



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

ÁREA TÉCNICA

TÍTULO DE INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Diseño de aplicaciones interactivas de carácter informativo y educativo (T-Learning) para su aplicación en el Sistema de Televisión Digital Terrestre del Ecuador.

TRABAJO DE TITULACIÓN

AUTOR: *Ambuludí León, John Paúl*

DIRECTOR: *Samaniego Armijos, Hernán Adolfo, M.Sc.*

LOJA – ECUADOR

2016



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>

2016

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN

Magíster.

Hernán Adolfo Samaniego Armijos.

DOCENTE DE LA TITULACIÓN

De mi consideración:

*El presente trabajo de titulación: **Diseño de aplicaciones interactivas de carácter informativo y educativo (T-Learning) para su aplicación en el Sistema de Televisión Digital Terrestre del Ecuador** realizado por **Ambuludí León John Paúl**, ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo.*

Loja, octubre de 2016

f)

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

*“ Yo **Ambuludí León John Paúl** declaro ser autor (a) del presente trabajo de titulación: **Diseño de aplicaciones interactivas de carácter informativo y educativo (T-Learning) para su aplicación en el Sistema de Televisión Digital Terrestre del Ecuador**, de la Titulación de **Electrónica y Telecomunicaciones**, siendo **Hernán Adolfo Samaniego Armijos** director (a) 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: *Ambuludí León John Paúl*

Cédula: **1104933633**

DEDICATORIA

A Dios mi Padre Celestial ya que sin su bendición no hubiera sido posible llegar hasta este momento, gracias al don de la fe que me supo regalar pude vencer las dificultades que se han presentado. A mis padres que me han enseñado desde pequeño a ser una persona fuerte y luchadora, pero en especial, a mi Madre, mi Marielita, siempre la he sentido conmigo bien cerca aunque se me haya ido, sé que junto a Dios, ella me ha derramado bendiciones en la salud, en el trabajo, en el estudio y sobre todo me ha regalado la fortaleza necesaria que he necesitado para cumplir con este trabajo y muchos objetivos más.

A mis abuelitos ya que gracias a sus enseñanzas y apoyo incondicional he podido cumplir con este objetivo importante en mi vida. A mi hermanita que ha sido un pilar fundamental para salir adelante y cumplir con esta meta.

A mi hija Marielita, desde que nació llegó una bendición enorme a mi vida, cada día me ha llenado de enorme felicidad con su sonrisa, sus travesuras y amor, son motivos que me han dado más fuerza y valor para luchar, salir adelante, y saber que tengo a alguien que se ha convertido en mi adoración y mi motor para ser vencedor ante las dificultades que se me presenten.

A mi familia, quienes me han sabido guiar y aconsejar para hacer mejorar las cosas, por las alegrías y momentos que siempre llenan el alma.

A mis amigos y compañeros, que estuvieron apoyándome siempre y que son fundamentales para llegar a alcanzar este objetivo.

AGRADECIMIENTO

A mi madrecita Mariela, ya que gracias a ella he aprendido a ser una persona fuerte y vencedora, me ha sabido guiar por el buen camino y siempre me ha enseñado a vivir junto con los problemas vencéndolos y superándolos para conseguir mis metas propuestas. A mis abuelitos que me acogieron cuando más los necesitaba, y aunque mi madre se fue antes de este mundo, aun quedaron ellos que me guiaron y apoyaron incondicionalmente.

A mi hermana que con sus ocurrencias y apoyo me ha sabido apoyar en todo, me ha brindado su amor y cariño alentándome a no dejar de soñar y apoyándonos mutuamente para superar cualquier adversidad.

A mi hija, mi Mariela, mi princesa, mi vida, mi motor, mi todo, gracias a ella he podido alentarme a mí mismo y saber hacer las cosas, sabiendo que todo lo que hago en mi vida profesional y personal siempre beneficie a la bendición que me llevo con anticipación pero a la vez que lleno mi vida.

A mi familia, que han sabido llenar de amor y felicidad todos mis años de vida, que me ha sabido aconsejar y halar las orejas cuando lo he necesitado para mejorar.

A mis profesores a quienes les debo muchos años de enseñanza y conocimiento, gracias a su paciencia y apoyo que me brindaron para llegar hasta este momento.

A mi director y amigo, Hernán Adolfo Samaniego Armijos, por guiarme en cada paso de este proyecto innovador.

A mis amigos quien han formado parte de las decisiones que he tomado en este caminar, por sus consejos y buenos momentos en las buenas y en las malas gracias, no podría nombrar a todos ya que la lista sería interminable.

ÍNDICE DE ABREVIATURAS

- AAC:** *Advanced Audio Compression – Codificador de Audio Avanzado.*
- ACAP:** *Advanced Common Application Platform – Plataforma de Aplicaciones Comunes Avanzadas.*
- ACATS:** *Advisory Committe on Advanced Television Systems – Comisión Asesora sobre Servicios de Televisión Avanzada.*
- ADSL:** *Asymmetric Digital Subscriber Line – Línea de Abonado Digital Asimétrica, tecnología de acceso a internet de banda ancha.*
- API:** *Application Programming Interface – Interfaz de Programación de Aplicaciones.*
- ARIB:** *Association of Radio Industries and Bussinesses – Asociación de Industrias y Empresas de Radiocomunicaciones.*
- AT&T:** *American Telephone and Telegraph Corporation – Corporación Estadounidense de Teléfono y Telégrafo.*
- ATSC:** *Advanced Television System Committee – Comité de Sistemas de Televisión Avanzada.*
- AVC:** *Advanced Video Coding – Códec de Vídeo Avanzado.*
- BML:** *Broadcast Markup Languaje – Lenguaje de Marcado de Difusión, estándar basado en XML como una especificación de difusión de datos para la televisión digital de radiodifusión.*
- BST-OFDM:** *Band Segmented Transmission Orthogonal Frequency Division Multiplexing – Banda de Transmisión Segmentada Multiplexación por División de Frecuencias Ortogonales.*
- CCREA:** *Asociación de Canales Comunitarios y Regionales del Ecuador.*
- COFDM:** *Coded Orthogonal Frequency Division Multiplexing – Multiplexación por División de Frecuencia Ortogonal Codificada.*
- CONARTEL:** *Consejo Nacional de Radiodifusión y Televisión.*
- CONATEL:** *Consejo Nacional de Telecomunicaciones.*
- DASE:** *Digital Television Applications Software Environment – Aplicación de Televisión Digital Mediante Software, norma del middleware americano desarrollado por la ATSC.*
- Dolby Digital AC-3:** *Nombre comercial para una serie de tecnologías de compresión de audio en su versión AC-3.*
- DQPSK:** *Differential Quadrature Phase-Shift Keying – Modulación Diferencial por Desplazamiento de Fase en Cuadratura.*
- DTMB:** *Digital Terrestrial Multimedia Broadcasting – Transmisión Digital Terrestre de Multimedia.*
- DVB-T:** *Digital Video Broadcasting-Terrestrial – Difusión de Vídeo Digital-Terrestre.*
- EPG:** *Electronic Program Guide – Guía Electrónica de Programas.*

ES: *Elementary Stream – Flujo elemental, es un protocolo de comunicación que contiene un solo tipo de datos (audio, vídeo o subtítulos).*

EWS: *Emergency Warning Systems – Sistema de Alerta Temprana.*

FCC: *Federal Communications Commission – Comisión Federal de Comunicaciones, regulación de las telecomunicaciones interestatales e internacionales por radio, televisión, redes inalámbricas, teléfonos satélite y cable.*

FEC: *Forward Error Correction – Corrección de errores hacia adelante, permite la corrección en el receptor sin retransmisión de la información original.*

FPGA: *Field Programmable Gate Array – Arreglo de Compuertas Programables en Campo.*

Ginga: *middleware abierto del sistema nipo-brasileño de televisión digital ISDB-Tb.*

GPLv2: *General Public License – Licencia Pública General versión 2.*

GPON: *Gigabit-capable Passive Optical Network – Red Óptica Pasiva con Capacidad de Gigabit, tecnología de acceso de telecomunicaciones que utiliza fibra óptica para llegar hasta el suscriptor.*

H.264 (MPEG-4 parte 10): *superior a MPEG-2 y H.263 (MPEG-4 parte 2), norma que define un códec de vídeo de alta compresión.*

HCJB: *Hoy Cristo Jesús Bendice. Primera estación televisora religiosa del Ecuador.*

HD: *High Definition – Alta definición.*

HDMI: *High-Definition Multimedia Interface – Interfaz Multimedia de Alta Definición.*

HDTV: *High Definition Television – Televisión de Alta Definición.*

HE-AAC: *High Efficiency Advanced Audio Coding – Codificador Avanzado de Audio de Alta Eficiencia.*

HFC: *Hybrid Fibre Coaxial – Híbrido de Fibra-Coaxial, red de fibra óptica que incorpora tanto fibra óptica como cable coaxial para crear una red de banda ancha.*

HTML: *HyperText Markup Lenguaje – Lenguaje de Marcas de Hipertexto.*

IIOP: *Internet Inter Orb Protocol – Protocolo de Internet entre ORBs*

ISDB-T: *Integrated Services Digital Broadcasting-Terrestrial – Radiodifusión Digital de Servicios Integrados-Terrestre.*

ISDB-Tb: *Integrated Services Digital Broadcasting-Terrestrial Brazilia version - Radiodifusión Digital de Servicios Integrados-Terrestre versión Brasileña.*

ITU: *International Telecommunication Union – Unión Internacional de Telecomunicaciones, regula las telecomunicaciones a nivel internacional.*

JDBC: *Java Database Connectivity – Conectividad con Bases de Datos desde Java.*

JPEG: *Joint Photographic Experts Group – Grupo Conjunto de Expertos en Fotografía.*

LDPC: *Low-Density Parity-Check – Comprobación de Paridad de Baja Densidad, código de corrección de error lineal que permiten transmitir un mensaje por un canal de comunicaciones ruidoso.*

MFN: *Multiple Frequency Network – Red de Frecuencia Múltiple.*

MHP: *Multimedia Home Platform – Plataforma Multimedia del Hogar, middleware diseñado por DVB y estandarizado por la ETSI (Instituto Europeo de Normas de Telecomunicaciones).*

MIMO: *Multiple input Multiple output – Múltiple entrada Múltiple salida.*

MPEG: *Moving Picture Experts Group – Grupo de Expertos en Imágenes en Movimiento.*

MUSE: *Multiple Sub-Nyquist Sampling Encoding – Codificación Múltiple de Muestreo Sub-Nyquist.*

NCL: *Nested Context Language – Lenguaje de Contextos Anidados.*

NCM: *Nested Context Model – Modelo de Contextos Anidados.*

NKH: *Nippon Hoso Kyokai - Asociación de Radiodifusión de Japón.*

NTSC: *National Television System Committee – Comité Nacional de Sistema de Televisión.*

OFDM: *Orthogonal Frequency Division Multiplexing – Multiplexación por División de Frecuencias Ortogonales.*

One Seg: *servicio de recepción parcial en la banda de televisión digital terrestre, diseñado para ser captado en dispositivos móviles.*

PCI: *Peripheral Component Interconnect – Interconexión de Componentes Periféricos.*

PDA: *Personal Digital Assistant – Asistente Digital Personal.*

PES: *Packetized Elementary Stream – Flujo de Paquetes Elementales.*

PTVE: *Primera Televisión Ecuatoriana, canal Guayaquileño.*

QAM: *Quadrature Amplitude Modulation – Modulación de Amplitud en Cuadratura.*

QPSK: *Quadrature Phase-Shift Keying – Modulación por Desplazamiento de Fase en Cuadratura.*

Reed-Solomon: *El código Reed-Solomon es un importante grupo de código de corrección de errores, es capaz de detectar y corregir varios errores de símbolos (grupos de bits).*

SBTVD: *Sistema Brasileiro de Televisão Digital – Sistema Brasileño de Televisión Digital*

SD: *Standard Definition – Definición estándar.*

SDTV: *Standard Definition Television – Televisión de Definición Estándar.*

SFN: *Single Frequency Network – Red de Frecuencia Única.*

SSL: *Secure Sockets Layer – Capa de Sockets Seguros.*

STB: *Set Top Box – Receptor o Decodificador de televisión.*

SUPERTEL: *Superintendencia de Telecomunicaciones*

TDS: *Televisión Digital por Satélite.*

TDS-OFDM: *Time Domain Synchronous Orthogonal Frequency Division Multiplexing – Dominio de Tiempo Síncrono-Multiplexación por División de Frecuencias Ortogonales.*

TDT: *Televisión Digital Terrestre.*

TIMI: *Terrestrial Interactive Multiservice Infrastructure – Infraestructura Multiservicios Terrestre Interactiva.*

TMCC: *Transmission and Multiplexing Configuration Control – Control de Configuración de Multiplexación y Transmisión.*

TS: *Transport Stream – Flujo de Transporte, protocolo de comunicación para audio, vídeo y datos especificados en los estándares de MPEG-2.*

TVDi: *Televisión Digital Interactiva.*

UHF: *Ultra High Frequency – Frecuencia ultra alta (300MHz a 3 GHz).*

UNP: *Unión Nacional de Periodistas.*

VGA: *Video Graphics Array – Matriz Gráfica de Vídeo.*

VHF: *Very High Frequency – Frecuencia muy alta (30 MHz a 300MHz).*

VSF: *Vestigial Sideband – Banda Lateral Vestigial.*

XML: *eXtensible Markup Language – Lenguaje de Marcas Extensible, utilizado para almacenar datos de forma legible.*

ÍNDICE DE CONTENIDOS

CARÁTULA	i
APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN.....	ii
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE DE ABREVIATURAS	vi
ÍNDICE DE CONTENIDOS.....	x
LISTA DE FIGURAS.....	xvii
LISTA DE TABLAS.....	xxiv
RESUMEN.....	1
ABSTRACT	2
INTRODUCCIÓN.....	3
CAPÍTULO I.....	4
1. ESTADO DEL ARTE.....	4
1.1. Historia de la televisión en Ecuador	5
1.1.1. La televisión en Loja.....	7
1.2. La Televisión Digital.....	8
1.2.1. Televisión Digital por Satélite.....	9
1.2.2. Televisión Digital por Cable.....	9
1.2.3. Televisión Digital por Banda Ancha	9
1.2.4. Televisión Digital por Redes Móviles	10
1.2.5. Televisión Digital Terrestre	10
1.3. Estándares internacionales de Televisión Digital Terrestre	11
1.3.1. Estándar Americano ATSC (Advanced Television System Committee)	11
1.3.2. Estándar Europeo DVB-T (Digital Video Broadcasting-Terrestrial)	12

1.3.3.	<i>Estándar Chino DTMB (Digital Terrestrial Multimedia Broadcasting)</i>	12
1.3.4.	<i>Estándar Japonés ISDB-T (Integrated Services Digital Broadcasting-Terrestrial)</i>	13
1.3.5.	<i>Estándar Japonés con variaciones brasileñas ISDB-Tb o SBTVD (Sistema Brasileño de Televisión Digital)</i>	14
1.3.5.1.	<i>Visión General</i>	17
1.3.5.2.	<i>Tipo de compresión y multiplexación</i>	17
1.3.5.3.	<i>Modulación BST-OFDM</i>	19
1.3.5.4.	<i>Transmisión jerárquica</i>	20
1.3.5.5.	<i>Flujo de transporte (TS)</i>	21
1.3.5.6.	<i>Recepción parcial</i>	22
1.3.6.	<i>Tabla comparativa de los estándares de Televisión Digital Terrestre</i>	23
1.4.	<i>Infraestructura de una red para televisión digital</i>	24
1.4.1.	<i>Infreestructura física</i>	24
1.4.1.1.	<i>Computador de producción y desarrollo</i>	25
1.4.1.2.	<i>Servidor de aplicaciones y contenidos</i>	25
1.4.1.3.	<i>Servidor de playout</i>	25
1.4.1.4.	<i>Set Top Box (STB) o equipo de usuario</i>	26
1.4.1.5.	<i>Receptor de televisor</i>	28
1.4.1.6.	<i>Tarjeta de televisión</i>	28
1.4.2.	<i>Infraestructura de software</i>	29
1.4.2.1.	<i>Software para el servidor de aplicaciones y contenidos</i>	29
1.4.2.2.	<i>Software para el servidor de playout</i>	30
1.5.	<i>Transición hacia la Televisión Digital Terrestre en Ecuador</i>	31
1.5.1.	<i>Plan Maestro de Transición a la Televisión Digital Terrestre en Ecuador</i>	36
1.5.1.1.	<i>Generalidades</i>	36
1.5.1.2.	<i>Implementación</i>	37

1.5.1.3.	<i>Espectro radioeléctrico</i>	37
1.5.1.4.	<i>Fases del apagón analógico</i>	37
1.5.1.5.	<i>Ventajas de la Televisión Digital</i>	38
CAPÍTULO 2.....		39
2.	<i>TELEVISIÓN DIGITAL INTERACTIVA</i>	39
2.1.	<i>Definición</i>	40
2.1.1.	<i>Elementos tecnológicos de la TVDi</i>	40
2.2.	<i>Interactividad</i>	41
2.2.1.	<i>Interactividad local</i>	41
2.2.2.	<i>Interactividad unidireccional</i>	42
2.2.3.	<i>Interactividad remota o con canal de retorno</i>	43
2.3.	<i>Servicios interactivos</i>	43
2.4.	<i>Características de diseño de una aplicación interactiva</i>	44
2.4.1.	<i>Layout de pantalla</i>	45
2.4.1.1.	<i>Superposición</i>	45
2.4.1.2.	<i>Pantalla competa con vídeo</i>	46
2.4.1.3.	<i>Pantalla completa sin vídeo</i>	47
2.4.2.	<i>Navegación</i>	47
2.4.2.1.	<i>Menú</i>	48
2.4.2.2.	<i>Vídeo Multi-pantalla</i>	48
2.4.2.3.	<i>Índice</i>	49
2.4.2.4.	<i>Paginado</i>	49
2.4.2.5.	<i>Tabs</i>	49
2.4.3.	<i>Teclas de control remoto</i>	50
2.4.4.	<i>Funciones básicas de los servicios interactivos</i>	52
2.4.5.	<i>Presentación de contenido</i>	52
2.4.6.	<i>Participación de usuario</i>	53

2.4.7.	<i>Entrada de textos</i>	53
2.4.8.	<i>Menú de ayuda</i>	53
2.4.9.	<i>Accesibilidad y personalización</i>	54
2.5.	<i>Middleware Ginga</i>	54
2.5.1.	<i>Definición</i>	54
2.5.1.1.	<i>Ventajas del middleware Ginga</i>	55
2.5.2.	<i>Arquitectura del middleware Ginga</i>	55
2.5.2.1.	<i>Ginga-CC</i>	57
2.5.2.1.1.	<i>Arquitectura</i>	57
2.5.2.2.	<i>Ginga-NCL</i>	60
2.5.2.2.1.	<i>Arquitectura</i>	60
2.5.2.2.2.	<i>Requerimientos</i>	63
2.5.2.3.	<i>Ginga-J</i>	63
2.5.2.3.1.	<i>Arquitectura</i>	63
2.5.2.3.2.	<i>Requerimientos</i>	64
2.5.2.4.	<i>Cuadro comparativo entre Ginga-NCL y Ginga-J</i>	64
2.5.2.5.	<i>LUA</i>	65
2.6.	<i>Canal de retorno</i>	65
2.6.1.	<i>Tecnologías de conectividad para el canal de retorno</i>	67
2.6.1.1.	<i>Canal de retorno mediante la red de PLC (Power Line Communication)</i>	67
2.6.1.2.	<i>Canal de retorno mediante tecnologías satelitales</i>	68
2.6.1.3.	<i>Canal de retorno mediante ADSL (Asymmetric Digital Subscriber Line)</i>	68
2.6.1.4.	<i>Canal de retorno mediante tecnologías de fibra óptica</i>	70
2.6.1.5.	<i>Canal de retorno mediante comunicaciones inalámbricas WiFi/WiMAX</i>	70

2.6.1.6.	Canal de retorno mediante tecnologías móviles	70
2.6.2.	Implementación del canal de retorno	71
2.6.2.1.	Simulación en Virtual Set Top Box Ginga-NCL.....	71
2.6.2.2.	Implementación en el Set Top Box para TV digital	71
2.7.	Set Top Box	72
2.7.1.	Definición	72
2.7.2.	Proceso de recepción y presentación de un STB	73
2.7.3.	Características de un Set Top Box para el estándar ISDB-Tb	73
CAPÍTULO 3.....		75
3.	PROGRAMACIÓN DE APLICACIONES INTERACTIVAS.....	75
3.1.	Lenguaje de programación NCL.....	76
3.1.1.	Estructura de un documento NCL.....	76
3.1.1.1.	Estructura inicial de un documento NCL.....	76
3.1.1.2.	Cabecera del documento NCL	77
3.1.1.3.	Cuerpo del documento NCL	83
3.1.1.3.1.	Nodos u objetos media.....	83
3.1.1.3.1.1.	Nodos de contenido.....	83
3.1.1.3.1.2.	Nodos de contexto	84
3.2.	Lenguaje de programación LUA.....	87
3.2.1.	Integración NCL-LUA	87
3.3.	Herramientas de desarrollo	89
3.3.1.	NCL Composer.....	89
3.3.2.	NCL Eclipse.....	92
3.3.3.	Ginga GUI o Ginga4Windows.....	93
3.3.4.	Ginga-NCL Virtual Set Top Box.....	93
CAPÍTULO 4.....		95
4.	DISEÑO DE APLICACIONES INTERACTIVAS.....	95

4.1.	<i>Diseño de la aplicación interactiva para Televisión Digital</i>	96
4.1.1.	<i>Parámetros generales de la aplicación interactiva</i>	96
4.1.1.1.	<i>Selección del tema para la aplicación interactiva</i>	96
4.1.1.2.	<i>Estándar de Televisión Digital Terrestre</i>	96
4.1.2.	<i>Middleware y plataforma de programación</i>	97
4.1.3.	<i>Interactividad y servicios interactivos</i>	97
4.1.4.	<i>Requerimientos de la interfaz de la aplicación interactiva</i>	98
4.2.	<i>Desarrollo de la aplicación interactiva</i>	99
4.2.1.	<i>Algoritmo: PSeInt</i>	99
4.2.2.	<i>Codificación: NCL Eclipse</i>	99
4.2.3.	<i>Layout de pantalla: NCL Composer</i>	101
4.3.	<i>Uso del teclado del control remoto</i>	102
4.4.	<i>Funcionamiento de la aplicación interactiva</i>	103
4.5.	<i>Caracterización del Set Top Box para reproducción de la aplicación</i>	104
4.6.	<i>Simulación y pruebas de la aplicación interactiva</i>	105
4.6.1.	<i>Simulación en Ginga GUI (Ginga4Windows)</i>	106
4.6.2.	<i>Simulación en Ginga-NCL Virtual Set Top Box</i>	112
4.6.3.	<i>Simulación en EITV Developer Box</i>	119
CAPÍTULO 5.....		126
5.	CONCLUSIONES Y RECOMENDACIONES.....	126
BIBLIOGRAFÍA.....		130
ANEXOS.....		137
ANEXO A.....		138
ALGORITMO DE LA APLICACIÓN INTERACTIVA DESARROLLADO EN PSEINT.....		138
ANEXO B.....		149
CODIFICACIÓN DE LA APLICACIÓN INTERACTIVA UTILIZANDO EL SOFTWARE NCL ECLIPSE.....		149

<i>ANEXO C.....</i>	<i>200</i>
<i> DISEÑO DEL LAYOUT DE PANTALLA Y VISUALIZACIÓN GRÁFICA DEL CÓDIGO COMPLETO DE LA APLICACIÓN INTERACTIVA MEDIANTE NCL COMPOSER</i>	<i>200</i>

LISTA DE FIGURAS

CAPITULO I

Figura 1.1. *Múltiple canalización del canal radioeléctrico de televisión digital.*

Figura 1.2. *Visión general del sistema de transmisión.*

Figura 1.3. *Capas de multiplexación y compresión del sistema brasileño.*

Figura 1.4. *Portadoras en los segmentos BST-OFDM.*

Figura 1.5. *Diagrama en bloques del sistema de transmisión.*

Figura 1.6. *Formación del flujo de transporte TS para ISDB-Tb.*

Figura 1.7. *Espectro de frecuencia del canal de TV.*

Figura 1.8. *Infraestructura física de la red de televisión digital.*

Figura 1.9. *Funciones de un equipo comercial de Playout profesional.*

Figura 1.10. *Set Top Box comercial.*

Figura 1.11. *Etapas de recepción en el STB.*

Figura 1.12. *Televisores con receptores externo e interno.*

Figura 1.13. *Tipos de tarjetas receptoras para televisión digital.*

Figura 1.14. *Transferencia de archivos a través del carrusel de objetos.*

Figura 1.15. *Porcentaje de intensidad de campo de los diferentes estándares.*

Figura 1.16. *Porcentaje de relación señal a ruido de los diferentes estándares.*

CAPITULO II

Figura 2.1. *Aplicación con interactividad local.*

Figura 2.2. *Aplicación con interactividad unidireccional.*

Figura 2.3. *Aplicación con interactividad remota.*

Figura 2.4. *Layout de pantalla con superposición.*

Figura 2.5. *Layout de pantalla completa con vídeo.*

Figura 2.6. *Layout de pantalla completa sin vídeo. Ginga Hero o Guitar Hero para televisión digital.*

Figura 2.7. *Layout de pantalla con navegación mediante listado y menú.*

Figura 2.8. *Tipos de controles para decodificador digital.*

Figura 2.9. *Arquitectura del middleware Ginga.*

Figura 2.10. *Arquitectura de Ginga-CC.*

Figura 2.11. *Arquitectura de Ginga-NCL.*

Figura 2.12. *Arquitectura de Ginga-J.*

Figura 2.13. *Diagrama de una red de televisión digital con canal de interactividad.*

Figura 2.14. *Canal de retorno mediante la red de PLC.*

Figura 2.15. *Canal de retorno mediante ADSL.*

Figura 2.16. *Escenario de simulación para la interactividad con canal de retorno.*

Figura 2.17. *Escenario de prueba real para la interactividad con canal de retorno.*

CAPITULO III

Figura 3.1. *Estructura inicial de un documento NCL.*

Figura 3.2. *Estructura de una base de regiones.*

Figura 3.3. *Atributos que pueden ser especificados en una región.*

Figura 3.4. Estructura de una base de descriptores.

Figura 3.5. Estructura de una base de conectores.

Figura 3.6. Línea de código para importar un documento de conectores.

Figura 3.7. Nodo de contenido de un archivo media de imagen en formato png.

Figura 3.8. Puerta de enlace o acceso.

Figura 3.9. Enlaces para sincronización de objetos media.

Figura 3.10. Ancla de contenido en un nodo media de vídeo.

Figura 3.11. Ancla de atributo para modificar las dimensiones y coordenadas de un nodo de media de un vídeo.

Figura 3.12. Estructura de un tratador de eventos.

Figura 3.13. Código para identificar los botones del control remoto en LUA.

Figura 3.14. Código para enviar los eventos desde LUA al documento NCL.

Figura 3.15. Visión estructural del software NCL Composer.

Figura 3.16. Visión de diseño del software NCL Composer.

Figura 3.17. Visión de esquema del software NCL Composer.

Figura 3.18. Visión temporal del software NCL Composer.

Figura 3.19. Visión textual del software NCL Composer.

Figura 3.20. Ambiente de programación en NCL Eclipse.

Figura 3.21. Simulador Ginga GUI (Ginga4Windows).

Figura 3.22. Emulador Ginga-NCL Virtual Set Top Box.

CAPITULO IV

Figura 4.1. Líneas de código de las regiones y descriptores de la aplicación interactiva.

Figura 4.2. Líneas de código de la importación del documento NCL de conectores de la aplicación interactiva.

Figura 4.3. Líneas de código de los archivos media de la aplicación interactiva.

Figura 4.4. Líneas de código de los enlaces de presentación de los archivos media de la aplicación interactiva.

Figura 4.5. Modelo básico de control remoto para televisión digital.

Figura 4.6. Configuración en Eclipse para utilizar el simulador Ginga4Windows.

Figura 4.7. Simulación en Ginga4Windows. (a) Presentación del botón INFO durante un programa de televisión. (b) Presentación de la aplicación interactiva (menú principal).

Figura 4.8. Simulación en Ginga4Windows. (a) Presentación del menú secundario ZOOLOGICO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo AVES. (c) Presentación de la galería de imágenes del botón verde MAMÍFEROS. (d) Presentación de la galería de imágenes del botón amarillo REPTILES. (e) Presentación de la galería de imágenes del botón azul NUEVA ESPECIE.

Figura 4.9. Simulación en Ginga4Windows. (a) Presentación del menú secundario ORQUIDEARIO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo NAVE 1. (c) Presentación de la galería de imágenes del botón verde NAVE 2. (d) Presentación de la galería de imágenes del botón amarillo NAVE 3. (e) Presentación de la galería de imágenes del botón azul NAVE 4.

Figura 4.10. Simulación en Ginga4Windows. (a) Presentación del menú secundario VIVERO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo ARBOLES. (c) Presentación de la galería de imágenes del botón verde FRUTALES. (d) Presentación de la galería de imágenes del botón amarillo ORNAMENTALES. (e) Presentación de la galería de imágenes del botón azul OTRAS ESPECIES.

Figura 4.11. Simulación en Ginga4Windows. Presentación del menú secundario CÓMO LLEGAR con una opción de retorno.

Figura 4.12. Configuración en Eclipse para utilizar el emulador Ginga-NCL Virtual Set Top Box.

Figura 4.13. Dirección IP del emulador Ginga-NCL Virtual Set Top Box.

Figura 4.14. Simulación en emulador Ginga-NCL Virtual Set Top Box. (a) Presentación del botón INFO durante un programa de televisión. (b) Presentación de la aplicación interactiva (menú principal).

Figura 4.15. Simulación en emulador Ginga-NCL Virtual Set Top Box. (a) Presentación del menú secundario ZOOLOGICO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo AVES. (c) Presentación de la galería de imágenes del botón verde MAMÍFEROS. (d) Presentación de la galería de imágenes del botón amarillo REPTILES. (e) Presentación de la galería de imágenes del botón azul NUEVA ESPECIE.

Figura 4.16. Simulación en emulador Ginga-NCL Virtual Set Top Box. (a) Presentación del menú secundario ORQUIDEARIO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo NAVE 1. (c) Presentación de la galería de imágenes del botón verde NAVE 2. (d) Presentación de la galería de imágenes del botón amarillo NAVE 3. (e) Presentación de la galería de imágenes del botón azul NAVE 4.

Figura 4.17. Simulación en emulador Ginga-NCL Virtual Set Top Box. (a) Presentación del menú secundario VIVERO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo ARBOLES. (c) Presentación de la galería de imágenes del botón verde FRUTALES. (d) Presentación de la galería de imágenes del botón amarillo ORNAMENTALES. (e) Presentación de la galería de imágenes del botón azul OTRAS ESPECIES.

Figura 4.18. Simulación en emulador Ginga-NCL Virtual Set Top Box. Presentación del menú secundario CÓMO LLEGAR con una opción de retorno.

Figura 4.19. Conexión de EITV Developer Box con pendrive mediante USB y televisor LG mediante HDMI.

Figura 4.20. Simulación en EITV Developer Box. (a) Presentación del botón INFO durante un programa de televisión (sin señal digital). (b) Presentación de la aplicación interactiva (menú principal).

Figura 4.21. Simulación en EITV Developer Box. (a) Presentación del menú secundario ZOOLOGICO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo AVES. (c) Presentación de la galería de imágenes del botón verde MAMÍFEROS. (d) Presentación de la galería de imágenes del botón amarillo REPTILES. (e) Presentación de la galería de imágenes del botón azul NUEVA ESPECIE.

Figura 4.22. Simulación en EITV Developer Box. (a) Presentación del menú secundario ORQUIDEARIO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo NAVE 1. (c) Presentación de la galería de imágenes del botón verde NAVE 2. (d) Presentación de la galería de imágenes del botón amarillo NAVE 3. (e) Presentación de la galería de imágenes del botón azul NAVE 4.

Figura 4.23. Simulación en EITV Developer Box. (a) Presentación del menú secundario VIVERO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo ARBOLES. (c) Presentación de la galería de imágenes del botón verde FRUTALES. (d) Presentación de la galería de imágenes del botón amarillo ORNAMENTALES. (e) Presentación de la galería de imágenes del botón azul OTRAS ESPECIES.

Figura 4.24. Simulación en EITV Developer Box. Presentación del menú secundario CÓMO LLEGAR con una opción de retorno.

ANEXO C

Figura 1. Layout de pantalla de presentación del botón INFO.

Figura 2. Programación gráfica de las acciones que realiza el botón INFO.

Figura 3. Layout de pantalla de la aplicación cuando se ha presionado el botón INFO.

Figura 4. Activación del botón EXIT para salir de la aplicación y reiniciar el botón INFO.

Figura 5. Activación del botón ZOOLOGICO.

Figura 6. Layout de pantalla de presentación del botón ROJO - ZOOLOGICO.

Figura 7. Layout de pantalla de presentación del botón VERDE - ORQUIDEARIO.

Figura 8. Layout de pantalla de presentación del botón AMARILLO - VIVERO.

Figura 9. Retorno al menú principal desde la opción BACK del ZOOLOGICO.

Figura 10. Navegación de las imágenes que se encuentre en la opción ROJO “Aves” del menú ZOOLOGICO.

Figura 11. Layout de pantalla de visualización de imágenes dentro de la opción ZOOLOGICO.

Figura 12. Layout de pantalla de visualización de imágenes dentro de la opción ORQUIDEARIO.

Figura 13. Layout de pantalla de visualización de imágenes dentro de la opción VIVERO.

Figura 14. Activación del botón AZUL.- CÓMO LLEGAR.

Figura 15. Layout de pantalla de presentación del botón AZUL - CÓMO LLEGAR.

Figura 16. Retorno al menú principal desde la opción BACK de la opción CÓMO LLEGAR.

LISTA DE TABLAS

CAPITULO I

Tabla. 1.1. Características del sistema de transmisión ISDB-Tb.

Tabla. 1.2. Resumen de las características técnicas de los estándares de televisión digital.

Tabla. 1.3. Resumen de la votación correspondiente a la calidad de vídeo, total 255 votos, modo fijo exterior.

Tabla. 1.4. Resumen de la votación correspondiente a la calidad de audio, total 255 votos, modo fijo exterior.

Tabla. 1.5. Resumen de la votación correspondiente a la calidad de audio, total 60 votos, modo fijo interior sin la presencia de ruido impulsivo.

Tabla. 1.6. Resumen de la votación correspondiente a la calidad de audio, total 60 votos, modo fijo interior con la presencia de ruido.

Tabla. 1.7. Resumen de la votación correspondiente a la calidad de vídeo, total 60 votos, modo fijo interior sin la presencia de ruido.

Tabla. 1.8. Resumen de la votación correspondiente a la calidad de vídeo, total 60 votos, modo fijo interior con la presencia de ruido.

Tabla. 1.9. Resumen de las fases del apagón analógico según el Plan de la Transición hacia la TDT.

CAPITULO II

Tabla. 2.1. Resumen de los diversos servicios interactivos para televisión digital.

Tabla. 2.2. Tipos de diseño de layout de pantalla.

Tabla. 2.3. Tipos de diseño de navegación en pantalla.

Tabla. 2.4. Teclado básico de un control remoto para televisión digital.

Tabla. 2.5. Cuadro comparativo entre Ginga-NCL y Ginga-J.

Tabla. 2.6. Resumen de las características técnicas de un STB para el estándar ISDB-Tb.

RESUMEN

A nivel mundial en muchos países la televisión digital ya ha reemplazado a la televisión convencional para tener más beneficios de ésta como es la mejora en calidad de audio y vídeo, visualización de imagen en HD, interactividad, entre otros. Con el presente trabajo se trata de realizar una investigación de esta tecnología en cuanto a características, estándar adoptado por Ecuador y herramientas de desarrollo de aplicativos interactivos, además se desarrolla una aplicación interactiva con contenido informativo y educativo (T-Learning) acerca de uno de los parques de la ciudad de Loja.

En conclusión, este trabajo propone realizar un estudio teórico y práctico de la tecnología de televisión digital y el desarrollo de aplicaciones interactivas que sean de beneficio informativo e interactivo para el usuario, de este modo se da paso al desenvolvimiento de la tecnología de televisión digital en nuestra ciudad tanto en implementación de ésta como en el desarrollo de aplicativos interactivos.

Keywords: *televisión digital, ISDB-Tb, modulación BST-OFDM, interactividad, aplicación interactiva, middleware ginga, canal de interactividad, set top box, NCL, LUA.*

ABSTRACT

In many countries worldwide, digital television has already replaced conventional television to have more benefits from this, like improving quality audio and video, HD image display, interactivity, among others. The present work is an investigation of this technology in terms of features, standard adopted by Ecuador and tools for developing interactive applications, plus the development of an interactive application with informative and educational content (T-Learning) about a Loja city park.

In conclusion, this work proposes a theoretical and practical study of digital television technology and the development of interactive applications that are of benefit informative and interactive for the user, thus it give way to the development of digital television technology in our city both in implementation this as in the development of interactive applications.

Keywords: *digital television, ISDB-Tb, BST-OFDM modulation, interactivity, interactive application, Ginga middleware, interactive channel, set top box, NCL, LUA.*

INTRODUCCIÓN

Las tecnologías enfocadas a la transmisión de información a través de redes de telecomunicaciones han ido mejorando tras el paso del tiempo, esto se lo hace con el fin de aprovechar el espectro radioeléctrico para los servicios de telecomunicaciones y esto es lo que pretende realizar la digitalización de la televisión.

La transición de la televisión analógica hacia la digital cada vez está más cerca de nuestras manos, en Ecuador en las grandes ciudades como Quito, Guayaquil y Cuenca esta tecnología se encuentra parcialmente implementada ya que aún se están realizando pruebas de transmisión y recepción de la señal digital de televisión. Esta tecnología trae diversos beneficios para el usuario final en las cuales se destacan la movilidad y portabilidad a través de one seg y la interactividad por medio del middleware Ginga, ésta última característica es la que causa mayor novedad ya que es una forma nueva de ver televisión.

El estándar adoptado por Ecuador ha sido el ISDB-Tb o SBTVD, éste es un estándar japonés con variaciones brasileñas que hacen que el sistema de televisión digital sea más robusto. Ginga es el middleware del SBTVD, éste compilador consta de un conjunto de tecnologías estandarizadas e innovaciones brasileñas que hacen que el decodificador sea una herramienta de recepción de la señal de televisión digital más avanzada y la mejor solución para el desarrollo de aplicaciones interactivas.

El proceso de transición hacia la televisión digital se encuentra en proceso de implementación en nuestro país, por lo tanto, el desarrollo de contenido interactivo surge como una necesidad inminente para las empresas que se dedican a la transmisión de canales de televisión, haciendo de este modo que exista una demanda de profesionales en la materia.

