Número de Casos de Uso.	4				
		Número de Actores.	3				
		Número de Clases.	5				
		Número de Paquetes.	-	-	-	-	-
		Número de Métodos por Clase.		6	-	7	2
		Número de Atributos por Clase.		5	10	6	9
		Historias de usuario	14				
		Tarjetas de tareas	14				

3.1.1 Evaluación de las métricas para el producto

A través de los datos obtenidos que se presentan en las tablas de documentos y los modelos utilizados en el proceso del desarrollo para el producto, se establece que:

- Los documentos definidos en este marco de trabajo son básicos, fáciles de entender y de elaborar.
- Los modelos utilizados en el desarrollo de este proyecto son fáciles de entender y la cantidad de modelos es aceptable y suficiente.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

3.2 Para el Proceso

Para caracterizar el proceso, las mediciones deben estar basadas en comparaciones entre los valores actuales y los estimados realizados. Algunas características y sus métricas asociadas se listan en la siguiente tabla:

Métricas de calidad

La **Mantenibilidad** permite medir la capacidad del producto (aplicación móvil) para ser modificado, en relación a incluir correcciones, mejoras o adaptaciones del software en función de los requerimientos y especificaciones funcionales.

ISO 9126-1 ISO 9126-2 Características	Sub-características	Métricas	Datos
Mantenibilidad	Estabilidad	Número de requerimientos estables	12
		Total de requerimientos	14
	Capacidad de ser analizado	Numero de requerimiento corregidos	3
		Total de requerimientos	14
	Capacidad de ser cambiado	Número de requerimientos nuevos	0
		Total de requerimientos	14

La **eficiencia** permite medir la capacidad del producto (aplicación móvil) para identificar errores bajo condiciones determinadas, en relación a identificar / detectar errores, corregir errores y validar los errores corregidos en función de los requerimientos y especificaciones funcionales.

ISO 9126-1 ISO 9126-2 Características	Sub-características	Métricas	Datos
Eficiencia	Eficiencia en la detección de errores	Numero de errores encontrados	5
		Numero de funcionalidades ejecutadas	13
	Eficiencia en la corrección de errores	Numero de errores corregidos	4
		Numero de funcionalidades ejecutadas	13
	Eficiencia en la verificación	Numero de funcionalidades ejecutadas	13
		Numero total de funcionalidades	13

3.2.1 Evaluación de las métricas de calidad en el proceso

A través de los datos obtenidos que se presentan en las tablas de mantenibilidad y eficiencia utilizados en el proceso del desarrollo, se establece que:

- Los requerimientos funcionales definidos en el documento de especificación de requerimientos (ERS) se han corregido en un 95% y han cumplido con las expectativas del grupo de interesados.

3.3 Métricas en el desarrollo de las aplicaciones móviles

Estas métricas se han seleccionado con relación al análisis de las características de las aplicaciones móviles en el entorno de desarrollo, están asociadas en las diferentes etapas del ciclo de vida del desarrollo del producto software basadas en el modelo de calidad ISO/EC 9126 en los tres aspectos de calidad: interna, calidad externa y calidad de uso.

El modelo de calidad asociado a las aplicaciones móviles se basa en las siguientes características

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

debidamente seleccionadas:

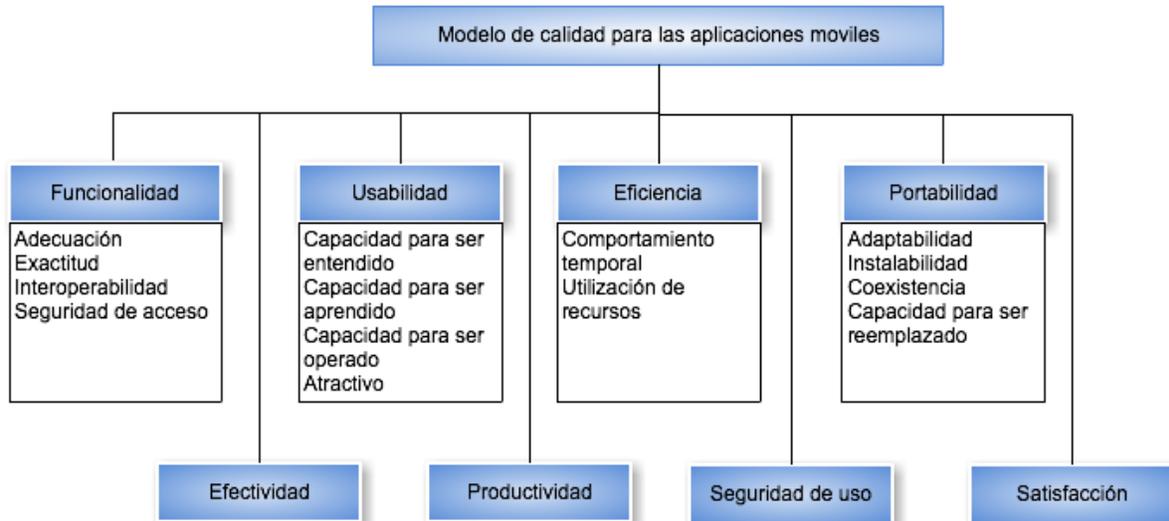


Figura 3.1. Modelo de calidad para las aplicaciones móviles.

Fuente: Modelo ISO/EC 9126

La estrategia de cada una de las pruebas del software mediante el uso del modelo de calidad ISO/EC 9126 permite medir la calidad de los datos obtenidos y así poder lograr obtener un producto de calidad a través de los siguientes campos definidos de la tabla de valoración 3.1.

- **Características de calidad de ISO/EC 9126:** se presentan los 8 atributos seleccionados tanto de calidad interna, externa y de uso.
- **Sub-características de calidad de ISO 9126:** se presentan los 14 atributos seleccionados y distribuidos específicamente en las características de la calidad interna y externa.
- **Atributo definido (estándar):** para que se pueda cumplir y considerar aceptable la validación.
- **Indicador:** magnitud y/o valores que permite identificar el cumplimiento de un atributo/estándar.
- **Datos:** Valores asignados en relación al indicador
- **Escala cualitativa:** a través de la valoración de los datos se evalúa en 5 categorías.

Valoración				
4 (100%)	3 (75%)	2 (50%)	1 (25%)	0 (0%)
Totalmente	En su mayor parte	Parcialmente	En desacuerdo	Ninguno
Excelente	Suficiente	Parcial	Insuficiente	Nada
Muy Satisfecho	Satisfecho	Regularmente satisfecho	Insatisfecho	Nada

- **Ponderación:** de 100 puntos desglosados para cada una de las características y sub-características.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Tabla 3.1. Validación del modelo ISO/EC 9126 en las aplicaciones móviles

Características de calidad de ISO/EC 9126	Sub-características de calidad de ISO/EC 9126	Atributo definido (estándar)	Indicador	Datos	Escala Cualitativa	Valor (0-4)	Ponderación
Funcionalidad	Adecuación	Numero de funcionalidades asignadas	Número (n)	7	En su mayor parte	3	25%
		Número total de funcionalidades	Número (n)	8			
	Exactitud	Numero de funcionalidades exitosas ejecutadas	Número (n)	8	En su mayor parte	3	
		Número total de funcionalidades	Número (n)	8			
	Interoperabilidad	Capacidad de interactuar con los diferentes sistemas operativos	Booleano (Si, No)	Si	Excelente	4	
	Seguridad de acceso	Seguridad de los datos del usuario	Booleano (Si, No)	Si	En su mayor parte	3	
Seguridad de los datos de la aplicación		Booleano (Si, No)	Si	En su mayor parte	3		
Usabilidad	Capacidad para ser entendido	Tiempo necesario para entender	Rango numero (<= 3 min)	2 min	Muy adecuado	4	10%
	Capacidad para ser aprendido	Tiempo necesario para aprender	Rango numero (<= 3 min)	2 min	Adecuado	3	
		Satisfacción mientras aprende	Booleano (Si, No)	Si	Satisfecho	3	
	Capacidad para ser operado	Tiempo necesario para utilizar	Rango numero (<= 3 min)	3 min	Muy adecuado	4	
		Satisfacción con la ayuda	Booleano (Si, No)	Si	Satisfecho	3	
	Atractivo	Satisfacción con la interfaz	Booleano (Si, No)	Si	Satisfecho	3	
Eficiencia	Comportamiento temporal	Tiempo necesario para completar una funcionalidad	Rango numero (<= 2 seg)	1 seg	Adecuado	3	10%
	Utilización de recursos	Porcentaje de batería utilizada	Booleano (Si, No)	No	Adecuado	3	
Portabilidad	Adaptabilidad	Se adapta de manera correcta a la versión del software del dispositivo	Booleano (Si, No)	Si	Excelente	4	25%
	Instalabilidad	Tiempo de instalación	Rango numero	60	Muy	4	

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Características de calidad de ISO/EC 9126	Sub-características de calidad de ISO/EC 9126	Atributo definido (estándar)	Indicador	Datos	Escala Cualitativa	Valor (0-4)	Ponderación
			(<= 60 seg)		adecuado		
		Proceso de instalación	Booleano (Fácil, Difícil)	Fácil	Satisfecho	3	
		Satisfacción en el proceso de instalación	Booleano (Si, No)	Si	Satisfecho	3	
		Numero de interacciones durante la instalación	Rango numero (<= 5)	4	Satisfecho	3	
	Coexistencia	Capacidad de interactuar con los diferentes servicios web que depende para su correcto funcionamiento	Booleano (Si, No)	Si	Parcial	2	
	Capacidad para ser reemplazado	Actualizar una nueva versión	Booleano (Si, No)	Si	Muy adecuado	4	
Efectividad		Cumple con los objetivos establecidos	Booleano (Si, No)	Si	En su mayor parte	3	30%
Productividad		Consumo de recursos adicionales para el funcionamiento del software	Booleano (Si, No)	No	Excelente	4	
Seguridad de uso		Seguridad con el uso de los datos del usuario	Booleano (Si, No)	Si	En su mayor parte	3	
Satisfacción		El uso y navegación del software	(Fácil, Difícil)	Fácil	Satisfecho	3	
							100%

Fuente: Modelo ISO/EC 9126, El autor.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 15/10/2016
Plan de medición de calidad	

Las ponderaciones asignadas en esta tabla en relación a las características de Funcionalidad) el 21% de 25%, Usabilidad) 8% de 10%, Eficiencia) 8.5% de 10%, Portabilidad) el 20% de 25%, Efectividad, Productividad, Seguridad de uso, Satisfacción) el 25% del 30%, esto quiere decir que el 82.5% del 100% se valora en la escala cualitativa de 4 como máxima en la escala. Y por consecuencia la evaluación del producto software a través de cada una de las métricas definidas se apoya en el modelo ISO/EC 9126 ya que los aspectos son cuantificables y pueden medirse. Al analizar las características y sub-características que componen el modelo, se determino que las métricas, los atributos y los datos son suficientes al momento de medir la validación de las aplicaciones móviles.

La aplicación móvil “Gestión de tutorías UTPL” ha sido asignada a estudiantes de la UTPL en carreras y materias específicas; donde el 95% de los usuarios utilizaron de manera completa todas las funcionalidades de la aplicación, y ” cumple con las necesidades y requerimientos establecidos.

4. GLOSARIO

Modelo ISO/EC 9126: es un estándar internacional para la evaluación de la calidad del software. Está dividido en cuatro partes las cuales dirigen, realidad, métricas externas, métricas internas y calidad en las métricas de uso.

Calidad interna y externa: categoriza los atributos de calidad de software en seis características: Funcionalidad, Adecuación, Exactitud, Interoperabilidad, Seguridad de acceso, Usabilidad, Capacidad para ser entendido, Capacidad para ser aprendido, Capacidad para ser operado, Atractivo, Eficiencia, Comportamiento temporal, Utilización de recursos, Portabilidad, Adaptabilidad, Inestabilidad, Coexistencia, Capacidad para ser reemplazado.