CAPÍTULO I

1. ESTADO DEL ARTE

1.1. Historia de la televisión en Ecuador.

La historia de la televisión en Ecuador comienza en el año de 1954 cuando el ingeniero estadounidense Gliford Hartwell plantea la idea de implementar en la ciudad de Quito la primera estación televisora religiosa del Ecuador en manos de la organización religiosa HCJB– “Hoy Cristo Jesús Bendice”. Para cumplir con este objetivo el ingeniero Hartwell intenta reparar un equipo transmisor de televisión que se encontraba abandonado en las bodegas de General Electric en Syracuse – New York, lo lleva al garaje de su casa donde pacientemente lo repara con ayuda de algunos trabajadores de la compañía hasta obtener resultados positivos. Luego de un arduo trabajo de cinco años de reparación los equipos (dos cámaras, una filmadora de slides y dos transmisores de audio y vídeo) quedan listos para su traslado a Ecuador. El 11 de julio de 1959 el ingeniero Hartwell decide trasladar los equipos a Quito junto con su esposa Virginia quien había obtenido conocimientos en la producción audiovisual de programas educativos y religiosos especialmente dedicados a niños, haciéndose así de esta manera parte del primer equipo de producción de la televisión ecuatoriana [1].

En este mismo año los equipos de transmisión mediante donación del ingeniero Hartwell pasaron a manos del ministerio de la HCJB, pero su instalación, dirección, producción de contenido estuvo a cargo del ingeniero Hartwell y su equipo, en un inicio se pensó que se llegaría directamente a instalar los equipos e iniciar con la transmisión, pero no fueron las cosas sencillas debido a que en el Ecuador no existían emisoras de televisión, por ende no existía un reglamento que las regule y que las controle motivo por el cual no se concedió el permiso para implementar esta nueva infraestructura. Sin embargo, esto no fue obstáculo para Hartwell ni para su equipo y aunque se negó el permiso, ellos construyeron una antena casera y la colocaron en las instalaciones de HCJB junto con los demás equipos para realizar pruebas mientras se conseguían los permisos [1].

El 10 de agosto de 1959, la Unión Nacional de Periodistas (UNP) pidió a la emisora que se lleven los equipos a una exposición en la feria desarrollada en los jardines del Colegio Americano conmemorando al Sesquicentenario de la Independencia del Ecuador para que los quiteños vean televisión en blanco y negro a través de una cobertura en circuito cerrado¹, esto fue considerado como el mayor atractivo de la feria tecnológica y paso a la historia porque fue la primera vez que

¹ Circuito cerrado de televisión: es un sistema que integra cámaras de vídeo ubicadas en lugares específicos que mediante una consola o switch, están conectados a televisores. Este sistema permite transmitir únicamente a los televisores conectados en el mismo sitio donde se desarrolla la programación.

se vio televisión en Ecuador. Este trabajo que realizó la HCJB en la feria tecnológica fue apoyado por una empresa privada de Guayaquil que se había formado recientemente, “Televisión Ecuatoriana”, ésta promocionó la feria y dio a conocer a todos el evento que se desarrollaría en aquel lugar, debido a esto en 1960 se llevó a cabo la feria de octubre en Guayaquil y con ayuda de la Casa de la Cultura la televisión llega al puerto principal y con la ayuda del ingeniero Hartwell realiza así mismo la instalación de una antena para realizar más pruebas [1][2].

Paralelamente a este acontecimiento, en 1959 la manabita Linda Zambrano oriunda de Bahía de Caráquez y su esposo el alemán Horts Michael Rosembau fueron quienes iniciaron con la primera televisión en Ecuador, ya que ellos como amantes de la tecnología y la innovación asistieron a la Feria Internacional de la Tecnología en Alemania y fue ahí donde conocieron la televisión, fue entonces cuando nace su gran interés de este producto innovador y deciden llevarlo a Ecuador. En ese año, la familia Rosembau Zambrano importa los primeros equipos profesionales para TV para instalar un canal en Guayaquil denominado “Primera Televisión Ecuatoriana” (PTVE), pero al igual que la HCJB se les negó el permiso de funcionamiento [2].

El 1 de junio de 1960 el gobierno de Camilo Ponce emitió la primera reglamentación y normativa para la regulación y funcionamiento de canales en Ecuador publicado en el registro N°. 985 [3]. Este permiso fue concedido a Linda Zambrano debido a que su esposo era de nacionalidad alemana motivo por el cual el permiso le queda negado a él. Una vez concedida la frecuencia y el permiso necesario, mediante un acuerdo con la Casa de la Cultura Núcleo del Guayas el canal 4 PTVE empieza a operar en el quinto piso de su edificio, luego por cambio de dueños este pasó a llamarse RedTeleSistema (RTS), este canal fue el primero en obtener un permiso para laborar como canal de carácter privado realizando su primera transmisión el 12 de diciembre de 1960 con una señal comercial para todo el país. Posteriormente a esto se integra también el canal 6 en la ciudad de Quito, ya con los permisos necesarios la HCJB se convierte en lo que hoy en día es Telemazonas [4].

El 1 de mayo de 1967, nace Ecuavisa con las instalaciones ubicadas en el Cerro del Carmen - Guayas, este canal transmitió la llegada del hombre a la luna mediante un reporte telefónico desde Miami el 21 de julio de 1969. En ese año 1967, la televisión llega a Cuenca con el surgimiento del canal 3 propiedad de Presley Norton y en 1969 se crea la Cadena Ecuatoriana de Televisión, Telecentro, que transmitió en el canal 10 con una señal en VHF [3].

El surgimiento de la televisión a color fue un gran acontecimiento en la televisión ecuatoriana, esto fue posible gracias al empresario Antonio Granda Centeno dueño de la televisora de Teleamazonas, donde el 22 de febrero de 1974 se transmitió el primer partido de fútbol nacional en vivo y a color. Para los años 70, nace la Televisora Nacional en su canal 8 y Gamavisión en el canal 2, ambas en Quito, ya en los años 80 Ecuavisa extendió su señal a Guayaquil y Cuenca. En la década de los 90 surgen los primeros canales en la banda de UHF y aparece la televisión por cable denominada Sistemas de audio y vídeo por suscripción [3]; por conmemoración de la Independencia de Cuenca surge la señal de ETV Telerama con ayuda del empresario Juan Eljuri propietario del Banco del Austro y además TC Televisión (Antes Telecentro) inicia sus transmisiones frente al aeropuerto José Joaquín de Olmedo [5].

El 27 de septiembre del 2004 Ecuavisa inicia sus transmisiones a nivel internacional mediante señal satelital; RTU nace en el año 2005 bajo el mando de Carlos Alarcón Costa y también transmite su señal vía satélite, en el mismo año nace “Canal 1 Internacional”. Para el año 2007 inicia sus transmisiones “TV Ecuador” con motivo de la instalación de la Asamblea Constituyente desde la Ciudad Alfaro de Montecristi – Manabí y en el 2008 con cobertura para toda América mediante señal satelital (Intelsat 1R) nace Teleamazonas Internacional [5].

1.1.1. La televisión en Loja [3].

La primera televisora en Loja es el canal 4 UV Televisión que empieza a transmitir en 1968, la misma que fue utilizada como una repetidora para transmitir contenidos de los monopolios de ese entonces de Guayaquil y Quito, esto sucedió debido a que Presley Norton tenía por idea crear varios canales locales en todo el país, de este modo conseguía tener una red nacional que sería dirigido por varios locutores de radio que posteriormente serían los pioneros de la televisión en Ecuador, en el caso de Loja el señor Norton trajo desde Guayaquil al locutor Walter Jaramillo, éste sería uno de los fundadores de la televisión en Loja además de aportar económicamente para la compra del equipo original necesario.

Las transmisiones en el canal se detuvieron en 1978 debido a la falta de recursos económicos lo que obligó al canal a cerrar durante 10 años, para ese entonces el canal reinicio sus actividades ahora ya formando parte de la Asociación de Canales Comunitarios y Regionales del Ecuador (CCREA). Por otro lado, el 10 de mayo de 1997 se inaugura ECOTEL TV.

En la provincia de Loja se autorizan 28 frecuencias de televisión de la cuales 24 son repetidoras y 4 son matrices (3 en VHF y 1 UHF) designadas a Televisión Educativa Calasancia, ubicada en el cantón Saraguro, UV Televisión y ECOTEL TV con matriz en la ciudad de Loja y la Televisora del Sur en el cantón calvas, quien por motivos económicos ante la inversión de la digitalización del canal se convirtió en un productor local y dejó de transmitir en el 2013.

1.2. La Televisión Digital.

La televisión analógica está a punto de llegar a su fin dando paso a la televisión digital, la difusión de las señales de televisión se lo realizará mediante tecnología digital proporcionando al contenido del programa una calidad mejorada en imagen y audio, además de tener acceso a información e interactividad. La señal analógica será convertida en digital mediante un conversor análogo/digital, este a su vez comprime la señal, la almacena y la transporta utilizando un mínimo de recursos en el espectro sin degradar la señal digital [6].

Los canales radioeléctricos de televisión digital utilizan el mismo ancho de banda (6 MHz) que los canales de televisión analógica, la diferencia radica en la utilización más eficiente de ese ancho de banda, mediante la tecnología digital y gracias a sus técnicas de compresión de imagen y audio y a su velocidad de transmisión de los canales digitales, se puede tener un solo programa de televisión en alta definición (High Definition – HD) o se puede tener cinco programas con calidad similar a la señal actual (Standard Definition – SD) [7]. En la figura 1.1 se muestra la múltiple canalización que se puede realizar mediante la tecnología digital de televisión.



Figura 1.1. Múltiple canalización del canal radioeléctrico de televisión digital.

Fuente: https://docs.google.com/file/d/0B3Hp29uy3q7_ODY0RnRMZnhBOTA/edit

1.2.1. Televisión Digital por Satélite.

La TDS utiliza como su nombre indica, satélites de comunicaciones para la transmisión de la señal de televisión en formato digital. Para este medio de transmisión se han distinguido dos tramos diferentes: enlace ascendente y enlace descendente, los mismos que utilizan bandas de frecuencias diferentes para evitar interferencias, estos son los encargados de enviar datos desde la televisora emisora al satélite y la transmisión de datos desde el satélite hacia la zona que cubre este en la superficie terrestre [6]. Los datos se encuentran en formato digital y esto hace que se pueda ofrecer más canales utilizando el mismo ancho de banda.

Este sistema de transmisión mediante satélite tiene como objetivo dar cobertura de la señal digital de televisión a aquellas zonas en las cuales la cobertura de la Televisión Digital Terrestre (TDT) es escasa debido a su zona geográfica o a la baja densidad de población.

1.2.2. Televisión Digital por Cable.

Este sistema se basa en la transmisión de señales digitales de televisión a través de las redes híbridas de fibra óptica y cable coaxial, además por estas redes también se ofrece servicios de radio, telefonía fija y acceso a Internet. Las redes utilizadas en la distribución de este tipo de servicios se divide en cuatro secciones: la cabecera, la red troncal, la red de distribución y la red de acometida hacia los abonados [8].

Este servicio de suscripción que se ofrece mediante cable ha tenido una gran acogida debido a que brinda múltiples servicios, esto lo hace diferente a los servicios que se reciben habitualmente aunque una de las limitaciones de éste es la posibilidad de pago.

1.2.3. Televisión Digital por Banda Ancha.

La televisión digital por banda ancha hace referencia a la transmisión de la misma mediante cualquier tipo de red que tenga una elevada capacidad para transportar información, entre ellas se encuentran las redes GPON, HFC, ADSL, otros. Estas redes además sirven como canal de interactividad mediante el cual se envía información hacia la televisora, por lo general para este canal se utiliza ADSL ya que este sistema se encuentra implementado en la mayoría de los hogares.

1.2.4. Televisión Digital por Redes Móviles.

Es un servicio de difusión de televisión con tecnología digital mediante ondas radioeléctricas hacia dispositivos móviles o portátiles utilizando una infraestructura de red celular.

En la antigua definición de la televisión para dispositivos móviles se establecía un solo canal exclusivo para cada dispositivo, es decir, existía una comunicación punto a punto, lo cual hacía que existiera un número limitado de usuarios de manera simultánea. La tecnología digital sobre la tecnología convencional en dispositivos móviles se lo realiza como un servicio de difusión más eficiente, en este caso existe una comunicación punto – multipunto, lo cual implica que el servicio no va a ser limitado, ya que este podrá abarcar a un gran número de usuarios con acceso al servicio de manera simultánea [40].

1.2.5. Televisión Digital Terrestre.

Este sistema nació como un intento de transmitir canales de televisión en alta definición (HDTV). HDTV fue creado por la NHK (Asociación de Radiodifusión de Japón) y fue presentado como un estándar a la ITU en 1973, el objetivo de este sistema era de dejar libres las bandas de 470 Hz a 890 Hz y también las frecuencias UHF y VHF para luego estas ser reasignadas a los canales de televisión digital. La NHK creó un sistema de compresión y transmisión MUSE (Codificación Múltiple de Muestreo Sub-Nyquist), esto se lo hizo debido a que se necesitaba de un ancho de banda considerable para poder transmitir la imagen en alta definición [9].

En 1990 la FCC (Comisión Federal de Comunicaciones) creó el ACATS (Comisión Asesora sobre Servicios de Televisión Avanzada), la cual sería la encargada de la creación de un estándar que regule y controle la transmisión de canales en alta definición en formato digital utilizando las tecnologías que existían hasta la fecha. En 1993, las empresas AT&T, General Instruments, Philips, Thomson Electronics, Zenith, el centro de investigación de David Sarnoff y el Tecnológico de Massachussets se unieron con el objetivo de realizar mejoras en el sistema MUSE que fue analizado por la ACATS, ellos introdujeron la compresión de datos de audio y vídeo con múltiples técnicas de multiplexación y modulación, debido a esto se dejó de llamar a estos sistemas como televisión de alta definición y junto con la transmisión de datos se lo llamo Televisión Digital Terrestre [9].

Los estándares de televisión digital terrestre que existen en la actualidad son los siguientes:

- *Estándar Americano ATSC (Advanced Television System Committee).*
- *Estándar Europeo DVB-T (Digital Video Broadcasting-Terrestrial).*
- *Estándar Chino DTMB (Digital Terrestrial Multimedia Broadcasting).*
- *Estándar Japonés ISDB-T (Integrated Services Digital Broadcasting-Terrestrial).*
- *Estándar Japonés con variaciones brasileñas ISDB-Tb o SBTVD (Sistema Brasileño de Televisión Digital).*

1.3. Estándares internacionales de Televisión Digital Terrestre.

Los estándares de televisión digital son un conjunto de especificaciones que se basan en un conjunto de requerimientos de la población en cuanto a difusión de televisión, las mismas que se usan como normas o criterios para asegurar que los procesos se den de acuerdo al propósito del estándar. Para elegir un estándar adecuado, se lo debe hacer de acuerdo a la realidad de cada país dependiendo de sus condiciones geográficas, densidad poblacional, entre otros [7].

1.3.1. Estándar Americano ATSC (Advanced Television System Committee).

ATSC es un estándar norteamericano que fue adoptado por USA, Canadá, Corea del Sur, México, Honduras, República Dominicana y El Salvador [10]. Fue creado con el fin de reemplazar al estándar analógico NTSC y de este modo poder transmitir señales en HDTV y SDTV a una velocidad de 19.39 Mbps con un ancho de banda de 6 MHz, brinda 6 veces mejor calidad de imagen que la televisión analógica actual y mejor calidad de sonido tipo cine Audio Dolby Digital (AC-3) [6].

Utiliza la modulación 8-VSB (level vestigial sideband – Banda Lateral Vestigial) basada en la modulación 8-QAM y 64-QAM con una codificación de Trellis, esta es una modulación mono portadora e independiente de fase para evitar la mayor cantidad de distorsiones [10]. La señal de vídeo se comprime con MPEG-2, la señal de audio se comprime con Audio Core 3 AC-3 que se compone de 6 canales discretos, existen servicios interactivos denominados DASE (Digital TV Applications Software Environment) [11].

Las aplicaciones interactivas se desarrollan en la plataforma conocida como ACAP (Advanced Common Application Platform – Plataforma de Aplicación Común Avanzada) [10].

1.3.2. Estándar Europeo DVB-T (Digital Video Broadcasting-Terrestrial).

Este sistema transmite audio, vídeo y otros datos a través de un flujo MPEG-2 (SDTV) y MPEG-4 (HDTV), diseñado para canales de 8 MHz, 7 MHz y 6 MHz, usando una modulación COFDM (Coded Orthogonal Frequency Division Multiplexing) [10]. Con el tiempo se crea DVB-T2 la cual incorpora MPEG-4 AVC (H.264) para la codificación de vídeo y Dolby Digital (AC-3) para el audio. El DVB-T fue adoptado por la comunidad Europea, Oceanía (Australia y Nueva Zelanda), Asia (Arabia Saudita, India, Israel, entre otros.), Colombia, Panamá, Trinidad y Tobago, Groenlandia, Bermudas y algunos países del sur de África (Egipto, Marruecos, Sudáfrica, entre otros.) [3].

Se puede transmitir cinco o más programas de SDTV a una velocidad de 20 a 30 Mbps, para el servicio móvil solamente se pueden transmitir dos canales SDTV. El uso de OFDM hace que se puedan obtener dos modos de transmisión: 2K (efecto Doppler) y el modo 8K (protección frente a trayectorias múltiples) portadoras ambas con modulación QAM y para la corrección de errores se emplea la convolucional FEC (1/2, 2/3, 3/4, 5/6, 7/8) y Reed-Solomon [7].

La plataforma para el desarrollo de aplicaciones interactivas es conocida como MHP (Multimedia Home Platform), es un middleware abierto [12].

1.3.3. Estándar Chino DTMB (Digital Terrestrial Multimedia Broadcasting) [6].

Es un estándar para televisión en terminales fijos y móviles adoptado por la República Popular de China, Hong Kong y Macao, utiliza la modulación TDS-OFDM (Time Domain Synchronous Orthogonal Frequency Division Multiplexing)

DTMB es una fusión de varias tecnologías que incluye derivaciones de los estándares ATSC y DVB-T, este estándar se aprobó en agosto del 2007, es un sistema muy flexible ya que soporta la combinación de redes de frecuencia única (SFN) y redes de frecuencia múltiple (MFN), utiliza MPEG-2 o MPEG-4 para la compresión de audio y vídeo y MPEG-2 para la multiplexación y transporte del flujo de datos que se realiza mediante paquetes denominados PES (Packetized

Elementary Stream), estos se multiplexan con otros programas y servicios de datos en el mismo ancho de banda y forman lo que se denomina TS (*Transport Stream*).

Para la corrección de errores utiliza la codificación LDPC (*Low-Density Parity-Check*) y una modulación TDS-OFDM para combinar los canales SD, HD y los demás servicios multimedia. Las aplicaciones interactivas se desarrollan en la plataforma TiMi (*Terrestrial Interactive Multiservice Infrastructure*).

1.3.4. Estándar Japonés ISDB-T (*Integrated Services Digital Broadcasting-Terrestrial*).

Este estándar nació en ARIB (*Asociación de industrias y Empresas de Radiodifusión*) el cual es un organismo de normalización en Japón [10].

La compresión de vídeo se lo hace con MPEG-2 y para la compresión de audio se utiliza MPEG-2 AAC para transmisión fija y MPEG-4 HE-AAC para dispositivos móviles [12]. Para la transmisión se utiliza una modulación OFDM en bandas segmentadas donde se utilizan algunas técnicas de modulación (QPSK, DQPSK, 16QAM 64QAM), utiliza la multiplexación MPEG-2 como medio de transporte de alta definición de diferentes señales en un mismo ancho de banda de 6 MHz a velocidades que oscilan entre 3.65 y 23.3 Mbps, pero su capacidad puede escalarse a cualquier anchura de banda de canal, con tan solo modificar la capacidad de datos; el hecho de utilizar OFDM en bandas segmentadas significa que las portadoras estarán agrupadas en 13 segmentos, de los cuales uno es usado específicamente para enviar la señal en baja definición para equipos portátiles y dispositivos móviles. Estos 13 segmentos son utilizados para transmitir un canal en HDTV y un canal para dispositivos móviles, o enviar tres canales en SDTV y a la vez también el canal para dispositivos móviles [7].

El estándar permite la recepción en dispositivos móviles, computadores portátiles y vehículos, gracias a un servicio que ofrece ISDB-T, 1seg (*One Seg*). Los países que adoptaron este estándar son Japón, Filipinas, Maldivas. En junio del 2006, Brasil adopta el estándar ISDB-T y realiza algunas modificaciones como H.264 o MPEG-4 (código de vídeo de alta compresión) y utiliza tecnologías como el middleware Ginga para el desarrollo de las aplicaciones interactivas dejando así de lado al middleware ARIB [10].

Para la transmisión de datos mediante las aplicaciones interactivas se utiliza el middleware ARIB, el mismo que se basa en XML y en el middleware DVB-MHP [12]. Se incorpora el sistema EWS – Sistema de Alerta Temprana.

1.3.5. Estándar Japonés con variaciones brasileñas ISDB-Tb o SBTVD (Sistema Brasileño de Televisión Digital) [5].

El SBTVD es una norma brasileña para la transmisión de la televisión digital terrestre. En el 2006 Brasil elige su sistema de TV digital, el cual está basado en la mezcla de las tecnologías del sistema japonés ISDB-T con mejoras tecnológicas brasileñas para acoplarse a su mercado y al mercado latinoamericano. Los países que adoptaron este estándar son Argentina, Brasil, Bolivia, Chile, Costa Rica, Ecuador, Guatemala, Nicaragua, Paraguay, Perú, Uruguay y Venezuela [13].

Las características del SBTVD son [13]:

- Multiprogramación, disponibilidad de hasta 4 canales.
- Interactividad que puede ser usada en distintos niveles.
- Interoperabilidad entre los diferentes patrones de televisión digital.
- Robustez que recibe distintas programaciones en todo el país.
- Movilidad.
- Portabilidad.
- Accesibilidad, para las personas con necesidades especiales.
- Está disponible tanto en HDTV como en SDTV.
- Utiliza MPEG-4, que tiene más recursos tecnológicos.

El sistema brasileño presenta una serie de mejoras tecnológicas, entre ellas se diferencia del sistema ISDB-T en el uso de tecnologías de compresión de vídeo y audio más avanzadas (H.264/HE-AAC), en Japón las tecnologías que se utilizan son MPEG-2 para vídeo y para audio MPEG L2. La modulación BST-OFDM en ambos sistemas se mantiene, al igual que el medio de transporte que está basado en el estándar MPEG-2 [13].

Otra mejora tecnológica del sistema brasileño se basa en la codificación de audio y vídeo para la transmisión para dispositivos móviles (one seg), presentando de esta manera una mejora en la calidad de la televisión portátil y móvil [13]:

- *Japón: Vídeo H.264 a 15 fps; Audio HE-AAC v.1 de baja complejidad.*
- *Brasil: Vídeo H.264 a 30 fps; Audio HE-AAC v.2 de baja complejidad.*

SBTVD puede transmitirse en 6 MHz, 7 MHz y 8 MHz, de esta manera se permite la interoperabilidad entre los sistemas norteamericano, europeo, japonés y el híbrido japonés-brasileño.

Se destaca también el empleo de sistemas MIMO (Multiple-input Multiple-output), esta cuenta con dos antenas de transmisión con el objetivo de que en un futuro se cuente también con múltiples antenas de recepción, esto para que el sistema cuente con un sistema más robusto de soporte en condiciones geográficas desfavorables de propagación y cuando se introduzca los receptores móviles [13].

El middleware Ginga es totalmente innovador y fue desarrollado en Brasil. Ginga presenta una mejor interacción con aplicaciones complejas, provee sincronismo espacio-temporal de objetos media y da un mejor soporte, permitiendo así el intercambio y gestión de aplicaciones de una manera más fácil para todos los desarrolladores, ya que está basado en NCL (Nested Context Lenguaje) que es declarativo y LUA que es de procedimiento, a diferencia del middleware japonés BML (Broadcast Markup Lenguaje) que sólo es declarativo [7].

Ginga permite que los contenidos creados para TV digital se puedan exhibir en diferentes sistemas de recepción, independientemente del fabricante o del tipo de receptor, debido a que este middleware acepta TV, celulares, computadoras de mano (PDAs) o TV pagada, como cable y satelital entre otros. Este middleware ofrece código abierto y libre, además de interfaz con Internet e interfaz gráfica [13].

En general el SBTVD es el estándar más moderno y tecnológicamente más avanzado, por esto es una alternativa excelente para llevar adelante procesos de investigación, además es de licencia libre, por lo que los costos de la red para el consumidor final disminuyen [9].

En la tabla 1.1 se presenta un cuadro con algunas características adicionales del Sistema Brasileño de Televisión Digital Terrestre:

Tabla. 1.1. Características del sistema de transmisión ISDB-Tb [14].

Parámetros de Transmisión	Características		
Modulación	BST – OFDM		
Número de segmentos (Ns)	13		
Modo	1 (2k)	2 (4k)	3 (8k)
Ancho de banda útil	5.575 MHz	5.573 MHz	5.572 MHz
Número total de portadoras	$108 * N_s + 1 = 1405$	$216 * N_s + 1 = 2809$	$432 * N_s + 1 = 5617$
Número de portadoras útiles	$96 * N_s = 1248$	$192 * N_s = 2496$	$384 * N_s = 4992$
AB por portadora	3.968 KHz	1.984 KHz	0.992 KHz
Número de símbolos/Trama	204		
Duración de símbolos activos	252 us	504 us	1008 us
Relación de intervalo de guardia	$\frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$ de la duración de símbolos activos		
Modulación de la portadora	QPSK, 16QAM, 64QAM, DPSK		
Corrección de errores	Inner: Código convolucional Tasa de codificación: $\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}, \frac{7}{8}$		
	Outer: Reed – Solomon (204, 188)		
Time Interleave	0 – 0.5 seg		
Tasa de bit de información	3.7 – 23.2 Mbps		

Elaborado: 13 diciembre 2015.

1.3.5.1. Visión General.

En la transmisión, una o más entradas conteniendo haz de datos TS, definidas en el sistema MPEG-2, se deben remultiplexar obligatoriamente para crear un único TS. Ese TS debe obligatoriamente ser sometido a la etapa de codificación de canal múltiple, de acuerdo con la intención de servicio y debe, obligatoriamente, ser entonces enviado como una señal OFDM común. En la figura 1.2 se muestra un esquema general del sistema de transmisión [15].

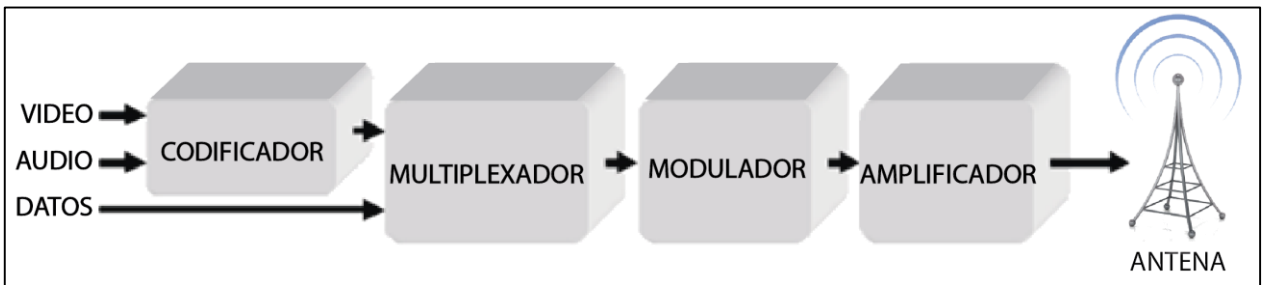


Figura 1.2. Visión general del sistema de transmisión [15].

Como se muestra en el esquema anterior, las señales de audio y vídeo deben estar codificadas antes de ser multiplexadas con los datos, para luego ser moduladas y así tener la señal de televisión digital lista para su transmisión.

El Time Interleaving se utiliza para obtener una codificación con la menor tasa de errores para la recepción móvil. El ancho de banda del canal de televisión es dividido en 13 segmentos OFDM sucesivos donde cada segmento ocupa 1/14 del ancho de banda del canal de televisión [15].

1.3.5.2. Tipo de compresión y multiplexación.

Se utiliza la codificación H.264 (MPEG-4 AVC) con el fin de fortalecer algunas debilidades presentes en las normas de compresión anteriores a esta. Trabaja con un código de vídeo de alta compresión y proporciona una imagen de alta calidad con tasas binarias muy inferiores a estándares anteriores, de este modo se obtiene más beneficios sin la necesidad de modificar su diseño [14].

Algunas características de H.264 son [14]:

- Reducción en la tasa de bit del 50 %, proporcionando una calidad de vídeo mejorada en comparación con cualquier otro estándar de vídeo.
- Los errores de transmisión sobre varias redes son toleradas.
- Sintaxis sencilla que simplifica las implementaciones.
- Decodificación exacta, los cálculos numéricos son realizados por un codificador y un decodificador para evitar la acumulación de errores.

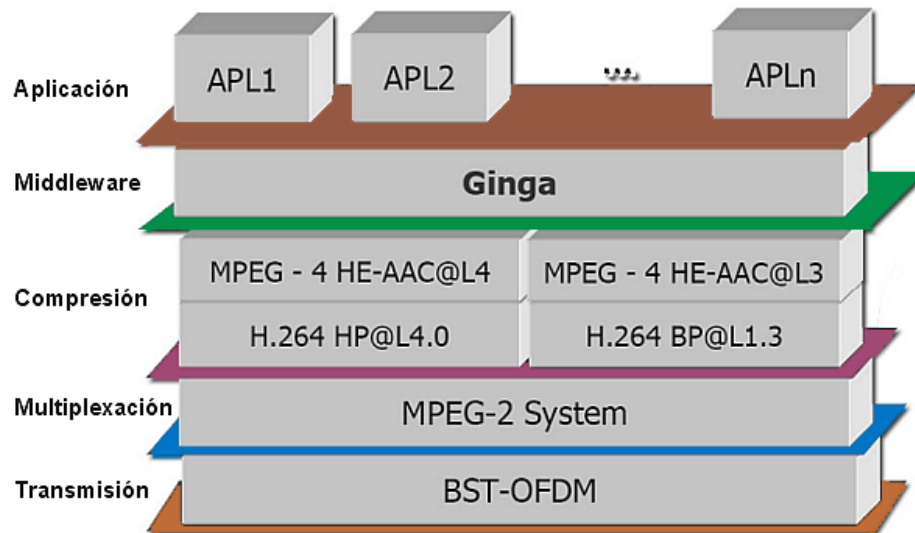


Figura 1.3. Capas de multiplexación y compresión del sistema brasileño.

Fuente: <http://alejandroayala.solmedia.ec/?p=491>

MPEG-4 AVC fue creado para la transmisión de contenido web para dispositivos con recursos limitados y conexiones de internet con tasas de bit bajas. Este códec presenta mejoras frente a MPEG2, se pueden hacer composiciones de vídeo sobre un fondo en tiempo real, mejor estimación de movimiento y filtraje de desbloqueo. Soporta interleaving, resoluciones de hasta 4096 pixeles x 4096 pixeles y un flujo de datos entre 5 Kbps y 10 Mbps, lo cual abarca un ancho de banda que soporta desde una red de telefonía móvil hasta vídeo en televisión en HD y permite interactividad [13].

El estándar MPEG-4 HE-AAC para audio soporta todas las herramientas definidas en MPEG-2 AAC y entrega nuevas herramientas como PNS (Perceptual Noise Substitution) para disminuir los efectos de ruido y LTP (Long Term Prediction) que es una herramienta de predicción. Estas dos herramientas utilizan menos procesamiento computacional que MPEG-2 y entregan mejores resultados [16].

1.3.5.3. Modulación BST-OFDM [7].

Transmisión de Banda Segmentada (BST) OFDM, consiste en un grupo de rangos de bloques de frecuencia comunes básicos, llamados "Segmentos BST". Subdivide la señal en 13 segmentos sucesivos OFDM, de estos cada uno cuenta con un ancho de banda de $BW/14$ MHz, donde BW es el ancho de banda de un canal normal de televisión analógica terrestre que puede ser 6, 7 u 8 MHz dependiendo del país. Un canal típico de 6 MHz, uno de sus segmentos ocupado $6/14 = 428.6$ KHz.

Al utilizar OFDM, las sub portadoras son ortogonales entre sí, la diafonía entre los sub canales es eliminado por lo que no es necesario dejar un espacio de ancho de banda entre sub canales, en la figura 1.4 se muestran las portadoras de los segmentos BST-OFDM.

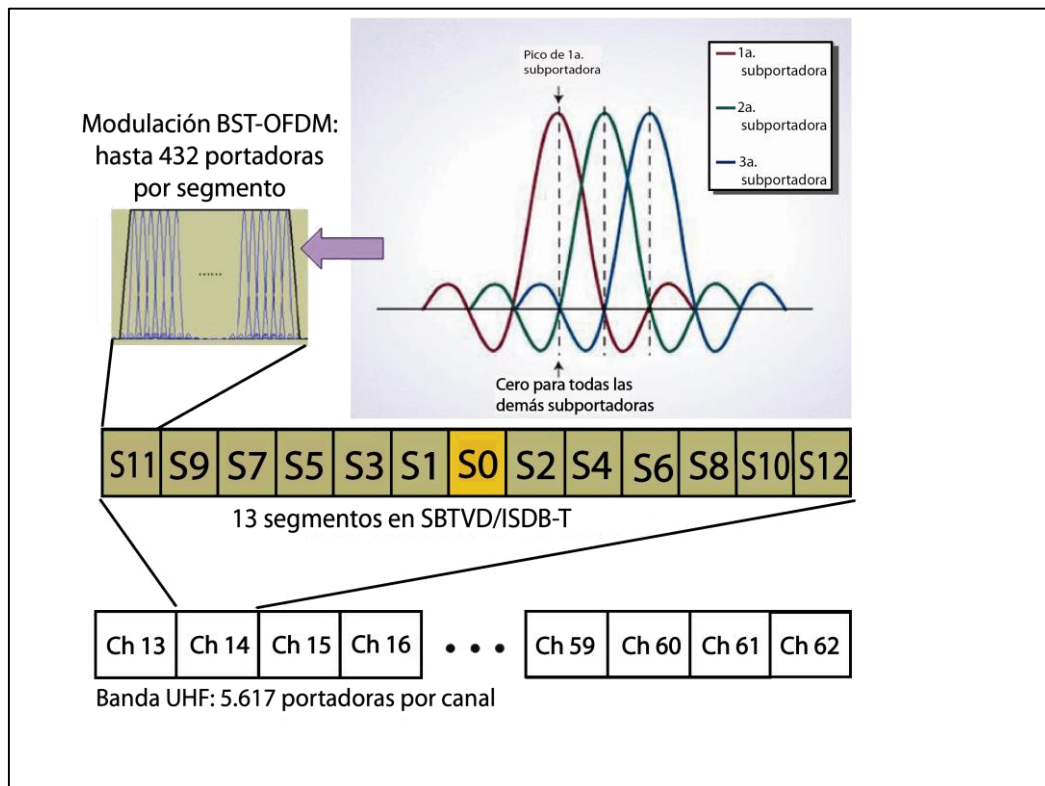


Figura 1.4. Portadoras en los segmentos BST-OFDM [7].

Las principales propiedades de modulación OFDM, BST-OFDM es que proveen transmisiones jerárquicas al usar diferentes tipos de modulación de portadoras y tasas de codificación de código interno para los diferentes segmentos que se pueda tener. Al dividir una señal en segmentos, cada

uno de estos segmentos de datos puede tener su propio método de protección contra errores y tipo de modulación (QPSK, DQPSK, 16-QAM o 64-QAM).

La tasa neta de bits para un canal digital varía entre 3.65 Mbps y 23.23 Mbps con un ancho de banda útil de 5.57 MHz, ya que $13 \times BW/14 = 5.57$ MHz, mientras que en un canal analógico la tasa de bits varía en el orden de los Kbps.

1.3.5.4. Transmisión jerárquica [15].

La codificación de canal debe obligatoriamente ser realizada en unidades de segmento OFDM. Un único canal de televisión debe obligatoriamente ser usado simultáneamente para servicio de recepción fija, recepción móvil y recepción portátil (transmisión jerárquica).

Cada capa jerárquica consiste en uno o más segmentos OFDM. Cada capa jerárquica puede especificar parámetros como: esquema de modulación de portadoras OFDM, tasa de código interno y del entrelazado de tiempo. Se puede definir hasta tres capas jerárquicas, de este modo se tiene un segmento que puede ser usado para recepción parcial, siendo también considerada como capa jerárquica.

El radiodifusor es el encargado de configurar el número de segmentos y el conjunto de parámetros de codificación de cada capa jerárquica. La señal TMCC (Transmission and Multiplexing Configuration Control) contiene la información necesaria para ayudar al receptor en la identificación de los modos de operación.

En la figura 1.5 observamos la codificación de datos, audio y vídeo para ser transmitido posiblemente hacia un receptor fijo, mientras tanto, existen más codificaciones que se están realizando para enviar la misma información hacia dispositivos móviles y portátiles, todos estos se multiplexan para formar el TS que será transmitido luego de ser modulado.

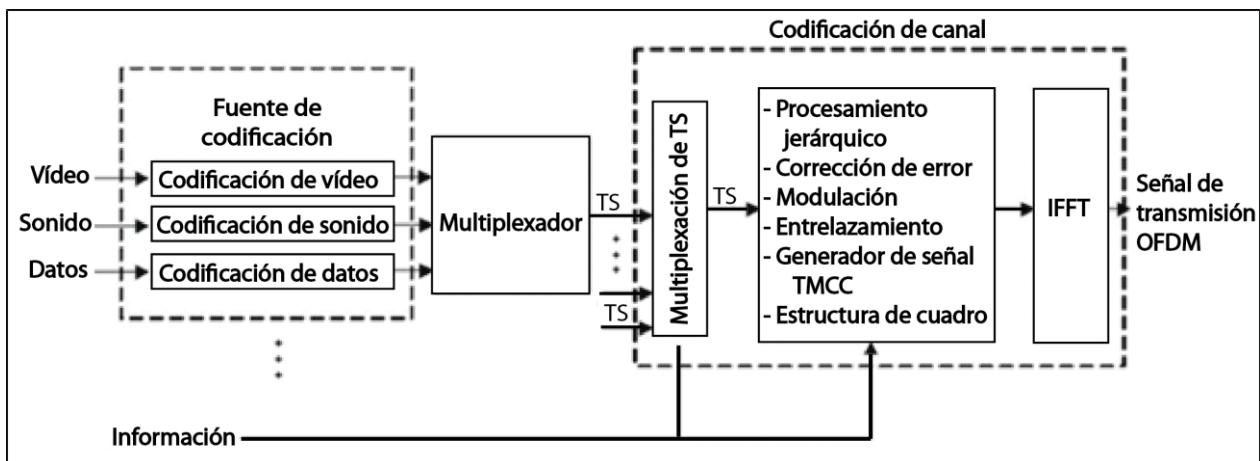


Figura 1.5. Diagrama en bloques del sistema de transmisión [15].

1.3.5.5. Flujo de transporte (TS) [13].

El transport stream – TS, es un formato especificado en MPEG-2 cuyo objetivo es multiplexar el vídeo, audio y datos, y sincronizarlos para obtener un solo flujo de transporte.

Primero se deben comprimir las señales de audio y vídeo, y se da formato a los datos, de este modo se obtienen el ES (Elementary Stream) de cada una de las tres señales; se crean tramas de datos de longitud variable PES y se realiza la multiplexación de audio, vídeo y datos en un solo flujo de paquetes de longitud constante de 188 bytes, que es el TS de un solo programa.

Si se requiere el envío de más programas en diferentes formatos (one-seg, señal SD, señal HD, datos e interactividad), se necesita un segundo nivel de multiplexación el cual se lo realiza entre diferentes TS. El proceso a seguir para la formación de archivos con formato TS se describe en la figura 1.6.

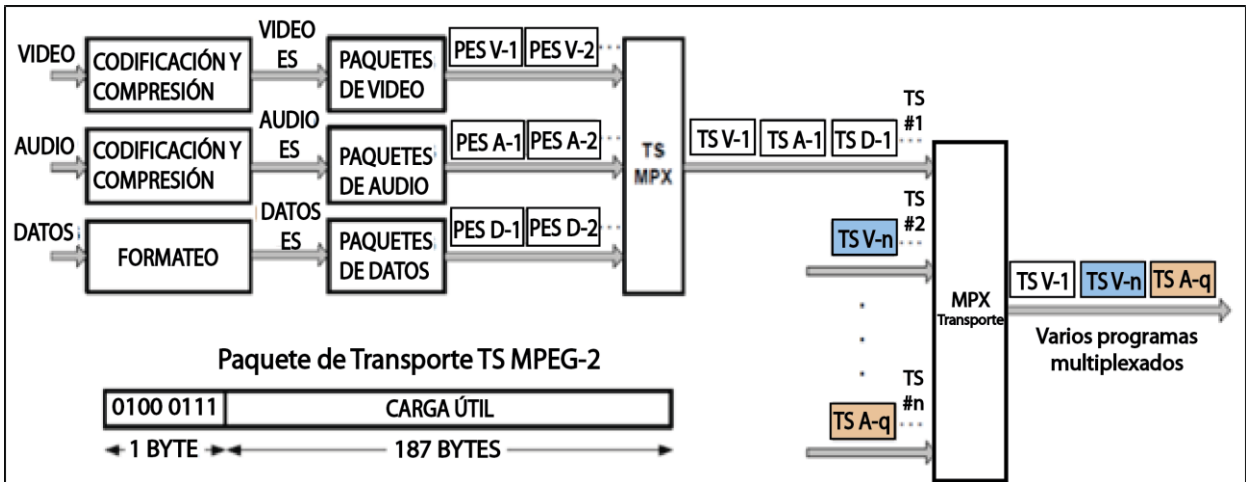


Figura 1.6. Formación del flujo de transporte TS para ISDB-Tb [13].

1.3.5.6. Recepción parcial [15].

El segmento central del espectro que consiste en 13 segmentos, puede ser sometido al proceso de entrelazamiento de frecuencia sin la participación de las demás porciones del espectro de radiodifusión. Ese tipo de configuración permite la creación de un servicio portátil (one-seg), que consiste en una de las capas del servicio de televisión.

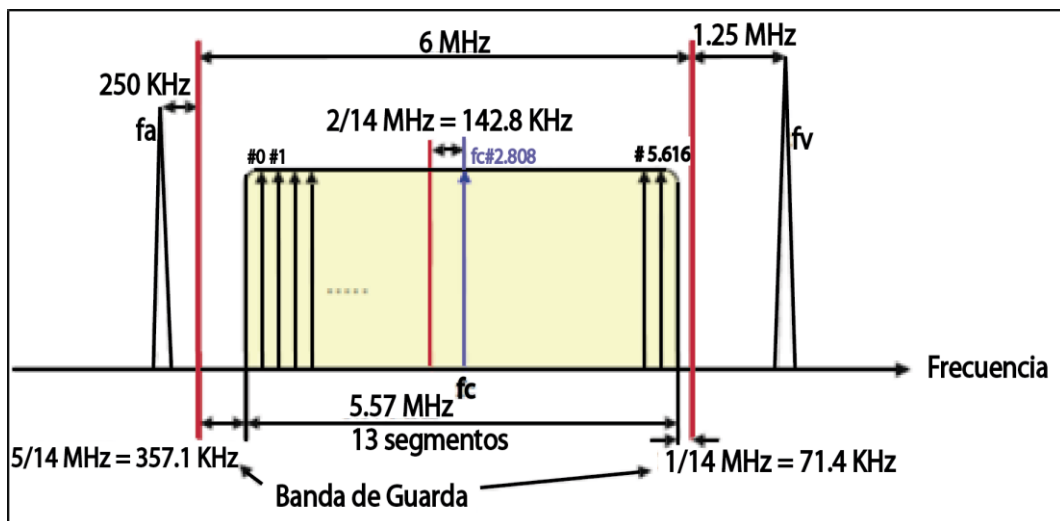


Figura 1.7. Espectro de frecuencia del canal de TV [13].

1.3.6. Tabla comparativa de los estándares de Televisión Digital Terrestre.

Tabla. 1.2. Resumen de las características técnicas de los estándares de televisión digital [11].

	ATSC	ISDB-T	DVB-T	SBTVD	DMBT
Compatibilidad con cable y satélite	NO	SI	SI	SI	NO
Codificación	MPEG-2	MPEG-2, MPEG-4 en desarrollo	MPEG-2 MPEG-4	MPEG-2 MPEG L2 MPEG-4	MPEG-2 MPEG-4
Interactividad Desarrollada	SI	SI	SI	SI	SI
Movilidad	No implementada	SI	SI	SI	SI
Canales de TV en celulares	0	1	Hasta 20	1	15
Carga digital	Fija 19 Mbps	Variable entre 4.98 y 31.67 Mbps	Variable entre 3.7 y 23.7 Mbps	Variable entre 3.65 y 23.23 Mbps	Variable entre 4 y 28 Mbps
Redes de frecuencia única	En desarrollo	SI	SI	SI	SI
Canales de Alta Definición	1	1/2	1/2	1	1
Recepción HDTV mientras se está en movimiento	Imposible	Posible	Imposible (solo con SDTV)	Posible	Posible
Recepción portátil utilizando el mismo sistema que recepción fija	Imposible	Posible	Imposible	Posible	
Ancho del canal	Canal de 6 MHz	Canal de 6 MHz	Canales de 8, 7 y 6 MHz	Canal de 6 MHz	Canal de 6 y 8 MHz
Modulación	8-VSB	Utiliza COFDM con QPSK/QAM	QPSK o diferentes niveles de QAM.	DQPSK, QPSK, 16-QAM y 64-QAM	TDS-OFDM (SD) y 8-VSB (HD)
Vídeo	MP@ML (SD) y MP@HL (HD)	MPEG-2	MPEG-2 MP@ML	H.264	MPEG-4 y MPEG-2
Audio	Sonido multicanal: 1, 2, 3, 4, 5.1 canales.	MPEG-2	MPEG Layer II. Dolby AC-3	AAC	MPEG-2 y AVS
Máxima resolución vertical	1080 líneas	1152 líneas	1152 líneas	1152 líneas	720p a 1080i
Máxima resolución horizontal	1920 pixeles	1920 pixeles	1920 pixeles	1920 pixeles	

Elaborado: 15 diciembre 2015.

1.4. Infraestructura de una red para televisión digital.

Para el desarrollo de la TVDi se requiere de ciertos componentes que conforman la infraestructura de red, estos son la infraestructura física y la infraestructura de software. Estos deben cumplir con ciertos requerimientos como bajo costo, deben ser orientados a manejar aplicaciones interactivas, emplear equipos de desarrollo, utilizar software libre y usar emuladores de elementos de hardware.

1.4.1. Infraestructura física.

Un sistema de televisión digital posee un mínimo de dos componentes necesarios: infraestructura del proveedor (Computador de producción y desarrollo, servidor de aplicaciones y contenidos, y el servidor de playout), y la infraestructura de usuario (Set Top Box, receptor de televisión).

En la figura 1.8 se muestra la infraestructura física de una red de televisión digital, cada stream de audio, vídeo y datos se multiplexan en un stream de transporte MPEG-2, éste se modula y se transmite para que sea recibida por una antena convencional, la señal se demodula y decodifica en el STB, el cual mediante el middleware reproduce el contenido del stream que ha recibido.

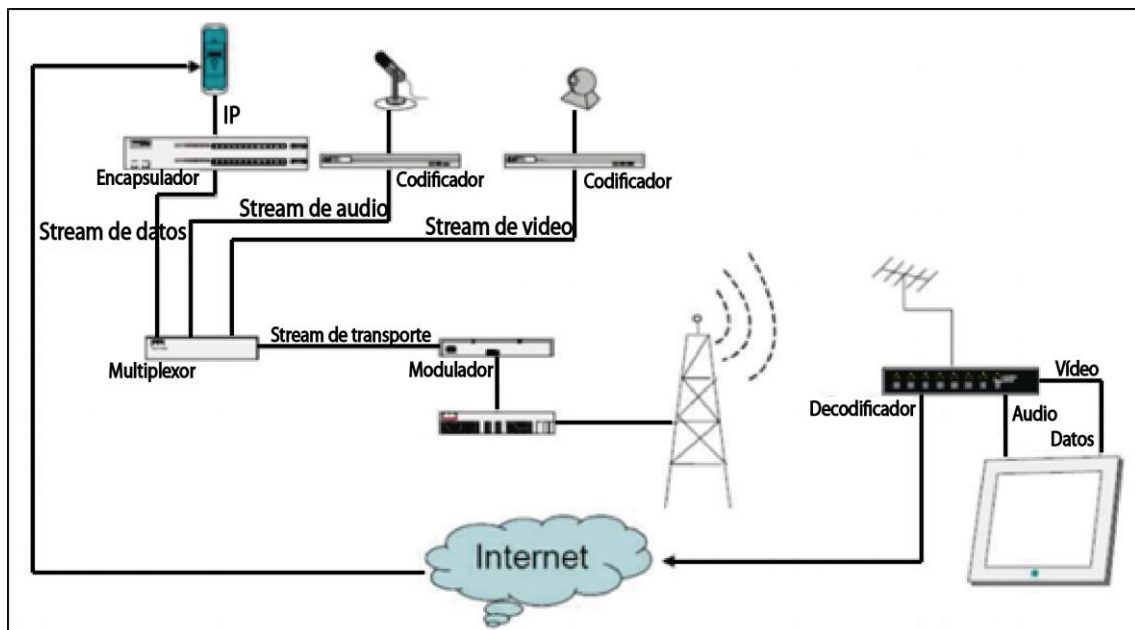


Figura 1.8. Infraestructura física de la red de televisión digital [9].

1.4.1.1. Computador de producción y desarrollo.

Es un computador donde se generan los contenidos digitales para TVDi que luego serán enviados al servidor de contenidos y aplicaciones. Se encarga de la elaboración, edición y producción de los contenidos y aplicaciones interactivas (menús, guías electrónicas de programación, T-Learning, entre otros).

Este computador es utilizado como un equipo de simulación, ya que al generar las aplicaciones correspondientes, estas se simulan en este equipo que cuenta con un emulador de STB para visualizar como se observaría la aplicación en el televisor. Para esto se requiere que el equipo cuente con un nivel de procesamiento y memoria elevada y una capacidad de almacenamiento adecuado, esto con el objetivo de que no existan errores de compilación del programa ni interrupciones en la reproducción de todos los archivos medios que serán utilizados. Es importante este punto ya que el computador debe almacenar gran cantidad de archivos de audio y vídeo pregrabados, imágenes e información adicional para el diseño de la aplicación [16].

1.4.1.2. Servidor de aplicaciones y contenidos.

Se trata de un equipo encargado del almacenamiento y distribución de las aplicaciones NCL que se enviarán en la transmisión de la señal de televisión digital, además es el encargado de la comunicación de la aplicación con lo que realice el usuario si se tratase de una aplicación que tenga interactividad remota (con canal de retorno), por lo que debe contener en su interior un software para realizar dicho proceso.

1.4.1.3. Servidor de playout.

Se encarga de la difusión y modulación de la señal de televisión. Genera el stream de transporte utilizando la norma MPEG-2 por medio de la multiplexación de datos provenientes de diferentes fuentes como son: streams de audio/vídeo, información de programa/servicio y el sistema de archivos de carrusel de objetos [9].

Este equipo es un conjunto de hardware muy robusto, pues demanda un procesamiento elevado, a altas tasas de transmisión y requiere funcionar continuamente. Es el encargado de unir la aplicación NCL con el flujo de audio y vídeo [9].

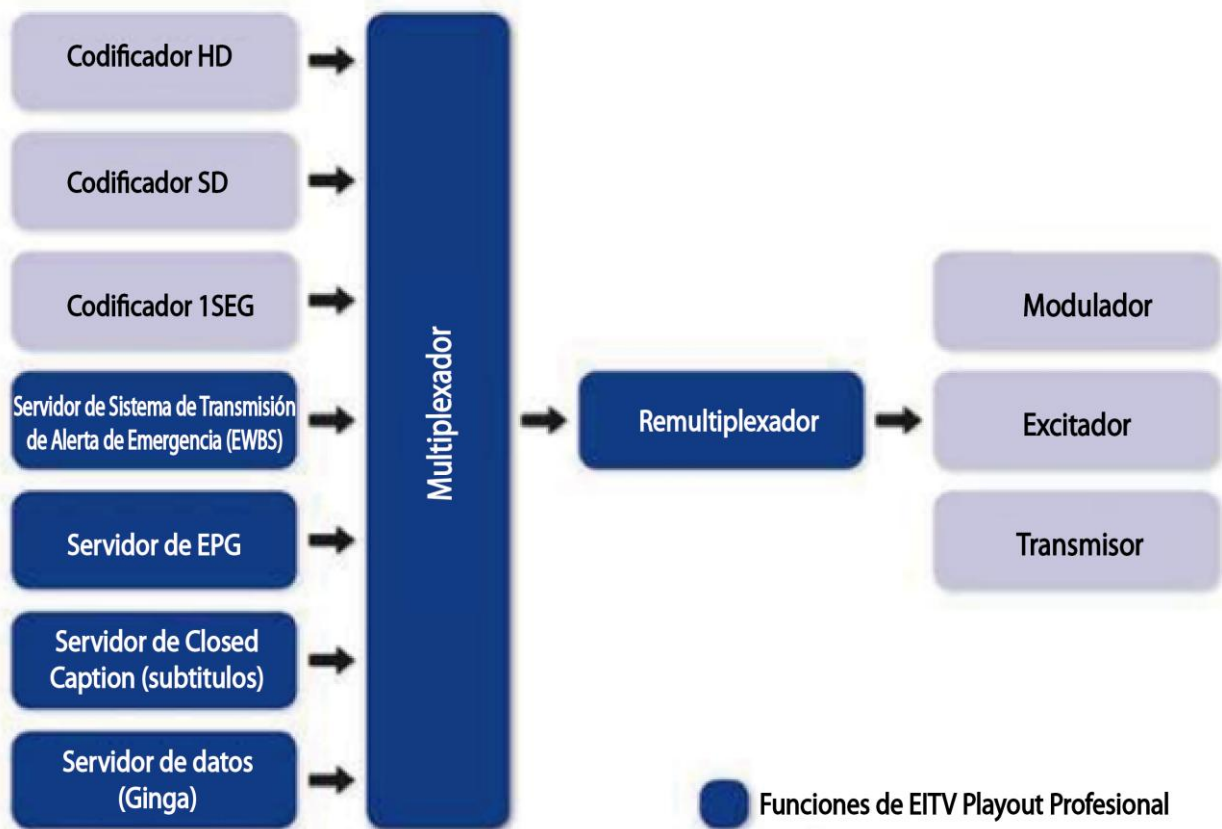


Figura 1.9. Funciones de un equipo comercial de PlayOut profesional [9].

En la figura 1.9 se muestran las funciones que puede hacer un servidor de PlayOut, aquí se puede codificar los distintos formatos de televisión a ser transmitidos en los cuales se destacan los formatos HD, SD y One Seg, además existe la opción de incluir los servicios interactivos como EPG, aplicaciones interactivas Ginga, otros.

Todos estos se multiplexan y se envían a un remultiplexador para luego ser modulados y transmitidos.

1.4.1.4. Set Top Box (STB) o equipo de usuario.

Este equipo es el encargado de recibir la señal digital de televisión, decodificarla y mostrarla en un televisor. Sobre este equipo se encuentra instalado el middleware necesario, el cual se encarga de la reproducción de aplicación que se ha decodificado con anterioridad.



Figura 1.10. Set Top Box comercial [9].

El STB tiene como función convertir las señales digitales de esta forma una TV analógica puede trabajar en un ambiente digital, además cuenta con lo que se denomina canal de retorno, el cual es una forma de comunicar al usuario con la televisora.

Las etapas involucradas en el procesamiento de la señal recibida por el STB inicia con la captación de la señal de televisión digital mediante el sintonizador, esta señal se demodula para eliminar el flujo de transporte MPEG-2. Cuando esto sucede el demultiplexor se encarga de extraer todos los elementary streams, estos pasan al decodificador, el cual se encarga de convertir las señales de audio, vídeo y datos en formato adecuado para su visualización.

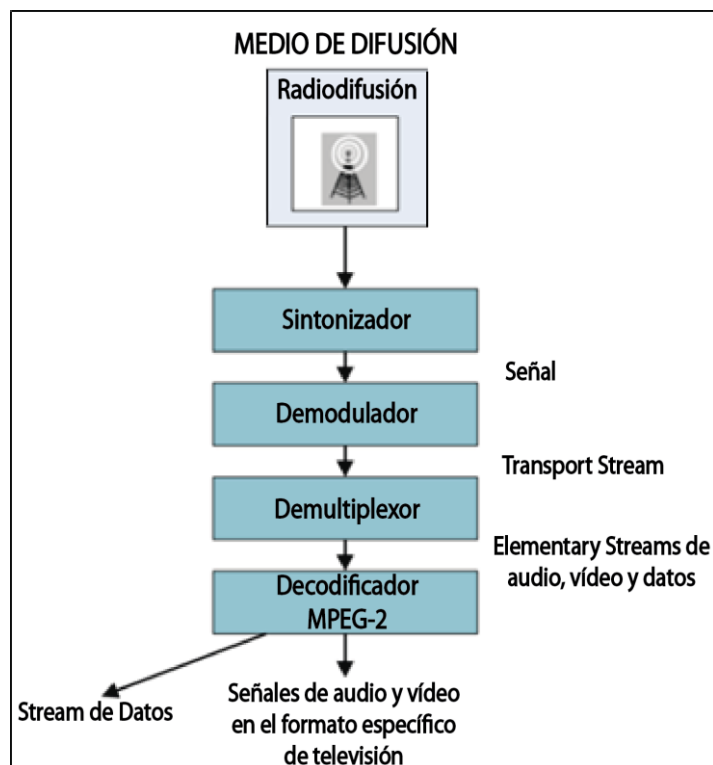


Figura 1.11. Etapas de recepción en el STB [16].

Existen otras opciones para receptores de televisión digital, los cuales pueden estar embebidos en el televisor, en el teléfono móvil (one-seg) o también pueden tener puertos USB que son utilizados para computadoras portátiles.

1.4.1.5. Receptor de televisión.

Los televisores que se pueden utilizar para acceder a esta tecnología digital pueden ser analógicos o digitales, para esto es necesario que cada uno de estos cuente con su respectivo receptor, estos pueden ser un receptor externo para TV analógica o un receptor digital integrado para TV digital.

- **Receptor externo o STB**, este se conecta al televisor analógico convencional.
- **Receptor digital integrado**, este receptor se incorpora dentro del chasis del televisor permitiendo así recibir tanto señales digitales como analógicas.



Figura 1.12. Televisores con receptores externo e interno [16].

Hay que tener en cuenta que los televisores que cuenten con el receptor digital integrado deben tener el estándar que ha sido adoptado por cada país, así mismo sucede con el STB. En nuestro caso es el estándar ISDB-Tb.

1.4.1.6. Tarjeta de televisión.

Este dispositivo permite sintonizar señales de televisión digital para poder visualizarlas en el computador. Este dispositivo permite sintonizar canales digitales de TV y de radio, graba la programación de un canal mientras se visualiza otro, puede pausar, retroceder y adelantar programas en vivo, guía electrónica de programación, entre otros [16].

Estas tarjetas de TV o sintonizadores son de dos tipos: sintonizadores de TV externos que utilizan para conexión un puerto USB y sintonizadores de TV internas, las cuales necesitan instalación en un slot PCI en el computador.

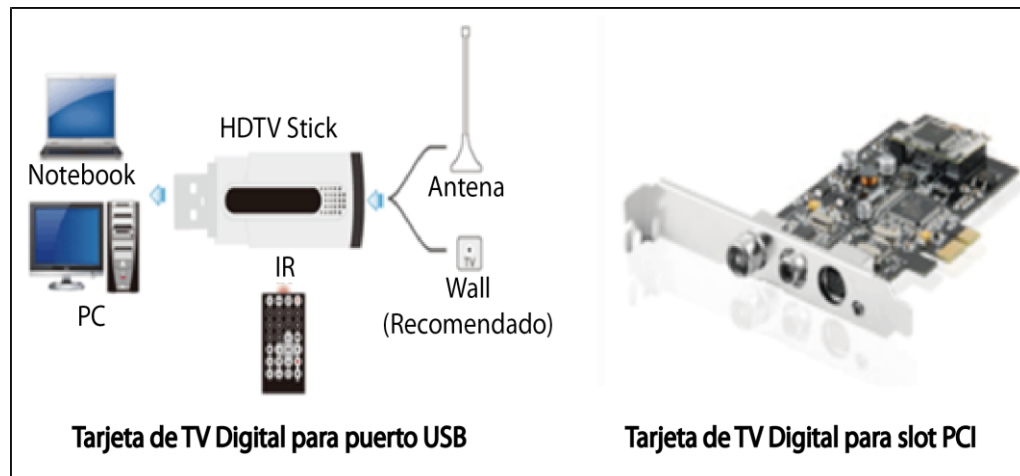


Figura 1.13. Tipos de tarjetas receptoras para televisión digital [16].

1.4.2. Infraestructura de software.

1.4.2.1. Software para el servidor de aplicaciones y contenidos.

Este software está encargado de [9]:

- *Empaquetamiento de aplicaciones NCL.*
- *Gestión del canal de difusión de aplicaciones (bajo demanda).*
- *Gestión de peticiones, almacenamiento y respuesta en el canal de retorno.*
- *Desarrollo de aplicaciones.*
- *Gestión de contenido multimedia.*
- *Proporcionan soporte a una variedad de estándares como HTML, XML, IIOP, JDBC, SSL, etc. los mismos que permiten su funcionamiento en ambientes web y aceptan conexión a una variedad de fuentes de datos, sistemas y dispositivos.*

Existen múltiples opciones para un servidor de contenidos, uno de ellos es usar VLC Media Player de la empresa Video Lan Organization, ya que este es un software libre que permite enviar un

stream de vídeo a una dirección IP, por telnet o a otro servidor, además puede convertir un vídeo en audio o vídeo en distintos formatos [9].

Hay que tener en cuenta que para que el servidor de contenidos tenga éxito, este debe cumplir con la normativa ISDB-Tb con la característica de poder transmitir más de un programa simultáneamente.

1.4.2.2. Software para el servidor de playout.

Es el encargado de codificar los contenidos, la multiplexación y la generación del carrusel de objetos, datos y eventos (similar a una máquina de estados).

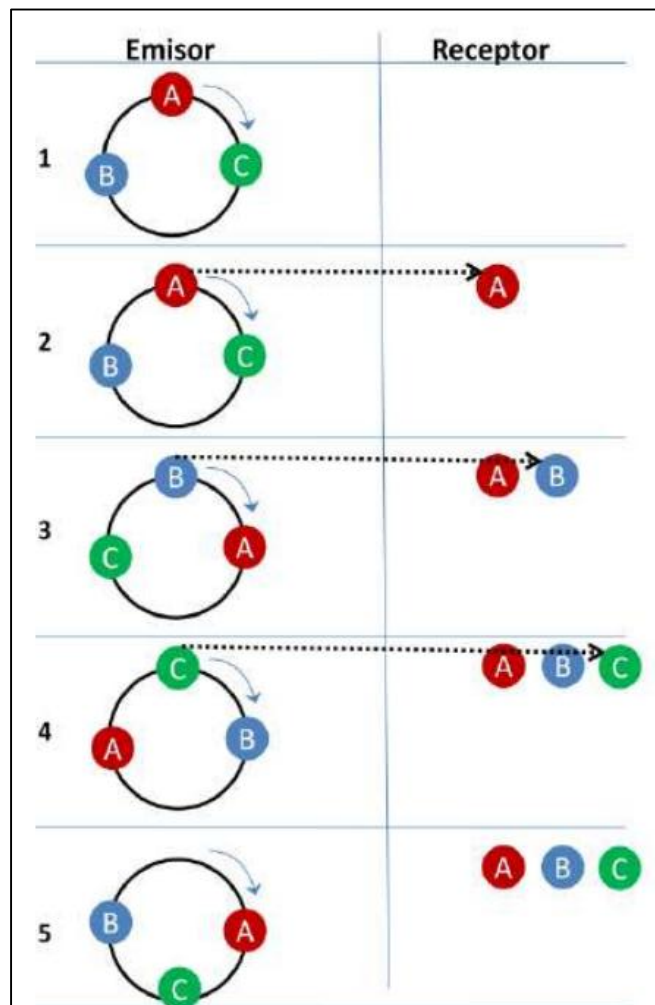


Figura 1.14. Transferencia de archivos a través del carrusel de objetos [9].

A continuación se realiza la descripción de los diferentes estados que cumple el carrusel de objetos de la figura 1.14:

- *Estado 1: Si el receptor está apagado como en este caso, no se recibirá ningún dato en la memoria del carrusel.*
- *Estado 2: El carrusel de objetos realiza su primer avance de giro y se transfiere el objeto A.*
- *Estado 3: Con un giro más del carrusel se transfiere el objeto B. El receptor lee su memoria y almacena los objetos que aún no han sido almacenados hasta que se carguen todos los archivos.*
- *Estado 4: El carrusel sigue girando y almacena el objeto C. El receptor repite el mismo procedimiento anterior comprobando en su memoria todos los registros hechos y los que faltan realizar.*
- *Estado 5: Cuando el receptor recibe el último identificador de archivo, la aplicación está lista para ejecutarse.*

OpenCaster es un software de la empresa Avalpa Digital Engineering que cumple con características de carrusel de objetos y multiplexación de contenidos [9].

1.5. Transición hacia la Televisión Digital Terrestre en Ecuador.

La televisión en Ecuador se ha convertido en el aparato máspreciado según un estudio socioeconómico de la ex SUPERTEL, y aunque para muchos no es novedad disponer de una guía electrónica de programación o servicios de información, para la gran mayoría de la población no es posible solventar los costos de estos servicios de televisión pagados, es ahí donde radica la importancia de la TDT, estos servicios que se podrán brindar mediante la tecnología digital serán totalmente abiertos a todo público sin necesidad de pago.

La ex SUPERTEL fue la delegada en realizar los análisis, pruebas y recomendaciones para definir el estándar más adecuado que beneficie de forma general a todas las brechas sociales, es así que Ecuador realizó el préstamo de equipos de transmisión de televisión digital desde el Gobierno de Japón y la Comunidad Europea que utilizan los estándares ISDB-T y DVB-T respectivamente [17][18].

Se realizaron las pruebas con los equipos importados y con cada estándar mencionado anteriormente, el 22 de abril del 2009, los equipos importados desde Japón fueron acondicionados para realizar pruebas con el sistema SBTVD (Sistema Brasileño de Televisión Digital) y luego el 24 de junio del 2009 se realizó pruebas con el estándar Chino DTMB; por problemas de coordinación y de no llegar a un acuerdo no se realizaron pruebas con el estándar norteamericano ATSC [18].

Luego de realizar los estudios, análisis y comparaciones necesarias entre los diferentes estándares que fueron probados, la SUPERTEL recomienda adoptar el estándar ISDB-Tb (estándar Japonés con variaciones brasileñas), es así que la CONARTEL el 26 de marzo del 2010 acepta el estándar por prestar una mejor calidad de señal, mejor rendimiento en condiciones indoor², mayor número de canales, interacción de usuarios y operadores, mayor flexibilidad en recepción móvil, entre otros [18].

Algunos de los parámetros que se tomaron en cuenta para evaluar cada estándar y elegir el mejor, son los siguientes:

- **Intensidad de campo** [19].

En la figura 1.2, se muestra los resultados obtenidos tras realizar las pruebas en 85 puntos en modo fijo exterior. El estándar chino DTMB es superior a otros estándares obteniendo el 39 % de los puntos que equivale a 33 puntos de medición, seguido del estándar ISDB-Tb con un 37 % que equivale a 32 puntos de medición y a continuación los estándares DVB-T e ISDB-T con el 12 % de los puntos que equivale a 10 puntos de medición cada uno.

² Condiciones indoor: interferencias y obstáculos que pueden atenuar la señal recibida.

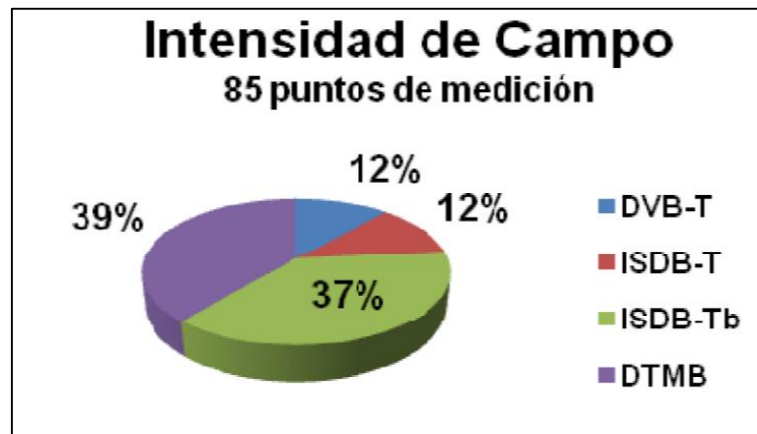


Figura 1.15. Porcentaje de intensidad de campo de los diferentes estándares [32].

- **Relación Señal a Ruido** [19].

En la figura 1.3, el estándar ISDB-Tb obtuvo un porcentaje del 64 % equivalente a 54 de los 85 puntos de medición, el estándar DTMB obtuvo el 22 % equivalente a 19 puntos de medición, el estándar ISDB-T obtuvo un 9 % equivalente a 8 puntos de medición, seguido por el estándar DVB-T con un porcentaje de 4 % equivalente a 3 puntos de medición y finalmente se hace referencia a un porcentaje de medición de 1 %, en el cual se indica que se ha obtenido un valor idéntico en un punto de medición tanto en ISDB-T e ISDB-Tb.

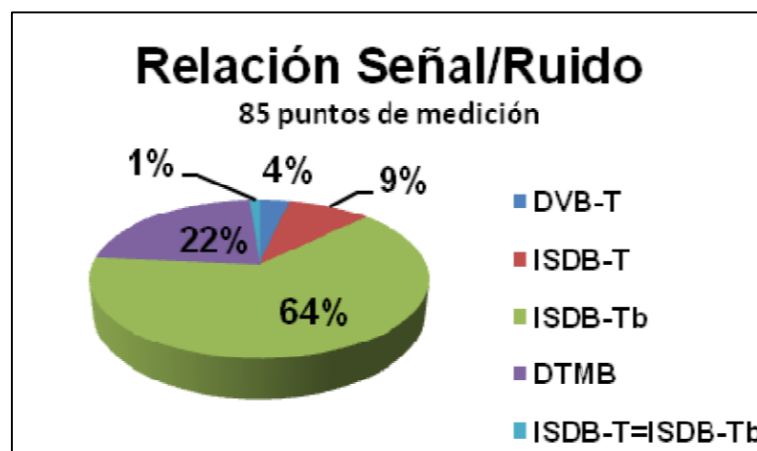


Figura 1.16. Porcentaje de relación señal a ruido de los diferentes estándares [32].

- **Evaluación subjetiva de la calidad de vídeo y audio en modo fijo exterior [19].**

En cada una de las siguientes tablas se muestran los resultados subjetivos realizados por la ex SUPERTEL en la evaluación de audio y vídeo.

Tabla. 1.3. Resumen de la votación correspondiente a la calidad de vídeo, total 255 votos, modo fijo exterior [19].

Calidad	DVB-T	ISDB-T	ISDB-Tb	DTMB
Excelente	83 (32.5%)	164 (64%)	216 (85%)	202 (79%)
Buena	96 (38%)	30 (12%)	23 (9%)	21 (8.2%)
Regular	12 (4.5%)	1 (0.4%)	8 (3%)	2 (0.8%)
Pobre	4 (1.5%)	1 (0.4%)	5 (2%)	6 (2.5%)
Mala	60 (23.5%)	59 (23.2%)	3 (1.2%)	24 (9.5%)

Elaborado: 20 diciembre 2015.

Tabla. 1.4. Resumen de la votación correspondiente a la calidad de audio, total 255 votos, modo fijo exterior [19].

Calidad	DVB-T	ISDB-T	ISDB-Tb	DTMB
Excelente	170 (66.6%)	163 (64%)	232 (91%)	216 (85%)
Buena	18 (7.1%)	27 (10.4%)	13 (5%)	7 (2.5%)
Regular	2 (0.8%)	5 (2%)	5 (2%)	2 (0.8%)
Pobre	2 (0.8%)	1 (0.4%)	2 (0.8%)	4 (1.5%)
Mala	63 (24.7%)	59 (23.2%)	3 (1.2%)	26 (10.2%)

Elaborado: 20 diciembre 2015.

Tabla. 1.5. Resumen de la votación correspondiente a la calidad de audio, total 60 votos, modo fijo interior sin la presencia de ruido impulsivo [19].

Calidad	DVB-T	ISDB-T	ISDB-Tb	DTMB
Excelente	22 (36.6%)	30 (50%)	42 (70%)	32 (53.3%)
Buena	12 (20%)	9 (15%)	5 (8.4%)	4 (6.7%)
Regular	4 (6.7%)	6 (10%)	1 (1.6%)	3 (5%)
Pobre	4 (6.7%)	0 (0%)	0 (0%)	3 (5%)
Mala	18 (30%)	15 (25%)	12 (20%)	18 (30%)

Elaborado: 20 diciembre 2015.

Tabla. 1.6. Resumen de la votación correspondiente a la calidad de audio, total 60 votos, modo fijo interior con la presencia de ruido [19].

Calidad	DVB-T	ISDB-T	ISDB-Tb	DTMB
Excelente	29 (33.3%)	30 (50%)	39 (65%)	31 (51.6%)
Buena	8 (13.3%)	5 (8.3%)	3 (5%)	4 (6.7%)
Regular	3 (5%)	7 (11.7%)	6 (10%)	3 (5%)
Pobre	8 (13.3%)	0 (0%)	0 (0%)	2 (3.4%)
Mala	21 (35%)	18 (30%)	12 (20%)	20 (33.4%)

Elaborado: 20 diciembre 2015.

Tabla. 1.7. Resumen de la votación correspondiente a la calidad de vídeo, total 60 votos, modo fijo interior sin la presencia de ruido [19].

Calidad	DVB-T	ISDB-T	ISDB-Tb	DTMB
Excelente	17 (28.3%)	28 (46.7%)	38 (63.3%)	30 (50%)
Buena	15 (25%)	10 (16.6%)	5 (8.3%)	6 (10%)
Regular	5 (8.3%)	6 (10%)	5 (8.3%)	3 (5%)
Pobre	6 (10%)	1 (1.6%)	0 (0%)	3 (5%)
Mala	17 (28.3%)	15 (25%)	12 (20%)	18 (30%)

Elaborado: 20 diciembre 2015.

Tabla. 1.8. Resumen de la votación correspondiente a la calidad de vídeo, total 60 votos, modo fijo interior con la presencia de ruido [19].

Calidad	DVB-T	ISDB-T	ISDB-Tb	DTMB
Excelente	15 (25%)	26 (43.3%)	37 (61.7%)	30 (50%)
Buena	9 (15%)	9 (15%)	3 (5%)	5 (8.3%)
Regular	7 (11.7%)	6 (10%)	3 (5%)	3 (5%)
Pobre	9 (15%)	1 (1.6%)	5 (8.3%)	3 (5%)
Mala	20 (33.3%)	18 (30%)	12 (20%)	19 (31.7%)

Elaborado: 20 diciembre 2015.

Algunas de las razones por las que se adopta este estándar se mencionan a continuación [10]:

- Menores costos de instalación de infraestructura, lo que hace que los consumidores y radiodifusores tengan acceso a los beneficios que este estándar ofrece.

- *Alta calidad de audio y vídeo en sus modalidades de alta definición HD y definición estándar SD, además de su eficiencia del uso del espectro (portabilidad) y su capacidad de convergencia de servicios (movilidad).*
- *Transmisión simultánea de canales HD y SD, además del servicio de alerta temprana y su canal para dispositivos móviles.*
- *Acceso a internet para tener un sistema interactivo con canal de retorno, y así hacer del usuario alguien más activo cuando se siente frente a la TV.*
- *Cada canal puede transmitir hasta ocho canales de televisión por lo cual se ofertaran una mayor cantidad de programas.*
- *Nuevos servicios como correo electrónico, información de cotizaciones de bolsa, EPG, vídeo pagado, etc.*
- *Oportunidad productiva y comercial para la industria por la demanda de decodificadores y receptores, además existirá una gran demanda de desarrolladores de interactividad para cada programa o cada proveedor que quiera ofertar algún tipo de información comercial o educativa en TV.*
- *Servicios de entretenimiento, informativos, de publicidad, inclusión económica y social, ya que la TDT está disponible para todo público sin excepción.*

1.5.1. Plan Maestro de Transición a la Televisión Digital Terrestre en Ecuador.

Según el plan maestro de transición a la TDT en Ecuador, se destacan los puntos más importantes:

1.5.1.1. Generalidades [20][3].

El objetivo general trata de establecer las condiciones ideales para el proceso de transición hacia la televisión digital.

El Plan Maestro de transición a la TDT en Ecuador está orientado a buscar la optimización del espectro radioeléctrico, realizando una mejor distribución y aprovechamiento del mismo, donde se piensa distribuir el 33% en canales públicos, 33% en canales privados y 34% en canales comunitarios, esto para reducir la brecha digital.