Calidad de uso: Conjunto de atributos relacionados con la aceptación por parte del usuario final y Seguridad: Efectividad, Productividad, Seguridad de uso, Satisfacción.

Anexo 7. Plan de pruebas

**“Análisis y desarrollo de un framework metodológico
para la innovación educativa enfocada a la acción
tutorial docente de la Modalidad Presencial de la
UTPL”**

Plan de Pruebas

Versión 1.0

[La versión del documento se actualizará según la iteración y la fase del proyecto]

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
25/10/2016	1.0	Versión preliminar como propuesta de desarrollo.	Gabriela Puglla

Información del proyecto

Empresa/Organización	UTPL
Nombre del proyecto	“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”
Fecha de elaboración	25/10/2016
Cliente	UTPL
Patrocinador	UTPL
Gerente del proyecto	Ing. Fernanda Soto
Analista de Pruebas	Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

Tabla de Contenidos

Plan de Pruebas	4
1. Introducción.....	4
1.1 <i>Resumen.....</i>	<i>4</i>
1.2 <i>Alcance</i>	<i>4</i>
1.3 <i>Lugar</i>	<i>4</i>
1.4 <i>Definiciones</i>	<i>4</i>
1.5 <i>Referencias</i>	<i>4</i>
2. Agenda Propuesta.....	5
3. Condiciones	5
3.1 <i>Condiciones de Entrada.....</i>	<i>5</i>
3.2 <i>Condiciones para Continuar</i>	<i>5</i>
3.3 <i>Condiciones de Salida</i>	<i>5</i>
4. Requerimientos del Entorno	5
5. Desarrollo de las pruebas.....	6
6. Ejecución de las pruebas	6
6.1 <i>Tipos de pruebas.....</i>	<i>6</i>
6.2 <i>Personas y roles.....</i>	<i>9</i>
6.3 <i>Requerimientos de Capacitación</i>	<i>9</i>
6.4 <i>Proceso de Seguimiento de Incidentes.....</i>	<i>9</i>
6.5 <i>Clasificación de los defectos.....</i>	<i>9</i>
6.6 <i>Administración de las Versiones.....</i>	<i>9</i>
6.7 <i>Ciclos.....</i>	<i>9</i>
7. Entregables.....	10
7.1 <i>Reporte de Evaluación de las pruebas.....</i>	<i>10</i>
7.2 <i>Otros Productos.....</i>	<i>10</i>

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

Plan de Pruebas

1. Introducción

1.1 Resumen

Este Plan de Pruebas documentado para el “Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL” tiene los siguientes objetivos:

- Se van a probar todos los requerimientos definidos en el documento de ERS (Anexo 3).
- A través de la definición de casos de prueba se van a ejecutar las pruebas de la aplicación móvil “Gestión de tutorías UTPL” directamente en los dispositivos móviles.

1.2 Alcance

Mediante la tabla se indican que funcionalidades están y cuales no están dentro del esfuerzo de evaluación.

Ingresar al QR
Ver componentes
Crear tutoría
Editar tutoría
Ver notificaciones
Ver participantes
Ver comentarios
Validar asistencia
Ver tutorías
Participar en tutoría
Comentar tutoría
Ver notificaciones
Ver participantes
Ver comentarios
Validar asistencia
Calificar tutoría
Ver estado de tutoría

1.3 Lugar

El lugar a ejecutar las pruebas se realizaran en el Campus de la UTPL, se asignará a un grupo de estudiantes, docente y Vicerrectorado de UTPL para que utilicen la aplicación y obtener los resultados de uso.

1.4 Definiciones

[Incluir todos los términos que se usen en este plan para que todos lo entiendan]

1.5 Referencias

- Aplicación móvil “Gestión de tutorías UTPL
- “Documento de especificación de requerimientos
- Prototipos del producto (Diseño e interfaces de usuario)
- Plan de pruebas actualizado
- Plantilla de historias de usuario y tarjetas de tareas
- Servicios Web que se utilizan para ejecutar la aplicación móvil.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

2. Agenda Propuesta

OBJETIVO	FECHA
Planificación del Proyecto	14-10-2016
Plan de pruebas completo	09-12-2016
Ambiente de pruebas configurado	09-12-2016
Diseño de las pruebas	28-10-2016
Casos de prueba	15-11-2016
Ejecución de las pruebas	12-12-2016
Inicio de las pruebas	12-12-2016
Ciclo 1	12-12-2016 / 02-02-2017
Ciclo 2	-

3. Condiciones

3.1 Condiciones de Entrada

- El sistema de seguimiento de incidentes fue definido
- Todos los componentes están bajo administración de la configuración
- El equipo de desarrollo configuró el entorno.
- El equipo de pruebas tiene los permisos adecuados de acceso al sistema
- El equipo de desarrollo ha completado todas las funcionalidades (y arreglado todos los defectos) establecidas para esta versión
- El equipo de desarrollo ha realizado las pruebas unitarias de todos los componentes de esta versión
- El equipo de desarrollo provee el software al equipo de pruebas 3 días calendario antes de comenzar las pruebas

3.2 Condiciones para Continuar

- Se hacen reuniones semanales de revisión de bugs para gestionar los bugs abiertos y los tiempos de cierre

3.3 Condiciones de Salida

- No se han hecho cambios excepto para arreglar defectos encontrados en las últimas reuniones
- El equipo de proyecto (Desarrollo) ha ejecutado todas las pruebas planificadas
- El equipo de pruebas ha verificado que todos los bugs en el sistema de seguimiento están cerrados o rechazados, verificado por medio de pruebas de regresión
- Se tiene una reunión entre el Equipo de dirección y el equipo de pruebas (calidad) para acordar que se ha finalizado la evaluación.

4. Requerimientos del Entorno

Presenta los recursos no humanos que se requieren para las pruebas, los recursos tanto de hardware y software en el ambiente de las aplicaciones móviles:

- En este caso los entornos y herramientas de pruebas debe considerarse el tipo de aplicación que se está probando, por ejemplo si es de aplicación web móvil, nativa o híbrida.
- Se define el uso de emuladores y dispositivos móviles identificando las principales características, considerando que existe una amplia gama de versiones que se deben considerar:

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

Entorno de desarrollo	
ID de dispositivo móvil	Detalle del dispositivo móvil
<DM01> Smartphone Android	Tipo: Smartphone/Tablet Marca: Independiente “Sistema operativo Android” Modelo del dispositivo móvil: Independiente “Sistema operativo Android” Versión del sistema operativo: A partir de la versión 4.1 (Jelly Bean)
<DM02> Smartphone iOS	Tipo: Smartphone/Tablet Marca: iOS Mac Modelo del dispositivo móvil: iPhone, iPad Versión del sistema operativo: A partir de la versión 7.
ID del emulador	Detalle del emulador
Simulador iOS	Nombre del emulador: Xcode Lenguaje de programación utilizado/Plataforma para el desarrollo del producto de software: Ionic Framework Simulador en línea, descargado: El simulador es parte del framework de desarrollo.
Simulador Android	Nombre del emulador: Android Studio Lenguaje de programación utilizado/Plataforma para el desarrollo del producto de software: Ionic Framework. Simulador en línea, descargado: El simulador es parte del framework de desarrollo.
Herramientas de pruebas de desempeño (Carga y estres), funcionalidad y seguridad.	Monkop1. : Herramienta para pruebas de desempeño, funcionalidad y seguridad, utilizando el archivo ejecutable. Registrado en versión libre por un periodo de 30 días. Testdroid cloud2. Herramienta para pruebas de desempeño, funcionalidad y seguridad, utilizando el archivo ejecutable. Registrado en versión libre por un periodo de 30 días. Test object.3 Herramienta para pruebas de desempeño, funcionalidad y seguridad, utilizando el archivo ejecutable. Registrado en versión libre por un periodo de 30 días.

5. Desarrollo de las pruebas

Describe la forma en que el equipo de pruebas va a crear el Testware y cómo se conseguirán los datos para las pruebas.

El equipo de calidad (Gerente de pruebas) solicita como entrada principal el documento de especificación de requerimientos (ERS) en el que detallan los requerimientos que ha solicitado el grupo de interesados, DGA UTPL, para poder ejecutar las pruebas.

6. Ejecución de las pruebas

Describir la forma en que serán manejados las versiones, las pruebas y los datos durante la ejecución de las pruebas.

6.1 Tipos de pruebas

Las consideraciones principales para la estrategia de pruebas son las técnicas a usarse y los criterios para determinar si la prueba fue completada. Además de las consideraciones provistas para cada prueba mencionada, las pruebas deberían ser únicamente ejecutadas usando bases de datos conocidas y controladas en entornos seguros.

¹ Monkop : <https://console.monkop.com>

² Testdroid cloud: <https://cloud.testdroid.com/>

³ Test object: <https://app.testobject.com/>

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

La siguiente estrategia de pruebas es genérica en su naturaleza y está dirigida a aplicarse sobre los requerimientos listados en la sección 3, de este documento. Para cada una de las pruebas a ejecutar se detalla los siguientes consideraciones, tomando en cuenta el ítem 3.2.2.1:

- **Tipos de pruebas**

- **Pruebas funcionales:** Permite, asegurar que las funciones de la aplicación cumplan con las especificaciones comerciales y los requerimientos que darán solución a la necesidad del cliente
- **Pruebas de Seguridad:** Buscan validar la resistencia de la aplicación a ataques por usuarios maliciosos.

Por ejemplo:

- Ataques vía la red
- Ataques al servidor.
- Ataques al dispositivo.

Con este tipo de pruebas también se busca validar que los programadores apliquen prácticas de seguridad informática en el código de programa.

- **Pruebas de interrupción:** Las pruebas de interrupción se incluyen y funcionan ejecutando otra actividad o suceso, por ejemplo:
 - Mensajes de error.
 - Llamada entrante.
 - SMS entrante.
 - Correo electrónico.
 - Notificaciones de social media.
 - Batería baja
 - Batería en estado crítico.
 - Apagado del equipo.
 - Caída de la conexión con la red.
- **Pruebas de Compatibilidad:** Consisten en validar la compatibilidad de la aplicación con:
 - Pruebas en distintos dispositivos y sistemas operativos.
 - Otras aplicaciones.
 - Plataformas.
 - Dispositivos.
 - Navegadores.
 - Redes de telecomunicaciones.
 - Versiones anteriores de la aplicación.
- **Pruebas de desempeño (carga y estrés):** Se realizan para determinar la rapidez y la capacidad para ejecutar una tarea de la aplicación móvil, en condiciones particulares de trabajo.
- **Pruebas de Interfaz con el usuario (UI):** Buscan evaluar la interfaz con el usuario en relación con estándares para la interacción humano-maquina aceptable. Entre las pruebas que podemos realizar tenemos:
 - Proceso de instalación y desinstalación correcta de la aplicación.
 - Verificación de logo y nombre de la aplicación
 - Dispositivos con pantalla táctil y sin pantalla táctil.
 - Organización de pantallas, alineación, colores, fondos y patrón de lectura.
 - Ajuste y rotación de la aplicación en relación a la dirección de la dirección del dispositivo (horizontal/vertical).
 - Posición, tamaño, datos de entradas y acciones.
 - Claridad, alineación y densidad de imágenes y símbolos.
 - Consistencia de la interfaz en toda la aplicación y entre dispositivos disímiles.
 - Presentación de la información en el diseño de página para móviles.
 - Facilidad para completar tareas.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