1.5.1.2. Implementación [20][3].

Los marcos regulatorios serán las leyes y reglamentos vigentes, Constitución de la República del Ecuador 2008, Ley Orgánica de Comunicación 2013 y los Reglamentos y Resoluciones implementadas por la CONATEL.

Se realiza el proceso de simulcast, el cual consiste en dar continuidad a los operadores del servicio televisivo hasta que llegue la fecha del apagón analógico, esto con el fin de que estos operadores puedan generar el contenido necesario que será utilizado en la televisión digital (aplicaciones de interactividad, los contenidos informativos, etc.).

1.5.1.3. Espectro radioeléctrico [20][3].

En este punto se realizará la asignación de canales dependiendo de las bandas de frecuencias dispuestas en el espectro radioeléctrico, según la Ley Orgánica de Telecomunicaciones en el artículo 18, el espectro es de dominio público del Estado considerado como un recurso natural limitado que necesita administración, control, gestión y regulación [21].

1.5.1.4. Fases del apagón analógico [20][3].

El Plan Maestro de transición de la TDT preverá de equipos de transmisión (canales) y recepción (usuario), esto con el fin de aprovechar esta tecnología al máximo mediante las aplicaciones que serán desarrolladas mediante entidades independientes, los mismos canales o cualquier ente que pueda ofrecer este servicio, siempre y cuando este regulado por el Consejo de Regulación, Desarrollo de la Información y Comunicación, según lo estipula la Ley Orgánica de Comunicación.

Las fases del apagón analógico según el Plan de la Transición hacia la TDT son las siguientes:

Tabla. 1.9. Resumen de Las fases del apagón analógico según el Plan de la Transición hacia la TDT [20].

	FASE 1	FASE 2	FASE 3
Fecha	31 de diciembre del 2016	31 de diciembre del 2017	31 de diciembre del 2018
Cobertura	Capital de provincia, cabecera cantonal o parroquia	Capital de provincia, cabecera cantonal o parroquia	Capital de provincia, cabecera cantonal o parroquia
Población	Mayor a 500 000 habitantes	De 500 000 y 200 000 habitantes	Menor a 200 000 habitantes

Elaborado: 20 de diciembre 2016.

1.5.1.5. Ventajas de la Televisión Digital.

Entre las principales ventajas que presenta la televisión digital con respecto a la analógica tenemos:

- *Permite la opción de “multitransmisión”, es decir, con la televisión digital se pueden acceder varios programas a la vez en un mismo canal.*
- *La imagen y audio son de mejor calidad gracias a su formato de compresión AVC y AAC; además la resolución en pantalla pasara de 720 x 480 pixeles (TV analógica) hasta 1920 x 1080 pixeles (TV digital).*
- *Las interferencias que se presentan actualmente, problemas con ruidos o presencia de dobles imágenes ya no existirán con la tecnología digital.*
- *Se utiliza el mismo ancho de banda de 6 MHz para transmitir más programas, los mismos que podrán ser accedidos mediante una Guía electrónica de Programación, habrá la posibilidad de que existan más operadoras de TV.*
- *El usuario podrá interactuar con la televisión dependiendo de los servicios que ofrezca cada programa (cambio de idioma, votaciones, compras por TV, acceso a información, etc.), para ello su televisor deberá tener un STB e incluir el estándar necesario.*
- *Permite interactividad, lo cual hace que los usuarios tengan acceso a información adicional a la que se observa en la televisión, con la posibilidad de participar e influir en los programas de televisión. En el apartado 2.2 se realiza un estudio de los aspectos que intervienen en la interactividad para televisión digital.*

CAPÍTULO II

2. TELEVISIÓN DIGITAL INTERACTIVA

2.1. Definición.

La TVDi es una realidad de la llamada “Sociedad de la Información”³ que funciona a partir de la difusión de la señal de televisión directa, de las redes por cable y de la televisión digital terrestre, junto con las mejoras en la calidad de recepción y visualización de contenido. Se espera que se permita el desarrollo de aplicaciones interactivas que se podrán utilizar en la vida cotidiana como en la educación formal y no formal [22].

Se pretende que la TVDi sea considerada como una herramienta tecnológica de recursos ilimitados que hace que el telespectador pueda hacer algo más que solo ver y oír televisión, pasando a ser un elemento activo en la red televisiva teniendo la posibilidad de realizar acciones que pueden cambiar el contenido mostrado por su televisor.

2.1.1. Elementos tecnológicos de la TVDi.

- **Interactividad.** *Permite la interacción entre el televisor y el televidente.*
- **Servicios interactivos.** *Almacenamiento y preparación de las aplicaciones interactivas que serán emitidas al aire, este puede estar conectado o no al canal de retorno, esto dependerá del tipo de interactividad que se esté ofreciendo en el programa de televisión*
- **Middleware.** *Forma parte del STB, es el encargado de reproducir y de dar vida a las aplicaciones interactivas.*
- **Canal de retorno.** *Permite el envío de datos desde el usuario hacia el servidor donde se encuentra la aplicación con la que el televidente esta interactuando.*
- **Set Top Box.** *Decodifica y demultiplexar la señal para convertirla en analógica (audio y vídeo) de alta definición, contiene el middleware y permite el uso del canal de retorno.*

En el presente capítulo se hará énfasis en dos de los elementos tecnológicos de la TVDi. En el apartado 2.2 se analizará el término de Interactividad, fundamental en el desarrollo de este Proyecto de Titulación y en el apartado 2.3 se estudiará los diferentes tipos de servicios interactivos que nos ofrece la TVDi.

³ Sociedad de la Información, es aquella que tiene la facultad de crear, distribuir y manipular la información haciendo de esta una herramienta muy importante en las actividades sociales, culturales y económicas.

2.2. Interactividad.

La interactividad es la capacidad para ofrecer al telespectador contenido adicional durante los programas de televisión, la misma que es enviada durante el proceso de emisión y que sólo es visible si el espectador lo desea.

Se puede intercambiar información accediendo a servidores a través de un canal de retorno el mismo que hace que el espectador participe activamente del programa según sea el uso que haga de la información. Las aplicaciones interactivas permiten al usuario observar información asociada al contenido audiovisual, programación de los canales (EPG), participar en concursos, votaciones, comprar productos o servicios e incluso participar del programa en tiempo real [23].

2.2.1. Interactividad local [24].

Son aquellas aplicaciones transmitidas por broadcast desde la estación de TV a cada uno de los STB donde las aplicaciones son descargadas automáticamente.

Este tipo de interactividad es accesible a todos los usuarios ya que no requiere de un canal de retorno (no requiere una conexión a internet), de esta manera se permitirá la inclusión social. En este nivel de interactividad se pueden cargar un sin número de aplicaciones informativas, publicitarias, EPG, juegos, entre otros, el usuario escoge cuando acceder a esta información.

La interactividad del usuario con la TV no cambia el contenido o su linealidad, sino solo la forma en que los usuarios controlan la visualización de ese contenido.



Figura 2.1. Aplicación con interactividad local [25].

2.2.2. Interactividad unidireccional.

En este tipo de interactividad solamente es posible enviar datos mediante el canal de retorno (ADSL, satélite, otros). Se puede tener sitios web y servicios de teléfonos móviles coordinados en simultáneo con los programas de TV, de este modo es posible enviar información a la televisora solo si el usuario lo requiere. Algunas aplicaciones para este tipo de interactividad son: votaciones (altera el resultado del programa de TV), encuestas, otros.



Figura 2.2. Aplicación con interactividad con unidireccional [26].

2.2.3. Interactividad remota o con canal de retorno.

La interactividad remota permite además del acceso al contenido interactivo y navegar en éste, es posible enviar solicitudes o respuestas a través del canal de interactividad. El usuario interactúa con el proveedor de servicios externo permitiendo de este modo navegar por la web, enviar y recibir mensajes, participar en concursos e incluso en la dirección que pueda tener ciertos programas. Se proporciona lo que se denomina “TV social”, ya que los televidentes de un programa pueden intercambiar datos entre sí [10].

En este tipo de interactividad se incluyen programas de televisión donde los espectadores son quienes eligen o influyen los detalles del contenido y la forma de transmisión; aplicaciones para redes sociales, compras por TV (T-Commerce), cuentas bancarias, vídeo bajo demanda, otros.



Figura 2.3. Aplicación con interactividad remota [34].

2.3. Servicios interactivos.

La TVDi soporta múltiples servicios para el televidente, estos se clasifican dependiendo de la información interactiva que contiene la aplicación. En la tabla 2.1 presenta los distintos servicios interactivos que existen para televisión digital.

Tabla. 2.1. Resumen de los diversos servicios interactivos para televisión digital.

Servicios	Definición	Servicios	Definición
Informativos	Proporcionan información independientemente de lo que se esté transmitiendo. Contenido meteorológico, avisos de emergencia, etc.	T-Voting	Permiten realizar cuadros estadísticos en base a la opinión ciudadana sobre temas de interés público. Aplicaciones de encuestas.
Ligados a la programación	Proporcionan información complementaria sobre el contenido que se está observando. EPG, material multimedia, otros.	T-Health	Proveen asistencia médica a través del televisor. Información sobre enfermedades, alimentación, solicitud de citas médicas, otros.
Transaccionales	Ofrecen la posibilidad de enviar y recibir información de forma personalizada y exclusiva.	T-Banking	Se puede realizar transacciones bancarias. Consulta de saldos, pago de servicios, transferencia de dinero, otros
T-Commerce	Se realizan compras de productos que se ofertan en televisión. Las empresas (PayPal, Amazon) se dedican al comercio electrónico.	T-Government	Se realizan encuestas de carácter político o se puede informar sobre los procesos del gobierno, informes, consultas o votaciones electrónicas.
T-Administration	Se realizan tramites de gestión de impuestos, solicitud de impresos a administraciones públicas, diligencias administrativas, otros.	De entretenimiento y comerciales	Aplicaciones interactivas que permiten jugar, participar en concursos, trivias, salas de chat, información sobre concursantes de algún programa, otros.
T-Learning	Se ofrece al televidente educación para niños de todas las edades a través del televisor. Las aplicaciones se basan en juegos interactivos con contenido didáctico, educativo y recreativo para la formación intelectual.		

Fuente: Creado por el autor.

Elaborado: 20 de diciembre 2016.

2.4. Características de diseño de una aplicación interactiva.

Para diseñar una aplicación interactiva, se deben tener en cuenta múltiples aspectos que ayudan a captar la atención del usuario, esto dependerá del tipo de programa que el usuario observe.

2.4.1. Layout de pantalla.

Existen algunos tipos de diseños para presentación en pantalla, estos se presentan en la tabla 2.2. Se diferencian entre sí dependiendo de la secuencia del programa de TV, aquí unas ventajas y desventajas de los mismos.

Tabla. 2.2. Tipos de diseño de layout de pantalla.

	Ventajas	Desventajas
<i>Superposición</i>	<i>El vídeo de fondo sigue en pantalla completa [27]. Mayor integración – aplicación de escritura de texto sobre vídeo.</i>	<i>Oculto partes importantes del vídeo. Distractor por cercanía, al usar un contraste inadecuado para el texto, no se visualizaría ni el vídeo ni el texto.</i>
<i>Pantalla completa con vídeo</i>	<i>Se puede seguir el vídeo o programa que se esté emitiendo. Facilita la atención dividida (multitarea), vídeo y texto [27].</i>	<i>Vídeo reducido, ¼ del total de pantalla (modificable). Gráficos en el flujo de vídeo son muy pequeños para ser leídos [27].</i>
<i>Pantalla completa sin vídeo</i>	<i>Adecuado para mostrar grandes contenidos [27]. Visualización en pantalla completa de la aplicación.</i>	<i>Pérdida total del programa que se está observando [27]. Son muy pocas las personas que utilizan este método.</i>

Fuente: Creado por autor.

Elaborado: 10 enero 2016.

2.4.1.1. Superposición.

Se utiliza para mostrar pequeñas cantidades de textos sobre el vídeo que se encuentra reproduciéndose.

- **Posición:** En la parte inferior de la pantalla con un ancho de pantalla completa, o a la derecha de la pantalla con toda su altura. Si los textos son pequeños, en la parte inferior de la pantalla o en la esquina superior derecha.
- **Tamaño:** Depende de la cantidad y legibilidad de los contenidos, y de la visibilidad del vídeo de fondo.
- **Transparencia:** Dependerá del contenido del texto y de la calidad del vídeo, ya que en base a esto se elegirá un nivel de transparencia.

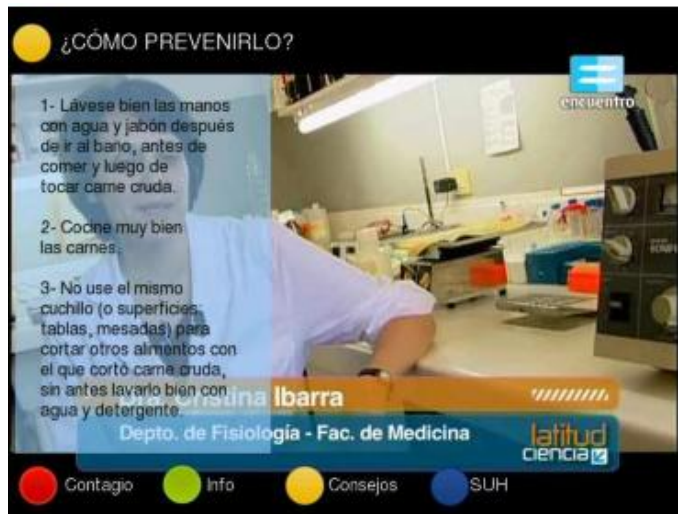


Figura 2.4. Layout de pantalla con superposición.

Fuente: <http://www.slideshare.net/movimientodisenoinclusivo/patrones-de-diseo-de-interaccin-para-tv-digital-interactiva>

2.4.1.2. Pantalla completa con vídeo.

En aplicaciones que pueden ser recibidos por un grupo de usuarios. Mientras un usuario está revisando la aplicación, los demás pueden seguir observando el programa de TV, aunque pueden existir problemas debido a que el vídeo es muy pequeño, la visibilidad de imágenes o logos que puedan salir en la TV, no será muy legible.



Figura 2.5. Layout de pantalla completa con vídeo.

Fuente: <http://www.slideshare.net/movimientodisenoinclusivo/patrones-de-diseo-de-interaccin-para-tv-digital-interactiva>

- **Posición:** En la parte superior izquierda o derecha de la pantalla se puede exhibir el vídeo y el resto de la aplicación en el espacio sobrante.
- **Tamaño:** Los tamaños comunes son: 1/3, 1/4, 1/8, y 1/16 del tamaño de la pantalla original, esto dependerá del punto principal de atención si el vídeo o la aplicación.

2.4.1.3. Pantalla completa sin vídeo.

Cubre totalmente la pantalla de la TV, aunque este método se usa muy poco es muy útil para aplicaciones de mejoras de televisión.

- **Audio:** Debe desaparecer completamente el audio del vídeo que se encuentra en la programación, dando paso solamente al audio de la aplicación.

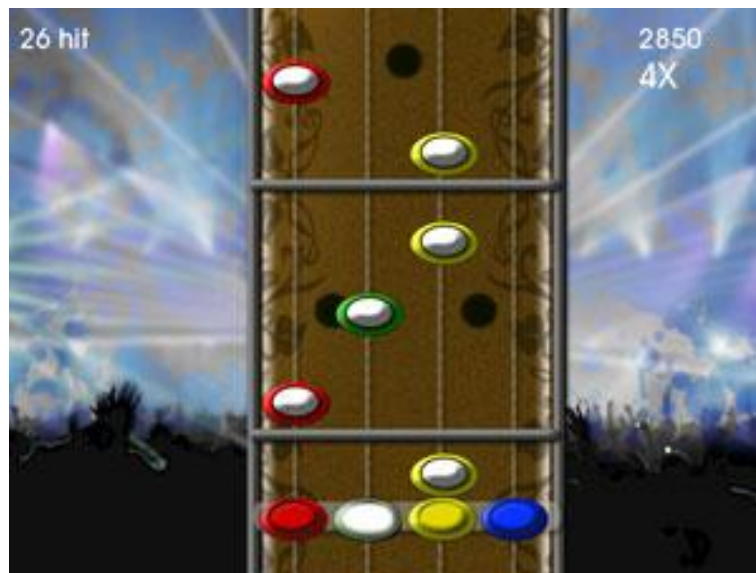


Figura 2.6. Layout de pantalla completa sin vídeo. Ginja Hero o Guitar Hero para televisión digital.

Fuente: <http://g1.globo.com/tecnologia/noticia/2010/11/estudantes-apostam-na-criacao-de-games-para-tv-digital-brasileira.html>

2.4.2. Navegación.

Cuando ya se haya elegido el layout, se procede a escoger el tipo de navegación con el cual la aplicación va a funcionar, esto dependerá del tipo de contenido que se va a ofrecer a los usuarios.



Figura 2.7. Layout de pantalla con navegación mediante listado y menú.

Fuente: <http://www.redusers.com/noticias/wp-content/uploads/2012/03/interactivo.jpg>

La navegación en pantalla se la puede diseñar de diversas formas y estilos, cada una de estas opciones de diseño cuenta con sus características.

2.4.2.1. Menú.

- **Presentación del menú:** Menú estático, hace visible todo el contenido pero necesita de más espacio en pantalla; menú dinámico, no requieren de mucho espacio en pantalla, aunque no se visualiza toda la información. Pueden ser presentados de forma horizontal o vertical.
- **Número de elementos:** Depende del espacio que se le dé al menú en pantalla, recomendable de 5 a 7 opciones por menú, ya que si se excede puede ocasionar confusiones.
- **Redacción de los elementos del menú:** Debido a que el espacio es reducido en las etiquetas del menú, los textos deben ser cortos, palabras clave.
- **Transparencia:** Texto legible acompañado de un buen contraste con el vídeo de fondo de pantalla que hay que tener en cuenta que es cambiante.

2.4.2.2. Vídeo Multi-pantalla.

- **Rectángulo:** Cuadrícula o tabla de vídeo simple.

- **Secuencia de vídeo grande:** Un vídeo más grande que otros sin dejar de ofrecer una visión general, se centra en el vídeo general.
- **Lista vertical:** Diseño para incluir información a cada secuencia de vídeo.
- **Tamaño de flujos de vídeo:** Depende del espacio en pantalla disponible y la visibilidad de las acciones del vídeo.
- **Encabezamiento:** Título corto y conciso para cada vídeo.
- **Audio:** Cada vez que se seleccione algún vídeo, debe tener el audio activado, esto como guía de lo que ofrece cada uno de ellos.

2.4.2.3. Índice.

- **Estructura:** Se presenta mediante una lista alfabética (A, B, C, etc.) y las clases mediante grupos de letras (ABC, DEF, etc.).
- **Destacando un elemento elegido:** Para usuarios que tengan un lugar específico al que siempre visitan, así se obtiene un acceso más rápido al contenido.

2.4.2.4. Paginado.

- **Números:** El usuario puede memorizar el número de página de sus contenidos favoritos, este no debe exceder de tres dígitos.
- **Indicador en pantalla:** Se debe mostrar siempre el número de página.

2.4.2.5. Tabs.

- **Presentación de tabs:** Horizontal o vertical, con etiquetas en cada tab.
- **Redacción de etiquetas de tabs:** Depende del contenido de la pestaña y el espacio limitado de pantalla.

En la tabla 2.3 se presenta un cuadro con las ventajas y desventajas de los tipos de diseño de navegación para aplicaciones interactivas

Tabla. 2.3. Tipos de diseño de navegación en pantalla [27].

	Ventajas	Desventajas
<i>Menú</i>	<i>Acceso al contenido mediante clases. Estructura jerárquica (menús y submenús).</i>	<i>Wording (redacción) del contenido puede ser confuso. Dificultad en la clasificación del contenido.</i>
<i>Vídeo Multi - pantalla</i>	<i>Conmutación eficiente entre flujos de vídeo. Soporte de vídeo basado en navegación.</i>	<i>El número de vídeos ofrecidos para la navegación se limita al espacio en pantalla para los mismos.</i>
<i>Índice</i>	<i>Acceso directo a los usuarios que saben que quieren observar. Comprueba si un tema en particular es proporcionado por la aplicación.</i>	<i>No todo el contenido puede ser mostrado en una página, por lo que se necesita usar más páginas.</i>
<i>Paginado</i>	<i>Atajo para usuarios regulares. Soporte para usuarios habituales que tienen su página favorita</i>	<i>Familiarizado con antiguos usuarios de teletexto.</i>
<i>Tabs</i>	<i>Acceso al contenido mediante clases. Navegación clásica.</i>	<i>Wording (redacción) del contenido puede ser confuso. Dificultad en la clasificación del contenido.</i>

Elaborado: 10 enero 2016.

2.4.3. Teclas de control remoto.

Una vez elegido el layout y el tipo de navegación en pantalla, hay que tomar en cuenta el tipo de teclado que tiene el control remoto y la funcionalidad de cada botón. Esto es necesario conocerlo para poder navegar en cualquier aplicación interactiva.



Figura 2.8. Tipos de controles para decodificador digital.

Fuente: <http://www.slideshare.net/movimientodisenoinclusivo/patrones-de-diseo-de-interaccin-para-tv-digital-interactiva>

Tabla. 2.4. Teclado básico de un control remoto para televisión digital [27].

	Ventajas	Desventajas
Teclas de flecha	Detectables por el tacto. Para moverse dentro de la interfaz.	Son teclas de navegación y no de selección.
Tecla OK	Detectable mediante el tacto. Para confirmar la elección.	Solo se puede elegir una opción de entre varias opciones.
Botones de colores	Un botón puede activar directamente una función. Elección de los elementos del menú principal.	Existen 4 opciones. Difícil de encontrar mediante el tacto.
Tecla numéricas	Una tecla puede activar directamente una función. Selecciona elementos de un menú, en modo de atajo.	Existen 10 opciones. Difícil de encontrar mediante el tacto.
Teclas especiales	Una tecla puede activar directamente una función. Realizan el mismo efecto en todas las aplicaciones.	No disponible en todos los controles. Difícil de encontrar por medio del tacto.

Elaborado: 10 enero 2016.

2.4.4. Funciones básicas de los servicios interactivos.

- **Métodos de acceso:** *Pull*, los usuarios controlan el inicio de la presentación de las aplicaciones interactivas; *push*, las aplicaciones son iniciados de forma automática.
- **Posición del indicador en pantalla:** *Por lo general este indicador se denomina INFO. Se ubica en el borde de la pantalla con el fin de no cubrir ninguna acción importante en el programa de TV.*
- **Duración del indicador en pantalla:** *El indicador no debe permanecer mucho tiempo en pantalla para no interrumpir la atención del usuario hacia el programa. Este indicador permanecerá en pantalla durante 10 segundos y se repetirá su presentación luego de 5 o 7 minutos.*
- **Disponibilidad:** *La opción para iniciar, ocultar y terminar una aplicación debe estar disponible y activo en todo momento, además de una opción de regresar a la página anterior.*
- **Control a distancia:** *Para iniciar una aplicación se puede hacer uso de algún color, tecla numérica, tecla OK o alguna tecla en especial (info).*
- **Indicador de estado en pantalla:** *Debe existir en pantalla un indicador de estado (elemento gráfico: estático o dinámico) que muestre al usuario el proceso de carga de la página solicitada.*
- **Opción cancelar:** *Permite al usuario cancelar la aplicación que este en pantalla, de este modo solo se necesitaría de una sola pulsación de un botón para salir a la pantalla de TV convencional.*

2.4.5. Presentación de contenido.

- **Tipo de letra:** *Debido a la baja resolución de las pantallas de televisión actuales es difícil leer caracteres, por lo que se recomienda utilizar un tipo de letra sans-serif como TIRESIAS.*
- **Tamaño de la fuente:** *Mientras más grande sean el tamaño de las letras será más visible, el problema ocurre debido al espacio en pantalla. Se recomienda utilizar un tamaño de fuente de 24.*

- **Color:** *Contrastar bien los colores con el fondo de pantalla del programa. Se recomienda utilizar un texto claro sobre fondo oscuro. Utilizar colores para el texto de un alto contraste de brillo.*
- **Longitud del texto:** *No mostrar más de 90 palabras por página.*
- **Estructura del texto:** *Usar frases cortas y sencillas, títulos y subtítulos. Solo textos importantes.*
- **Transparencia:** *La transparencia del cuadro de contenido dependerá de la legibilidad del texto, color del texto y visibilidad de las imágenes.*

2.4.6. Participación de usuario.

- **Votaciones y preguntas de opción múltiple:** *En la mayoría de aplicaciones, se permite solo la elección de una respuesta.*
- **Completar texto:** *Combinación de preguntas de opción múltiple. Proporciona información para los ejercicios o pruebas de aprendizaje.*
- **Aprobación para la conectividad:** *Para realizar un chat con un grupo de usuarios, se debe aprobar la conectividad de los usuarios que requieren unirse a la conexión.*

2.4.7. Entrada de textos.

- **Usos:** *Múltiples métodos para introducir caracteres.*
- **Teclado en pantalla QWERTY:** *Se ofrece a usuarios con ordenadores personales.*
- **Teclado del teléfono móvil:** *Para usuarios que utilizan teléfonos móviles.*

2.4.8. Menú de ayuda.

- **Usos:** *Proporciona instrucciones en pantalla sobre cómo lograr el máximo de objetivos comunes de los usuarios.*
- **Wording:** *Usar instrucciones de uso breves y precisas. Referenciar con los botones del control remoto de manera diferente al resto de instrucciones.*
- **Posición:** *Colocar las instrucciones en pantalla constantemente y de forma estática en la parte inferior de la página.*

- **Ayuda activa vs pasiva:** *Ayuda activa, se realiza el seguimiento de lo que el usuario está realizando para mostrar una ayuda contextual automática; ayuda pasiva, es necesario que el usuario acceda manualmente al bloque de ayuda.*
- **Acceso y salida de la sección ayuda:** *La sección ayuda debe ser accesible desde todas las páginas de la aplicación.*

2.4.9. Accesibilidad y personalización.

- **Tamaño de la fuente:** *Opción de ampliar el tamaño de la fuente para todos los textos que se muestren en pantalla.*
- **Subtítulos:** *Opción de visualización de subtítulos.*
- **Idioma:** *Permite cambiar el idioma del audio del programa y de cualquier texto de la aplicación.*

2.5. Middleware Ginga.

2.5.1. Definición.

Ginga es el nombre del middleware de la Recomendación ITU-T [13] para servicios de IPTV y del Sistema Nipo-Brasileño de TV Digital Terrestre (ISDB-Tb). El nombre fue escogido en reconocimiento a la cultura, arte y continua lucha por la libertad e igualdad del pueblo brasileño [24].

Es una capa de software intermedio (middleware) entre el hardware/Sistema Operativo del STB y las aplicaciones. Ofrece una serie de facilidades para el desarrollo de contenidos y aplicaciones para TV Digital, permitiendo la posibilidad de poder presentar los contenidos en distintos receptores independientemente de la plataforma de hardware del fabricante y el tipo de receptor (TV, celular, PDAs, etc.) [24].

La finalidad de la capa del middleware es ofrecer un servicio estandarizado para las aplicaciones, de este modo se ocultan las diferencias de capas inferiores que existen entre los diferentes fabricantes de receptores. Un middleware estandarizado facilita la portabilidad de las aplicaciones y permite que sean ejecutadas en cualquier receptor digital que soporte el middleware adoptado.

Este middleware fue desarrollado en el TeleMídia Lab (Departamento de Informática de la Pontificia Universidad Católica de Río de Janeiro, Brasil) junto con la ayuda de LAViD (Laboratorio de Aplicaciones de Vídeo Digital, Universidad Federal de Paraíba, Brasil) y LIFIA (Laboratorio de Investigación y Formación en Informática Avanzada, Universidad Nacional de la Plata, Argentina).

Ginga es una especificación abierta de fácil aprendizaje y libre de royalties⁴, de este modo se permite la explotación del middleware para que todos puedan producir contenido interactivo, esto ayudará a canales comunitarios y a grandes televisoras. TeleMídia garantiza el acceso permanente a toda evolución del código publicado en la Comunidad Ginga, esto es posible gracias a la adopción de la licencia de software libre GPLv2 [24].

2.5.1.1. Ventajas del middleware Ginga.

- *Facilita la interactividad, cualquier sistema receptor podrá reproducir las aplicaciones y contenidos interactivos ya que el middleware define una interfaz estandarizada.*
- *Flexibilidad en el momento de generar contenido, este puede ser creado o modelado de acuerdo a las necesidades de cada radiodifusor.*
- *Portabilidad de las aplicaciones, las aplicaciones interactivas pueden reproducirse en cualquier receptor independiente de la plataforma, modelo y/o fabricante.*
- *Permite la interacción simultánea de varios usuarios utilizando diferentes dispositivos.*
- *Facilita la integración del receptor de TV digital con otros dispositivos a su alrededor.*

2.5.2. Arquitectura del middleware Ginga.

La arquitectura de Ginga comprende dos ambiente de ejecución de aplicaciones: uno declarativo, Ginga-NCL para la presentación de aplicaciones NCL, y uno procedural Ginga-J pensado para controlar la ejecución de entidades escritas en Java.

Esta arquitectura posee un subsistema denominado Ginga-CC que proporciona el soporte básico de ejecución de los lenguajes antes mencionados.

⁴ Royalties, es el pago que se efectúa al titular de derechos de autor, patentes, marcas a cambio del derecho a usarlos o explotarlos.

Además, el middleware Ginga cuenta con un lenguaje de scripting denominado LUA, el cual tiene por objeto hacer que las aplicaciones basadas en Ginga-NCL sean extendidas, éste realiza procedimientos matemáticos, animaciones, utiliza el canal de retorno y otras características que NCL no puede realizar.

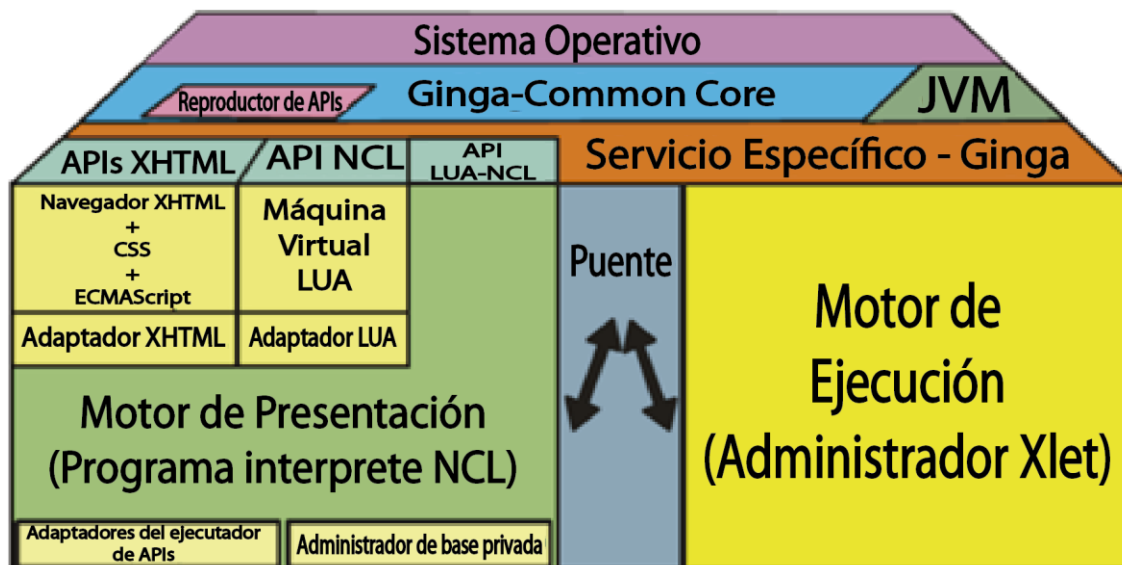


Figura 2.9. Arquitectura del middleware Ginga [24].

El diseño del lenguaje NCL se basó en distintos requisitos que el SBTVD adoptó y mejoró, algunos de estos requisitos fueron: establecer un ambiente declarativo, sincronización con objetos multimedia, adaptabilidad de contenido y la presentación, soporte para múltiples dispositivos de visualización y soporte que permita generar aplicaciones de TV Digital “en vivo” [28].

NCL fue adoptado como lenguaje declarativo y LUA como lenguaje de scripting, es decir, que el motor de Ginga-NCL incluye una API entre NCL y objetos LUA. Estos lenguajes son de código abierto (open source), libre de regalías, independiente de la red de distribución de TV Digital utilizada, a diferencia de Java, estos son fáciles de usar y no se requiere conocimientos profundos de programación para poder desarrollar aplicaciones interactivas. Otro de los módulos importantes es el agente de usuario del modelo XHTML, este incluye un stylesheet (CSS) y un interpretador ECMAScript [24].

Ginga-J es el subsistema lógico del sistema Ginga que procesa el contenido de los objetos Xlet, incorpora tecnología Java y mantiene su compatibilidad con muchos middlewares para TV Digital

ya que cuenta con GEM (Globally Executable MHP – Plataforma Multimedia de Hogar Mundialmente Ejecutable)

Ginga-NCL y Ginga-J son construidos sobre servicios ofrecidos por el modulo del núcleo común de Ginga (Ginga-Common Core), este está compuesto por decodificadores de contenido común y procedimientos para obtener contenidos. Decodifica contenido como JPEG y MPEG, permite el acceso condicional a las aplicaciones, acceso al canal de retorno y la interfaz de usuario [24].

2.5.2.1. Ginga-CC.

Es el núcleo común del middleware Ginga que integra las APIs necesarias para ejecutar tanto aplicaciones declarativas como procedurales.

Este subsistema integra la interfaz directa con el sistema operativo haciendo un puente estrecho con el hardware. Esta capa permite acceder al sintonizador de canales, sistemas de archivos, terminales gráficos, acceso a datos obtenidos a través del canal de retorno, entre otros. Está compuesto por decodificadores de contenido común tanto para aplicaciones procedimentales como para las declarativas, que necesitan decodificar y presentar objetos de formato PNG, JPEG, MPEG y otros [24].

2.5.2.1.1. Arquitectura.

Los principales componentes de Ginga-CC, se encuentran representados en la siguiente figura.



Figura 2.10. Arquitectura de Ginga-CC [29].

- *Sintonizador.*

Es el módulo encargado de sintonizar un canal, seleccionando un canal físico y los flujos de transporte que están siendo enviados por éste.

- *Filtros de sección.*

Estos se encargan de realizar la búsqueda de la parte exacta de las aplicaciones interactivas que las APIs necesitan para su ejecución.

- *Procesador de datos.*

Es el responsable de acceder, procesar, notificar y transferir los datos recibidos por la capa física. Este componente también es el encargado de notificar a los otros componentes acerca de cualquier evento que haya sido recibido.

- *Persistencia.*

Se encarga del almacenamiento de los datos solicitados por las aplicaciones. Este módulo brinda soporte para que el archivo sea abierto en otra ocasión.

- *Administrador de aplicaciones.*

Realiza las unciones de cargar, configurar, inicializar, eliminar, controlar y ejecutar cualquier aplicación, ya sea de tipo declarativo o de procedimiento.

- *Adaptador principal de A/V.*

Este adaptador sirve para que las aplicaciones Ginga permitan ver el flujo de la aplicación en audio y vídeo. Esto es necesario cuando se necesite controlar sus acciones, de acuerdo con lo que se está transmitiendo.

- *Administrador de gráficos.*

Las imágenes, vídeos, datos, son exhibidos de acuerdo a las normas del middleware Ginga, es decir, este administrador define el plano de exhibición del vídeo principal H.264, los planos de exhibición de los otros objetos de media que componen la aplicación y cómo estos planos se sobreponen.

- *Administrador de actualizaciones.*

Este componente gestiona, controla y descarga las actualizaciones del middleware, esto para corregir los errores de versiones anteriores.

Los STB tienen la capacidad de actualizar su firmware mediante dos métodos: actualización vía broadcasting (por aire) o mediante un dispositivo de almacenamiento externo conectado al puerto usb.

- *Reproductor de archivos multimedia*

Se compone de las herramientas necesarias para presentar los archivos multimedia recibidos. Estos players son exhibidores de audio, vídeo, texto e imagen, exhibidores MPEG-4/H.264 y exhibidor HTML.

- *Interface de usuario.*

Es la responsable de captar e interpretar los eventos generados por los usuarios.

- *Administrador de contextos.*

Es el encargado de recopilar la información en el dispositivo receptor, información sobre el perfil de usuario y su localización, esto permitirá que Ginga-NCL y Ginga-J puedan realizar adaptaciones a los contenidos de la forma cómo deben ser presentados, según lo determinado por las aplicaciones.

- *Canal de interactividad.*

Es el que proporciona la interfaz de las capas superiores con el canal de interacción. Gestiona el canal de retorno de manera que los datos se transmitan cuando el canal esté disponible, o se fuerza una transmisión en caso de que el televidente o una aplicación tengan establecido un horario predeterminado.

- *Acceso condicional.*

Es el encargado de restringir contenidos inapropiados, proporcionando seguridad al middleware.

2.5.2.2. Ginga-NCL.

Ginga-NCL fue desarrollado por la Pontificia Universidad Católica de Rio de Janeiro PUC-Rio, es el subsistema encargado de procesar y ejecutar aplicaciones interactivas de tipo declarativas escritas en lenguaje NCL (Nested Context Lenguaje) y del tipo imperativas en lenguaje LUA, para esto utiliza la funcionalidad que Ginga-CC expone en forma de APIs.

NCL es una aplicación de XML (eXtensible Markup Lenguaje) con facilidades para los aspectos de interactividad, sincronismo, espacio-temporal entre otros objetos de media, adaptabilidad, soporte a múltiples dispositivos y soporte a la producción de programas interactivos en vivo no-lineares [14].

NCL es un lenguaje basado en la estructura que separa el contenido y la estructura de un aplicativo, permitiendo definir objetos de media estructurados y relacionados tanto en tiempo y espacio [14].

2.5.2.2.1. Arquitectura

El subsistema Ginga-NCL está formado por los siguientes componentes:



Figura 2.11. Arquitectura de Ginga-NCL [42].

- *Formateador.*

Es el motor de ejecución del contenido declarativo NCL, recibe y controla la ejecución de una aplicación NCL, garantizando que las relaciones espacio-temporales definidas por el programador sean respetadas.

- *Administrador de diseño.*

Soporta múltiples dispositivos de presentación al mismo tiempo ya que se asignan todas las regiones definidas en una aplicación para cada pantalla de dispositivo en donde será exhibida la aplicación.

- *Administrador de contexto NCL.*

Soporta el contenido NCL incluyendo sus adaptaciones de presentación, de acuerdo a la información procedente del Gestor de contexto del Ginga-CC.

- *Planificador y Administrador de Reproducción.*

La aplicación debe estar traducida a un lenguaje entendible para el planificador, esto lo realiza el conversor.

Una vez sucedido esto, el planificador es iniciado para organizar el orden de la presentación del documento NCL.

- ✓ *Captura de los objetos de media, previa a su exhibición.*
- ✓ *Evaluación de las condiciones de enlace de un objeto media con otro.*
- ✓ *Planificación de las acciones a ejecutarse, esto dependerá del tipo de condición de cada enlace, de este modo se guiara el flujo de la presentación.*

Una vez realizada la planificación, el administrador de reproducción exhibe el tipo de archivo media en el momento indicado.

- *Analizador de XML y Conversor.*

Cuando se recibe una aplicación, el formateador solicita al analizador XML y al conversor que realice la traducción de la aplicación NCL a la estructura interna de datos Ginga-NCL necesaria para controlar la presentación de la aplicación.

- *Administrador de recuperación.*

El administrador de recuperación detecta, controla e implementa los procedimientos de recuperación de fallas.

- *Administrador de base privada.*

Este administrador recibe comandos para activar y manipular las aplicaciones en tiempo real, estos comandos son:

- **Primer Grupo de Comandos:** *habilita o deshabilita las aplicaciones de un determinado canal de televisión.*
- **Segundo Grupo de Comandos:** *inicia, pausa, reanuda, detiene y remueve aplicaciones NCL.*
- **Tercer Grupo de Comandos:** *actualiza una aplicación en tiempo real, permitiendo agregar o remover elementos NCL y admite que se asignen valores a las propiedades de los objetos de media.*

2.5.2.2.2. Requerimientos [10].

Debemos contar con un computador con memoria RAM no menor a un 1GB y espacio en disco no menor a 5GB, donde previamente se encuentre instalado Linux y VMware, y sobre este último se instala la máquina virtual Ginga-NCL STB Virtual.

2.5.2.3. Ginga-J.

Ginga-J fue desarrollado por la Universidad Federal de Paraíba UFPB, para proveer una infraestructura de ejecución de aplicaciones basadas en lenguaje Java, llamadas Xlet, con facilidades y detalles para el ambiente de TV digital proporcionada por la especificación GEM que fueron sustituidos más tarde con la especificación abierta Java DTV desarrollada por Sun Microsystems, ahora Oracle Corporation [24].

Es el subsistema lógico del sistema Ginga que procesa aplicaciones interactivas de tipo procedimental (Xlets Java) en una máquina virtual de Java. Ginga-J al estar basado en GEM, mantiene la compatibilidad con la mayoría de middlewares de TV Digital actuales [28].

2.5.2.3.1. Arquitectura [30].

Está basado en tres APIs llamados Verde, Amarillo y Azul.

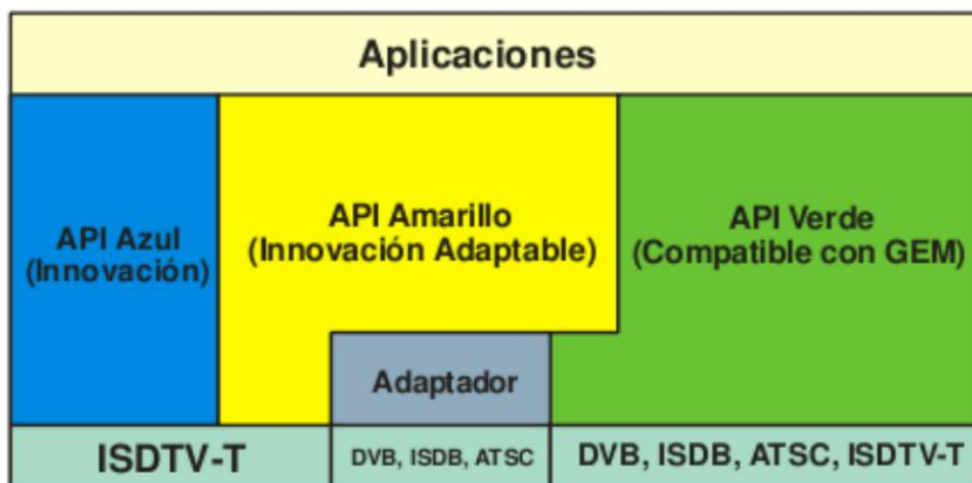


Figura 2.12. Arquitectura de Ginga-J [30].

En esta imagen se puede observar los tres grupos de APIs con el cual funciona Ginga-J.

- API Verde (compatible con GEM), incluye las APIs provenientes de los paquetes de Sun JavaTV.
- API Amarillo (compuesto por el JMF 2.1 API), este es necesario para el desarrollo de aplicaciones con captura de sonido.
- API Azul, permite al receptor de TV Digital comunicarse con cualquier dispositivo con una interfaz compatible (Ethernet, PLC, infrarrojos o Bluetooth). Aquí se encuentra la API que permite el desarrollo de las aplicaciones Ginga-J que tengan Ginga-NCL (API puente).

Estos APIs fueron diseñados para proporcionar toda la funcionalidad necesaria para el despliegue de aplicaciones para TV Digital, desde el manejo de datos multimedia a los protocolos de acceso.

2.5.2.3.2. Requerimientos [10].

Se necesita tener un computador que cuente con un procesador Intel o AMD a 1GHz, con memoria RAM de 512 MB, una tarjeta gráfica de 512 Mb o superior y memoria en disco duro de 20 GB.

Un software que permite trabajar con Ginga-J es Open Ginga, el cual está basado en código libre que actualmente está siendo desarrollado por el laboratorio LAVID de la Universidad Federal de Paraíba. Para que Open Ginga funcione correctamente en el computador se necesita tener un sistema operativo Linux Ubuntu 10.0.4 o superior, la máquina virtual Java J2SDK 1.4.0.2 y XML.

2.5.2.4. Cuadro comparativo entre Ginga-NCL y Ginga-J.

Tabla. 2.5. Cuadro comparativo entre Ginga-NCL y Ginga-J [10].

Ginga-NCL	Ginga-J
Software libre.	Necesita incluir bibliotecas extras.
Menor capacidad de procesamiento.	Necesita mayor capacidad de procesamiento.
Lenguaje con poca difusión.	Lenguaje popular.
La norma especifica su presencia en todo tipo de dispositivos.	Se especifica en dispositivos fijos y móviles.
Lenguaje maduro.	Lenguaje en desarrollo.

Elaborado: 20 enero 2016.

2.5.2.5. LUA.

En el lenguaje declarativo NCL la realización de muchas tareas se vuelve complicada a la hora de realizar alguna tarea como el procesamiento matemático, manipulación de texto, uso del canal de interactividad, animaciones, otros. Es por esto que se requiere de la ayuda de algún lenguaje imperativo.

LUA fue creado en 1993 por miembros del Grupo de Tecnología en Computación Grafica (Tecgraf) en la PUC-Rio, es un lenguaje planeado para ser usado en conjunto con otros lenguajes de programación y es utilizado como un script para permitir que una aplicación sea extendida. Por ejemplo en los videojuegos el script de Lua es utilizado para describir el comportamiento de un personaje, uno de los principales juegos que utiliza este lenguaje es Worl of Warcraft [31].

LUA se caracteriza por ser simple y eficiente ya que requiere menos costos de hardware. Es portable, ya que puede ejecutarse en diversas plataformas como computadores personales, celulares, sistemas embarcados y consolas de videojuegos. Además, es un software libre que calza perfectamente al escenario de TV Digital ya que el middleware está desarrollado para que se ajuste a las características complejas de los equipos receptores [32].

Es así que LUA fue extendido en el área de televisión digital, con nuevas funcionalidades para poder integrarse con NCL y saber cuándo un objeto NCLua es activado en un enlace para su ejecución. Un objeto NCLua también puede responder a las teclas del control remoto. LUA es reconocido por su alto rendimiento y bajo consumo de recursos, en comparación con otros lenguajes es en promedio, 7 veces más rápido y con una memoria utilizable 40 veces menor [31].

2.6. Canal de retorno.

Es un camino de transmisión de datos que establece una comunicación entre el proveedor de servicio interactivo y el receptor de TV. Permite a los usuarios ser participantes activos del servicio permitiéndoles el envío y recepción de datos, permite la recepción de datos adicionales al servicio principal.

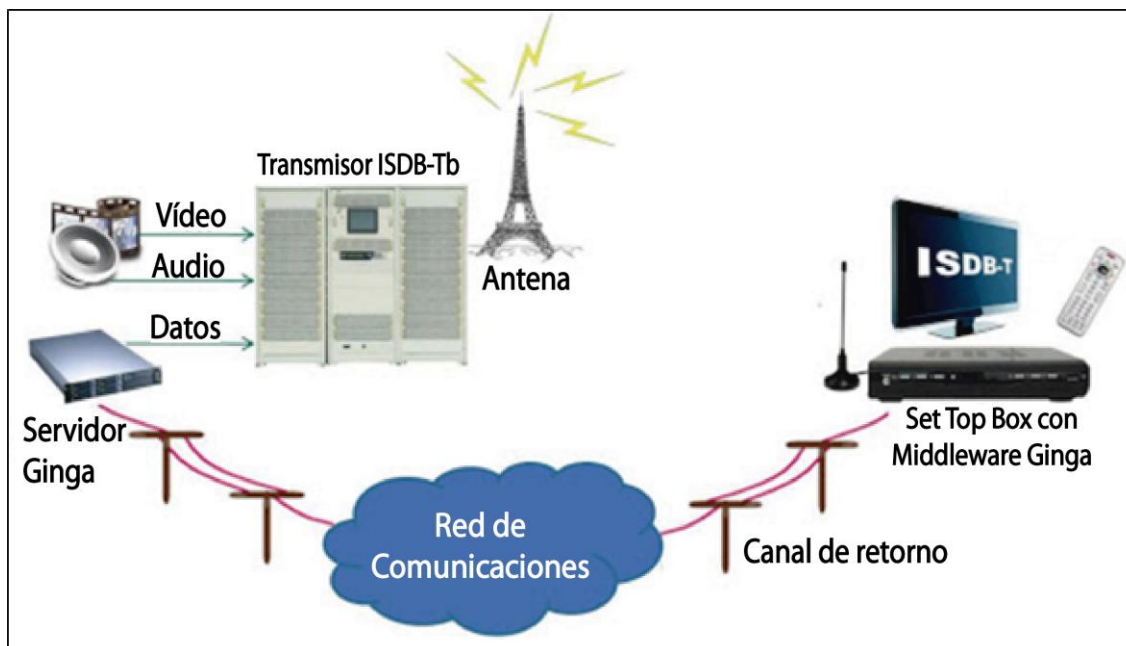


Figura 2.13. Diagrama de una red de televisión digital con canal de interactividad.

Fuente: <https://lh4.googleusercontent.com/Wf17muwm2inowEtSjciR0u0SI4i3E5fRXxTQD56HQMZC6W4DuZh-UBBPw9m4EBBgz0rGbmMCKUhrWnEUvMgwLrAhQ=s512>

El canal de retorno puede ser: unidireccional, se permite solo el envío de datos como en el caso de una votación de algún tema; y bidireccional, permite que el receptor realice la descarga de los datos que el televidente requiera desde el servidor de aplicaciones.

Los componentes del canal de interactividad son:

- **Canal de retorno:** está conectado en el flujo de transporte del canal de radiodifusión, utilizado para el envío de solicitudes y respuestas desde las emisoras hacia los usuarios.
- **Gateway:** une las redes de comunicaciones con las emisoras, recibe los flujos de datos de estas redes y las adapta como un flujo de datos capaz de ser recibido y procesado por el servidor de aplicaciones.
- **Servidor de aplicaciones:** realiza la interpretación y el debido tratamiento de las peticiones de cada usuario, este servidor puede realizar la búsqueda de la información en otros servidores para generar una respuesta para cada televidente, genera nuevo contenido para actualizar las aplicaciones según las sugerencias de los usuarios.

- **Unidad de adaptación de radiodifusión:** se encarga de la adaptación de la información que será enviada como respuesta a lo que el usuario solicitó. Para esto se puede necesitar de un procesador o generador de carrusel de contenidos on-line.

Algunas de las aplicaciones que tienen acceso a un canal de retorno son aquellas en las que el televidente requiere ver información adicional, manipular la programación de los canales, comprar servicios o productos por televisión, participar en concursos, votaciones, entre otros.

Estas aplicaciones son utilizadas para diversos fines, pero siempre mantienen el objetivo de brindar un beneficio a la sociedad, la inclusión social y digital, es entonces que se existen diferentes topologías de comunicación dependiendo del tipo de aplicación que se desea realizar: T-Voting, T-Learning, T-Commerce, T-Bank, T-Government, entre otros [10].

2.6.1. Tecnologías de conectividad para el canal de retorno.

El canal de retorno se puede implementar mediante el uso de diferentes tecnologías, cada una de ellas presenta diversas características que pueden ser mejores para determinadas zonas, regiones o países, a continuación se realizara un análisis de los mismos. Las tecnologías de acceso para el canal de retorno se especifican en la norma brasileña ABNT NBR 15607-1 [33].

Las tecnologías de acceso definidas son: módems discados, Ethernet, ISDN, GSM, CDMA, WiMAX, WiFi, entre otros.

2.6.1.1. Canal de retorno mediante la red de PLC (Power Line Communication).

Esta tecnología permite la utilización de la red de energía eléctrica para transmitir datos y es muy conveniente porque está disponible hasta en los lugares más remotos.

Esta tecnología de transmisión de datos a través de las líneas de distribución de energía eléctrica es una buena solución debido a que ofrece el alcance deseado, soporta transmisiones de 14 Mbps (pudiendo llegar a 200 Mbps) y su instalación no sería muy costoso [16].

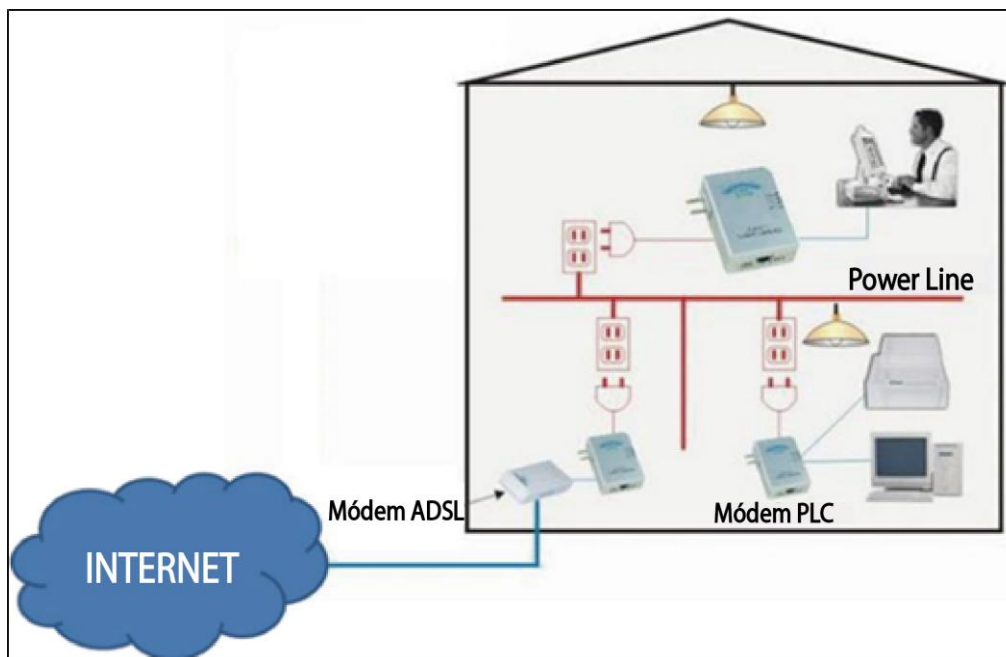


Figura 2.14. Canal de retorno mediante la red de PLC [10].

2.6.1.2. Canal de retorno mediante tecnologías satelitales.

La televisión satelital ofrece una rápida instalación de los equipos, brinda una gran cobertura la cual no depende de cables ni distancias, además, el ancho de banda que tiene es muy alto llegando al orden de los Mbps [10].

Debido a estas ventajas de la tecnología satelital, los servicios están disponibles tanto en dispositivos fijos como móviles, esto permite una comunicación más eficiente ya que el canal de retorno no dependerá de ninguna otra red que la satelital.

Uno de los principales beneficios de utilizar tecnología satelital es la gran cobertura que ofrece, pero así como existen beneficios, también las desventajas de este diseño son considerables, el tiempo en que demora en llegar la información transmitida desde el usuario al proveedor de interactividad es mayor, esto ocasiona problemas en las aplicaciones de tiempo real, además la implementación de este sistema sería muy costoso para el operador.

2.6.1.3. Canal de retorno mediante ADSL (Asymmetric Digital Subscriber Line).

El sistema de acceso ADSL se compone de dos módems en cada extremo de la línea telefónica, creándose tres canales de información: uno descendente a alta velocidad, otro ascendente dúplex

a velocidad media, y el del servicio telefónico básico, el cual es separado del módem digital mediante filtros, garantizando así la continuidad del servicio telefónico ante una caída o fallo de dicho módem [10].

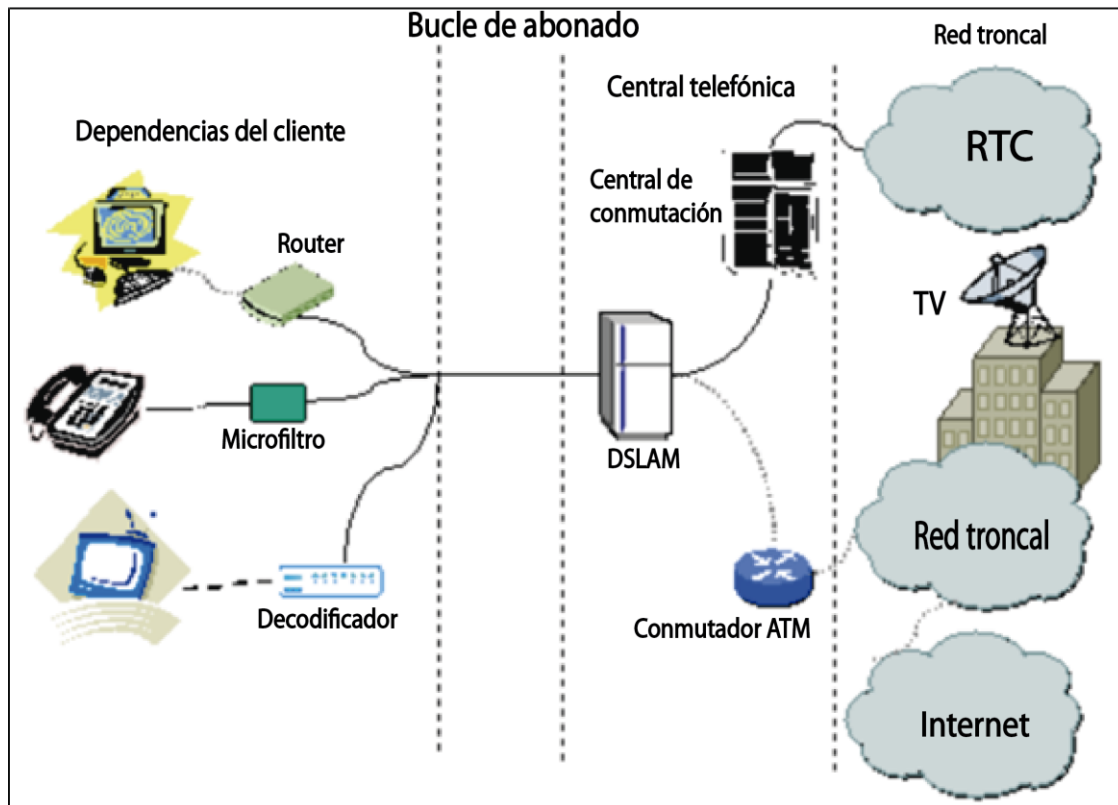


Figura 2.15. Canal de retorno mediante ADSL [10].

ADSL es reconocido como el sistema más desplegado por el mundo debido a que es un sistema de gran velocidad en el servicio de aplicaciones multimedia, sus velocidades de transmisión de datos descendente son de hasta 1.5 Mbps sobre distancias de 6 Km, y de hasta 8 Mbps para distancias de 3 Km; y las velocidades ascendentes alcanzan los 16 a 640 Kbps sobre los mismos tramos [10].

Es recomendable utilizar esta tecnología en el momento de implementar un canal de retorno, porque además de que es muy eficiente, los costos de implementación disminuyen debido a que se utiliza la infraestructura técnica ya existente.

Para que exista una buena comunicación entre el televidente y la emisora y viceversa, debe existir en la red por parte del usuario un módem y del lado de la central telefónica un DSLAM (Digital

Subscriber Line Access Multiplexer – Multiplexor de acceso a la línea digital de abonado) que normalmente está instalado en lugares o ciudades de mayor densidad poblacional. Esto ocasiona problemas para las zonas rurales o poblaciones con bajos recursos económicos.

2.6.1.4. Canal de retorno mediante tecnologías de fibra óptica.

La televisión por cable se distribuye por medio de redes híbridas (fibra óptica y cable coaxial – HFC) las cuales son introducidas a la red de acceso para la transmisión por fibra óptica, esto permite reducir perturbaciones del sistema por cable y cubrir grandes distancias (100 Km sin amplificadores). La televisión digital también puede distribuirse mediante una red GPON, la misma que está constituida puramente de fibra óptica lo cual hace que esta red tenga un ancho de banda mayor y con menores interferencias.

2.6.1.5. Canal de retorno mediante comunicaciones inalámbricas WiFi/WiMAX.

Las redes inalámbricas WiMAX IEEE 802.16 y WiFi IEEE 802.11 son alternativas tecnológicas para el canal de retorno. Las ventajas de estas tecnologías se deben a que por ser redes inalámbricas facilitan la instalación en los hogares de los usuarios. Tiene altas tasas de transferencia, pero puede ser limitado si la misma banda se divide por un número muy grande de usuarios. Permite el canal de retorno en los dispositivos móviles mientras están en movimiento. Las desventajas que presenta esta tecnología se presenta a nivel de equipos o accesorios en el lado del terminal que se utiliza para conectarse a la red, su costo aumenta considerablemente [34].

La tecnología WiFi cuenta con dos alternativas de operación:

- *Red en modo Ad Hoc.*
- *Red en modo AP (Access Point).*

2.6.1.6. Canal de retorno mediante tecnologías móviles.

Las tecnologías para comunicaciones móviles solo permitían acceso a la telefonía móvil, pero estas redes no estaban limitadas, con el avance de las nuevas generaciones de telefonía móvil, se permitió la transmisión de audio y vídeo en tiempo real, capacidad multimedia, velocidad de

acceso a Internet y la transmisión de voz en alta calidad. Es por esta razón que la normativa brasileña ABNT NBR 15607-1 utiliza las redes de comunicaciones celulares de GSM (GPRS, EDGE) y CDMA (1xRTT, EVDO) como canales de interactividad. En la actualidad se puede implementar canales de retorno utilizando las tecnologías de 3G, 4G y LTE [33].

2.6.2. Implementación del canal de retorno.

2.6.2.1. Simulación en Virtual Set Top Box Ginga-NCL.

El funcionamiento del canal de retorno se lo puede realizar en el virtual Set Top Box Ginga NCL instalado en un computador con sistema operativo Ubuntu. Este estará conectado a una red LAN Ethernet de 10/100 Mbps [35].

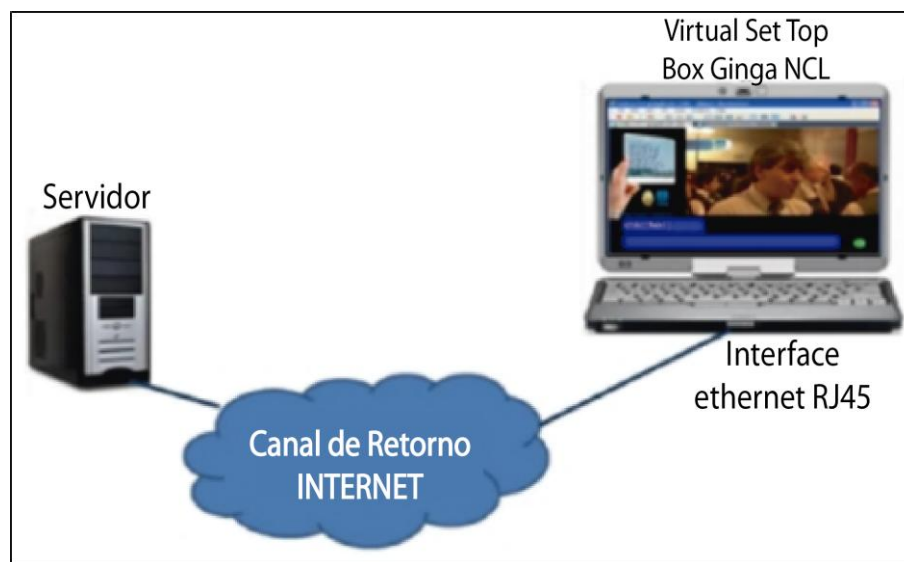


Figura 2.16. Escenario de simulación para la interactividad con canal de retorno [35].

2.6.2.2. Implementación en el Set Top Box para TV digital.

Para realizar la implementación de este tipo de simulación, se utiliza un STB con soporte de middleware Ginga NCL, MPEG-2, H.264 HD/SD, Flash Memory de 63 MB y 256 MB SDRAM, un televisor digital y un dispositivo de memoria externa. La interfaz Ethernet del STB se conecta a internet para que funcione como canal de retorno para la interactividad [35].

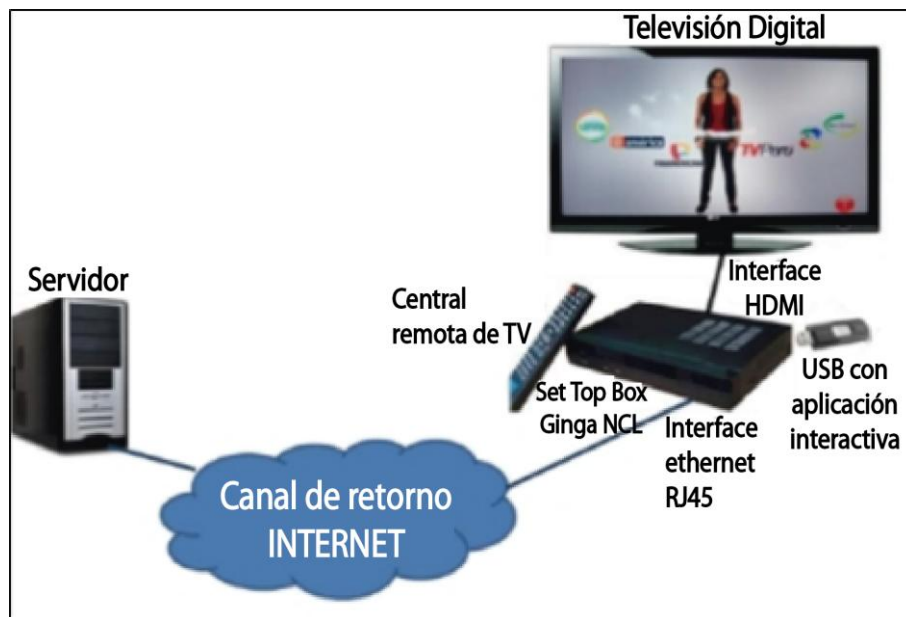


Figura 2.17. Escenario de prueba real para la interactividad con canal de retorno [35].

El STB va enlazado con el punto de conexión a la red, con lo cual por medio de broadcast se llegaría a cargar aplicaciones desde la central televisiva, el usuario podrá navegar desde su televisor y formar parte de las aplicaciones en tiempo real.

2.7. Set Top Box para el estándar ISDB-Tb.

2.7.1. Definición.

La normativa ABNTNBR15604 define al Set-top Box como un dispositivo de recepción y decodificación de señales de televisión digital que está conectado a un televisor por medio de cables o cualquier otro tipo de conexión y que, por lo tanto, proporciona interfaces de salida de audio y vídeo, ya sean analógicas o digitales [36].

Este decodificador puede ofrecer muchos servicios dependiendo de las características del mismo y de la aplicación interactiva que se vaya a reproducir, además muchos de estos STB nos dan la opción de conectarles dispositivos externos como podrían ser videocámaras, impresoras, discos externos para almacenar los programas grabados, entre otros.

Para cumplir todas estas funciones, el STB contiene un microprocesador, una memoria, interfaces de entrada para periféricos, tarjetas, conexión al canal de retorno (los más comunes, ADSL o telefonía móvil), y un módem que permite el acceso a internet y a servicios interactivos [37].

2.7.2. Proceso de recepción y presentación de un STB.

El proceso de recepción del STB se describe a continuación:

- *Sintonización de la señal digital, incluye la información de vídeo en sus diferentes formatos, información de audio e información de datos.*
- *Separa los tres tipos de información que recibimos para hacer un tratamiento de los mismos por separado.*
- *El sistema de acceso condicional decidirá cuales son los permisos que tiene el suscriptor para poder ver los contenidos que está recibiendo.*
- *Una vez decodificada la información, los paquetes de vídeo y audio son enviados al televisor.*
- *Los paquetes de datos recibidos se ejecutarán en caso de ser necesarios o solicitados por el consumidor*
- *El STB mediante el canal de retorno puede enviar datos a la cabecera (Back Channel).*

2.7.3. Características de un Set Top Box para el estándar ISDB-Tb.

Para reunir las características necesarias para un STB para el estándar ISDB-Tb, se realizó una comparación entre el EITV DeveloperBox de la empresa brasileña EITV y el Set Top Box COL601M de la empresa COLABLE. Se presenta un cuadro con los requisitos mínimos que un decodificador debe tener para éste estándar.

Tabla. 2.6. Resumen de las características técnicas de un STB para el estándar ISDB-Tb [21][23].

Característica	Especificación	Característica	Especificación
Sintonizador y Demodulador [38][39]		Decodificación de vídeo[38]	
Frecuencia de entrada	UHF: 470 MHz – 806 MHz VHF: 174 MHz – 216 MHz	Patrón	Rec. ITU-T H.264 (MPEG-4 AVC)
Ancho de banda	5 – 6 MHz	Perfil	HP@L4.0
Nivel de señal	-85 dBm a -20 dBm	Formatos	480i y 1080i
Impedancia de entrada	50 Ohms	Frame rate	25, 30, 50, y 60 MHz
Conexión de E/S	Conector F	Frecuencia de vídeo	50 y 60 MHz
Unidad de procesamiento [38][39]		Interfaces de comunicación [38]	
Procesador	STi 7105 (CPU 450 MHz)	USB	High Speed 2.0
Memoria RAM	Mayor a 512 MB	Salida de audio y vídeo	HDMI
Memoria flash	Mayor a 128 MB	Salida de vídeo componente	100 Mbps (RJ45), YPbPr (Verde+Azul+Vermelho)
Decodificación de audio [38]		Salida de audio estéreo 1 y 2	1) 2 RCA (Blanco+Rojo) 2) 2 RCA (Blanco+Amarillo)
Patrón	ISO/IEC 14496-3 (MPEG-4 AAC) (SBR)	Salida de vídeo compuesto	1 RCA (Amarillo)
Perfil	AAC@L4 y HE-AAC@L4	Salida de audio digital	SPIDF Coaxial (Naranja)

Fuente: Creado por el autor.

Elaborado: 12 abril 2016.

CAPÍTULO III

3. PROGRAMACIÓN DE APLICACIONES INTERACTIVAS

3.1. Lenguaje de programación NCL.

El lenguaje NCM (Modelo de Contextos Anidados) es el lenguaje en el que se basó NCL. NCM contiene elementos multimedia que pueden ser anidados, de este modo se segmenta y estructura el documento [40].

El lenguaje de programación NCL permite el desarrollo de aplicaciones multimedia con facilidades para los aspectos de interactividad, sincronismo espacio-temporal con medios como vídeo, audio, imágenes, texto y objetos imperativos LUA, soporte a múltiples dispositivos y soporte a la producción de programas interactivos en vivo no-lineares [41].

NCL es fácil de aprender, y más aún si se complementa el aprendizaje con algunas herramientas que generan automáticamente el código NCL especificando los parámetros necesarios para la construcción del documento mediante la ayuda de interfaces gráficas, estas herramientas son CreaTV Digital y NCL Composer.

NCL sirve para dar vida a los objetos multimedia, es decir, se define el lugar en el que se van a presentar, dónde, cómo y cuándo. La estructura de un documento NCL se divide en tres partes: la estructura inicial, la cabecera <head>, lugar donde se encuentra la información que especifica dónde y cómo se va a exhibir el contenido, y el cuerpo <body>, espacio del documento donde se define qué contenidos se presentarán y cuándo.

3.1.1. Estructura de un documento NCL.

3.1.1.1. Estructura inicial de un documento NCL.

Cuando se crea un nuevo archivo NCL, estas características iniciales del documento se crean automáticamente.

En la figura 3.1 podemos apreciar que, en la primera línea de código del documento NCL se define la versión de XML y el tipo de codificación utilizada, en nuestro caso será ISO-8859-1. En la segunda línea de código, se encuentran dos atributos: id, que se utiliza para identificar la aplicación (nombre del archivo NCL) y xmlns, que sirve para identificar el tipo de perfil o versión de NCL que se está utilizando.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="ejemplo03a" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
```

Figura 3.1. Estructura inicial de un documento NCL.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

3.1.1.2. Cabecera del documento NCL.

Al igual que en la estructura de un documento HTML, la cabecera está delimitada por las etiquetas `<head>` y `</head>`. Está conformada por tres elementos básicos: base de regiones, base de descriptores y base de conectores.

- Base de regiones.

Las regiones son etiquetas que sirven para especificar la posición de los objetos media (vídeo, imágenes, texto) en la pantalla. Otra característica importante de las regiones, es que es aquí donde se especifica el dispositivo en el cual la aplicación será exhibida, es por esto que es posible interactuar con la TV mediante un dispositivo móvil.

Cuando se definen las dimensiones de las regiones se puede utilizar valores absolutos (50 pixels) o valores relativos al tamaño de la pantalla (porcentaje). Las regiones pueden ser anidadas, es decir, se puede declarar regiones dentro de otra región, las regiones que fueron declaradas dentro de otras tiene como límite las dimensiones de la región madre.