- Eficiencia y exactitud.
- Minimizar que el usuario tenga que recordar información.
- Tamaño de pantalla.
- Bloqueo de teclado.
- Condiciones de iluminación.
- Tamaño de la interfaz táctil.
- Integración de aplicaciones nativas, aplicaciones web móvil e híbridas.
- Salir de la aplicación sin ningún inconveniente.
- Inicio/reinicio de la aplicación menor 25 segundos.
- Comportamiento de la aplicación al llegar una llamada o mensaje.
- La aplicación no debe bloquearse/colgarse.
- **Pruebas de Movilidad:** Consiste en probar el desempeño de la aplicación cuando el dispositivo móvil se encuentra en movimiento, con una persona o vehículos. Esto es más importante en aplicaciones que utilizan la localización para mostrar información relevante al usuario. Posibles pruebas a realizar:
 - Geo localización
 - Atravesar zonas de la red móvil (por ejemplo ir de una celda a otra).
 - Señal de GPS, Wi-Fi o celular débil o intermitente (se interrumpe/corta).
- **Pruebas de Conectividad:** Desempeño de la aplicación al conectarse a las redes en distintos protocolos y distintas condiciones, por ejemplo:
 - Wi-Fi.
 - Bluetooth.
 - Red analógica, 3G o 4G.
 - Señales de distinta intensidad (sin señal, conexión intermitente, señal fuerte, intensidad de señal variable).
- **Pruebas de usabilidad:** son ejecutadas directamente con los evaluadores y usuarios finales, en este caso a través de una encuesta permite tomar nota de los errores y dificultades que se identifican los usuarios.
- **Pruebas de aceptación:** son ejecutadas directamente con los evaluadores y usuarios finales con el objetivo de obtener un producto de excelente calidad.
- **Requerimientos:** Se listan los requisitos a los que se irá haciendo un seguimiento a lo largo de todo el desarrollo del proyecto.
 - **Ingresar al QR:** permite ingresar al sistema de gestión de tutorías especificando el rol del usuario (Docente, Estudiante, DGA) respectivamente para gestionar, acceder y generar reportes de las tutorías.
 - **Ver componentes:** presenta los componentes educativos para crear, ver y generar reportes de las tutorías.
 - **Crear tutoría:** permite al usuario con rol específico (Docente) crear una tutoría.
 - **Editar tutoría:** permite al usuario con rol específico (Docente) modificar los datos de la tutoría creada.
 - **Ver notificaciones:** permite al usuario con rol (Docente / Estudiante) informar a través del correo, sobre los comentarios y participantes de una tutoría.
 - **Ver participantes:** permite al usuario con rol específico (Docente) ver a los estudiantes que asistirán a la tutoría.
 - **Ver comentarios:** permite al usuario con rol específico (Docente) ver los comentarios de cada estudiante sobre el tema de la tutoría creada.
 - **Validar asistencia:** permite al usuario con rol específico (Docente) validar la asistencia del estudiante a la tutoría.
 - **Ver tutorías creadas:** permite al usuario con rol (Docente / Estudiante) ver las tutorías creadas.
 - **Participar en tutoría:** permite al usuario con rol específico (Estudiante) participar en una tutoría.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

- **Comentar tutoría:** permite al usuario con rol específico (Estudiante) opinar/comentar sobre un tema de la tutoría.
- **Calificar tutoría:** permite al usuario con rol específico (Estudiante) calificar una tutoría asistida.
- **Id del dispositivo/Simulador:** DM01, DM02/ SM iOS, SM Android.
- **Objetivo de la prueba:** comprobar, verificar o validar el sistema con la aplicación de las pruebas establecidas
- **Técnica:** verificar cada una de las funcionalidades.
- **Criterio de cumplimiento:** se requiere tener acceso a toda la aplicación.

Además se debe tener en cuenta que al momento de desarrollar la aplicación móvil debe cumplir los lineamientos, condiciones y características de diseño exigido por las App Stores de Android⁴ e iOS⁵ y que debe tener la aplicación.

6.2 Personas y roles

Rol	Persona	Responsabilidades
Analista de pruebas	Gabriela Puglla	<ul style="list-style-type: none"> ▪ Establecer el plan de pruebas y detallar y especificar los tipos de pruebas que se van a ejecutar para validar al aplicación móvil “Gestión de tutorías UTPL”.

6.3 Requerimientos de Capacitación

Describir los requerimientos de capacitación necesarios para realizar las pruebas: No se requiere capacitación para ejecutar las pruebas en este proyecto.

6.4 Proceso de Seguimiento de Incidentes

Describir el sistema usado para administrar y seguir los casos de prueba y los bugs: en este caso el proceso de seguimiento se administra a través del documento que detalla los casos de prueba para la aplicación y la matriz de reporte y bugs que se presenta en el anexo 9 de este trabajo.

6.5 Clasificación de los defectos

Describir la forma en que se clasifican los defectos encontrados: se establecen a través de estados de prioridad que se asignan a los errores encontrados, que se detalla en la matriz de reporte y bugs.

6.6 Administración de las Versiones

Describir quien será el encargado de manejar las versiones en el ambiente de pruebas: El encargado de manejar las versiones en el desarrollo de la aplicación móvil la manejan el equipo del proyecto y el gerente de calidad quienes consideran cuando se considera el control de versiones del producto.

6.7 Ciclos

Describir la forma en que serán administrados los ciclos de prueba se detallan en el apartado 2 de este plan.

⁴ Guía de diseño de Android. <http://developer.android.com/design/patterns/navigation-drawer.html>

⁵ iOS Human Interface Guidelines. <https://developer.apple.com/library/ios/documentation/UserExperience/Conceptual/MobileHIG/index.html#>

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 25/10/2016
Plan de pruebas	

7. Entregables

Establece que se entregará como parte de la ejecución del plan, por ejemplo:

- Plan de Pruebas,
- Casos de Pruebas,
- Especificación de Diseño de Casos,
- Logs de errores, de incidencias,
- Evidencias de prueba.

7.1 Reporte de Evaluación de las pruebas

El reporte de ejecución de las pruebas se presenta en el anexo 11 del desarrollo de este proyecto.

7.2 Otros Productos

Herramientas web seleccionadas y utilizadas para ejecutar pruebas de desempeño (carga y estrés), seguridad y funcionalidad de las aplicaciones móviles en la que se utiliza específicamente el archivo ejecutable .apk o .app para obtener resultados de las pruebas (Apartado 4).

Anexo 8. Casos de pruebas

Aplicación móvil “Gestión de tutorías UTPL”

Casos de prueba

Rol Usuario final (Docente)

Tabla 1. Caso de prueba. Ingresar al QR

Id Caso de prueba	001		
Nombre del Caso de Prueba	Ingresar a la aplicación “Gestión de tutorías UTPL”	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) va ingresar a la aplicación móvil para poder gestionar las tutorías.		
Funcionalidad a probar:	Ingresar al QR		
Pre-requisitos	<ul style="list-style-type: none"> • Disponer de una conexión de internet estable. • Descargar la aplicación móvil • Instalar la aplicación 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Abrir la aplicación móvil • El usuario pertenece a la UTPL • Ingresar los datos del usuario (nombre y contraseña del EVA) para ingresar al sistema de gestión de tutorías. 		
Resultados esperados:	El usuario debe haber iniciado sesión en el sistema de gestión de tutorías a través de la aplicación móvil “Gestión de Tutorías UTPL”	Resultados obtenidos:	Efectivamente el usuario con rol de docente ha podido ingresar al sistema de gestión de tutorías a través de la aplicación móvil.
Errores asociados:	Se puede mejorar la interfaz de usuario con un tamaño estatico de las ventana a partir de un estandar de diseño de android 4.0.		
Estado	Ejecutado (X)	Exitoso (X)	
	Pendiente en ejecución ()	Fallido ()	
	En construcción ()	Parado ()	

Tabla 2: Caso de prueba. Ver componentes

Id Caso de prueba	002		
Nombre del Caso de Prueba	Ver componentes	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) verifica que puede ver todos los componentes educativos que tiene asignado.		
Funcionalidad a probar:	Ver componentes educativos asignados		
Pre-requisitos	<ul style="list-style-type: none"> • Haber ingresado correctamente al sistema de gestión de tutorías. 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Visualizar todos los componentes educativos. 		
Resultados esperados:	El usuario puede visualizar todos los componentes educativos asignados	Resultados obtenidos:	El usuario a podido visualizar de manera correcta
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
	Pendiente en ejecución ()	Fallido ()	
	En construcción ()	Parado ()	

Tabla 3: Caso de prueba. Crear tutorías

Id Caso de prueba	003		
Nombre del Caso de Prueba	Crear tutorías	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) permite crear tutorías dentro del componente educativo seleccionado.		
Funcionalidad a probar:	<ul style="list-style-type: none"> • Crear tutorías • Ver tutorías creadas 		
Pre-requisitos	<ul style="list-style-type: none"> • Visualizar todos los componentes educativos asignados • Seleccionar el componente educativo. • Ingresar al componente educativo 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Seleccionar la opción crear tutoría • Ver la nueva ventana para crear la tutoría • Seleccionar la fecha de tutoría del desplegable "Fecha de tutoría" (las fechas son específicas en relación al plan de estudios del componente educativo) • Seleccionar el horario de inicio de la tutoría • Seleccionar el horario de salida de la tutoría • Seleccionar el número de aula donde se va realizar la tutoría • Incluir y establecer el tema sobre la tutoría. (se relaciona con los temas dictados en la última clase) • Incluir y detallar los contenidos a tratar en la tutoría • Incluir y detallar observaciones que se pueden considerar en la tutoría. • Seleccionar y presionar el botón de crear tutoría. • Ver la tutoría creada 		
Resultados esperados:	<ul style="list-style-type: none"> • El docente puede crear una tutoría dentro del componente educativo y poder seleccionar y llenar todos los campos requeridos. • El usuario puede ver la tutoría creada dentro del componente educativo. 	Resultados obtenidos:	<p>El docente puede crear de manera correcta una tutoría dentro del componente educativo y poder seleccionar y llenar todos los campos requeridos.</p> <p>El usuario puede ver correctamente la tutoría creada y cada uno de los campos llenados dentro del componente educativo.</p>
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
	Fallido ()	Parado ()	
	Pendiente en ejecución ()		
	En construcción ()		

Tabla 4: Caso de prueba. Editar tutorías

Id Caso de prueba	004		
Nombre del Caso de Prueba	Editar tutorías	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) puede editar cada uno de los campos establecidos en la tutoría si es necesario.		
Funcionalidad a probar:	<ul style="list-style-type: none"> • Editar tutorías • Ver tutorías 		
Pre-requisitos	<ul style="list-style-type: none"> • Haber creado una tutoría dentro de un component educativo 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Ingresar a la tutoría creada • Seleccionar la opción de editar dentro de la tutoría • Ver la nueva ventana que muestra los campos y que en este caso se pueden editar y/o modificar lo que sea necesario. • Seleccionar y presionar el botón de crear tutoría, una vez que los campos hayan sido modificados. • Ver la tutoría con los campos modificados 		
Resultados esperados:	<ul style="list-style-type: none"> • El usuario puede editar cada uno de los campos establecidos en la tutoría si es necesario. 	Resultados obtenidos:	<ul style="list-style-type: none"> • El usuario puede editar de manera correcta cada uno de los campos establecidos en la tutoría si es necesario.
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
		Fallido ()	
	Pendiente en ejecución ()	Parado ()	
	En construcción ()		

Tabla 5: Caso de prueba. Ver notificaciones

Id Caso de prueba	005		
Nombre del Caso de Prueba	Ver notificaciones	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) recibe una notificación de que se ha creado una tutoría que se envía al correo electrónico para su verificación.		
Funcionalidad a probar:	<ul style="list-style-type: none"> • Ver notificaciones 		
Pre-requisitos	<ul style="list-style-type: none"> • El usuario ha creado una tutoría en el componente educativo seleccionado 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Selecciona y acepta enviar un correo de notificación a través de una nueva ventana que permite ingresar los datos obligatorios de correo electrónico y recibir la notificación. • El usuario ha enviado una notificación a la lista de los participantes del componente educativo seleccionado. 		
Resultados esperados:	<ul style="list-style-type: none"> • El usuario debe recibir una notificación de que la tutoría ha sido creada correctamente a través de un correo electrónico en el que se presentan los campos que han sido llenados. 	Resultados obtenidos:	<ul style="list-style-type: none"> • El usuario ha recibido una notificación de que la tutoría ha sido creada correctamente a través de un correo electrónico en el que se presentan los campos que han sido llenados.
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
		Fallido ()	
	Pendiente en ejecución ()	Parado ()	
	En construcción ()		

Tabla 6: Caso de prueba. Ver participantes

Id Caso de prueba	006		
Nombre del Caso de Prueba	Ver participantes	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) puede ver la lista de los participantes que se han unido a la tutoría.		
Funcionalidad a probar:	<ul style="list-style-type: none"> Ver notificaciones Ver participantes 		
Pre-requisitos	<ul style="list-style-type: none"> El usuario ha creado una tutoría en el componente educativo seleccionado. El usuario ha enviado una notificación a través de correo electrónico a la lista de los participantes del componente educativo seleccionado. 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> El usuario puede ver la lista de los participantes que van a asistir a la tutoría. El usuario puede ver la fecha, hora de inicio y fin en la que va a participar en la tutoría. 		
Resultados esperados:	<ul style="list-style-type: none"> El usuario con rol de docente puede ver la lista de los participantes que van a asistir a la tutoría creada. 	Resultados obtenidos:	<ul style="list-style-type: none"> El usuario con rol de docente puede ver la lista de los participantes que van a asistir a la tutoría creada.
Errores asociados:			
Estado	Ejecutado (X)	Exitoso (X)	
	Pendiente en ejecución ()	Fallido ()	
	En construcción ()	Parado ()	

Tabla 7: Caso de prueba. Ver comentarios

Id Caso de prueba	007		
Nombre del Caso de Prueba	Ver comentarios	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) puede ver los comentarios que han emitido los participantes (estudiantes) en la tutoría creada.		
Funcionalidad a probar:	<ul style="list-style-type: none"> Ver comentarios Ver participantes 		
Pre-requisitos	<ul style="list-style-type: none"> El usuario ha creado una tutoría en el componente educativo seleccionado. El usuario puede ver la lista de los participantes que van a asistir a la tutoría. 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> El usuario puede ver los comentarios que han emitido los participantes (estudiantes) en la tutoría creada. 		
Resultados esperados:	<ul style="list-style-type: none"> El usuario con rol de docente puede ver todos los comentarios de los participantes que se han incluido en la tutoría creada. 	Resultados obtenidos:	<ul style="list-style-type: none"> El usuario con rol de docente puede ver todos los comentarios de los participantes que se han incluido en la tutoría creada.
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
	Pendiente en ejecución ()	Fallido ()	
	En construcción ()	Parado ()	

Tabla 8: Caso de prueba. Caso de prueba. Validar asistencia

Id Caso de prueba	008		
Nombre del Caso de Prueba	Validar asistencia	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Docente) puede validar la asistencia a cada uno de los participantes que se ha unido a la tutoría creada.		
Funcionalidad a probar:	<ul style="list-style-type: none"> • Ver participantes • Validar asistencia 		
Pre-requisitos	<ul style="list-style-type: none"> • El usuario (docente) puede ver la lista de todos los participantes que se han unido a la tutoría creada. • El usuario (docente) puede ver la lista de participantes 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Seleccionar a cada uno de los participantes que se presenta en una nueva ventana. • El usuario puede seleccionar la opción de “Si asistió” y “No asistio” para validar la asistencia. 		
Resultados esperados:	<ul style="list-style-type: none"> • El usuario (Docente) puede validar la asistencia. 	Resultados obtenidos:	<ul style="list-style-type: none"> • El usuario (Docente) puede validar la asistencia, de manera correcta.
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X) Fallido () Parado ()	
	Pendiente en ejecución ()		
	En construcción ()		

Rol Usuario final (Estudiante)

Tabla 9. Caso de prueba. Ingresar al QR

Id Caso de prueba	009		
Nombre del Caso de Prueba	Ingresar a la aplicación "Gestión de tutorías UTPL"	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Estudiante) va ingresar a la aplicación móvil para poder gestionar las tutorías.		
Funcionalidad a probar:	Ingresar al QR		
Pre-requisitos	<ul style="list-style-type: none"> • El usuario pertenece a la UTPL • Disponer de una conexión de internet estable. • Descargar la aplicación móvil • Instalar la aplicación 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Abrir la aplicación móvil • Ingresar los datos del usuario (nombre y contraseña del EVA) para ingresar al sistema de gestión de tutorías. 		
Resultados esperados:	El usuario debe haber iniciado sesión en el sistema de gestión de tutorías a través de la aplicación móvil "Gestión de Tutorías UTPL"	Resultados obtenidos:	Efectivamente el usuario con rol de estudiante ha podido ingresar al sistema de gestión de tutorías a través de la aplicación móvil.
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
	Pendiente en ejecución ()	Fallido ()	
	En construcción ()	Parado ()	

Tabla 10: Caso de prueba. Ver componentes

Id Caso de prueba	010		
Nombre del Caso de Prueba	Ver componentes	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (Estudiante) verifica que puede ver todos los componentes educativos que tiene asignado.		
Funcionalidad a probar:	Ver componentes educativos asignados		
Pre-requisitos	<ul style="list-style-type: none"> • Haber ingresado correctamente al sistema de gestión de tutorías. 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> • Visualizar todos los componentes educativos. 		
Resultados esperados:	El usuario puede visualizar todos los componentes educativos asignados	Resultados obtenidos:	El usuario ha podido visualizar de manera correcta
Errores asociados:	Ninguno		
Estado	Ejecutado (X)	Exitoso (X)	
	Pendiente en ejecución ()	Fallido ()	
	En construcción ()	Parado ()	

Tabla 11: Caso de prueba. Ver tutorías

Id Caso de prueba	011		
Nombre del Caso de Prueba	<ul style="list-style-type: none"> Ver notificaciones Ver tutorías 	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (estudiante) permite crear tutorías dentro del componente educativo seleccionado.		
Funcionalidad a probar:	<ul style="list-style-type: none"> Ver tutorías creadas 		
Pre-requisitos	<ul style="list-style-type: none"> Revisar a través del correo electrónico la notificación para participar en una tutoría creada por el docente. Visualizar todos los componentes educativos asignados Seleccionar el componente educativo. Ingresar al componente educativo 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> Ver la tutoría creada Ver la información de la tutoría creada (Nombre de la tutoría, fecha, hora, numero de aula) 		
Resultados esperados:	<ul style="list-style-type: none"> El usuario (Estudiante) puede ver la tutoría creada dentro del componente educativo. 	Resultados obtenidos:	El usuario (estudiante) puede ver correctamente la tutoría creada, con la información necesaria para poder asistir.
Errores asociados:			
Estado	Ejecutado (X)	Exitoso (X)	
		Fallido ()	
	Pendiente en ejecución ()	Parado ()	
	En construcción ()		

Tabla 12: Caso de prueba. Participar en tutoría

Id Caso de prueba	012		
Nombre del Caso de Prueba	Ver tutorías	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (estudiante) permite crear tutorías dentro del componente educativo seleccionado.		
Funcionalidad a probar:	<ul style="list-style-type: none"> Ver tutorías creadas 		
Pre-requisitos	<ul style="list-style-type: none"> Visualizar todos los componentes educativos asignados Seleccionar el componente educativo. Ingresar al componente educativo 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> Ver la tutoría creada Ver la información de la tutoría creada (Nombre de la tutoría, fecha, hora, numero de aula) Seleccionar el horario a través de una nueva ventana que permite elegir la hora y minutos en el que va asistir a la tutoría. 		
Resultados esperados:	<ul style="list-style-type: none"> El usuario (Estudiante) puede ver la tutoría creada dentro del componente educativo. 	Resultados obtenidos:	El usuario (estudiante) puede ver correctamente la tutoría creada, con la información necesaria para poder asistir.
Errores asociados:			
Estado	Ejecutado (X)	Exitoso (X)	
		Fallido ()	
	Pendiente en ejecución ()	Parado ()	
	En construcción ()		

Tabla 13: Caso de prueba. Comentar tutorías

Id Caso de prueba	013		
Nombre del Caso de Prueba	<ul style="list-style-type: none"> Comentar tutorías Ver comentarios 	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (estudiante) permite comentar una tutoría creada dentro del componente educativo seleccionado.		
Funcionalidad a probar:	<ul style="list-style-type: none"> Ver tutorías Comentar tutorías 		
Pre-requisitos	<ul style="list-style-type: none"> Visualizar todos los componentes educativos asignados Seleccionar el componente educativo. Ingresar al componente educativo 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> Ver la tutoría creada Ver la información de la tutoría creada (Nombre de la tutoría, fecha, hora, numero de aula) Comentar sobre el tema de la tutoría creada Escribir en el campo de texto asignando para los comentarios, sobre un tema específico a tratarse en la tutoría. El estudiante puede ver el comentario que ha incluido dentro de la tutoría creada. 		
Resultados esperados:	<ul style="list-style-type: none"> El usuario (Estudiante) puede ver la tutoría creada dentro del componente educativo. 	Resultados obtenidos:	El usuario (estudiante) puede ver correctamente la tutoría creada, con la información necesaria para poder asistir.
Errores asociados:			
Estado	Ejecutado (X)	Exitoso (X)	
		Fallido ()	
	Pendiente en ejecución ()	Parado ()	
	En construcción ()		

Tabla 14: Caso de prueba. Calificar tutorías

Id Caso de prueba	015		
Nombre del Caso de Prueba	<ul style="list-style-type: none"> Calificar tutorías 	Prioridad	Alta (X) Media () Baja ()
Descripción	El usuario (estudiante)		
Funcionalidad a probar:	<ul style="list-style-type: none"> Ver tutorías Comentar tutorías Ver estado de tutoría 		
Pre-requisitos	<ul style="list-style-type: none"> Ver la tutoría creada Haber creado un horario de participación en la tutoría Ver la información de la tutoría creada (Nombre de la tutoría, fecha, hora, número de aula). 		
Pasos y condiciones a ejecutar	<ul style="list-style-type: none"> Puede calificar la tutoría a la que asistió a través de una nueva ventana que muestra 5 estados. Debe seleccionar un estado para calificar la tutoría Ver el estado de asistencia a la tutoría 		
Resultados esperados:	<ul style="list-style-type: none"> El usuario (Estudiante) ha seleccionado un estado para calificar la tutoría a la que asistió. 	Resultados obtenidos:	El usuario (estudiante) puede ver la calificación que ha asignado a la tutoría a la que asistió.
Errores asociados:			
Estado	Ejecutado (X)	Exitoso (X)	
		Fallido ()	
	Pendiente en ejecución ()	Parado ()	
	En construcción ()		