```
<!--REGIONES-->
<regionBase>
  <region id="rgTV" zIndex="0">
    <region id="rgVideo" zIndex="1"/>
    <region id="rgInfo" width="6%" height="7%" left="3%" bottom="5%" zIndex="2"/>

    <region id="rgfondo1" height="75%" width="25%" zIndex="1">
      <region id="rgTituloPr" height="12%" width="80%" top="3%" left="10%" zIndex="2"/>
      <region id="rgTextoPr" height="85%" width="92%" left="4%" top="15%" zIndex="2"/>
      <region id="rgImg1" width="80%" height="20%" top="18%" left="10%" zIndex="2"/>
      <region id="rgImg2" width="80%" height="20%" top="43%" left="10%" zIndex="2"/>
      <region id="rgImg3" width="80%" height="20%" top="68%" left="10%" zIndex="2"/>
    </region>
  </region>
</regionBase>
```

Figura 3.2. Estructura de una base de regiones.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

En la figura anterior podemos observar los diferentes atributos que pueden especificar en la base de regiones, a continuación se presenta su definición para mayor comprensión:

- ✓ **id**: nombre que identifica la región de forma única a lo largo de todo el programa.
- ✓ **height**: representa la altura de la región.
- ✓ **width**: representa el largo de la región.
- ✓ **left**: es la distancia que existe entre el lado izquierdo de la región y el lado izquierdo de la pantalla del dispositivo de exhibición.
- ✓ **right**: es la distancia que existe entre el lado derecho de la región y el lado derecho de la pantalla del dispositivo de exhibición.
- ✓ **top**: es la distancia que existe desde el margen superior de la región y el margen superior de la pantalla del dispositivo de exhibición.
- ✓ **bottom**: es la distancia que existe desde el margen inferior de la región y el margen inferior de la pantalla del dispositivo de exhibición.
- ✓ **zIndex**: es un número entre 0 y 255 que representa la prioridad de superposición de cada región. Las regiones con un **zIndex** mayor se superponen a las regiones de menor valor haciendo que estas últimas regiones queden más atrás.

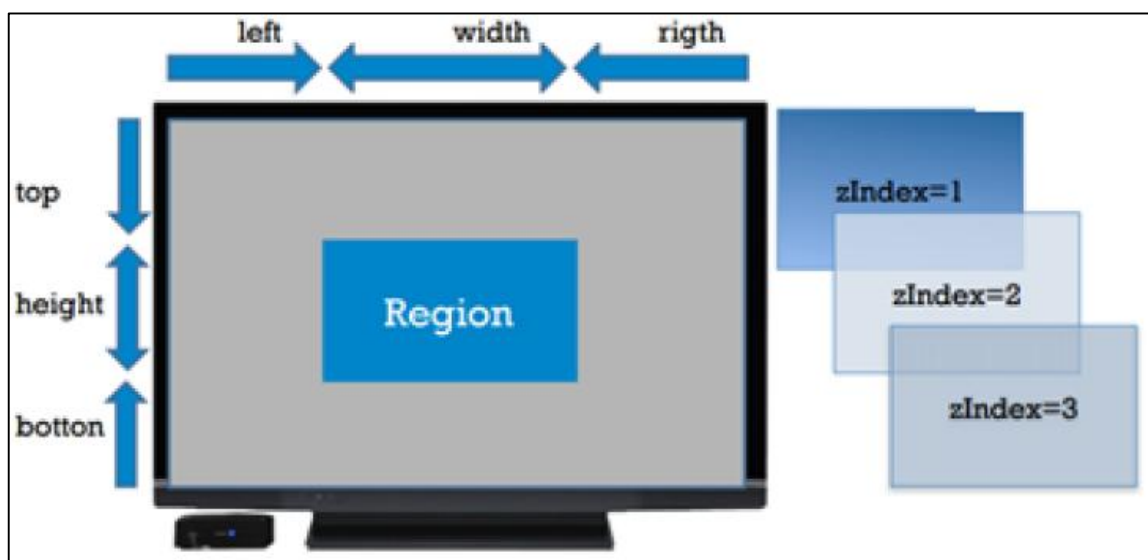


Figura 3.3. Atributos que pueden ser especificados en una región.

Fuente: <http://www.televisiondigital.gub.uy/download.php?m=n&i=31>

- Base de descriptores.

Los descriptores definen cómo los archivos medios serán presentados, esto significa que cuando declaramos una entrada de un archivo medio tenemos que asociarle a un descriptor para presentar algunas características auxiliares para la exhibición del mismo. Cada descriptor determina la región donde será presentado el archivo medio, como será el final de la presentación del archivo medio, los efectos de transición y también propiedades como volumen, transparencia y otros.

La navegación dentro de la aplicación se la puede hacer con ayuda de los descriptores, para esto se utiliza un foco que se utiliza para identificar el medio seleccionado en ese momento. A diferencia de las regiones, los descriptores no se pueden ser anidados.

En la siguiente figura se muestra una base de descriptores con algunos de sus atributos, los más importantes y obligatorios son "id" y "región", los demás son opcionales:

```

<!--DESCRIPTOR-->
<descriptorBase>
  <descriptor id="desTV" region="rgTV"/>
  <descriptor id="desVideo" region="rgVideo"/>
  <descriptor id="desInfo" region="rgInfo" explicitDur="20s" transIn="trin1"/>
  <descriptor id="desExit" region="rgExit" transIn="trin1"/>
  <descriptor id="desBr" region="rgBr" transIn="trin1"/>
  <descriptor id="desBg" region="rgBg" transIn="trin1"/>
  <descriptor id="desBy" region="rgBy" transIn="trin1"/>
  <descriptor id="desBb" region="rgBb" transIn="trin1"/>
  <descriptor id="desImg1" region="rgImg1" transOut="trin2"/>
  <descriptor id="desImg2" region="rgImg2" transOut="trin1"/>
  <descriptor id="desImg3" region="rgImg3" transOut="trin2"/>
  <descriptor id="desTituloPr" region="rgTituloPr" transIn="trin2" />
  <descriptor id="desTextoPr" region="rgTextoPr" transIn="trin1"/>
  <descriptor id="desTitulo" region="rgTitulo" transIn="trin1"/>
  <descriptor id="desTexto" region="rgTexto" transIn="trin1"/>
</descriptorBase>

```

Figura 3.4. Estructura de una base de descriptores.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

- ✓ **id**: nombre que identifica al descriptor de forma única a lo largo de todo el programa.
- ✓ **explicitDur**: tiempo de duración de exhibición del objeto medio.
- ✓ **region**: enlaza a una región definida en los elementos de la <regionBase>.
- ✓ **freeze**: especifica lo que ocurre con un objeto al final de su presentación.

- ✓ **transIn:** se referencia a un efecto de transición de cierto tiempo, que especifica cómo deberá iniciar el objeto media.
- ✓ **transOut:** se referencia a un efecto de transición de cierto tiempo, que especifica cómo deberá iniciar el final de un objeto media.

Los atributos que se muestran a continuación sirven para dar foco a los objetos media, de este modo se puede realizar la navegación en la aplicación:

- ✓ **moveLeft:** especifica el cambio de foco o de selección del objeto media cuando se pulse la flecha en dirección hacia la izquierda del control remoto.
- ✓ **moveRight:** especifica el cambio de foco o de selección del objeto media cuando se pulse la flecha en dirección hacia la derecha del control remoto.
- ✓ **moveDown:** especifica el cambio de foco o de selección del objeto media cuando se pulse la flecha en dirección hacia abajo del control remoto.
- ✓ **moveUp:** especifica el cambio de foco o de selección del objeto media cuando se pulse la flecha en dirección hacia arriba del control remoto.
- ✓ **focusIndex:** especifica el índice o valor del foco del objeto media que será utilizado para realizar la navegación sobre los mismos.
- ✓ **focusBorderColor:** define el color que se presentara en el borde del media cuando este sea seleccionado en la navegación, los colores reservados son: “white”, “black”, “silver”, “gray”, “red”, “maroon”, “fuchsia”, “purple”, “lime”, “green”, “yellow”, “olive”, “blue”, “navy”, “aqua”, “teal”.
- ✓ **focusBorderWidth:** establece el ancho del color de borde del objeto media cuando este se encuentre seleccionado.
- ✓ **focusBorderTransparency:** establece el nivel de transparencia del objeto media cuando se encuentra seleccionado.
- ✓ **focusSrc:** especifica un contenido alternativo que se presentará en lugar del que se encuentra en la presentación actual, solo sucede cuando el objeto media es seleccionado o está en foco.
- ✓ **focusSelSrc:** especifica un contenido alternativo que será presentado cuando un elemento en foco sea seleccionado presionando la tecla de activación “OK”.
- ✓ **selBorderColor:** especifica un color de borde cuando el objeto media se encuentra seleccionado.

Cuando se declara un descriptor, este puede poseer algunos parámetros (*descriptorParam*) que poseen dos atributos:

- ✓ **name:** especifica el nombre el parámetro o atributo.
- ✓ **value:** establece un valor inicial para el parámetro o atributo definido.

Dependiendo del tipo de archivo media que se utilice en el programa, se utilizará ciertos parámetros para cada uno de ellos, por ejemplo:

- Para archivos media de audio: “*soundLevel*”, “*balanceLevel*”, “*trebleLevel*”, “*bassLevel*”.
 - Para archivos media visuales: “*size*” (*w*, *h*), “*bounds*” (*l*, *t*, *w*, *h*), “*backGround*”, “*visible*”, “*transparency*”, “*fit*” (*fill*, *hidden*, *meet*, *meetBest*, *slice*).
- Base de conectores.

Los conectores definen cómo se activan los enlaces y lo que hace cada uno de ellos. Con ayuda de estos conectores se puede hacer que cuando inicie un objeto media, simultáneamente inicie o termine otro, esto dependerá de cómo este formado cada conector. Existe también la posibilidad de crear nuestros propios conectores dependiendo del uso que se les vaya a dar a lo largo de la creación de la aplicación interactiva.

A continuación se presenta una figura con una base de conectores, en la cual se definen acciones como “*onBeginStartNPauseN*” es decir cuando se inicie algún nodo se iniciará otro simultáneamente y otro puede pausarse su reproducción.

```

<causalConnector id="onBeginStartNPauseN">
  <simpleCondition role="onBegin"/>
  <compoundAction operator="seq">
    <simpleAction role="start" max="unbounded" qualifier="par"/>
    <simpleAction role="pause" max="unbounded" qualifier="par"/>
  </compoundAction>
</causalConnector>

<causalConnector id="onBeginStartNResumeN">
  <simpleCondition role="onBegin"/>
  <compoundAction operator="seq">
    <simpleAction role="start" max="unbounded" qualifier="par"/>
    <simpleAction role="resume" max="unbounded" qualifier="par"/>
  </compoundAction>
</causalConnector>

```

Figura 3.5. Estructura de una base de conectores.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Los elementos hijos utilizados dentro de `causalConnector` son:

- ✓ **connectorParam:** establece parámetros cuyos valores deberán ser definidos por los enlaces que utiliza el conector.
- ✓ **simpleConditionn:** define las condiciones simples de activación del enlace que utiliza el conector.
- ✓ **simpleAction y compoundAction:** definen acciones simples o complejas que se realizan cuando el enlace que utiliza el conector es activado.

Dentro de estos elementos de un `causalConnector` se definen ciertos atributos:

- ✓ **name:** nombre del parámetro.
- ✓ **role:** nombre del papel o acción que va a realizarse. Este atributo es obligatorio y debe ser único dentro del conector.
- ✓ **key:** indica el código de la tecla del control remoto que activa el enlace ("link").
- ✓ **delay:** tiempo en segundos recorrido desde que se inicia la condición verdadera, o desde que se activa un role o acción.
- ✓ **max:** establece el número máximo de medias que pueden ser asociados a una misma acción.

- ✓ **qualifier:** establece que todas las condiciones o acciones deben ser ligadas al rol (“and”) o es suficiente que una de estas lo esté (“or”).
- ✓ **value:** valor de atribución de las propiedades asociadas al rol.
- ✓ **operator:** establece el orden de disparo de las acciones, es decir, en forma paralela (“par”) o en forma secuencial (“seq”).

Generalmente los conectores están definidos en un archivo externo que se encuentra en el mismo directorio del archivo ncl, el cual luego es importado por el programa principal para su uso a lo largo del código principal. La siguiente figura muestra las líneas de código necesario para realizar la importación del archivo de conectores al programa principal.

```

<!--IMPORTAR CONECTORES-->
<connectorBase>
  <importBase documentURI="ConnectorBase.ncl" alias="conector"/>
</connectorBase>
</head>
```

Figura 3.6. Línea de código para importar un documento de conectores.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

3.1.1.3. Cuerpo del documento NCL.

3.1.1.3.1. Nodos u objetos media.

Los nodos son referencias para cada archivo media. Funcionan como enlaces, cuando son activados presentan en pantalla un determinado objeto media. Los nodos pueden ser de dos tipos: nodo de contenido y nodo de contexto.

3.1.1.3.1.1. Nodos de contenido.

A este tipo de nodo se le asocia un tipo de archivo media (txt, html, jpeg, mpeg, etc.). Un nodo de contenido se inicia a través de la etiqueta <media>, por ejemplo:

```

<media id="brojo" type="image/png" src="media/botonrojo.png" descriptor="desBr"/>
```

Figura 3.7. Nodo de contenido de un archivo media de imagen en formato png.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

- **id:** identificador único y obligatorio.
- **src:** especifica la localización del objeto media, incluido el nombre y extensión del archivo.
- **descriptor:** identificador del descriptor al cual está asociado el objeto de media.
- **type:** es opcional pero, si un media no posee el atributo src, es obligatorio, ya que el programador puede definir medias para almacenar valores resultantes de la aplicación.

Es importante que los objetos media se encuentren en el mismo directorio del programa principal. Es necesario que se referencie éste media a través de un descriptor, el mismo que contiene información de dónde y cómo se lo presentará.

En el atributo type de este nodo, debe especificar el tipo de media utilizado, los tipos de media más comunes son: image/gif, image/jpeg, image/bmp, video/mpeg, text/plain, text/html, audio/mp3.

3.1.1.3.1.2. Nodos de contexto.

Estos nodos constan de los siguientes componentes:

- Puertas de acceso.

Una puerta se representa mediante la etiqueta <port>. Es un punto de interface de un contexto a través del cual podemos tener acceso al contenido del mismo, es decir, para que un enlace pueda reproducir un nodo de contexto interno, éste debe tener declarado una puerta que lo lleve a ese nodo interno.

```
<!--PUERTAS-->
<port id="pEntrada" component="video"/>
```

Figura 3.8. Puerta de enlace o acceso.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

En el ejemplo anterior, "id" representa el nombre del atributo puerta, y "component" define el nodo que será iniciado en un determinado contexto.

- Enlaces.

Los enlaces funcionan de manera similar a un link en un archivo html. A través de los enlaces es posible crear sincronización, por ejemplo, se inicia un objeto media e inmediatamente se inicia otro, y al finalizar este inmediatamente se puede iniciar o terminar otro. Estos enlaces funcionan de acuerdo a los conectores que han sido creados en la Base de Conectores que se encuentre en la cabecera del programa.

```
<link xconnector="conector#onKeySelectionStartNStopNSetN">
  <linkParam name="keyCode" value="INFO"/>
  <bind role="onSelection" component="info"/>
  <bind role="set" component="video" interface="bounds">
    <bindParam name="var" value="25%, 0%, 100%, 75%"/>
  </bind>
  <bind role="stop" component="info"/>
  <bind role="start" component="exit"/>
  <bind role="start" component="titulopr"/>
</link>
```

Figura 3.9. Enlaces para sincronización de objetos media.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

El atributo `xconnector` consta de dos partes las mismas que están separadas por el carácter "#". El primer valor "conector" hace referencia a la base de conectores donde se encuentra el conector que será utilizado (atributo "alias" de la base de conectores), y el segundo valor "onKeySelectionStartNStopNSetN" indica el conector que será utilizado del documento de conectores importado.

La etiqueta `<bind>` indica el papel que desempeña cada nodo en ese enlace, es decir, que nodo de contenido se activa o desactiva. En la figura anterior, si se presiona la tecla "info" del control remoto la pantalla se redimensiona, esto gracias al atributo "set" mediante la interface "bounds", además se detiene el nodo "info" y se activan los botones de "exit" y "titulo", todo esto de manera simultánea.

- Áreas o anclas.

Las anclas son puntos de entrada para los nodos de media de contextos. El objetivo de estos, es utilizar segmentos de un nodo de media de contexto, sea como enlaces de origen o de destino.

Existen dos tipos de anclajes: ancla de contenido y ancla de atributo.

- ✓ **Ancla de contenido:** define un segmento del media (intervalo de tiempo y/o región en el espacio) que se pueden utilizar como punto de activación de los enlaces. Un ancla de contenido es definido con una etiqueta <area> dentro de un nodo de contenido <media>.

```
<media id="video" src="media/drone.mp4" descriptor="desVideo">
  <area id="aVideoLeyenda01" begin="5s" end="9s"/>
  <area id="aVideoLeyenda02" begin="10s" end="14s"/>
  <area id="aVideoLeyenda03" begin="15s" end="19s"/>
</media>
```

Figura 3.10. Ancla de contenido en un nodo media de vídeo.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

En la figura anterior, se observa tres anclas definidas para la presentación de una leyenda a lo largo del vídeo. El atributo “begin” indica el tiempo que debe transcurrir desde que se inicia el vídeo para que el archivo media sea activado, y el atributo “end” indica el tiempo en que el objeto media será desactivado.

- ✓ **Ancla de atributo:** son atributos de un nodo de origen o de destino que pueden ser manipulados por los enlaces, por ejemplo: se puede manipular el volumen de un archivo media de audio o vídeo, o se puede modificar las coordenadas y dimensiones de algún archivo media visual.

Esta ancla es definida mediante la etiqueta <attribute> dentro de un nodo de contenido <media> o <context>.

```
<media id="video" src="media/drone.mp4" descriptor="desVideo">
  <property name="top" value="25%"/>
  <property name="left" value="25%"/>
  <property name="width" value="50%"/>
  <property name="height" value="50%"/>
</media>
```

Figura 3.11. Ancla de atributo para modificar las dimensiones y coordenadas de un nodo de media de un vídeo.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

3.2. Lenguaje de programación LUA.

En NCL es difícil realizar tareas que consisten en la manipulación de texto, uso del canal de interactividad, animaciones y tareas que requieran algoritmos y estructuración de datos, para esto necesitamos la ayuda de un lenguaje imperativo.

La integración de NCL con un ambiente imperativo, da como resultado la creación de una clase de objetos de media tipo LUA llamados NCLua. Un objeto NCLua además de realizar las tareas anteriormente mencionadas, también puede responder a las teclas del control remoto.

3.2.1. Integración NCL-LUA.

El lenguaje de programación fue extendido con nuevas funcionalidades para poder integrarse con NCL y de este modo saber cuándo un objeto NCLua es activado en un enlace para su ejecución. Los siguientes módulos están disponibles para scripts NCLua:

- **Módulo event:** permite que los objetos NCLua se comuniquen a través de eventos con el documento NCL, y también con entidades externas como el control remoto y el canal de interactividad.
- **Módulo canvas:** proporciona una API para dibujar gráficos e imágenes simples en la región definida.
- **Módulo settings:** permite acceder a las variables definidas en el objeto de tipo "application/x-ncl-settings" del documento NCL.
- **Módulo persistent:** exporta una tabla con variables persistentes, que están disponibles para la manipulación solo de objetos imperativos.

Un objeto NCLua debe registrar una función que procesa eventos para que los eventos externos (lenguaje NCL) sean recibidos a través del llamado a la función `event.register`. Las líneas de código que existen entre “function handler (evt)” y “event.register”, se ejecutan antes de que NCL envíe otro evento, incluso antes de la presentación de cualquier objeto.

```
... -- código de inicialización
function handler (evt)
  ... -- código de un tratador
end
event.register(handler) -- registro de un tratador
```

Figura 3.12. Lenguaje de programación de un tratador de eventos.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

A continuación se muestran las líneas de código en LUA, mediante el cual se indica que se ha presionado la tecla roja.

```
evt = {
  class = 'key',
  type = 'press',
  key = 'RED'
}
```

Figura 3.13. Código para identificar los botones del control remoto.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Además de lo mencionado anteriormente, se necesita que LUA sea capaz de publicar los eventos que ocurren en el objeto NCLua, esto se lo consigue a través de la función `event.post`. Esta función se la describe en las siguientes líneas de código.

```
event.post {
  class = 'ncl',
  type = 'presentation',
  action = 'stop'
}
```

Figura 3.14. Código para enviar los eventos al documento NCL.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Se debe tener en cuenta que cuando se escriben las tablas en LUA, se menciona el campo “class” el cual tiene como finalidad separar los eventos en categorías, es así que tenemos los siguientes tipos de clases de eventos:

- **Clase ncl:** para comunicar LUA con un documento NCL.
- **Clase key:** indica que se ha presionado un botón del control remoto.
- **Clase tcp:** permite acceder al canal de interactividad por medio del canal de retorno.
- **Clase user:** permite que el programador cree sus propios eventos, de este modo LUA puede extenderse en cuanto a su funcionalidad.

3.3. Herramientas de desarrollo.

Las herramientas usadas para el desarrollo de aplicaciones interactivas en el entorno ginga-NCL para televisión digital son: NCL Composer y Eclipse (con el plugin NCL).

3.3.1. NCL Composer.

NCL Composer v.0.2.1 es una herramienta que permite construir programas audiovisuales interactivos para TVDi en lenguaje NCL. Este software fue desarrollado por el laboratorio TeleMídia de la Pontificia Universidad Católica de Río de Janeiro.

Posee una interfaz gráfica y brinda la capacidad de añadir plugins, los cuales se basan en dos archivos principales en C++. En esta herramienta el documento NCL puede presentarse de diferentes puntos de vista que son:

- **Visión estructural:** permite la creación de nodos para los diferentes tipos de media (audio, vídeo, imágenes, otros) así como definir sus propiedades, además se permite crear sincronismo entre ellos.

Se presenta mediante una interfaz gráfica donde se observa los nodos, cada uno con sus enlaces especificados en la programación de los mismos.

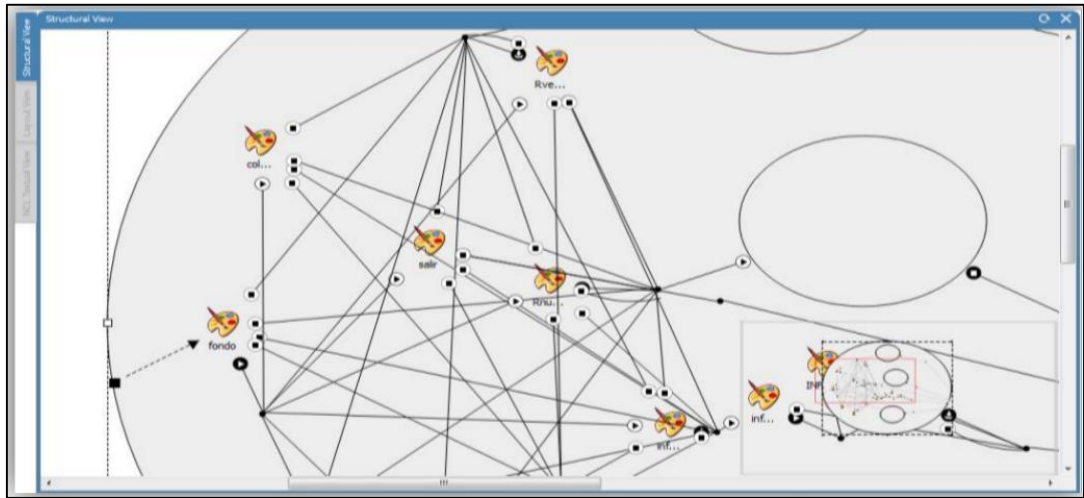


Figura 3.15. Visión estructural del software NCL Composer [27].

- **Visión de diseño:** permite representar las regiones en la pantalla donde se presentará cada objeto media, los mismos que son presentados con sus propiedades.

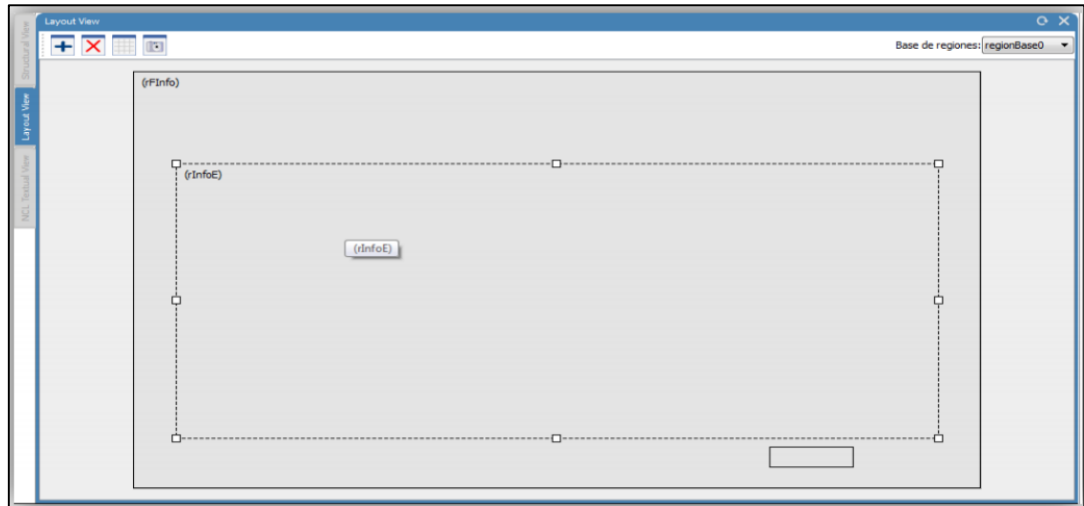


Figura 3.16. Visión de diseño del software NCL Composer [27].

- **Visión de esquema:** muestra todos los componentes que existen en el documento hipertexto, es decir todos los componentes que se generan en la cabecera y en el cuerpo del documento ncl.

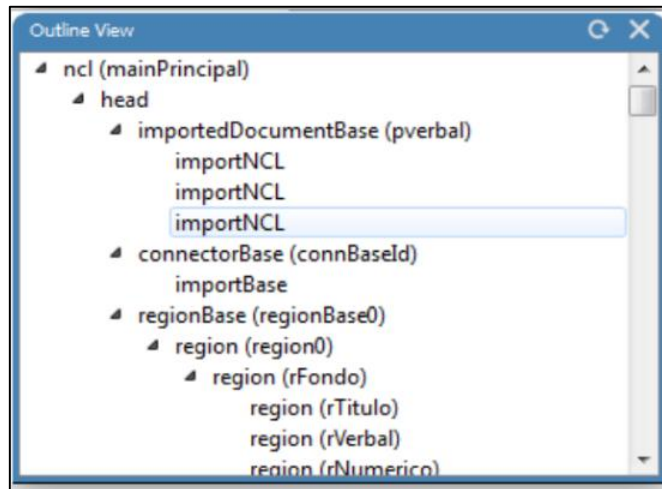


Figura 3.17. Visión de esquema del software NCL Composer [27].

- **Visión temporal:** permite visualizar el sincronismo temporal entre los nodos multimedia y los momentos de interactividad que puedan existir.

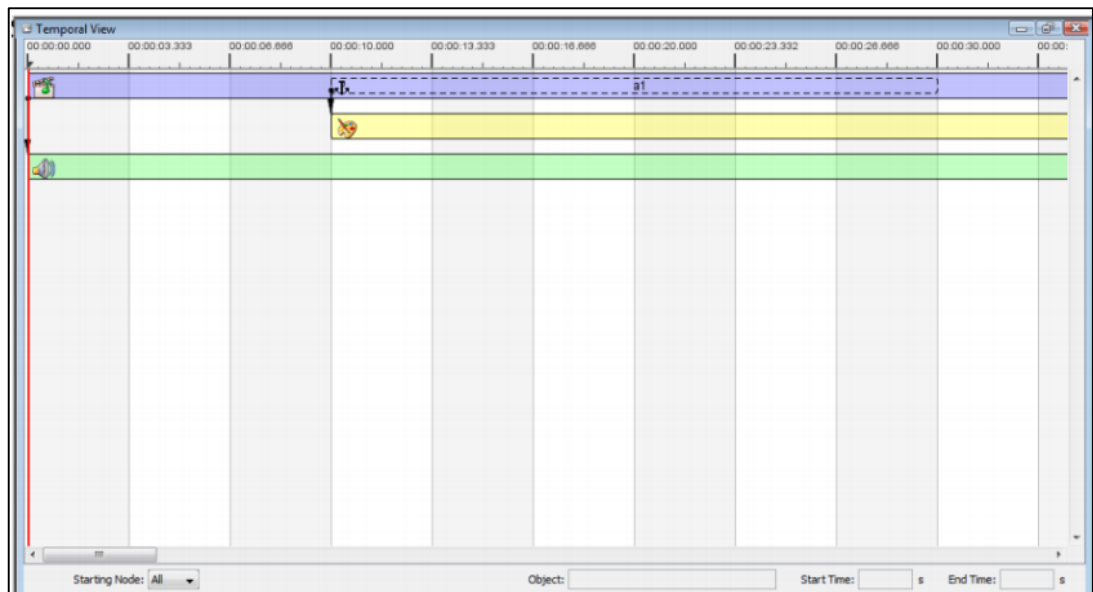


Figura 3.18. Visión temporal del software NCL Composer [27].

- **Visión textual:** permite visualizar el código NCL que se genera cuando manejamos las visiones anteriores, el programador tiene también la opción de editar directamente este código generado.


```

1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <!-- Generated by NCL Composer -->
3 <nd id="mainPrincipa">
4 <head>
5 <importedDocumentBase id="pverbal">
6 <importNCL alias="pVerbal" documentURI="media/pVerbal/PVerbal/PVerbal.ncl"/>
7 <importNCL alias="pNumerico" documentURI="media/pNumerico/PNumerico/PNumerico.ncl"/>
8 <importNCL alias="pAbstracto" documentURI="media/pAbstracto/PAbstracto/PAbstracto.ncl"/>
9 </importedDocumentBase>
10 <connectorBase id="connBaseId">
11 <importBase alias="conn" documentURI="defaultConnBase.ncl"/>
12 </connectorBase>
13 <regionBase id="regionBase0">
14 <region height="100.00%" id="rRegion0" left="0.00%" top="0.00%" width="100.00%" zIndex="1">
15 <region height="100.00%" id="rFondo" left="0.00%" top="0.00%" width="100.00%" zIndex="2">
16 <region height="15.00%" id="rTitulo" left="30.00%" top="2.00%" width="40.00%" zIndex="3"/>
17 <region height="8.00%" id="rVerbal" left="5.00%" top="25.00%" width="25.00%" zIndex="4"/>
18 <region height="8.00%" id="rNumerico" left="5.00%" top="38.00%" width="25.00%" zIndex="4"/>
19 <region height="8.00%" id="rAbstracto" left="5.00%" top="51.00%" width="25.00%" zIndex="4"/>
20 <region height="37.00%" id="rCollage" left="38.00%" top="22.00%" width="55.00%" zIndex="4"/>
21 <region height="7.00%" id="rInicio" left="38.00%" top="22.00%" width="4.00%" zIndex="5"/>
22 <region height="7.00%" id="rCerrar" left="93.00%" top="23.00%" width="4.00%" zIndex="5"/>
23 <region height="100.00%" id="rInfo" left="0.00%" top="0.00%" width="100.00%" zIndex="6"/>
24 <region height="8.00%" id="rInicio" left="75.00%" top="90.00%" width="10.00%" zIndex="7"/>
25 <region height="66.00%" id="rInfo" left="5.00%" top="22.00%" width="90.00%" zIndex="8"/>
26 </region>
27 </region>
28 </regionBase>
29 <descriptorBase id="descriptorBase1">
30 <descriptor id="dCerrar" region="rCerrar"/>
31 <descriptor id="dCollage" region="rCollage"/>
32 <descriptor id="dFondo" region="rFondo"/>
33 <descriptor id="dInfo" region="rInfo"/>
34 <descriptor id="dAbstracto" region="rAbstracto" focusIndex="31" mouseClick="1" mouseLine="33"/>

```

Figura 3.19. Visión textual del software NCL Composer [27].

3.3.2. NCL Eclipse.

NCL Eclipse es un plugin para el IDE Eclipse, fue desarrollado por el Laboratorio de Sistemas Avanzados de la Web de la Universidad Federal de Maranhão, posee licencias GPLv2, es comercial y tiene como objetivo acelerar el desarrollo de aplicaciones interactivas para TVD, dando formato a los documentos NCL y scripts LUA [32]. Esta herramienta se utiliza para facilitar el desarrollo de aplicaciones declarativas, ya que presenta facilidades para la edición de documentos NCL y NCLua, ya que posee también un plugin LuaEclipse.

```

1 <?xml version="1.0" encoding="ISO-8859-1"?>
2 <!-- Generated by NCL Eclipse -->
3 <nd id="main" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
4 <head>
5 <!--EFECTOS DE TRANSICION-->
6 <transitionBase>
7 <transition id="trIn1" type="fade" dur="2s"/>
8 <transition id="trIn2" type="barsipe" dur="2s"/>
9 </transitionBase>
10 <!--REGIONES-->
11 <regionBase>
12 <region id="rgVideo" zIndex="0">
13 <region id="rgInfo" width="68" height="78" left="38" bottom="58" zIndex="2"/>
14 <region id="rgFondo1" height="78" width="258" zIndex="1">
15 <region id="rgTituloPr" height="128" width="808" top="38" left="108" zIndex="2"/>
16 <region id="rgTextOp" height="858" width="928" left="48" top="158" zIndex="2"/>
17 <region id="rgImg1" width="808" height="208" top="158" left="108" zIndex="2"/>
18 <region id="rgImg2" width="808" height="208" top="438" left="108" zIndex="2"/>
19 <region id="rgImg3" width="808" height="208" top="668" left="108" zIndex="2"/>
20 </region>
21 <region id="rgFondo2" height="258" width="1008" top="758" zIndex="1">

```

Figura 3.20. Ambiente de programación en NCL Eclipse.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

3.3.3. *Ginga GUI o Ginga4Windows.*

Ginga4Windows es una herramienta desarrollada por el laboratorio *TeleMídia* del departamento *Informático* de la *PUC-Río*, que permite ejecutar y visualizar las aplicaciones *NCL*, es de código abierto y trabaja sobre la plataforma de *Windows* [27].



Figura 3.21. Simulador *Ginga GUI* (*Ginga4Windows*).

Fuente: Creado por el autor mediante captura de pantalla del simulador *Ginga4Windows*.

3.3.4. *Ginga-NCL Virtual Set Top Box.*

Es una aplicación desarrollada por el Laboratorio *TeleMídia* del Departamento *Informático* de la *Pontificia Universidad Católica de Río de Janeiro* y se ejecuta mediante una máquina virtual, en este caso, por medio de la herramienta *VMware Workstation 12 Player*.

Es un emulador del decodificador/receptor de televisión digital del estándar *ISDB-Tb*, posee las mismas características que un *STB* físico con características avanzadas para obtener un buen rendimiento y mayor proximidad a una aplicación real embebida en un *STB*. Para la visualización de las aplicaciones interactivas se utiliza la máquina virtual "ubuntu-server10.10-ginga-v.0.12.4-i386".

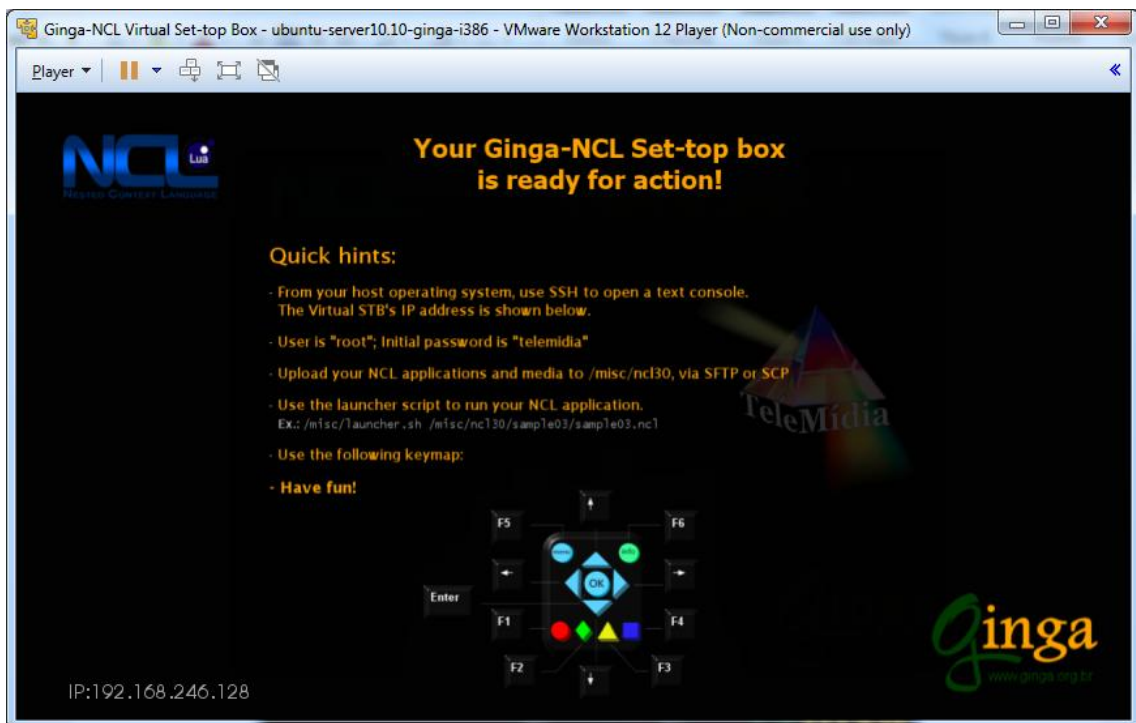


Figura 3.22. Emulador Ginga-NCL Virtual Set-top box

Fuente: Creado por el autor mediante captura de pantalla del emulador Ginga-NCL Virtual Set-top box

CAPÍTULO IV

4. DISEÑO DE APLICACIONES INTERACTIVAS

4.1. Diseño de la aplicación interactiva para Televisión Digital.

El diseño de una aplicación interactiva depende del fin de la misma y de los requerimientos del televidente, no existe un modelo a seguir para poder desarrollar una aplicación, el diseño se lo realiza en base a lo que el canal decida que es más importante y que le dará más utilidades. Por lo tanto, toda aplicación interactiva deberá regirse a la única regla de ser fácil de usar, entretenida y amigable con el usuario final.

En este proyecto se especifican las consideraciones que fueron tomadas en cuenta para el diseño de la aplicación interactiva.

4.1.1. Parámetros generales de la aplicación interactiva.

4.1.1.1. Selección del tema para la aplicación interactiva.

El tema elegido para el desarrollo de la aplicación interactiva nace de la necesidad de instruir e informar al público en general en cuanto respecta a nuestro mundo animal y vegetal a nivel de la provincia de Loja y del Ecuador. El Parque Recreacional Orillas del Zamora cuenta con una diversidad de lugares en los cuales el público se distrae, relaja y goza de las diferentes atracciones que ofrece este sitio, se cuenta con diferentes lugares como los baños de agua sulfurosa, kartódromo, orquideario, sendero ecológico, vivero y zoológico.

Para la aplicación a desarrollar en este proyecto se ha seleccionado los lugares más relevantes del Parque Recreacional Orillas del Zamora, el ZOOLOGICO MUNICIPAL, VIVERO MUNICIPAL y el ORQUIDEARIO MUNICIPAL. Se tratará de motivar, informar y educar mediante la aplicación interactiva a los televidentes para que tengan una idea clara de lo que existe en este lugar, de las diferentes especies animales y vegetales, de sus hábitos y cuidados, lugar donde habitan, características de cada especie, entre otros.

4.1.1.2. Estándar de Televisión Digital Terrestre.

El estándar de televisión digital terrestre es elegido por cada país en base a sus características, ventajas y desventajas, nivel de tecnología, entre otros aspectos, en base a esto los canales de televisión deben acoplarse al estándar elegido. Es necesario para el programador conocer el

estándar para tener conocimiento del middleware que se va a utilizar y de este modo conocer la plataforma de programación para generar la aplicación interactiva.

En la transición hacia la TDT, Ecuador ha elegido el estándar de televisión digital ISDB-Tb o SBTVD el cual se especifica en el apartado 1.3.5, en éste se definen las características del estándar, su tecnología, su fusión con el estándar japonés ISDB-T y su acogida en los diferentes países en su mayoría de Sudamérica.

4.1.2. Middleware y plataforma de programación.

Para el desarrollo de la aplicación interactiva, el estándar de televisión digital terrestre nos da a conocer la plataforma de programación en donde se desarrollará la misma, en este caso el estándar ISDB-Tb utiliza el middleware GINGA, éste se especifica a detalle en el apartado 2.5 mencionado sus ventajas y desventajas, sus ambientes de programación y ejecución de aplicaciones, tanto declarativas (Ginga-NCL-Lua) como procedurales (Ginga-J).

La aplicación interactiva del Parque Recreacional Orillas del Zamora será desarrollada como una aplicación declarativa, por lo tanto, el motor de ejecución del mismo corresponde a Ginga-NCL. El código se basa en el lenguaje NCL.

4.1.3. Interactividad y servicios interactivos.

Se utilizará una interactividad local, este está especificado en el apartado 2.2.1. Este tipo de interactividad es accesible al público en general por el hecho que no se necesita de un canal de retorno. En el futuro se podría implementar un canal de retorno para tener una comunicación bidireccional, esto dependerá de la acogida de la aplicación y de los servicios adicionales que se puedan implementar.

Los servicios interactivos que puede ofrecer una aplicación interactiva se especifican en el apartado 2.3, la aplicación actual ofrece servicios de información ya que se hace referencia a los diferentes tipos de animales del zoológico, las variedades de orquídeas y de plantas frutales, ornamentales y de otras especies. Se hace una recopilación de información acerca de cada espécimen y se la detalla en la forma más adecuada posible para que al televidente se le haga fácil entender y conocer a cada uno de estos animales y plantas.

Además de esto integra el servicio de T-Learning o educación mediante la televisión ya que, además de brindar la información necesaria acerca de los elementos del parque recreacional, también ayuda a que los niños aprendan de las diferentes especies animales y vegetales que existen, es decir, de donde provienen, cuáles son sus hábitos, como conviven, de que se alimentan y otras características, haciendo de este modo que se tenga una imagen de lo que se encuentra en el parque y así se incentiva a que acudan al lugar para que conozcan estas diferentes especies.

4.1.4. Requerimientos de la interfaz de la aplicación interactiva.

En apartados anteriores se mencionó que ninguna aplicación interactiva tiene normativas de diseño ni parámetros a seguir, la presente aplicación se desarrolló con el objetivo de que sea de agradable visualización, de contenidos fácilmente legibles y entendibles.

Se han considerado los siguientes parámetros basándonos en la información recopilada en el apartado 2.4:

- **Layout de pantalla:** se utilizó el modo de pantalla completa sin vídeo, debido a que es una aplicación compleja, necesita de un espacio amplio para poder visualizar todo el contenido que se ha introducido en la aplicación.
- **Navegación:** se ha utilizado la navegación mediante menú combinado con navegación mediante las teclas de colores y las teclas direccionales del control remoto. La programación de cada tecla se la realiza en el documento NCL donde se especifica el acceso a información dependiendo de la tecla que se presione.
- **Funciones básicas:** el método de acceso para la aplicación se la realiza mediante la manipulación de las teclas del control remoto. Se presenta un indicador en pantalla denominado INFO, el cual solo es accesible al ambiente aplicativo si el televidente presiona dicha tecla. Se cuenta también con otras funciones básicas como la opción de regresar a la página anterior (BACK), salir de la aplicación interactiva (EXIT), selección de contenido mediante las teclas principales del control remoto (RED, GREEN, YELLOW, BLUE) y visualización de imágenes mediante las teclas direccionales (LEFT, RIGHT).
- **Presentación de contenido:** Para tener una mejor visualización del contenido se utilizó el tipo de letra Comic Sans MS y Arial, con un tamaño de fuente 30 para títulos principales, 25 para los títulos secundarios y 21 para los contenidos informativos. Los colores de fuente

utilizados varían de acuerdo al área de presentación del contenido dependiendo si es más oscura o más clara la imagen de fondo.

4.2. Desarrollo de la aplicación interactiva.

4.2.1. Algoritmo: PSeInt.

Esta herramienta es utilizada para desarrollar el algoritmo mediante diagramas de flujo, esto con el fin de indicar el flujo de la información mediante el teclado del control remoto y funcionamiento de las teclas de acceso como de salida de la aplicación interactiva. El diagrama de flujo se encuentra en el ANEXO A.

4.2.2. Codificación: NCL Eclipse.

NCL Eclipse es necesario para desarrollar el código en lenguaje declarativo NCL. De acuerdo con el apartado 3.1.1 se ha ido desarrollando la aplicación siguiendo cada una de las estructuras de las que se conforma un documento NCL.

Se ha declarado cada región de visualización de imágenes, textos y botones, programación de cada descriptor para saber cómo se tiene que presentar los archivos media.