Anexo 9. Matriz de reportes de bugs e issues

Nombre de proyecto o programa:		"Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL"				Nombre de la aplicación:	Gestión de tutorías UTPL	Versión: 1.0.0 Android, iOS		Numero de informe: 1	
Tipo de Prueba	Id Funcionalidad	Funcionalidad	Hallazgo				Corrección		Fecha de corrección: DD/MM/AAAA	Observaciones	Estado
			Descripción	Gravedad de error	Fecha de reporte: DD/MM/AAAA	Responsable del error encontrado	Personal asignado para corrección del error	Medidas correctivas			
Pruebas de interfaz de usuario	1	Ingresar al QR	La primera ventana que permite ingresar al usuario a la aplicación se ajusta deacuerdo al tamaño del dispositivo y ergonomicamente al usuario no le agrada.	Bajo	10/10/2016	Usuarios Finales	Diego Moreno	Ajustar a un tamaño estatico la ventana y cada uno de los botones que se presentan en esta funcionalidad	22/11/2016	Se ha modificado en el proceso de desarrollo, se justifica que la interfaz esta diseñada deacuerdo al tamaño del dispositivo movil.	Corregido
	2	Ver componentes									
	3	Crear tutorias									
	4	Ver tutorias creadas									
	5	Ver participantes									
	6	Ver comentarios									
	7	Validar asistencia									
	8	Ver notificaciones									
Pruebas de interfaz de usuario	9	Calificar tutoria	El uso de los iconos no esta acorde al tipo de aplicación, se solicita cambiar la apariencia de los mismos	Medio	10/10/2016	Vicerrectorado de UTPL	Diego Moreno	Modificar y utilizar emoticons	22/11/2016	Se ha validado la corrección	Corregido
	10	Participar en tutoria									
	11	Comentar tutoria									
	12	Ver estado de tutoria									
Pruebas de funcionalidad	13	Generar reportes	Esta funcionalidad es de uso exclusivo de Vicerrectorado de UTPL como un servicio web para generar reportes estadísticos de la usabilidad de la aplicación movil y su proposito	Medio	10/10/2016	Vicerrectorado de UTPL	Diego Moreno	Se desarrollo en su totalidad	15/12/2016	Cuminando el proceso de desarrollo	Corregido

Anexo 10. Pruebas de usabilidad/Encuestas Usuarios finales

Validación de la aplicación “Gestión de tutorías UTPL”

Pruebas de Usabilidad/Usuarios Finales

Modelo de Dispositivo Móvil:

Versión de sistema operativo:

El proceso de instalación es:	Fácil ()	Difícil ()		
La instalación de la aplicación se ha ejecutado de manera correcta	Si ()	No ()		
Ha ingresado de manera correcta a la aplicación	Si ()	No ()		
Le agrada la apariencia de la aplicación	Si ()	No ()		
Le parece que la posición y funcionalidad de los botones/iconos son adecuados				
Las pantallas contiene los datos/información y funciones suficientes para ejecutar la aplicación				
El uso de la aplicación es una experiencia	Fácil ()	Difícil ()		
La navegación en la aplicación es	Fácil ()	Difícil ()		
El tiempo de ejecución de las funcionalidades es:	Rápida ()	Normal ()	Lenta ()	Muy lenta ()
El uso aplicaciones adicionales/ gestor de correo electrónico funciona correctamente	Si ()	No ()		
La documentación la infografía de la aplicación es fácil de entender	Si ()	No ()		
El tiempo de ejecución de múltiples consultas de la aplicación es:	Rápida ()	Normal ()	Lenta ()	Muy lenta ()
Se ha producido un considerable consumo de batería al ejecutar la aplicación	Si ()	No ()		
Los datos insertados del usuario se han almacenado correctamente	Si ()	No ()		
El usuario registrado puede ejecutar todas las funcionalidades	Si ()	No ()		
Se han presentado mensajes de error al momento de ejecutar la aplicación	Si ()	No ()		
La calidad de la conexión con los servicios web/internet.	Excelente()	Buena ()	Mala ()	Muy mala ()
La aplicación móvil se ha cerrado de manera correcta	Si ()	No ()		
EL inicio y reinicio para ejecutar la aplicación es	Rápida ()	Normal ()	Lenta ()	Muy lenta ()

Proceso / Funcionalidad	Funciona correctamente		Detalle del error	Solución
	SI	NO		

Validación de la aplicación “Gestión de tutorías UTPL”

Pruebas de Funcionalidad/Usuarios Finales – DOCENTES

Titulación:

El nombre de usuario y contraseña del usuario se han ingresado correctamente	Si ()	No ()
Puede ver todos los componentes educativos correspondientes	Si ()	No ()
Puede ingresar de manera correcta a la ventana de crear tutoría dentro del componente educativo seleccionado	Si ()	No ()
Ha logrado seleccionar de manera correcta: <ul style="list-style-type: none"> • Nombre de la tutoría • Fecha de asistencia a la tutoría • Horario de inicio de tutoría • Horario de fin de la tutoría • El numero de aula para realizar la tutoría • Contenido a tratar en la tutoría • Observaciones a considerar en la tutoría 	Si ()	No ()
Se ha creado correctamente la tutoría	Si ()	No ()
Se puede ver la(s) tutoría(s) creadas dentro del componente educativo seleccionado	Si ()	No ()
Se puede ingresar a la tutoría creada	Si ()	No ()
Se puede editar y/o modificar un campo seleccionado de la tutoría creada <ul style="list-style-type: none"> • Nombre de la tutoría • Fecha de asistencia a la tutoría • Horario de inicio de tutoría • Horario de fin de la tutoría • El numero de aula para realizar la tutoría • Contenido a tratar en la tutoría • Observaciones a considerar en la tutoría 	Si ()	No ()
Se puede ver los campos de la(s) tutoría(s) modificadas dentro del componente educativo seleccionado	Si ()	No ()
Se puede ver de manera correcta la lista de los participantes que forman parte del componente educativo	Si ()	No ()
Se ha enviado correctamente la notificación a los participantes a través del correo electrónico	Si ()	No ()
Se puede ver de manera correcta las dudas y temáticas por la que requieren asistir los participantes de la tutoría que forman parte del componente educativo	Si ()	No ()
Se ha validado de manera correcta la asistencia del participante en la tutoría	Si ()	No ()
Funciona correctamente la aplicación Gestión de tutorías UTPL	Si ()	No ()

Proceso / Funcionalidad	Funciona correctamente		Detalle del error	Solución
	SI	NO		

Validación de la aplicación “Gestión de tutorías UTPL”

Pruebas de Funcionalidad/Usuarios Finales – ESTUDIANTES

Titulación:

El nombre de usuario y contraseña del usuario se han ingresado correctamente	Si ()	No ()
Puede ver todos los componentes educativos correspondientes	Si ()	No ()
Puede ingresar a los componentes educativos correspondientes	Si ()	No ()
Puede ver la(s) tutoría(s) creadas dentro de los componentes educativos correspondientes	Si ()	No ()
Puede ver la información de la(s) tutoría(s) creadas dentro de los componentes educativos correspondientes. <ul style="list-style-type: none"> • Nombre de la tutoría • Fecha de asistencia a la tutoría • Hora de inicio de la tutoría • Hora de fin de la tutoría • Numero de aula asignado • Contenido • Observaciones 	Si ()	No ()
Ha recibido la notificación de la tutoría a través del correo electrónico	Si ()	No ()
Ha podido seleccionar de manera correcta el horario de asistencia a la tutoría. <ul style="list-style-type: none"> • Hora de inicio de la tutoría • Hora de fin de la tutoría 	Si ()	No ()
Ha ingresado de manera correcta el comentario dentro de la tutoría	Si ()	No ()
Ha podido seleccionar y calificar de manera correcta la tutoría	Si ()	No ()
Funciona correctamente la aplicación Gestión de tutorías UTPL	Si ()	No ()

Proceso / Funcionalidad	Funciona correctamente		Detalle del error	Solución
	SI	NO		

Anexo 11. Informe de ejecución de pruebas

**“Análisis y desarrollo de un framework metodológico
para la innovación educativa enfocada a la acción
tutorial docente de la Modalidad Presencial de la
UTPL”
Informe de ejecución de pruebas**

Versión 1.0

[La versión del documento se actualizará según la iteración y la fase del proyecto]

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
02/12/2016	1.0	Versión preliminar como propuesta de desarrollo.	Gabriela Puglla

Información del proyecto

Empresa/Organización	UTPL
Nombre del proyecto	“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”
Fecha de elaboración	02/12/2016
Cliente	UTPL/DGA
Patrocinador	Escuela de Sistemas Informáticos y Computación
Gerente del proyecto	Ing. Fernanda Soto
Analista de Pruebas	Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Tabla de Contenidos

1.	Introducción.....	4
2.	Objetivos.....	4
3.	Requerimientos para la ejecución de las pruebas.....	4
4.	Diseño de casos de prueba.....	4
5.	Ejecución de casos de prueba y reporte de bugs.....	5
6.	Resultado de las pruebas.....	6
6.1	<i>Bugs detectados.....</i>	7
7.	Observaciones.....	9
8.	Conclusiones.....	9
9.	Recomendaciones.....	9
10.	Aprobación de la ejecución de las pruebas.....	9
11.	Anexos.....	9

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Informe de ejecución de pruebas

1. Introducción

Los objetivos del control de calidad constituyen un compromiso de validar una aplicación móvil a través de cada uno de los requerimientos y funcionalidades definidas, con el propósito de identificar incidencias en el funcionamiento de la aplicación.

2. Objetivos

- Validar la aplicación móvil a través de sus funcionalidades establecidas, “Gestión de tutorías UTPL”.
- Para cumplir con este objetivo se involucró un grupo de estudiantes de la utpl y a dos docentes que participaron en el proceso de validación y usabilidad de la aplicación móvil.

3. Requerimientos para la ejecución de las pruebas

- *Requerimientos*
 - Aplicación móvil “Gestión de tutorías UTPL
 - Casos de prueba
- *Ambiente de pruebas*
 - Dispositivos móviles de los usuarios finales en los que como principal requerimiento solicita que el sistema operativo supere la versión de 4.1. de android y 7.0 de iOS.
 - Servicios web solicitados y disponibles de la DGA UTPL, con los que funciona la aplicación móvil.
 - data.utpl.edu.ec/wsqr/móvil
 - <http://carbono.utpl.edu.ec:8080/wscodigosqr/webresources/entidades.qrusuarios/userlogin?usuario=>
 - <https://sica.utpl.edu.ec/auth?user&password>
- Herramientas para pruebas de desempeño (carga y estrés) y funcionalidad de aplicaciones móviles utilizando el archivo ejecutable
 - Monkop¹. Registrado en versión libre por un periodo de 30 días.
 - Testdroid cloud². Registrado en versión libre por un periodo de 30 días.
 - Test object.³ Registrado en versión libre por un periodo de 30 días.

4. Diseño de casos de prueba

Se diseñaron los casos de prueba para validar la aplicación móvil “Gestión de tutorías UTPL” de acuerdo a la documentación entregada por parte del equipo de proyecto a través del documento de especificación de requerimientos y las historias de usuario, las mismas que se detallan en el anexo 7, almacenado en el repositorio Git Taw UTPL:

<https://git.taw.utpl.edu.ec/dimoreno/Tesis/tree/f3a368a0fdbd98c64fa7deafddf51a91d7ba2ab3>.

¹ Monkop : <https://console.monkop.com>

² Testdroid cloud: <https://cloud.testdroid.com/>

³ Test object: <https://app.testobject.com/>

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Requerimiento	Numero de casos de prueba
Ingresar al QR	1
Ver componentes	1
Crear tutoría	1
Editar tutoría	1
Ver turarías creadas	1
Participar en tutorías	1
Comentar tutoría	1
Ver notificaciones	1
Ver participantes	1
Ver comentarios	1
Validar asistencia	1
Calificar tutoría	1
Ver estado de tutoría	1

5. Ejecución de casos de prueba y reporte de bugs

En el proceso de la ejecución de las pruebas se detalla y reporta al equipo de desarrollo, los bugs encontrados a través de la matriz de reporte de issues y bugs (Anexo 8), que fueron corregidos en su totalidad.