```
66     <region id="rgAppmapa" zIndex="3"> <!--REGIONES BOTON AZUL MAPA-->
67         <region id="rgimg_mapa" width="68.5%" height="82%" left="3%" top="2.5%" zIndex="4"/>
68         <region id="rgtxt_mapa" width="23%" height="60%" left="72%" top="23%" zIndex="4"/>
69         <region id="rgtxt_mapa1" width="63%" height="13%" left="18%" top="85.5%" zIndex="4"/>
70     </region>
71 </region>
72 </regionBase>
73 <descriptorBase>
74     <descriptor id="dTv" region="rgTV"/>
75     <descriptor id="dApp" region="rgApp"/>
76     <descriptor id="dmusic"/>
77     <!--DESCRIPTOR BOTON INICIAL-->
78     <descriptor id="dreinicioinfo" region="rgreiniciainfo"/>
79     <descriptor id="dinfo" region="rginfo"/>
80     <!--DESCRIPTORES PANTALLA INICIAL-->
81     <descriptor id="dloading" region="rgloading" explicitDur="3s"/>
82     <descriptor id="dloadingmap" region="rgloading" explicitDur="1.5s"/>
83     <descriptor id="dtxt_infoprin" region="rgtxt_infoprin"/>
```

Figura 4.1. Líneas de código de las regiones y descriptores de la aplicación interactiva.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Se ha importado un documento de conectores base que serán necesarios para enlazar una acción con otra dependiendo del flujo de información de la aplicación.

```
141      <!--IMPORTACIÓN DEL DOCUMENTO DE CONECTORES-->
142      <connectorBase>
143          <importBase documentURI="ConnectorBase.ncl" alias="conector"/>
144      </connectorBase>
```

Figura 4.2. Líneas de código de la importación del documento NCL de conectores de la aplicación interactiva.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Se han declarado los diversos archivos media con los cuales se va a trabajar a lo largo del documento NCL, estos son necesarios para su presentación en pantalla.

```
<!--ARCHIVOS MEDIA-->
    <media id="video" src="media/video.mp4" descriptor="dTv">
        <property name="soundLevel" value="1"/>
    </media>
<media id="music" src="media/relax.mp4" descriptor="dmusic"/>
<!--MEDIA BOTON INICIAL-->
<media id="reinicio_info" src="media/reinicio_info.png" descriptor="dreinicioinfo"/>
<media id="info" src="media/info.png" descriptor="dinfo"/>
<!--MEDIAS PANTALLA INICIAL-->
<media id="loading" src="media/fondo_load.png" descriptor="dloading"/>
<media id="fondo" src="media/fondo_prin.png" descriptor="dApp"/>
<media id="txt_infoprin" src="media/info_txt_prin.png" descriptor="dtxt_infoprin"/>
<media id="txt_descParque" src="media/txt_descParque.png" descriptor="dtxt_descParque"/>
<media id="img_inicial" src="media/img_prin.png" descriptor="dimginicial"/>
<media id="exit" src="media/exit.png" descriptor="dexit"/>
<media id="bot_redprin" src="media/bot_red.png" descriptor="dbotred"/>
<media id="bot_greenprin" src="media/bot_green.png" descriptor="dbotgreen"/>
```

Figura 4.3. Líneas de código de los archivos media de la aplicación interactiva.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Por último se ha programado cada enlace dependiendo si se quiere presentar o detener algún tipo de presentación de los archivos media.

```

<!--ENLACE-->
<!-- Inicio aplicación -->
<link xconnector="conector#onBeginStartNDelay">
  <bind role="onBegin" component="video"/>
  <linkParam name="delay" value="5s"/>
  <bind role="start" component="info"/>
</link>
<!-- Reinicio FONDO MUSICAL -->
<link xconnector="conector#onEndStart">
  <bind role="onEnd" component="music" />
  <bind role="start" component="music" />
</link>
<!-- Reinicio INFO -->
<link xconnector="conector#onEndStartNDelay">
  <bind role="onEnd" component="reinicio_info" />
  <bind role="start" component="info" />
  <linkParam name="delay" value="300s"/>
</link>
<!-- Presentación INFO -->
<!-- Cambio de audio video a fondo musical -->
<link xconnector="conector#onKeySelectionSetN">
  <bind role="onSelection" component="info" />
  <linkParam name="keyCode" value="INFO"/>
  <bind role="set" component="video" interface="soundLevel" >
    <bindParam name="var" value="0"/>
  </bind>
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="info" />
  <linkParam name="keyCode" value="INFO"/>
  <bind role="stop" component="info" />
  <bind role="start" component="reinicio_info" />
  <bind role="start" component="loading" />

```

Figura 4.4. Líneas de código de los enlaces de presentación de los archivos media de la aplicación interactiva.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

Los archivos multimedia fueron desarrollados mediante la herramienta Adobe Illustrator CS5.1 y Adobe Photoshop CS5.1 a excepción de las fotografías de los animales y plantas del parque recreacional y del archivo de audio. El código completo de la aplicación se encuentra en el ANEXO B.

4.2.3. Layout de pantalla: NCL Composer.

NCL Composer es utilizado para obtener una visión estructural y una visión de diseño de la interfaz de la aplicación, de este modo se puede manipular las regiones inicialmente declaradas para ubicarlas en el lugar que se desee en pantalla, consiguiendo así una visualización más agradable para el televidente. La visualización estructural y de diseño se muestra en el ANEXO C.

4.3. Uso del teclado del control remoto.

En la normativa brasileña ABNT 15604 [18] se especifican ciertas características y funciones mínimas que los fabricantes de STB deben tomar en cuenta para la fabricación del control remoto de su equipo, por ejemplo, teclas numéricas, teclado de control de canal, teclado de control de volumen, tecla de acceso a EPG, tecla de información, teclas de colores, entre otros.

Para la presente aplicación se ha utilizado el teclado interactivo (de colores, de dirección, de retorno y de salida) y teclado de navegación (tecla de información), estos se pueden observar en la imagen 4.5:

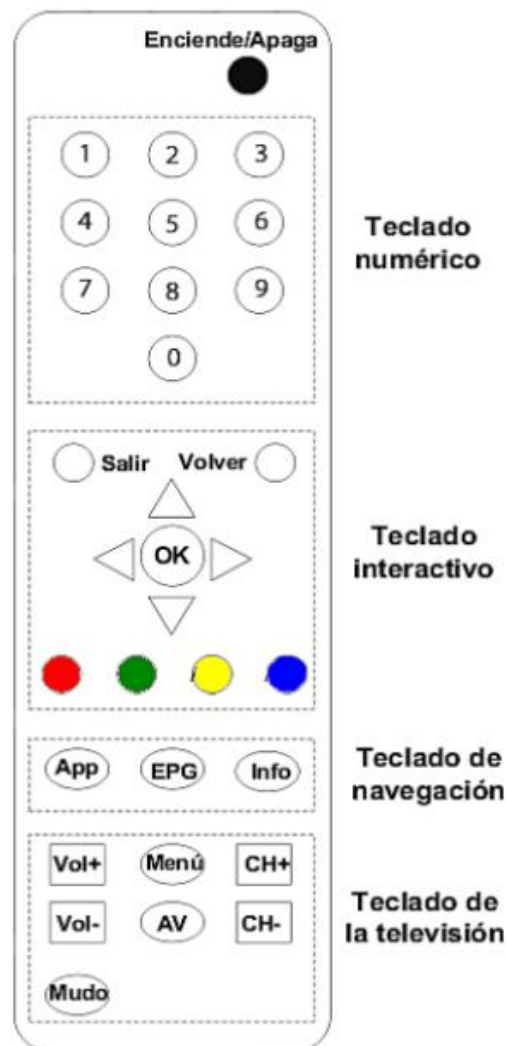


Figura 4.5. Modelo básico de control remoto para televisión digital [36].

4.4. Funcionamiento de la aplicación interactiva.

La aplicación debe permitir una fácil interacción con el usuario, para lograr esto, se presentará un botón de acceso a la información del Parque Recreacional en la parte inferior izquierda de la pantalla, el cual solo permitirá el acceso al menú de información solo si el usuario lo requiere, esto será posible si se presiona el botón interactivo llamado "INFO" el mismo que se presentará 5 segundos después de que el programa de televisión inicie.

Al presionar "INFO", se inicia la aplicación interactiva del Parque Recreacional Orillas del Zamora conjuntamente con un fondo musical que hace que la aplicación sea más llamativa, además se presenta un menú con cuatro opciones:

- Opción uno: "ZOOLOGICO", se ingresará a esta información mediante la interacción con el botón "ROJO" del control remoto.*
- Opción dos: "ORQUIDEARIO", se ingresará a esta información mediante la interacción con el botón "VERDE" del control remoto.*
- Opción tres: "VIVERO", se ingresará a esta información mediante la interacción con el botón "AMARILLO" del control remoto.*
- Opción cuatro: "¿CÓMO LLEGAR?", se ingresará a esta información mediante la interacción con el botón "AZUL" del control remoto.*

Al seleccionar la opción "ZOOLOGICO", se despliega un nuevo menú con cuatro nuevas opciones: se utiliza el botón "ROJO", "VERDE", "AMARILLO" y "AZUL" para acceder a la información de las "AVES", "MAMÍFEROS", "REPTILES" y "NUEVA ESPÉCIE" del zoológico respectivamente. Cada vez que se acceda a cada una de estas opciones, se habilita el teclado de navegación mediante el "CURSOR_LEFT" y el "CURSOR_RIGHT", esto para poder visualizar las diferentes imágenes e información de cada espécimen mostrado en cada opción.

La opción "ORQUIDEARIO", despliega un menú con cuatro opciones: se utilizan los botones "ROJO", "VERDE", "AMARILLO" y "AZUL" para acceder a la información de la "NAVE 1", "NAVE 2", "NAVE 3" y "NAVE 4" del orquideario respectivamente, al igual que en la opción anterior se habilita los botones "CURSOR_LEFT" y "CURSOR_RIGHT" para navegar por las diferentes imágenes.

La opción “VIVERO”, también despliega un menú con cuatro opciones: los botones “ROJO”, “VERDE”, “AMARILLO” y “AZUL” se utilizan para acceder a la información de “ÁRBOLES”, “FRUTALES”, “ORNAMENTALES”, “OTRA ESPECIE” del vivero respectivamente, al igual que las opciones anteriores se habilitan los botones “CURSOR_LEFT” y “CURSOR_RIGHT” para la visualización de las diferentes imágenes.

El número de especies animales, de orquídeas y de plantas del vivero es muy grande y esto hace que el código NCL se incremente cada vez más, es por esto que para desarrollar la aplicación interactiva como un “demo”, se utiliza cinco imágenes, cada una con su respectiva descripción para la visualización y navegación de cada una de las opciones anteriores: “ZOOLOGICO”, “ORQUIDEARIO” y “VIVERO”.

La última opción “¿CÓMO LLEGAR?”, presenta una imagen en pantalla con la dirección del Parque Recreacional Orillas del Zamora, un mapa del mismo y datos complementarios como contactos, precio, página web, entre otros.

Al ingresar a cada opción que existe en la aplicación existe el botón “BACK” que se utiliza para regresar al menú anterior, y el botón “EXIT” que se utiliza para salir por completo de la aplicación, una vez que sucede esto se vuelve a visualizar el botón “INFO” luego de 5 min con el fin de que la aplicación se encuentre disponible cada vez que el televidente necesite hacer uso del mismo.

4.5. Caracterización del Set Top Box para reproducción de la aplicación.

Las aplicaciones que existen para teléfonos celulares, para IPTV o para otros equipos son de gran utilidad para el usuario final, aunque dependerá de las necesidades del mismo para seleccionar su aplicación. Cada una de estas aplicaciones demanda de ciertas características para acoplarse al equipo necesario de ejecución, las aplicaciones para televisión digital también demandan de ciertas características para poder ejecutarse en el STB las mismas que dependerán de la complejidad de la aplicación.

Para reunir las características necesarias que debe cumplir el STB para la ejecución del presente proyecto, se realizó una prueba con una aplicación de mayor tamaño (20.5 MB) que la actual (18.2 MB), las mismas que estuvieron almacenadas en un pendrive de 8 GB. Estas pruebas se las

realizaron en el decodificador EITV Developer Box el mismo que tiene las siguientes características:

- *Procesador STi 7105 de 450 MHz.*
- *Memoria RAM de 256 MB.*
- *Memoria Flash de 128 MB.*
- *Soporta norma digital ISDB-T, IPTV y normas analógicas NTSC y PAL.*
- *Ancho de banda de 5 MHz y 6 MHz.*
- *Interfaces de comunicación USB 2.0 y Ethernet 100Mbps.*
- *Carga de aplicativos por aire, vía Internet o red local.*
- *Middleware GINGA.*

En las pruebas de funcionamiento el único problema que existió fue en la presentación de las imágenes, es decir, las imágenes se presentaban con un retardo de hasta dos segundos y en otras ocasiones no se presentaban. En lo que respecta al funcionamiento de los botones programados, no existió ningún inconveniente.

En base a esto, podemos decir que la aplicación interactiva del Parque Recreacional Orillas del Zamora al tener un tamaño menor a la aplicación de prueba, funcionará correctamente sin tener problemas de ejecución de los botones programados, pero puede que existan algunos problemas de reproducción de los archivos media; ésta duda se despejará luego de realizar las pruebas pertinentes.

Además de este decodificador de prueba existen múltiples decodificadores digitales que cumplen con los requisitos mínimos que la aplicación requiere, he aquí algunos ejemplos:

4.6. Simulación y pruebas de la aplicación interactiva.

El objetivo de simular la aplicación es para comprobar el funcionamiento total de la aplicación interactiva, es decir, revisar detalles de layout de pantalla, tamaños de fuente e imágenes, distribución de menús e información adicional, entre otros.

Tendremos un entorno similar al de un STB con Ginga cuando reproduce la aplicación en una televisión, para esto se ha utilizado algunos software como Ginga GUI o Ginga4Windows y Ginga-NCL Virtual Set Top Box, los cuales se han mencionado en los apartados 3.3.3 y 3.3.4 respectivamente.

4.6.1. Simulación en Ginga GUI (Ginga4Windows).

Esta simulación se la realiza con un computador que tiene las siguientes características:

- *Intel(R) Core(TM) i5-3210M CPU @ 2,50GHz.*
- *Memoria RAM de 4 GB.*
- *Disco duro con 5 Gb de espacio libre.*
- *Sistema operativo Windows 7 Ultimate.*

Este escenario de prueba se la realiza utilizando el software IDE Eclipse NCL en el cual se escribe la aplicación interactiva en lenguaje NCL, posteriormente ésta es compilada y reproducida mediante el software Ginga GUI. Para lograr esto, Ginga GUI debe estar incluido dentro de Eclipse.

En el layout de Eclipse vamos a Run, External Tools, External Tools Configurations, luego de esto se abre una ventana en la cual se realizan las siguientes configuraciones:

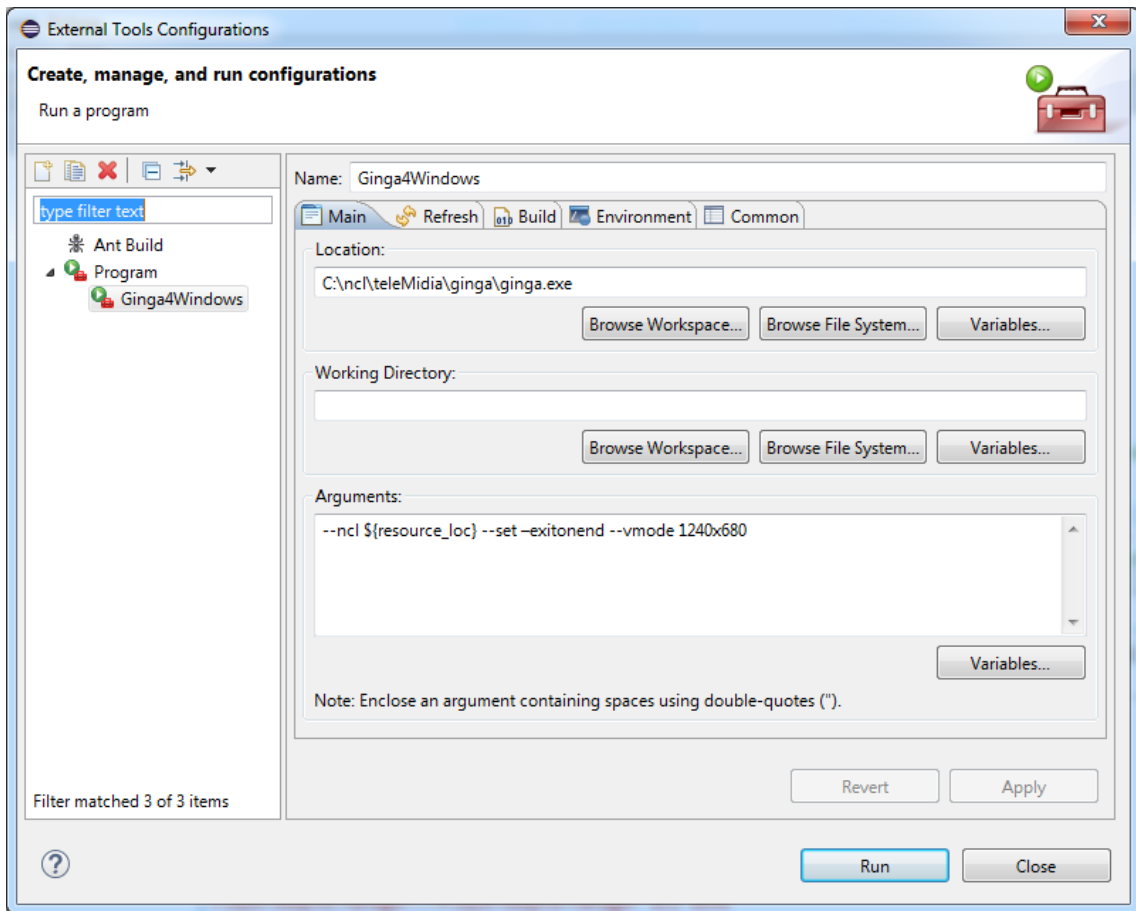


Figura 4.6. Configuración en Eclipse para utilizar el simulador Ginga4Windows.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

El documento NCL principal debe tener como nombre 'main.ncl', los archivos de imágenes en formato png y el archivo de audio en formato mp3.

Una vez realizado esto, se ejecuta la reproducción de la aplicación y los resultados obtenidos son los siguientes:



(a)



(b)

Figura 4.7. Simulación en Ginga4Windows. (a) Presentación del botón INFO durante un programa de televisión. (b) Presentación de la aplicación interactiva (menú principal).

Fuente: Creado por el autor mediante captura de pantalla del simulador Ginga4Windows.



(a)



(b)



(c)



(d)



(e)

Figura 4.8. Simulación en Gingga4Windows. (a) Presentación del menú secundario ZOOLOGICO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo AVES. (c) Presentación de la galería de imágenes del botón verde MAMÍFEROS. (d) Presentación de la galería de imágenes del botón amarillo REPTILES. (e) Presentación de la galería de imágenes del botón azul NUEVA ESPECIE.

Fuente: Creado por el autor mediante captura de pantalla del simulador Gingga4Windows.



(a)



(b)



(c)



(d)



(e)

Figura 4.9. Simulación en Gingga4Windows. (a) Presentación del menú secundario ORQUIDEARIO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo NAVE 1. (c) Presentación de la galería de imágenes del botón verde NAVE 2. (d) Presentación de la galería de imágenes del botón amarillo NAVE 3. (e) Presentación de la galería de imágenes del botón azul NAVE 4.

Fuente: Creado por el autor mediante captura de pantalla del simulador Gingga4Windows.



(a)



(b)



(c)

(d)



(e)

Figura 4.10. Simulación en Gingga4Windows. (a) Presentación del menú secundario VIVERO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo ARBOLES. (c) Presentación de la galería de imágenes del botón verde FRUTALES. (d) Presentación de la galería de imágenes del botón amarillo ORNAMENTALES. (e) Presentación de la galería de imágenes del botón azul OTRAS ESPECIES.

Fuente: Creado por el autor mediante captura de pantalla del simulador Gingga4Windows.



Figura 4.11. Simulación en Gingga4Windows. Presentación del menú secundario CÓMO LLEGAR con una opción de retorno.

Fuente: Creado por el autor mediante captura de pantalla del simulador Gingga4Windows.

Una vez realizada la simulación en el ambiente de Ginga4Windows se realizan las siguientes observaciones:

- *La respuesta de presentación de las imágenes es menor a un segundo.*
- *La reproducción del archivo de audio nunca se detuvo durante la ejecución de la aplicación, éste audio se encuentra en formato mp3 ya que es un formato que admite Ginga4Windows.*
- *El archivo original de audio debe estar en formato aac ya que es el estandarizado para la televisión digital terrestre, pero éste simulador no lo admite.*
- *El audio de vídeo de fondo que sirve como simulador para un programa de televisión, se detuvo.*
- *Todas las imágenes se presentaron en cada región designada según lo establecido en NCL Composer para el layout de pantalla.*
- *El control remoto funciona correctamente en la ejecución del programa.*
- *La señal digital no se puede utilizar en esta simulación es por tanto que se utiliza un vídeo en formato mp4 para simular el programa de televisión en ejecución normal.*
- *El reinicio de la aplicación interactiva luego de presionar el botón EXIT, funcionó correctamente.*

4.6.2. Simulación en Ginga-NCL Virtual Set Top Box.

Esta simulación se realiza con el computador de características indicado en el punto 4.6.1 y utilizando el software Eclipse NCL conjuntamente con el Ginga-NCL Virtual Set Top Box que es un emulador de STB el cual funciona en una máquina virtual en este caso VMware Workstation 12 Player. Para lograr esto, Ginga-NCL Virtual Set Top Box debe estar incluido dentro de Eclipse.

En el layout de Eclipse vamos a Window, Preferences, NCL, Remote Ginga-NCL Play, luego de esto se abre una ventana en la cual se realizan las siguientes configuraciones:

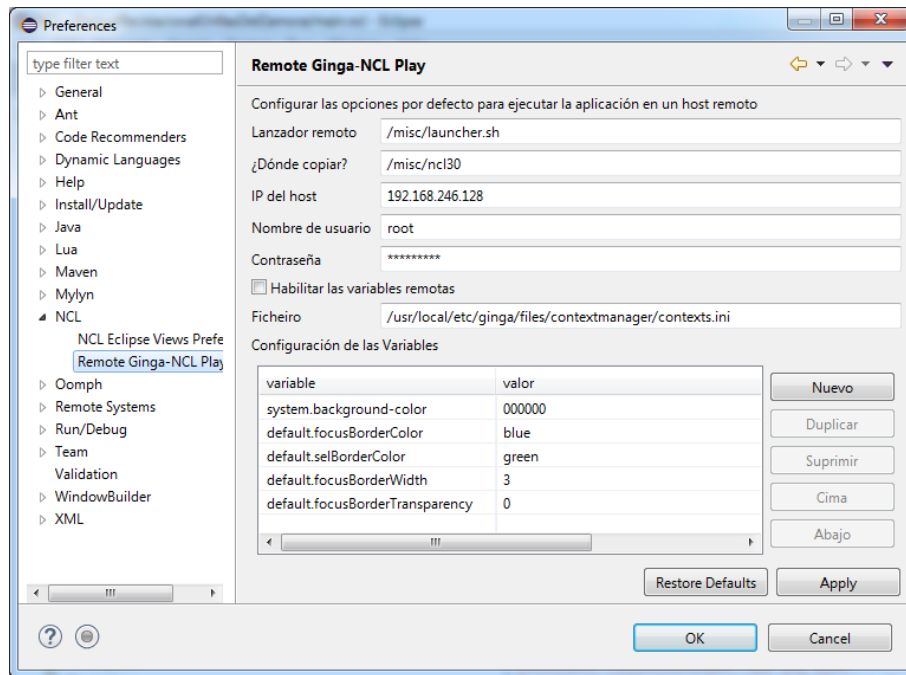


Figura 4.12. Configuración en Eclipse para utilizar el emulador Ginga-NCL Virtual Set Top Box.

Fuente: Creado por el autor mediante captura de pantalla del IDE de Eclipse.

En el recuadro “IP del host” hay que ingresar la dirección IP del emulador, ésta aparece en la pantalla principal de la máquina virtual, siempre y cuando el emulador se esté ejecutando.

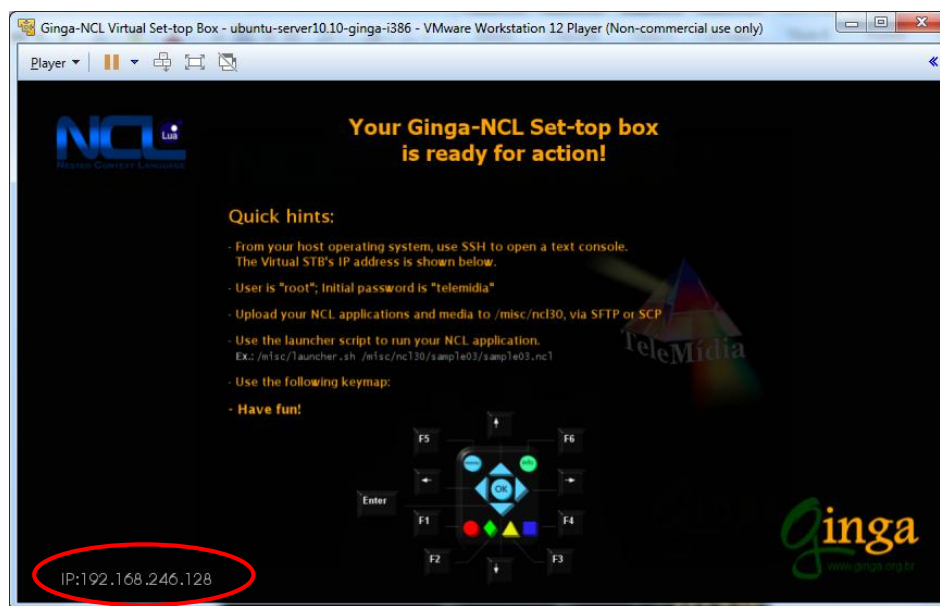


Figura 4.13. Dirección IP del emulador Ginga-NCL Virtual Set Top Box.

Fuente: Creado por el autor mediante captura de pantalla de la máquina virtual VMware Workstation 12 Player.

Una vez realizado esto, realizamos la reproducción de la aplicación y los resultados obtenidos son los siguientes:



(a)



(b)

Figura 4.14. Simulación en emulador Gingga-NCL Virtual Set Top Box. (a) Presentación del botón INFO durante un programa de televisión. (b) Presentación de la aplicación interactiva (menú principal).

Fuente: Creado por el autor mediante captura de pantalla del emulador Gingga-NCL Virtual Set Top Box.



(a)

(b)



(c)

(d)



(e)

Figura 4.15. Simulación en emulador Gingga-NCL Virtual Set Top Box. (a) Presentación del menú secundario ZOOLOGICO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo AVES. (c) Presentación de la galería de imágenes del botón verde MAMÍFEROS. (d) Presentación de la galería de imágenes del botón amarillo REPTILES. (e) Presentación de la galería de imágenes del botón azul NUEVA ESPECIE.

Fuente: Creado por el autor mediante captura de pantalla del emulador Gingga-NCL Virtual Set Top Box.



(a)

(b)



(c)

(d)



(e)

Figura 4.16. Simulación en emulador Ginga-NCL Virtual Set Top Box. (a) Presentación del menú secundario ORQUIDEARIO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo NAVE 1. (c) Presentación de la galería de imágenes del botón verde NAVE 2. (d) Presentación de la galería de imágenes del botón amarillo NAVE 3. (e) Presentación de la galería de imágenes del botón azul NAVE 4.

Fuente: Creado por el autor mediante captura de pantalla del emulador Ginga-NCL Virtual Set Top Box.



(a)

(b)



(c)

(d)



(e)

Figura 4.17. Simulación en emulador Ginga-NCL Virtual Set Top Box. (a) Presentación del menú secundario VIVERO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo ARBOLES. (c) Presentación de la galería de imágenes del botón verde FRUTALES. (d) Presentación de la galería de imágenes del botón amarillo ORNAMENTALES. (e) Presentación de la galería de imágenes del botón azul OTRAS ESPECIES.

Fuente: Creado por el autor mediante captura de pantalla del emulador Ginga-NCL Virtual Set Top Box.



Figura 4.18. Simulación en emulador Ginga-NCL Virtual Set Top Box. Presentación del menú secundario CÓMO LLEGAR con una opción de retorno.

Fuente: Creado por el autor mediante captura de pantalla del emulador Ginga-NCL Virtual Set Top Box.

Una vez realizada la simulación en el emulador Ginga-NCL Virtual Set Top Box se realizan las siguientes observaciones:

- La respuesta de presentación de las imágenes es menor a un segundo.
- La reproducción del archivo de audio nunca se detuvo durante la ejecución de la aplicación interactiva, éste audio se encuentra en formato mp4 ya que es un formato que admite el emulador Ginga-NCL Virtual Set Top Box.
- El archivo original de audio debe estar en formato aac ya que es el estandarizado para la televisión digital terrestre, pero éste emulador no lo admite.
- El audio de vídeo de fondo que sirve como simulador para un programa de televisión, no se detuvo.
- Todas las imágenes se presentaron en cada región designada según lo establecido en NCL Composer para el layout de pantalla, a excepción de la imagen del botón BACK.
- El control remoto funciona correctamente en la ejecución del programa. Aunque el botón BACK no se presente en pantalla la programación del mismo sigue funcionando.
- La señal digital no se puede utilizar en esta simulación es por tanto que se utiliza un vídeo en formato mp4 para simular el programa de televisión en ejecución normal.
- El reinicio de la aplicación interactiva luego de presionar el botón EXIT, funcionó correctamente.

4.6.3. Simulación en EITV Developer Box.

Esta simulación se la realiza utilizando un STB híbrido que sirve para la recepción de señales digitales con la norma ISDB-T e IPTV y es construido y distribuido por EITV en Brasil y por ADVICOM S.A. en Ecuador, éste decodificador está enfocado para desarrolladores de aplicaciones Ginga ya que no es necesario receptor señales digitales ya que el STB puede reproducir las aplicaciones interactivas sin ningún problema. Las características del EITV Developer Box son las siguientes:

- *Procesador de 450 MHz.*
- *Memoria RAM de 256 MB.*
- *Memoria Flash de 128 MB.*
- *Soporta norma digital ISDB-T y normas analógicas NTSC y PAL.*
- *Ancho de banda de 6 MHz.*
- *Interfaces de comunicación USB 2.0 y Ethernet 100Mbps.*
- *Middleware GINGA.*

Para este escenario de prueba es necesario cargar las aplicaciones en el STB, existe un aplicativo gráfico vía interfaz web que permite la instalación y configuración de los aplicativos ginga y canales IPTV. Las aplicaciones también se pueden embeber en el STB mediante la carga desde un pendrive, éste último método se utilizará para la visualización de la aplicación y es necesario que la pendrive este de preferencia solo con los archivos media y los documentos NCL utilizados.

El documento NCL principal debe tener como nombre 'main.ncl', los medias de imágenes en formato png y el archivo de audio en formato aac. El televisor utilizado para la conexión con el STB es un LG de 32”.

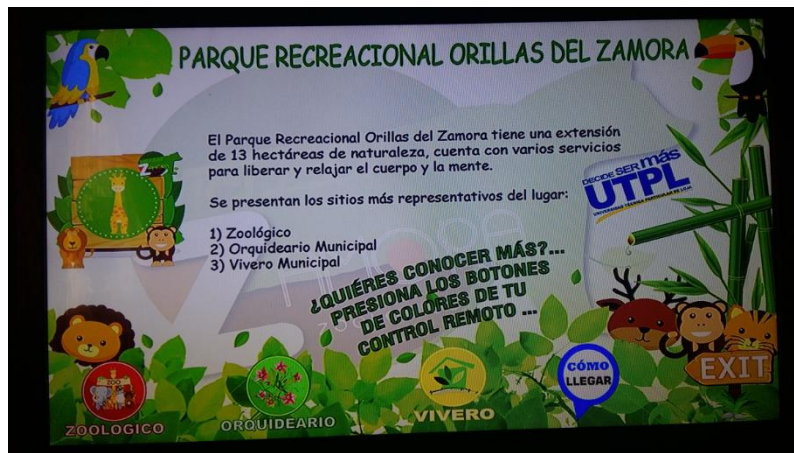
Los resultados obtenidos son los siguientes:



Figura 4.19. Conexión de EITV Developer Box con pendrive mediante USB y televisor LG mediante HDMI.
 Fuente: Creado por el autor mediante fotografías del ambiente de ejecución, STB conectado a LG.



(a)



(b)

Figura 4.20. Simulación en EITV Developer Box. (a) Presentación del botón INFO durante un programa de televisión (sin señal digital). (b) Presentación de la aplicación interactiva (menú principal).
 Fuente: Creado por el autor mediante fotografías del ambiente de ejecución, EITV Developer Box conectado a televisor LG.



(a)



(b)



(c)



(d)



(e)

Figura 4.21. Simulación en EITV Developer Box. (a) Presentación del menú secundario ZOOLOGICO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo AVES. (c) Presentación de la galería de imágenes del botón verde MAMÍFEROS. (d) Presentación de la galería de imágenes del botón amarillo REPTILES. (e) Presentación de la galería de imágenes del botón azul NUEVA ESPECIE.

Fuente: Creado por el autor mediante fotografías del ambiente de ejecución, EITV Developer Box conectado a televisor LG.



(a)



(b)



(c)



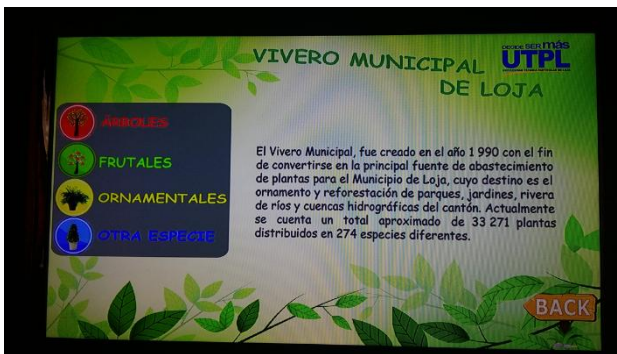
(d)



(e)

Figura 4.22. Simulación en EITV Developer Box. (a) Presentación del menú secundario ORQUIDEARIO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo NAVE 1. (c) Presentación de la galería de imágenes del botón verde NAVE 2. (d) Presentación de la galería de imágenes del botón amarillo NAVE 3. (e) Presentación de la galería de imágenes del botón azul NAVE 4.

Fuente: Creado por el autor mediante fotografías del ambiente de ejecución, EITV Developer Box conectado a televisor LG.



(a)



(b)



(c)



(d)



(e)

Figura 4.23. Simulación en EITV Developer Box. (a) Presentación del menú secundario VIVERO con cuatro opciones de acceso y uno de retorno. (b) Presentación de la galería de imágenes del botón rojo ARBOLES. (c) Presentación de la galería de imágenes del botón verde FRUTALES. (d) Presentación de la galería de imágenes del botón amarillo ORNAMENTALES. (e) Presentación de la galería de imágenes del botón azul OTRAS ESPECIES.

Fuente: Creado por el autor mediante fotografías del ambiente de ejecución, EITV Developer Box conectado a televisor LG.



Figura 4.24. Simulación en EITV Developer Box. Presentación del menú secundario CÓMO LLEGAR con una opción de retorno.

Fuente: Creado por el autor mediante fotografías del ambiente de ejecución, EITV Developer Box conectado a televisor LG.

Una vez realizada la simulación en el STB EITV Developer Box se puede realizar las siguientes observaciones:

- La respuesta de presentación de las imágenes es menor a tres segundos.
- El archivo de audio nunca se reprodujo durante la ejecución de la aplicación interactiva pese a que se utilizó diferentes formatos de audio (aac, mp3, m4a, wav, wma, mp4, entre otros).
- El decodificador pudo reconocer y reproducir individualmente el archivo de audio que se encuentra en formato AAC, pero cuando se ejecuta la aplicación interactiva conjuntamente con el audio especificado éste último no se reproduce.
- Todas las imágenes se presentaron en cada región designada según lo establecido en NCL Composer para el layout de pantalla.
- El control remoto funciona correctamente en la ejecución del programa
- A diferencia de las simulaciones anteriores que utilizaron un vídeo en formato mp4 para que sirva como fondo de un programa de televisión en ejecución normal, en este ambiente de simulación no es posible utilizarlo ya que se debería utilizar una señal digital de televisión pero como en Loja no existe aún señales digitales, solo se especificará la línea de código que se debe utilizar para poder receptar estas señales, "[sbtvd-ts://](#)".

- *El reinicio de la aplicación interactiva no funcionó pese a que se utilizó el mismo código, se puede pensar que el motivo por el cual sucedió esto es por la capacidad del STB utilizado para las pruebas.*
- *Como se mencionó en el punto 4.4 al realizar la simulación de prueba, la respuesta de presentación de las imágenes en algunas ocasiones demoró dos segundos en presentarse y en raras ocasiones (2 de cada 10) no se presentaban. Por lo tanto se puede concluir que la RAM de este decodificador no la indicada y por ende debe ser una RAM mayor a 256 MB para poder ejecutar esta aplicación interactiva.*

CAPÍTULO V

5. CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

- *La televisión digital presenta una mejora al uso eficiente del espectro radioeléctrico frente a la televisión analógica como por ejemplo la múltiple canalización que permite enviar hasta cuatro canales en SD o uno en HD o una combinación de canales en SD y HD en el mismo ancho de banda de 6 MHz sin perder la calidad del audio y vídeo, además se incluyen datos que realizan la interacción con el usuario y siempre está disponible el envío de canales para terminales móviles mediante el canal One Seg que forma parte del estándar ISDB-Tb.*
- *La resolución de vídeo de la televisión digital pasará de 720 x 480 pixels (HD) a 1920 x 1080 pixels (full HD) esto gracias a su formato de compresión AVC (H.264) y la mejora de la calidad de audio pasará de MPEG-2 AAC a MPEG-4 HE-AAC para disminuir los efectos de ruido.*
- *El estándar ISDB-Tb incluye el Sistema de Alerta Temprana que es de gran ayuda en momentos de catástrofes naturales, ya que permite que las personas, comunidades y organizaciones amenazadas por catástrofes naturales se informen, preparen y actúen de forma apropiada con un tiempo de anticipación y así hacer posible la minimización de pérdidas humanas.*
- *Para el diseño de aplicaciones interactivas hay que tener en cuenta la aplicabilidad, funcionalidad, facilidad de navegación y estética de los mismos ya que estos caracteres hacen que un programa de televisión tenga éxito.*
- *La importancia del lenguaje de programación NCL frente a los otros lenguajes de programación radica en la flexibilidad del lenguaje, fácil de entender, soporte multi-dispositivo, capacidad para definir una sincronización espacio-temporal entre los archivos media, incluidas las interacciones del espectador, es excelente para todo tipo de sistemas de televisión digital.*
- *La televisión digital crea un ambiente de inclusión social, las televisoras ofrecerán programas educativos que podrán ser de gran ayuda para personas que no puedan acceder a la educación, ya que por medio de su televisor o teléfono celular tendrán acceso a contenidos interactivos.*
- *El middleware Ginga es un software libre de gran importancia tecnológica en nuestro país, ya que al ser una tecnología abierta nos brinda la opción de compartir ideas entre desarrolladores, instituciones científicas, privadas o universidades con el fin de realizar*

trabajos de investigación y así aportar mejoras y corregir errores para el beneficio de todos los involucrados.

- *En el presente proyecto toda la programación de la aplicación interactiva se encuentra en lenguaje NCL debido a que los simuladores no eran capaces de reproducir diversos archivos LUA a la vez, además el lenguaje de programación LUA se encuentra en proceso de aprendizaje para la televisión digital.*
- *El ambiente de programación LUA hace que las aplicaciones interactivas sean más ligeras, existe la disminución de líneas de código, se realiza un depurado total del código al realizar llamados de imágenes, texto, audios o vídeos de una forma más eficiente y fácil de realizar para el compilador de NCL y LUA.*
- *Las imágenes que se utilicen dentro de la aplicación para el layout de pantalla deben ser lo más ligero posible sin perder la calidad de las imágenes.*
- *Debido a la falta de cobertura digital de televisión en Loja, la aplicación interactiva no se transmitió al aire para realizar pruebas, sin embargo este proyecto de titulación queda a disposición de las futuras investigaciones que se realicen en este tema y a la posible transmisión de la aplicación interactiva de este proyecto cuando la televisión digital terrestre se implemente en nuestra ciudad.*

RECOMENDACIONES

- *La aplicación interactiva dio como resultado 2419 líneas de código en lenguaje NCL, esto se debe a que se realizó el llamado a cinco imágenes por cada botón de cada menú lo cual ocasionaba el incremento significativo de líneas de código. Se recomienda utilizar LUA, ya que con esta herramienta de programación hace posible realizar el llamado a un número mayor de imágenes por cada opción con la utilización de menos líneas de código ya que utiliza un método más sencillo para realizar este proceso.*
- *Para que la aplicación interactiva sea más eficiente en su reproducción, se deben utilizar imágenes ligeras, para fondos se utilizar una resolución de 1500 x 1000 pixeles, para botones 300 x 300 pixels y para imágenes de las diferentes especies 500 x 300 pixeles.*
- *Para desarrollar una aplicación interactiva es recomendable crear un grupo inter disciplinario entre ingenieros, comunicadores sociales y diseñadores gráficos para trabajar en equipo. El diseñador gráfico creará las imágenes de botones, textos, formas, entre otros, el profesional en comunicación social ayudará a introducir lo más destacable del tema de la aplicación para que ésta tenga éxito frente a los televidentes y el diseñador de la aplicación interactiva se encarga de enlazar todos los contenidos y generar la App que será transmitida.*
- *Es importante simular las aplicaciones interactivas para tener una visión de la aplicación interactiva final, pero para garantizar el funcionamiento de la aplicación interactiva se debe simular ésta en un decodificador real que tenga los parámetros mínimos de funcionamiento, en especial la memoria RAM que sea de mayor capacidad que la mencionada en el punto 4.4. Esto con el fin de que exista una buena ejecución de las aplicaciones en cuanto respecta a imágenes de gran tamaño, reproducción de audios y vídeos adicionales, scripts LUA, entre otros.*

BIBLIOGRAFÍA

- [1] Montero, M., (2012). *Análisis sobre la producción audiovisual de programas infantiles en la Ciudad de Quito entre los años 2010 – 2011*. Universidad Internacional del Ecuador. [online]. Quito, p. 37, 38. Disponible en: <http://repositorio.uide.edu.ec/bitstream/37000/479/1/T-UIDE-0445.pdf>. [Consulta 16 oct. 2015].
- [2] Córdoba, R., (n.d.). *Historia de la televisión en el Ecuador y en la ciudad de Loja*. Universidad Técnica Particular de Loja. [online]. Loja, p. 2. Disponible en: http://www.utpl.edu.ec/jorgeluisjaramillo/wp-content/uploads/2010/06/roberto_guerrero-historia-de-la-TV-en-Ecuador-y-en-Loja.pdf. [Consulta 16 oct. 2015].
- [3] Urrego, B., (2013). *El apagón analógico y las oportunidades de incorporación de la televisión digital terrestre en el Ecuador*. Universidad Técnica Particular de Loja. [online]. Loja, pp. 6, 19. Disponible en: <http://dspace.utpl.edu.ec/bitstream/123456789/8243/1/Urrego%20Castillo,%20Byron-%20Tesis.pdf>. [Consulta 16 oct. 2015].
- [4] Murga, P., (2009). *Historia de la Televisión Ecuatoriana*. Sitio Web Medios de Comunicación. [online]. Disponible video en: <http://mediosdecomunicacionayh.blogspot.com/2009/11/historia-de-la-television-ecuatoriana.html>. [Consulta 16 oct. 2015].
- [5] Poma, F., (2008). *La Historia de la Televisión en Ecuador*. Sitio Web Televisión Satelital News. [online]. Disponible en: <http://megadigitalnews24.blogspot.com/2008/11/la-historia-de-la-television-en-ecuador.html>. [Consulta 16 oct. 2015].
- [6] Cevallos, A., (2014). *Estudio del estándar adoptado para la televisión digital terrestre en el Ecuador*. Universidad Católica de Santiago de Guayaquil. [online]. Guayaquil, p. 11, 21. Disponible en: <http://repositorio.ucsg.edu.ec/bitstream/3317/1731/1/T-UCSG-PRE-TEC-ITEL-41.pdf>. [Consulta 17 oct. 2015].
- [7] Valencia, N., & Constante, R., (2012). *Transmisión de señales de TV digital para dispositivos móviles*. Escuela Superior Politécnica del Litoral. [online]. Guayaquil, p. 4, 29, 41, 42. Disponible en: http://www.cib.espol.edu.ec/Digipath/D_Tesis_PDF/D-94219.pdf. [Consulta 27 oct. 2015].
- [8] *Televisión por cable*. Sitio Web Televisión Digital. [online]. Disponible en: <http://www.televisiondigital.gob.es/TelevisionDigital/formas-acceso/Paginas/tv-cable.aspx>. [Consulta 27 oct. 2015].

- [9] Dávila, M., (2012). *Diseño de una plataforma de software para televisión digital interactiva de un canal de deportes utilizando Gingga-NCL-Lua*. Universidad Politécnica Salesiana. [online]. Cuenca, pp. 8, 9, 25. Disponible en: <http://dspace.ups.edu.ec/bitstream/123456789/1736/13/UPS-CT02324.pdf>. [Consulta 27 oct. 2015].
- [10] Paredes, A. & Tonguino, N., (2012). *Análisis de desempeño del canal de retorno basado en el desarrollo y transmisión de aplicaciones interactivas de TV digital para el Sistema ISDB-Tb*. Escuela Politécnica del Ejército. [online]. Sangolquí, pp. 5, 6, 7, 9 14, 27, 28, 32. Disponible en: <http://repositorio.espe.edu.ec/bitstream/21000/5448/1/T-ESPE-033354.pdf>. [Consulta 12 nov. 2015].
- [11] Calero, A. & Villacrés, C., (2009). *Análisis y estudio de ingeniería para la selección del estándar de televisión digital más apropiado para Ecuador bajo la supervisión de la "SUPERTEL"*. Escuela Superior Politécnica de Chimborazo. [online]. Riobamba, pp. 122, 164. Disponible en: <http://dspace.espech.edu.ec/bitstream/123456789/179/1/38T00167.pdf>. [Consulta 15 nov. 2015].
- [12] Zaidán, M., (2010). *Análisis del dividendo digital resultante de la migración de la televisión analógica a digital en el Ecuador*. Escuela Politécnica Nacional. [online]. Quito, p. 35. Disponible en: <http://bibdigital.epn.edu.ec/bitstream/15000/2463/1/CD-3168.pdf>. [Consulta 15 nov. 2015].
- [13] Saca, A., (2013). *Análisis de cobertura del sistema de televisión digital terrestre para el canal 47 en la ciudad de Quito*. Universidad Politécnica Salesiana sede Quito. [online]. Quito, pp. 7, 8, 11, 12, 13, 14. Disponible en: <http://dspace.ups.edu.ec/bitstream/123456789/5783/1/UPS-ST001062.pdf>. [Consulta 27 nov. 2015].
- [14] Moreno, B., & Salazar, J., (2011). *Estudio y análisis de factibilidad para la implementación de la televisión digital terrestre en el Ecuador*. Universidad Politécnica Salesiana sede Cuenca. [online]. Cuenca, pp. 8, 40. Disponible en: <http://dspace.ups.edu.ec/bitstream/123456789/1435/13/UPS-CT002165.pdf>. [Consulta 17 dic. 2015].
- [15] Asociación Brasileña de Normas Técnicas ABNT, (2007). *Televisión digital terrestre – Sistema de transmisión ISDB-Tb*. [online]. Rio de Janeiro, pp. 4, 5. Disponible en: <http://docplayer.es/2497852-Abnt-nbr-15601-norma-brasilena-television-digital-terrestre-sistema-de-transmision-isdb-tb.html>. [Consulta 17 dic. 2015].

- [16] Morales, A., (2010). *Diseño de la red para interactividad en televisión digital terrestre e IPTV en el campus ESPE Sangolquí. Escuela Politécnica del Ejército. [online]. Sangolquí, p. 78, 98, 139, 140. Disponible en: <http://repositorio.espe.edu.ec/bitstream/21000/2645/1/T-ESPE-029824.pdf>. [Consulta 17 dic. 2015].*
- [17] Valverde, C., (2014). *Planificación de un programa de tv con la interacción de TDT aplicada a un prototipo de paquetes turísticos. Universidad del Azuay. [online]. Cuenca, p. 35. Disponible en: <http://dspace.uazuay.edu.ec/bitstream/datos/3707/1/10376.pdf>. [Consulta 02 ene. 2016].*
- [18] Alulema, D., (2012). *La Televisión Digital Terrestre en el Ecuador es interactiva. Universidad Tecnológica Equinoccial. [online]. Quito. Disponible en: <https://www.ute.edu.ec/posgrados/eidos5/art-2.html>. [Consulta 02 ene. 2016].*
- [19] Sotomayor, P., (n.d.). *Análisis de los estándares de televisión digital terrestre (TDT) y pruebas de campo utilizando los equipos de comprobación técnica de la Superintendencia de Telecomunicaciones. Escuela Politécnica Nacional. [online]. Quito, p. 8. Disponible en: <http://bibdigital.epn.edu.ec/bitstream/15000/1159/1/CD-2628.pdf>. [Consulta 02 ene. 2016].*
- [20] Informe CITDT-GATR-2012-004, (2012). *Plan Maestro de Transición a la Televisión Digital Terrestre en el Ecuador. [online]. MINTEL. Disponible en: <http://www.telecomunicaciones.gob.ec/wp-content/uploads/downloads/2013/02/Informe-CITDT-GATR-2012-004.pdf>. [Consulta 02 ene. 2016].*
- [21] *Ley Orgánica de Telecomunicaciones, (2015). [online]. Quito, p. 8. Disponible en: <http://www.arcotel.gob.ec/wp-content/uploads/downloads/2016/01/ley-organica-de-telecomunicaciones.pdf>. [Consulta 12 ene. 2016].*
- [22] González, A. & Jiménez, K., (2006). *La televisión digital interactiva y sus aplicaciones educativas. Revista Científica de Comunicación y Educación, Comunicar. [online]. Disponible en: <http://www.revistacomunicar.com/verpdf.php?numero=26&articulo=26-2006-15>. [Consulta 12 ene. 2016].*
- [23] Campos, P., (2010). *Estudio del estándar de televisión digital terrestre DTMB (Digital Terrestrial Multimedia Broadcasting), y propuesta de reglamento para la prestación del servicio de televisión digital terrestre en el Ecuador. Escuela Politécnica Nacional [online] Quito. Disponible en: <http://bibdigital.epn.edu.ec/bitstream/15000/2129/1/CD-2889.pdf>. [Consulta 04 feb. 2016].*

- [24] Villamarín, D., (2014). *Estudio comparativo y de integración para las plataformas de televisión interactiva europea HBBTV y latinoamericana Ginga*. Universidad Politécnica de Madrid. [online]. Madrid, pp. 44, 45, 51, 52. Disponible en: <http://repositorio.educacionsuperior.gob.ec/bitstream/28000/1482/1/T-SENESCYT-00616.pdf>. [Consulta 10 feb. 2016].
- [25] Zambrano, A., (n.d.). *Introducción a la TV Digital Interactiva y Ginga.ar*. Universidad Nacional de La Plata. [online]. La Plata, p. 39. Disponible en: <http://tvd.lifia.info.unlp.edu.ar/ginga.ar/images/diapositivas/introa-tvd.pdf>. [Consulta 22 feb. 2016].
- [26] Zambrano, A., (n.d.). *Introducción a la Televisión Digital Interactiva y desarrollo de software embebido para TVD*. Universidad Nacional de La Plata. [online]. La Plata, p. 11, 13. Disponible en: https://pleiad.cl/_media/research/adi/tvd-desarrollo.pdf. [Consulta 22 feb. 2016].
- [27] Kubert, T., (2009). *User-Centered Interaction Design Patterns for Interactive Digital Television Applications*. [online]. London, p. 108, 113. Disponible en: https://books.google.com.ec/books?id=3IMbP0T43bEC&pg=PA99&lpg=PA99&dq=User-Centered+Interaction+Design+Patterns+for+Interactive+Digital+Television+Applications&source=bl&ots=2FjQWykdRy&sig=ji9JqUeYm5DZX4-b8QJnsq7KxAE&hl=es&sa=X&ei=oAiDU_iQC-nisASthYKoAw#v=onepage&q=User-Centered%20Interaction%20Design%20Patterns%20for%20Interactive%20Digital%20Television%20Applications&f=false. [Consulta 26 feb. 2016].
- [28] Canchi, S. & González, M., (2013). *Arquitectura de Software para aplicaciones de publicidad interactivas en ISDB-Tb, usando Ginga-NCL y servicios web*. Universidad Nacional de Córdoba. [online]. Córdoba, p. 22. Disponible en: http://www.famaf.unc.edu.ar/wp-content/uploads/2014/04/9-Canchi_Gonzalez.pdf. [Consulta 22 feb. 2016].
- [29] Carrillo, C., (2014). *Desarrollo de una aplicación interactiva para TDT basado en middleware ginga en el instituto de nivelación IPREX*. Escuela Superior Politécnica de Chimborazo. [online]. Riobamba, p. 48. Disponible en: <http://dspace.espace.edu.ec/bitstream/123456789/3564/1/98T00055.pdf>. [Consulta 02 mar. 2016].
- [30] Quingaluisa, A., Torres, J., Martínez, D. & Salvador, S., (n.d.). *Estudio e Investigación del Middleware Ginga-J del estándar Brasileño de televisión digital. Caso Práctico: Desarrollo de una aplicación interactiva aplicando la metodología OpenUP/Basic como parte del*

- Proyecto ESPE-GINGA. Escuela Politécnica del Ejército. [online]. Quito, p. 3. Disponible en: <http://repositorio.espe.edu.ec/bitstream/21000/4748/2/T-ESPE-032865-A.pdf>. [Consulta 02 mar 2016].*
- [31] *Valencia, J., (2013). Diseño y desarrollo de aplicaciones interactivas para el middleware Ginga de televisión digital de la norma ISDB-Tb para brindar información de los protocolos de prevención a la población en lugares de alto riesgo de erupciones volcánicas, sismos y tsunamis. Escuela Politécnica Nacional. [online]. Quito. Disponible en: <http://bibdigital.epn.edu.ec/bitstream/15000/6218/1/CD-4840.pdf>. [Consulta 16 mar 2016].*
- [32] *Muñoz, T. & Sigüenza, L., (2012). TV Digital fija utilizando middleware Ginga-NCL aplicado a un noticiero digital. Universidad de Cuenca. [online]. Cuenca. Disponible en: <http://dspace.ucuenca.edu.ec/bitstream/123456789/645/1/ts199.pdf>. [Consulta 16 mar 2016].*
- [33] *Associação Brasileira de Normas Técnicas ABNT, (2008). Televisão digital terrestre – Canal de interatividade Parte 1: Protocolos, interfaces físicas e interfaces de software. [online]. Río de Janeiro, p. 18. Disponible en: http://gingacdn.lavid.ufpb.br/attachments/305/ABNTNBR15607_2D1_2008Ed1.pdf. [Consulta 01 abr. 2016].*
- [34] *Paucar, R., (2010). Análisis y modelamiento de las técnicas de canal de retorno e interactividad para el estándar de Televisión Digital Terrestre ISDB-T. Universidad Nacional Mayor de San Marcos. [online]. Lima. Disponible en: http://cybertesis.unmsm.edu.pe/bitstream/cybertesis/2338/1/Paucar_cr.pdf. [Consulta 02 abr. 2016].*
- [35] *Paucar, R., Velásquez, C., Mendoza, E. & Díaz, D., (2009). Análisis del Canal de Retorno para la Televisión Digital Interactiva utilizando la Clase TCP-Lua. Universidad Nacional de Ingeniería INICTEL-UNI. [online]. Lima. Disponible en: <http://www.revista.unisal.br/sj/index.php/123/article/viewFile/146/147>. [Consulta 02 abr. 2016].*
- [36] *Associação Brasileira de Normas Técnicas ABNT, (2008). Televisão digital terrestre – Receptores. [online]. Río de Janeiro, p. 3, 22. Disponible en: http://gingacdn.lavid.ufpb.br/attachments/292/ABNTNBR15604_2007Vc_2008.pdf. [Consulta 04 abr. 2016].*
- [37] *Set Top Box Descodificador para televisión. Sitio web Wikitel. [online]. Disponible en: http://wikitel.info/wiki/Set_Top_Box. [Consulta 05 may. 2016].*

- [38] *Entretenimiento e Interactividad para la Televisión Digital – EITV*, (n.d.). *EITV DeveloperBox Set-top Box para Desarrollo en el Ginga, Manual de usuario*. [online]. Disponible en: http://profesores.elo.utfsm.cl/~agv/TVD/manuales/EITV%20Developer%20Box%20-%20v1_1_SPA_FINAL.pdf. [Consulta 05 may. 2016].
- [39] COLABLE, (n.d.). *COL601M ISDB-T Set Top Box*. [online]. Disponible en: <http://f02.s.alicdn.com/kf/HTB1sXwHHXXXXXaUXVXX.PRXFXXp.pdf>. [Consulta 05 may. 2016].
- [40] Gomes, L. & Ferreira, R., (n.d.). *Nested Context Model 3.0 Part 1 – NCM Core*. Pontificia Universidade Católica do Rio de Janeiro. [online]. Río de Janeiro. Disponible en: ftp://ftp.inf.puc-rio.br/pub/docs/techreports/05_18_soares.pdf. [Consulta 15 may. 2016].
- [41] Gomes, L. & Ferreira, R., (n.d.). *Nested Context Lenguaje 3.0 Part 8 – NCL Digital TV Profiles*. Pontificia Universidade Católica do Rio de Janeiro. [online]. Río de Janeiro. Disponible en: <http://www.ncl.org.br/documentos/NCL3.0-DTV.pdf>. [Consulta 05 may. 2016].
- [42] Bertullo, S., Bolioli, F., Carriquiry, G., & Gutierrez, D., (2006). *Televisión Digital Estándar Americano (ATSC)*. Universidad de la República. [online]. Montevideo. Disponible en: <http://www.ccee.edu.uy/ensenian/catcomp/material/TELEVISION%20DIGITAL%20STD%20Amer.pdf>. [Consulta 20 may. 2015].
- [43] Rodrigo, F., (2009). *DVB, Digital Video Broadcasting*. [online]. España. Disponible en: <http://www.coyan.es/Blog/?s=DVB>. [Consulta 20 may. 2015].
- [44] DiBEG Digital Broadcasting Experts Group, (n.d.). *Transmisión de Televisión Digital Terrestre ISDB-T*. [online]. Disponible en: <http://www.ar.emb-japan.go.jp/Notas/090901TelevisionDigital.pdf>. [Consulta 20 may. 2015].
- [45] Pisciotta, N., (2010). *Sistema ISDB-Tb (primera parte)*. Universidad Blas Pascal. [online]. Córdoba. Disponible en: http://www2.elo.utfsm.cl/~elo341/SistemaISDB_Tb.pdf. [Consulta 20 may. 2015].
- [46] H.761: *Lenguaje de contexto anidado (NCL) y Ginga-NCL*, (2015). Sitio web Unión Internacional de Telecomunicaciones – ITU. [online]. Disponible en: <https://www.itu.int/rec/T-REC-H.761-201411-l/es>. [Consulta 03 ene. 2016].
- [47] Ojeda, R., (n.d.). *Set-Top Box Microprocesadores para Comunicaciones*. [online]. Disponible en: <http://www.iuma.ulpgc.es/~nunez/clases-micros-para-com/mpc0809-trabajos/mpc0809RuymanOjedaSTBs.pdf>. [Consulta 05 jun. 2016].

- [48] EITV Developer Box, (n.d.). Sitio Web Entretenimiento e Interactividad para la Televisión Digital – EITV. [online]. Disponible en: <http://www.eitv.com.br/es/produtos/eitv-developer-box/>. [Consulta 05 jun. 2016].
- [49] ISDB-Tb Set Top Box, (2015). Sitio Web COLABLE ELECTRONICS. [online]. Disponible en: <http://colable.com/Product/9254182253.html>. [Consulta 05 jun. 2016].
- [50] IHD, AVC, Set-Top Box For Satellite/Terrestrial Operators, (n.d.). Sitio Web ARRIS. [online]. Disponible en: <http://www.arris.com/globalassets/resources/data-sheets/zh4105.pdf>. [Consulta 05 jun. 2016].
- [51] Model DXD7025 Dual Tuner Hybrid HD STB with CI+, (2013). Sitio Web EKT Innovable Digital Media Solutions. [online]. Disponible en: <http://ekt-pim.s3.amazonaws.com/specsheets/DXD7025%20Dual%20Multi-format%20Tuner%20HYBRID%20HD%20STB.pdf>. [Consulta 05 jun. 2016].
- [52] Set Top Box OTT ISDB-T inteligente de doble núcleo DV7204-IS, (n.d.). Sitio Web Shenshen SDMC Tecnología S.A. [online]. Disponible en: <http://es.sdmctech.com/product/Receptor-multimedia-digital-OTT-ISDB-T-inteligente-de-doble-n%C3%BAcleo-DV7204-IS.html>. [Consulta 18 jun. 2016].
- [53] Ayala, M., (2011). Diseño de aplicaciones interactivas T-Government, T-Health y T-Learning para su aplicación en el sistema de televisión digital terrestre del Ecuador (SBTVD), para la empresa televisión del Pacífico S.A. GAMATV. Universidad Politécnica Salesiana. Quito, p. 114. Disponible en: <http://dspace.ups.edu.ec/bitstream/123456789/4095/1/UPS-ST000088.pdf>. [Consulta 18 jun. 2016].
- [54] Cabezas, G. & Quezada, M., (2012). Diseño e implementación de un prototipo para un sistema de generación de aplicaciones interactivas con Ginga-NCL para la evaluación de servicios masivos. Escuela Politécnica Nacional. [online]. Quito. Disponible en: <http://bibdigital.epn.edu.ec/bitstream/15000/5167/1/CD-4559.pdf>. [Consulta 18 jun. 2016].
- [55] Guisández, R., (2003). Una plataforma para el desarrollo de aplicaciones de TVi Social. Universidad Nacional de La Plata. [online]. Buenos Aires. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/47043/Documento_completo__.pdf?sequence=1. [Consulta 18 jun. 2016].

ANEXOS

ANEXO A

***ALGORITMO DE LA APLICACIÓN INTERACTIVA
DESARROLLADO EN PSEINT***

En este anexo se presenta el código desarrollado en PSeInt, esto se desarrolló con el fin de presentar un diagrama de flujo que describa la aplicación interactiva. Se especifica el estado inicial, las opciones disponibles, el contenido del menú de imágenes y como finaliza la aplicación.

Proceso Parque_Recreacional_Orillas_del_Zamora