Nombre del proyecto	Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL
Fecha comienzo planificada	12-12-2016
Fecha de finalización planificada	02-02-2017
Casos de prueba (Total)	14 casos de prueba.
Casos planificados	14 casos de prueba
Casos exitosos	14 casos de prueba
% avance planificado	14 / 14 casos de prueba. Equivale al 100% del avance planificado.
% avance real	14 / 14 casos de prueba. Equivale al 100% del avance planificado.
% desviación	0% desviación.
Días de desviación	0 días de desviación
Fecha fin pronóstico	Se finalizo en la fecha planificada. 02-02-2017.
Casos con incidencia	No se presentaron casos de prueba con incidencias.
% casos con incidencias	0% de casos de pruebas con incidencias. 0 / 14 casos de prueba con incidencias.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

6. Resultado de las pruebas

Requerimiento	Número de casos de prueba	# ciclo de pruebas
Ingresar al QR	1	1
Ver componentes	1	1
Crear tutoría	1	1
Editar tutoría	1	1
Ver turarías creadas	1	1
Participar en tutorías	1	1
Comentar tutoría	1	1
Ver notificaciones	1	1
Ver participantes	1	1
Ver comentarios	1	1
Validar asistencia	1	1
Calificar tutoría	1	1
Ver estado de tutoría	1	1
Total del proyecto	100%	100%

Situación actual de casos de prueba	
Exitosos	14 / 14 casos de prueba exitosos.
Con defectos	0 / 14 casos de prueba con defectos.
Bloqueados	0 / 14 casos de prueba bloqueados.
Diferidos	0 / 14 casos de prueba diferidos.
Pendientes	0 / 14 casos de prueba pendientes.
Situación actual de defectos	
Reportados	19 defectos reportados por parte de los usuarios finales.
En análisis	19 defectos en análisis.
Descartados	13 defectos descartados
En proceso	6 defectos en proceso que fueron analizados y se han considerado para ser corregidos en una segunda versión de la aplicación móvil.
Corregidos	6 defectos en proceso que fueron analizados y se han considerado para ser corregidos en una segunda versión de la aplicación móvil.
Resultados de la jornada	
Casos del día	13 casos descartados se han considerado como casos del día resueltos como exitosos.
Meta diaria	6 casos se esperan ejecutar cada día, en relación a los 6 casos en proceso.
Observaciones	
Puntos de atención y observaciones	El numero de defectos reportados: en proceso y corregidos, se han considerado resolver en el proceso de desarrollo de la segunda versión de la aplicación móvil Gestión de Tutorías UTPL.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

6.1 Bugs detectados

A continuación se detallan los bugs detectados y reportados a través de los resultados generados en las encuestas de los usuarios que utilizaron la aplicación móvil:

Id Func.	Funcionalidad	Detalle de bugs			Asignado a
		Fecha de ejecución	Detalle de bugs	# de bugs reportados	
1	Ingresar al QR	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
2	Ver componentes	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
3	Crear tutorías	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
4	Ver tutorías creadas	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
5	Editar tutorías	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
6	Ver participantes	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
7	Ver comentarios	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
8	Validar asistencia	19 de diciembre de 2016 al 03 de febrero de 2017	El no enviar mensajes de aviso en el que el usuario debe esperar para validar su asistencia.	1	Equipo de proyecto/ desarrollo
9	Ver notificaciones	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
10	Calificar tutoría	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
11	Participar en tutoría	19 de diciembre de 2016 al 03 de febrero de 2017	<ul style="list-style-type: none"> No se puede seleccionar el horario de la tutoría. Es necesario que se pueda escoger otra fecha para la tutoría. No se puede seleccionar el horario de la tutoría. No se ha creado la participación a la tutoría. No se guardan los datos. 	5	Equipo de proyecto/ desarrollo

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

12	Comentar tutoría	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		
13	Ver estado de tutoría	19 de diciembre de 2016 al 03 de febrero de 2017	No se reportaron bugs		

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

7. Observaciones

- La aplicación móvil “Gestión de Tutorías Utpl” se realizó en un entorno real con alumnos que están matriculados en este periodo académico.

8. Conclusiones

- A través de cada una de las pruebas se ejecutó el proceso de control de calidad en un 100% del primer ciclo de pruebas.
- En este proceso de pruebas el equipo de control de calidad establecido en el MTCC con el equipo de desarrollo lo realizaron de manera organizada cumpliendo con los objetivos establecidos que tiene este proyecto.

9. Recomendaciones

- La participación de los proveedores y clientes definidos en el MTCC desde el inicio de desarrollo del proyecto permite obtener un conocimiento más claro de los requerimientos en el proyecto.
- Dar a conocer a todo el equipo del proyecto MTCC la planificación de cada una de las fases de liberación de los requerimientos del proyecto.
- Cada uno de los artefactos definidos en el MTCC estén revisados y aprobados por el proveedor y el cliente antes de su liberación con el propósito de reducir retrasos en la ejecución de cada una de las fases de desarrollo del proyecto.

10. Aprobación de la ejecución de las pruebas

Aprobado y revisado por:	[Equipo de dirección de Mobile-D]	Firma: Diego Moreno Israel Rey	Fecha:
Evaluador por:	[Evaluador de control de calidad]	Firma Gabriela Puglla	Fecha: 06-02-2017
Elaborado por:	Evaluador de control de calidad]	Firma: Gabriela Puglla	Fecha 06-02-2017

11. Anexos

- Evidencias de la ejecución de las pruebas. Matriz de reporte y bugs.
- Encuestas de los usuarios finales
- Informe final de las pruebas generado en la herramienta www.e-encuestas.com

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pruebas aplicadas en el ambiente de desarrollo de aplicaciones móviles

Pruebas funcionales

Permite, asegurar que las funciones de la aplicación cumplan con las especificaciones y los requerimientos que darán solución a la necesidad del cliente. Definido en dos tipos de usuarios: Docentes y estudiantes.

Rol: Docente

Funcionalidad		Funciona correctamente		Rectificaciones/Modificaciones
		SI	NO	
Ingresar tutoría	El nombre de usuario y contraseña del usuario se han ingresado correctamente	X		No es necesario
Ver componentes	Puede ver todos los componentes educativos correspondientes	X		No es necesario
Crear tutoría	Puede ingresar de manera correcta a la ventana de crear tutoría dentro del componente educativo seleccionado	X		No es necesario
	Nombre de la tutoría	X		
	Fecha de asistencia a la tutoría	X		
	Horario de inicio de tutoría	X		
	Horario de fin de la tutoría	X		
	El número de aula para realizar la tutoría	X		
	Contenido a tratar en la tutoría	X		
	Observaciones a considerar en la tutoría	X		
Ver tutoría	Se ha creado correctamente la tutoría	X		No es necesario
	Se puede ver la(s) tutoría(s) creadas dentro del componente educativo seleccionado.	X		
Editar tutorías	Se puede ingresar a la tutoría creada	X		No es necesario
	Se puede editar y/o modificar un campo seleccionado de la tutoría creada			
	Nombre de la tutoría			
	Fecha de asistencia a la tutoría			
	Horario de inicio de tutoría			
	Horario de fin de la tutoría			
	El número de aula para realizar la tutoría			
	Contenido a tratar en la tutoría			
Observaciones a considerar en la tutoría				
Ver participantes	Se puede ver los campos de la(s) tutoría(s) modificadas dentro del componente educativo seleccionado	X		No es necesario
	Se puede ver de manera correcta la lista de los participantes que forman parte del componente educativo			
	Se ha enviado correctamente la notificación a los			

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

	participantes a través del correo electrónico			
Ver comentarios	Se puede ver de manera correcta las dudas y temáticas por la que requieren asistir los participantes de la tutoría que forman parte del componente educativo	X		No es necesario
Validar asistencia	Se ha validado de manera correcta la asistencia del participante en la tutoría	X		No es necesario

Rol: Estudiante

Funcionalidad		Funciona correctamente		Rectificaciones/ Modificaciones
		SI	NO	
Ingresar tutoría	El nombre de usuario y contraseña del usuario se han ingresado correctamente	X		No es necesario
Ver componentes	Puede ver todos los componentes educativos correspondientes	X		No es necesario
Ver tutoría	Puede ingresar de manera correcta a la ventana de crear tutoría dentro del componente educativo seleccionado	X		No es necesario
	Nombre de la tutoría	X		
	Fecha de asistencia a la tutoría	X		
	Horario de inicio de tutoría	X		
	Horario de fin de la tutoría	X		
	El número de aula para realizar la tutoría	X		
	Contenido a tratar en la tutoría	X		
Participar en tutoría	Observaciones a considerar en la tutoría	X		No es necesario
	Ha recibido la notificación de la tutoría a través del correo electrónico	X		
	Ha podido seleccionar de manera correcta el horario de asistencia a la tutoría.	X		
	Hora de inicio de la tutoría	X		
Comentar tutorías	Hora de fin de la tutoría	X		No es necesario
	Ha ingresado de manera correcta el comentario dentro de la tutoría	X		
Calificar tutorías	Ha podido seleccionar y calificar de manera correcta la tutoría	X		No es necesario

Como se ha mostrado en la tabla las pruebas de la aplicación móvil “Gestión de tutorías UTPL” en los distintos dispositivos han resultado satisfactorios para todas las funcionalidades implementadas.

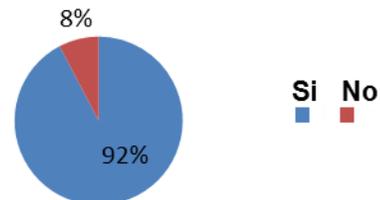
“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pruebas de instalación

Pregunta: El proceso de instalación es

Fácil () Difícil ()

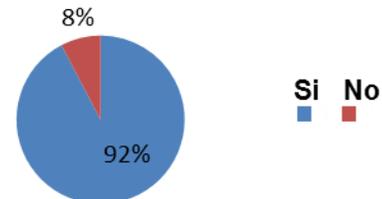
	Fácil	Difícil	
Docente	2	0	
Estudiantes	34	3	
Total de usuarios	36	3	39
Porcentaje	92%	8%	100%



Pregunta: La instalación de la aplicación se ha ejecutado de manera correcta

Si () No ()

	Si	No	
Docente	2	0	
Estudiantes	34	3	
Total de usuarios	36	3	39
Porcentaje	92%	8%	100%



El 92% de los usuarios han realizado una instalación exitosa de la aplicación móvil “Gestión de tutorías UTPL” y se ejecuta correctamente, en relación al 8% que no han realizado este proceso; es decir que las pruebas de instalación se han validado exitosamente. En cada una de las 3 herramientas web utilizadas para ejecutar las pruebas utilizando el archivo ejecutable (*.apk) la aplicación móvil “Gestión de tutorías UTPL” se ha instalado de manera correcta

Pruebas de Compatibilidad

Consisten en validar la compatibilidad de la aplicación con: La aplicación móvil “Gestión de tutorías UTPL” especifica como uno de los principales requerimientos para su correcto funcionamiento verificar la versión del sistema operativo del dispositivo móvil en Android e iOS.

Modelo de dispositivo móvil	Versión del sistema operativo del dispositivo móvil	Numero de dispositivos móviles
Android	> 4.0	36
iOs	> 7.0	3
Total de dispositivos móviles:		39

El 100% de los dispositivos móviles utilizados por los usuarios soportan las principales características y requerimientos para el correcto funcionamiento de la aplicación móvil ya que en las Pruebas de usabilidad (encuesta) se detallan el modelo y versión del sistema operativo de cada uno de los dispositivos móviles. En cada una de las 3 herramientas web utilizadas para ejecutar las pruebas utilizando el archivo ejecutable (*.apk) la aplicación móvil “Gestión de tutorías UTPL” es compatible con el 100% de los dispositivos móviles disponibles en la versión libre.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pruebas de interrupción

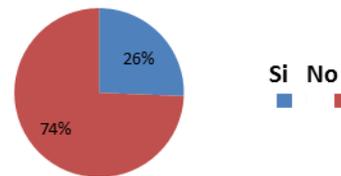
Las pruebas de interrupción se incluyen y funcionan ejecutando otra actividad o suceso, por ejemplo:

- Mensajes de error

Pregunta: ¿Se han presentado mensajes de error al momento de ejecutar la aplicación?

Si () No ()

	Respuestas		
	Si	No	
Docente	1	1	
Estudiantes	9	28	
Total de usuarios	10	29	39
Porcentaje	26%	74%	100%



El 74 % de los usuarios no han reportado ningún tipo de mensajes de error cuando utilizaron la aplicación móvil “Gestión de tutorías UTPL”; en comparación del 26% de los usuarios que indican que si se han presentado mensajes de error al momento de ejecutar las funcionalidades relacionados con “detención inesperada”.

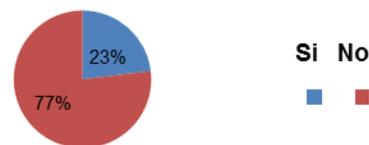
Este tipo de errores han sido reportados y evaluados por el equipo de desarrollo, y para esta versión se han descartado ya que no afectan en la ejecución y validación de la aplicación móvil.

- Consumo de batería del dispositivo móvil, Batería baja / En estado crítico.

Pregunta: ¿Se ha producido un considerable consumo de batería al ejecutar la aplicación?

Si () No ()

	Respuestas		
	Si	No	
Docente	0	2	
Estudiantes	9	28	
Total de usuarios	9	30	39
Porcentaje	23%	77%	100%



En el uso de la aplicación móvil “Gestión de tutorías UTPL”, el 77% de los usuarios han considerado que no existe un consumo considerable de la batería del dispositivo móvil, y se logra ejecutar en su totalidad la aplicación si la batería del dispositivo es baja o en estado crítico. En comparación del 9% de los usuarios que consideran que al ejecutar la aplicación móvil produce un consumo de batería pero que no afecta en que permanezca instalada y disponible en el dispositivo para ser utilizada.

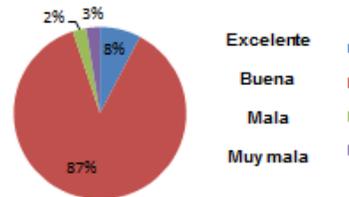
- Caída de la conexión con la red

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pregunta: La calidad de la conexión con los servicios web/internet es:

- Excelente (0 – 0.30 seg)
- Buena (0.30 – 1.00 seg)
- Mala (1.00 – 2.00 seg)
- Muy mala (≥ 2.00 seg)

	Respuestas				
	Excelente	Buena	Mala	Muy mala	
Docente	1	1	0	0	
Estudiantes	2	33	1	1	
Total de usuarios	3	34	1	1	39
Porcentaje	8%	87%	2%	3%	100%



El 84% de los usuarios considera que la calidad de la conexión con los servicios de web/internet para el correcto funcionamiento de la aplicación móvil es buena, del lado de la aplicación y del servidor, en este caso no se considera los demás porcentajes para ejecutar este tipo de prueba y considerar como exitosa.

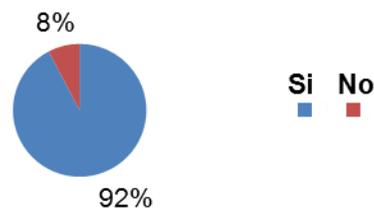
Pruebas de Seguridad

En este caso las pruebas de seguridad validan los datos de los usuarios para ingresar al sistema y poder ejecutar la aplicación móvil Gestión de tutorías UTPL.

Pregunta: Los datos insertados del usuario se han almacenado correctamente:

Si () No ()

	Respuestas		
	Si	No	
Docente	2	0	
Estudiantes	34	3	
Total de usuarios	36	3	39
Porcentaje	92%	8%	100%

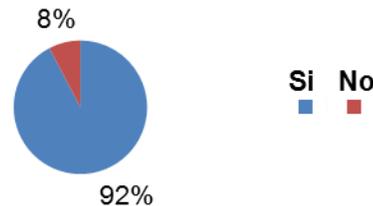


El 92% de los usuarios han utilizado e ingresado de manera correcta a la aplicación móvil “Gestión de tutorías UTPL” con los datos de nombre de usuario y contraseña utilizados en el entorno EVA y el 8% de los usuarios no han utilizado la aplicación móvil desde el proceso de instalación.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pregunta: El usuario registrado puede ejecutar todas las funcionalidades
 Si () No ()

	Respuestas		
	Si	No	
Docente	2	0	
Estudiantes	34	3	
Total de usuarios	36	3	39
Porcentaje	92%	8%	100%



El 92% de los usuarios que han utilizado e ingresado de manera correcta a la aplicación móvil “Gestión de tutorías UTPL” han ejecutado satisfactoriamente todas las funcionalidades asignadas a su rol y el 8% de los usuarios no han podido ejecutar la aplicación móvil incluso desde el proceso de instalación.

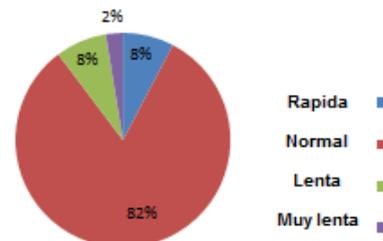
Pruebas de desempeño (carga y estrés)

Se realizan para determinar la rapidez y la capacidad para ejecutar una tarea de la aplicación móvil, en condiciones particulares de trabajo.

- Desempeño.
- Utilización de procesador, memoria y espacio de almacenamiento.
- Pruebas de carga del lado de servidor (pruebas de estrés).

El tiempo de ejecución de las funcionalidades

- Rápida () (0 – 0.30 seg)
- Normal () (0.30 – 1.00 seg)
- Lenta () (1.00 – 2.00 seg)
- Muy lenta () (≥ 2.00 seg)



	Respuestas				
	Rápida (0 – 0.30 seg)	Normal (0.30 – 1.00 seg)	Lenta (1.00 – 2.00 seg)	Muy lenta (≥ 2.00 seg)	
Docente	1	1	0	0	
Estudiantes	2	31	3	1	
Total de usuarios	3	32	3	1	39
Porcentaje	8%	82%	8%	2%	100%

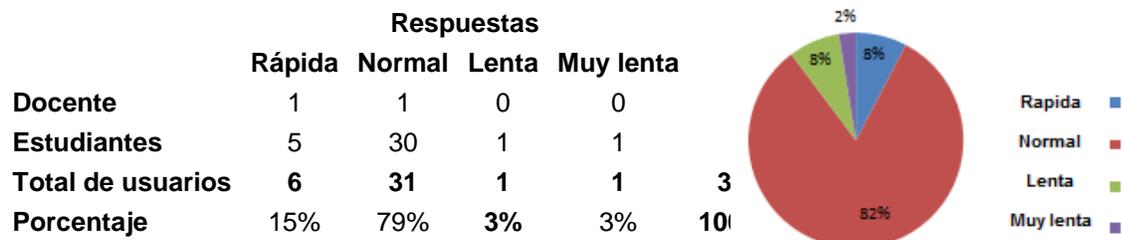
El 82% de los usuarios que han utilizado la aplicación móvil validan el tiempo de ejecución de las funcionalidades como normal en relación al tiempo de respuesta establecido de 0.30 – 1.00 seg. Y se considera que su funcionamiento es apropiado y correcto dentro del tiempo establecido. Es decir el rendimiento de cada una de las funcionalidades de la aplicación móvil depende del Web-Service.

En cada una de las 3 herramientas web utilizadas para ejecutar las pruebas utilizando el archivo ejecutable (*.apk) la aplicación móvil “Gestión de tutorías UTPL” el tiempo utilizado es normal en la ejecución de lado de los usuarios finales.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

El tiempo de ejecución de múltiples funcionalidades

- Rápida () (0 – 0.30 seg)
- Normal () (0.30 – 1.00 seg)
- Lenta () (1.00 – 2.00 seg)
- Muy lenta () (≥ 2.00 seg)



El 79% de los usuarios que han utilizado la aplicación móvil validan el tiempo de ejecución de las funcionalidades como normal en relación al tiempo de respuesta establecido de 0.30 – 1.00 seg. Y consideran que su funcionamiento es apropiado y correcto dentro del tiempo establecido, relacionando el tiempo de respuesta del lado del servidor. Cabe recordar que el tiempo de carga de cada uno de los servicios depende de la conexión de datos del dispositivo. Puesto que los datos se deben consultar desde el Webservice de pruebas de la UTPL.

Pruebas de Interfaz con el usuario (UI)

Buscan evaluar la interfaz con el usuario en relación con estándares para la interacción humano-maquina aceptable. Entre las pruebas que podemos realizar tenemos:

- **Verificación de logo y nombre de la aplicación**

En cada uno de los dispositivos móviles se ha instalado de manera correcta la aplicación “Gestión de tutorías UTPL” y por lo tanto el icono y logo se visualizan de manera correcta.

- **Dispositivos con pantalla táctil y sin pantalla táctil.**

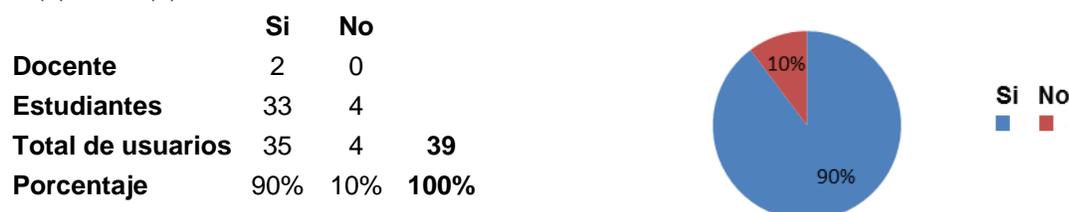
El 100% de los dispositivos móviles en el que se ha utilizado la aplicación “Gestión de tutorías UTPL” es de pantalla táctil y se ajusta al tamaño correctamente.

- **Ajuste y rotación de la aplicación en relación a la dirección del dispositivo (horizontal/vertical).**

En este caso para utilizar la aplicación móvil se ha restringido la rotación únicamente para posición vertical y facilitar la navegación de cada una de las funcionalidades de la aplicación móvil.

Pregunta: Le agrada la apariencia de la aplicación

Si () No ()

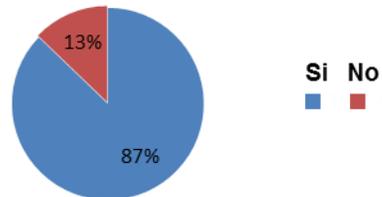


“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

El 90% de los usuarios que han utilizado la aplicación móvil le agrada la apariencia de la aplicación en relación a la conformidad del diseño la misma que sigue los estándares de estilo propio en cuanto al uso de iconos y colores de la organización “UTPL”.

Pregunta: Le parece que la posición y funcionalidad de los botones/iconos son adecuados
Si () No ()

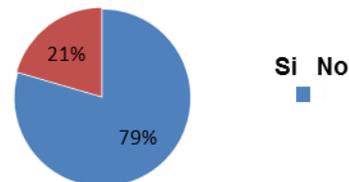
	Si	No	
Docente	2	0	
Estudiantes	32	5	
Total de usuarios	34	5	39
Porcentaje	87%	13%	100%



El 87% de los usuarios le parece que la navegación de la aplicación es adecuada a través de los objetos (botones/iconos) que se presentan en cada ventana y su acceso (táctil).

Pregunta: Las pantallas contiene los datos/información y funciones suficientes para ejecutar la aplicación.
Si () No ()

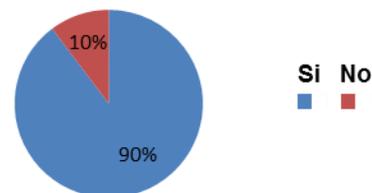
	Si	No	
Docente	2	0	
Estudiantes	29	8	
Total de usuarios	31	8	39
Porcentaje	79%	21%	100%



El 79% de los usuarios consideran que cada una de las pantallas (ventanas) contiene la información suficiente para ejecutar la aplicación en su totalidad.

Pregunta: La navegación en la aplicación es
Fácil () Difícil ()

	Fácil	Difícil	
Docente	2	0	
Estudiantes	33	4	
Total de usuarios	35	4	39
Porcentaje	90%	10%	100%



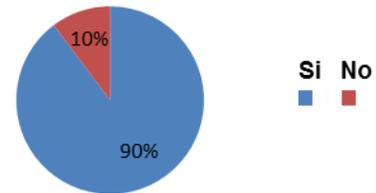
EL 90% de los usuarios aprueba que la navegación en la aplicación sea fácil con respecto a la secuencia de cada una de las ventanas y la información que presenta para ejecutar de manera correcta la aplicación móvil.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pregunta: La documentación la infografía de la aplicación es fácil de entender

Si () No ()

	Fácil	Difícil	
Docente	2	0	
Estudiantes	33	4	
Total de usuarios	35	4	39
Porcentaje	90%	10%	100%

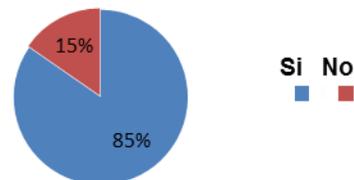


El 90% de los usuarios evalúa la documentación del usuario con respecto al uso de la aplicación móvil y especifica que la información que se presenta es fácil de entender.

Pregunta: La aplicación móvil se ha cerrado de manera correcta

Si () No ()

	Si	No	
Docente	2	0	
Estudiantes	31	6	
Total de usuarios	33	6	39
Porcentaje	85%	15%	100%

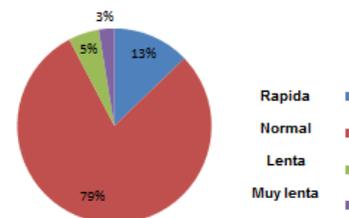


El 85% de los usuarios han cerrado de manera correcta la aplicación móvil independientemente si han ejecutado todas las funcionalidades de la aplicación. En relación al 15% de los usuarios que han reportado que la aplicación se ha cerrado de manera inesperada.

Pregunta: El inicio y reinicio para ejecutar la aplicación

- Rápida () (0 – 0.30 seg)
- Normal () (0.30 – 1.00 seg)
- Lenta () (1.00 – 2.00 seg)
- Muy lenta () (≥ 2.00 seg)

	Respuestas				
	Rápida	Normal	Lenta	Muy lenta	
Docente	1	1	0	0	
Estudiantes	4	30	2	1	
Total de usuarios	5	31	2	1	39
Porcentaje	13%	79%	5%	3%	100%



El 79% de los usuarios han iniciado y reiniciado la aplicación móvil de manera correcta en un tiempo de ejecución normal (0.30 – 1.00 seg) que consideran como adecuado para ejecutar la aplicación.

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

Pruebas de Movilidad y Conectividad

Pruebas de Movilidad: Consiste en probar el desempeño de la aplicación cuando el dispositivo móvil se encuentra en movimiento, con una persona o vehículos. Esto es más importante en aplicaciones que utilizan la localización para mostrar información relevante al usuario. Posibles pruebas a realizar:

- Geo localización
- Atravesar zonas de la red móvil (por ejemplo ir de una celda a otra).
- Señal de GPS, Wi-Fi o celular débil o intermitente (se interrumpe/corta).

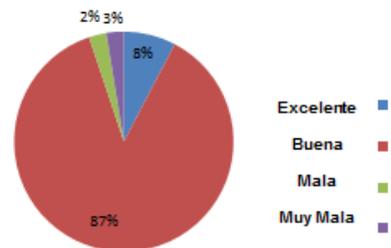
Pruebas de Conectividad: Desempeño de la aplicación al conectarse a las redes en distintos protocolos y distintas condiciones, por ejemplo:

- Wi-Fi.
- Bluetooth.
- Red analógica, 3G o 4G.
- Señales de distinta intensidad (sin señal, conexión intermitente, señal fuerte, intensidad de señal variable).

Pregunta: La calidad de la conexión con los servicios web/internet es:

- Excelente () (0 – 0.30 seg)
- Buena () (0.30 – 1.00 seg)
- Mala () (1.00 - 2.00seg)
- Muy mala (\geq 2.00 seg)

	Respuestas				
	Excelente	Buena	Mala	Muy mala	
Docente	1	1	0	0	
Estudiantes	2	33	1	1	
Total de usuarios	3	34	1	1	39
Porcentaje	13%	79%	5%	3%	100%



El 92% de los usuarios consideran que la calidad de la conexión con los servicios web/internet es normal (0.30 – 1.00 seg) en relación a la conectividad y movilidad (señal de la red wifi) al momento de utilizar la aplicación móvil

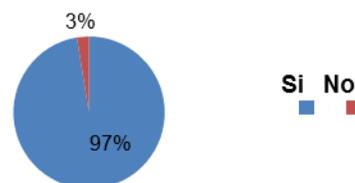
Pruebas de usabilidad

Notificaciones de social media. Correo electrónico.

Pregunta: Ha recibido la notificación de la tutoría a través del correo electrónico

Si () No ()

	Respuestas		
	Si	No	
Docente	2	0	
Estudiantes	36	1	
Total de usuarios	38	1	39
Porcentaje	97%	3%	100%



“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Pruebas- Control de calidad	Fecha: 02/12/2016
Informe de ejecución de pruebas	

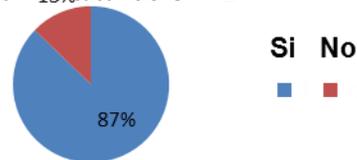
En este caso especifica el uso de hardware/software adicional para el correcto funcionamiento de la aplicación móvil.” Gestión de tutorías UTPL” utiliza necesaria y específicamente utiliza la aplicación de gestor de correo electrónico que permite enviar una notificación de confirmación de que los usuarios han utilizado la aplicación.

El 97% de los usuarios han utilizado correctamente el gestor de correo electrónico en función de la aplicación móvil.

Pregunta: Funciona correctamente la aplicación móvil “Gestión de tutorías UTPL”

Si () No ()

	Si	No	
Docente	2	0	
Estudiantes	32	5	
Total de usuarios	34	5	39
Porcentaje	87%	13%	100%



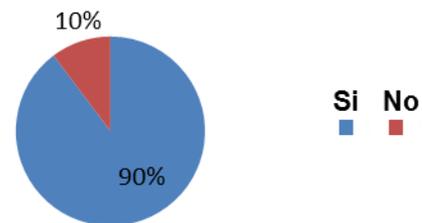
El 87% de los usuarios ha concluido que la aplicación móvil funciona correctamente desde el proceso de instalación y cada una de las funcionalidades que la contiene.

Pruebas de aceptación

Pregunta: El uso de la aplicación es una experiencia

Fácil () Difícil ()

	Fácil	Difícil	
Docente	2	0	
Estudiantes	33	4	
Total de usuarios	35	4	39
Porcentaje	90%	10%	100%



El 90% de los usuarios determinan que el uso de la aplicación móvil es una experiencia fácil. Aceptan continuar con el uso de la aplicación ya que satisfacen con las necesidades de los usuarios en relación a la complejidad de uso.

Anexo 12. Informe de cierre de proceso

**“Análisis y desarrollo de un framework metodológico
para la innovación educativa enfocada a la acción
tutorial docente de la Modalidad Presencial de la
UTPL”**

**Acta de cierre del proyecto, fase, proceso.
Proceso de cierre del control de calidad**

Versión 1.0

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 12/12/2016
Informe de cierre del proyecto	

Historial de Revisiones

Fecha	Versión	Descripción	Autor
12/12/2016	1.0	Versión preliminar como propuesta de desarrollo.	Gabriela Puglla

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 12/12/2016
Informe de cierre del proyecto	

Tabla de Contenidos

1.	Información de proyecto	4
1.1	<i>Datos</i>	4
1.2	<i>Patrocinadores</i>	4
2.	Razón de cierre	4
2.1	<i>Aceptación de los productos o entregables</i>	4
3.	Aprobaciones	5

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 12/12/2016
Informe de cierre del proyecto	

Informe de cierre del proyecto

1. Información de proyecto

1.1 Datos

Empresa / Organización	UTPL
Proyecto	“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”
Fecha de preparación	
Cliente	Vicerrectorado de-UTPL
Patrocinador principal	UTPL
Gerente de Proyecto	Ing. Fernanda Soto, Ing, Audrey Romero

1.2 Patrocinadores

Nombre	Cargo	Departamento / División
Ing. Audrey Romero	Director	“Marco de trabajo para el control de calidad en el desarrollo de aplicaciones móviles bajo el uso de metodologías ágiles”
Ing. Fernanda Soto	Director	“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”
Vicerrectorado de UTPL	Vicerrectorado de UTPL	Vicerrectorado de UTPL

2. Razón de cierre

En la siguiente lista se certifica las razones del cierre del proyecto o fase, específicamente si se entregó todos los componentes del producto, si algunos componentes fueron entregados y otros cancelados, o si se cancelaron todos los entregables

Por medio de la presente, se da cierre formal al proyecto, por las razones especificadas en la siguiente ficha:

Marcar con una “X” la razón de cierre:

Entrega de todos los productos de conformidad con los requerimientos del cliente.	X
Entrega parcial de productos y cancelación de otros de conformidad con los requerimientos del cliente.	
Cancelación de todos los productos asociados con el proyecto.	

2.1 Aceptación de los productos o entregables

A continuación se establece cuales entregables de proyecto han sido aceptados, haciendo referencia a las entregables, que pueden ser documentos o componentes del producto:

“Análisis y desarrollo de un framework metodológico para la innovación educativa enfocada a la acción tutorial docente de la Modalidad Presencial de la UTPL”	Versión: 1.0
Visión	Fecha: 12/12/2016
Informe de cierre del proyecto	

Entregable	Aceptación (Si o No)	Observaciones
Plan del proyecto	Si	-
Documento de especificación de requerimientos	Si	-
Prototipos del producto (Diseño e interfaces de usuario)	Si	-
Historias de usuario, Diagramas de clases, casos de uso	Si	-
Documento de estándares de código desarrollado/Manual de usuario del desarrollador	Si	-
Plantilla de checklist	Si	-
Plan de calidad	Si	-
Plan de medición	Si	-
Plan de pruebas	Si	-
Casos de pruebas	Si	-
Matriz de reporte de bugs e issues	Si	-
Informe de pruebas del sistema	Si	-
Informe de cierre del proceso de control de calidad	Si	-

Para cada entregable aceptado, se da por entendido que:

- El entregable ha cumplido los criterios de aceptación establecidos en la documentación de requerimientos y definición de alcance.
- Se ha verificado que los entregables cumplen los requerimientos.
- Se ha validado el cumplimiento de los requerimientos funcionales y de calidad definidos.
- Se ha realizado la transferencia de conocimientos y control al área operativa.
- Se ha concluido el entrenamiento que se definió necesario.
- Se ha entregado la documentación al área operativa.

Se autoriza al Gerente de Proyecto a continuar con el cierre formal del proyecto o fase, lo cual deberá incluir:

- Evaluación post-proyecto o fase.
- Documentación de lecciones aprendidas.
- Liberación del equipo de trabajo para su reasignación.
- Cierre de todos los procesos de procura y contratación con terceros.
- Archivo de la documentación del proyecto.

Una vez concluido el proceso de cierre, el Patrocinador (Sponsor) del proyecto deberá ser notificado para que el Gerente de Proyectos sea liberado y reasignado.

3. Aprobaciones

Patrocinador	Cargo	Fecha	Firma
Ing. Audrey Romero	Director		
Ing. Fernanda Soto	Director		
Vicerrectorado de UTPL			