```
Definir info,sel,opzoo,oporq,opviv,loop,loop1 Como Real;

Escribir ('REPRODUCCIÓN NORMAL DE TV.....');

Escribir ('Selecciona info (1) para ingresar a la aplicación');

Leer info;

Escribir ('***PARQUE RECREACIONAL ORILLAS DEL ZAMORA***');

loop <- 1;

Mientras loop=1 Hacer

    Borrar Pantalla;

    Escribir ('SELECCIONE LA OPCIÓN QUE DESEA OBSERVAR: ');

    Escribir ('1) Zoológico');

    Escribir ('2) Orquideario');

    Escribir ('3) Vivero');

    Escribir ('4) ¿Cómo llegar?');

    Escribir ('5) Regresar a reproducción de TV');

    Leer sel;

    Segun sel Hacer

        1:

            loop1 <- 1;

            Mientras loop1=1 Hacer

                Borrar Pantalla;

                Escribir ('***ZOOLOGICO ORILLAS DEL ZAMORA***');

                Escribir ('Seleccionar opción:');
```


Escribir ('1 Aves');

Escribir ('2 Mamíferos');

Escribir ('3 Reptiles');

Escribir ('4 Nueva Especie');

Escribir ('5 Regresar al menú principal');

Leer opzoo;

Borrar Pantalla;

Si opzoo=1 Entonces

Escribir ('** AVES DEL ZOOLOGICO MUNICIPAL**');

Escribir ('- Amazona festiva');

Escribir ('- Guacamayo azuliamarillo');

Escribir ('- Guacamayo escarlata');

Escribir ('- Loro macareño');

Escribir ('- Pavo real');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opzoo=2 Entonces

Escribir ('** MAMIFEROS DEL ZOOLOGICO MUNICIPAL**');

Escribir ('- Ardilla sabanera');

Escribir ('- Cabeza de mate');

Escribir ('- León africano');

Escribir ('- Mono capuchino blanco');

Escribir ('- Oso de anteojos');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opzoo=3 Entonces

Escribir (** REPTILES DEL ZOOLOGICO MUNICIPAL**);

Escribir ('- Boa constrictor');

Escribir ('- Boa mataballo');

Escribir ('- Tortuga galápagos');

Escribir ('- Tortuga mordedora');

Escribir ('- Tortuga motelo');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opzoo=4 Entonces

MUNICIPAL**);

Escribir (** NUEVO ESPÉCIMEN DEL ZOOLOGICO

Escribir ('- Coati amazónico');

Escribir ('- Colambo de cola negra');

Escribir ('- Espatula rosada');

Escribir ('- Tucán piquiacanalado');

Escribir ('- Tucán swainson');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opzoo=5 Entonces

loop1 <- 0;

FinSi

FinMientras

2:

loop1 <- 1;

Mientras loop1=1 Hacer

Borrar Pantalla;

Escribir ('***ORQUIDEARIO MUNICIPAL***');

Escribir ('Seleccionar opción:');

Escribir ('1) Nave 1');

Escribir ('2) Nave 2');

Escribir ('3) Nave 3');

Escribir ('4) Nave 4');

Escribir ('5) Regresar al menú principal');

Leer oporq;

Borrar Pantalla;

Si oporq=1 Entonces

Escribir ('** ORQUIDEARIO NAVE 1 **');

Escribir ('- Angulo virginalis');

Escribir ('- Brassia');

Escribir ('- Cattleya alba');

Escribir ('- Cymbidium');

Escribir ('- Drácula');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si oporq=2 Entonces

Escribir (** ORQUIDEARIO NAVE 2 **);

Escribir ('- Epidendrum');

Escribir ('- Helcia sanguinolenta');

Escribir ('- Huntleya');

Escribir ('- Ida');

Escribir ('- Maxillaria luteoalba');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si oporq=3 Entonces

Escribir (** ORQUIDEARIO NAVE 3 **);

Escribir ('- Maxillaria splendens');

Escribir ('- Odontoglossum');

Escribir ('- Oncidium excavatum');

Escribir ('- Phragmipedium caudatum');

Escribir ('- Pleurothallis');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si oporq=4 Entonces

Escribir (** ORQUIDEARIO NAVE 4 **);

Escribir ('- Rodriguezia secunda');

Escribir ('- Sobralia fimbriata');

Escribir ('- Stanhopea connata');

Escribir ('- Stelis');

```
    Escribir ('- Zootrophion');  
    Escribir ('Presione ENTER para regresar');  
    Esperar Tecla;
```

```
FinSi
```

```
Si oporq=5 Entonces
```

```
    loop1 0;
```

```
FinSi
```

```
FinMientras
```

3:

```
loop1 <- 1;
```

```
Mientras loop1=1 Hacer
```

```
    Borrar Pantalla;
```

```
    Escribir ('***VIVERO MUNICIPAL***');
```

```
    Escribir ('Seleccionar opción:');
```

```
    Escribir ('1) Árboles');
```

```
    Escribir ('2) Frutales');
```

```
    Escribir ('3) Ornamentales');
```

```
    Escribir ('4) Otra Especie');
```

```
    Escribir ('5) Regresar al menú principal');
```

```
    Leer opviv;
```

```
    Borrar Pantalla;
```

```
    Si opviv=1 Entonces
```

```
        Escribir ('** ÁRBOLES **');
```

```
        Escribir ('- Bugarvilla');
```

```
        Escribir ('- Calistemo');
```

Escribir ('- Ceibo');

Escribir ('- Farol chino');

Escribir ('- Tulipán africano');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opviv=2 Entonces

Escribir ('** FRUTALES **');

Escribir ('- Capulí');

Escribir ('- Guabo');

Escribir ('- Naranja agría');

Escribir ('- Níspero');

Escribir ('- Uva');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opviv=3 Entonces

Escribir ('** ORNAMENTALES **');

Escribir ('- Achira ornamental');

Escribir ('- Amarilis');

Escribir ('- Dalia');

Escribir ('- Lágrima de María');

Escribir ('- Lirio');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opviv=4 Entonces

Escribir (** OTRA ESPECIE **);

Escribir ('- Boxus');

Escribir ('- Cactus');

Escribir ('- Costilla de Adán');

Escribir ('- Hiedra');

Escribir ('- Siempreviva');

Escribir ('Presione ENTER para regresar');

Esperar Tecla;

FinSi

Si opviv=5 Entonces

loop1 <- 0;

FinSi

FinMientras

4:

Borrar Pantalla;

Escribir (**MAPA DEL LUGAR**);

Escribir ('Se visualizará un mapa del Parque Recreacional Orillas del Zamora');

Escribir ('--PRESIONE ENTER PARA REGRESAR AL MENU PRINCIPAL--');

Esperar Tecla;

5:

Escribir ('FINALIZANDO APLICACIÓN...');

Esperar 1 segundos;

Borrar Pantalla;

```
Escribir ('REPRODUCCIÓN NORMAL DE TV.....');
```

```
Esperar 2 segundos;
```

```
loop <- 0;
```

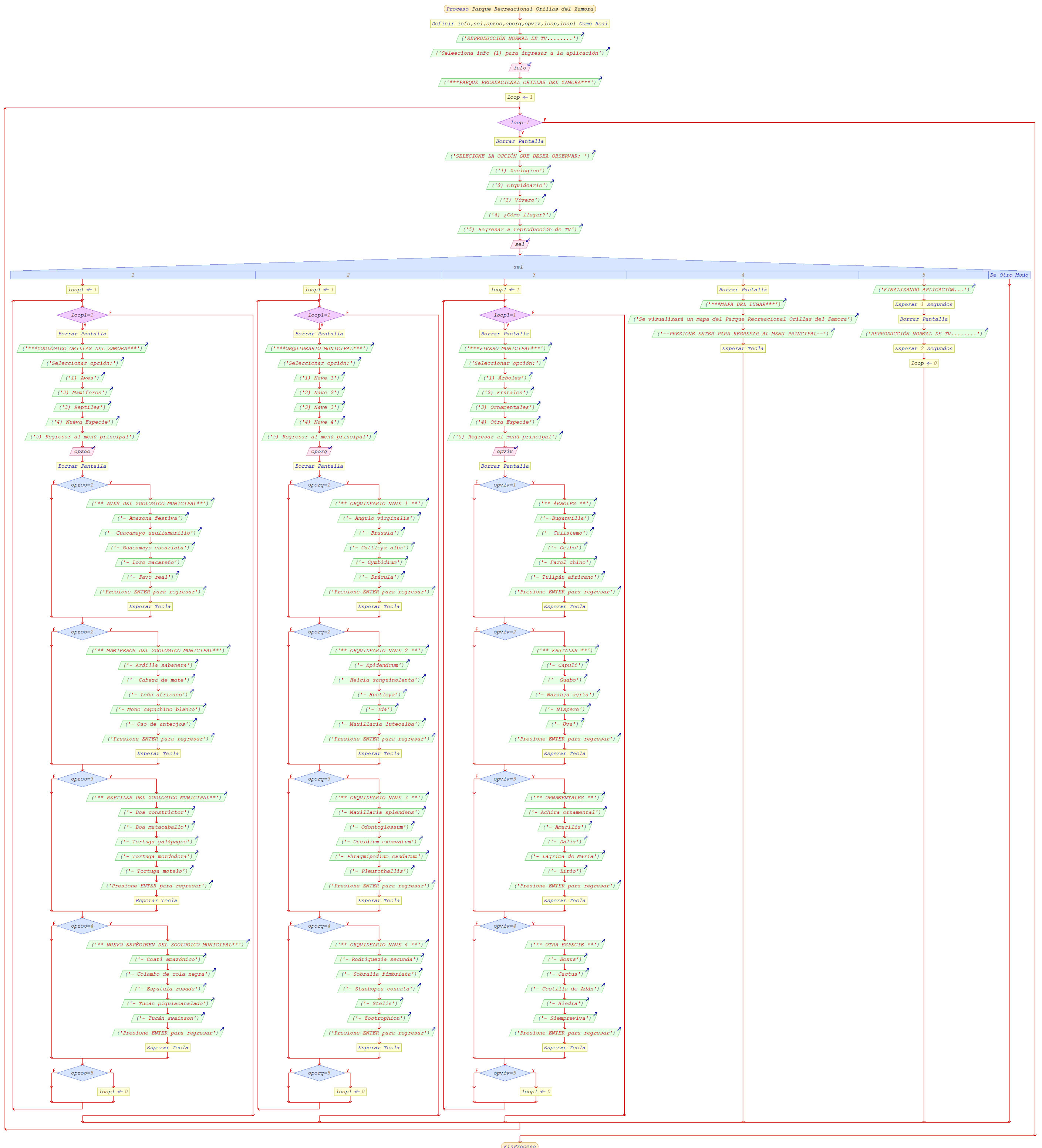
```
FinSegun
```

```
FinMientras
```

```
FinProceso
```


DIAGRAMA DE FLUJO DE LA APLICACIÓN INTERACTIVA PARA TELEVISIÓN DIGITAL TERRESTRE

“PARQUE RECREACIONAL ORILLASDEL ZAMORA”



ANEXO B

***CODIFICACIÓN DE LA APLICACIÓN INTERACTIVA
UTILIZANDO EL SOFTWARE NCL ECLIPSE***

Aquí se presenta el código desarrollado en el software Eclipse en el cual se ha instalado el plugin de NCL Eclipse, gracias a éste último se puede escribir el código de la aplicación interactiva. El código define las regiones de presentación de cada imagen, los descriptores de cómo se presentará la imagen, los archivos media (imágenes en formato png, vídeo en formato mp4, y audio en formato aac) y los enlaces que definen que se presentará después de que algún media inicie o en el momento de accionar cada botón.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Generated by NCL Eclipse -->
<ncl id="main" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <regionBase>
      <region id="rgTV" zIndex="0">
        <region id="rgreinicioinfo" zIndex="1"/>
        <region id="rgloading" zIndex="2"/>
        <region id="rgApp" zIndex="3">
          <!--REGIONES BOTONE INICIAL-->
          <region id="rginfo" width="54%" height="14%"
left="4%" top="84%" zIndex="4"/>
          <!--REGIONES PANTALLA INICIAL-->
          <region id="rgtxt_infoprin" width="34%" height="26%"
left="36%" top="54%" zIndex="5"/>
          <region id="rgtxt_descParque" width="55%"
height="33%" left="23%" top="27%" zIndex="4"/>
          <region id="rgimgincial" width="18%" height="28%"
left="2%" top="30%" zIndex="4"/>
          <region id="rgbotred" width="13%" height="18%"
left="4%" top="79%" zIndex="4"/>
          <region id="rgbotgreen" width="15%" height="18%"
left="24%" top="79%" zIndex="4"/>
          <region id="rgbotyellow" width="11%" height="18%"
left="49%" top="78.5%" zIndex="4"/>
          <region id="rgbotblue" width="9%" height="19%"
left="68%" top="78%" zIndex="4"/>
          <region id="rgexit" width="13%" height="18%"
left="85%" top="80%" zIndex="6"/>
          <region id="rgutpl" width="16%" height="19%"
left="72%" top="32%" zIndex="6"/>
        </region>
        <region id="rgAppzoo" zIndex="3"> <!--REGIONES BOTON ROJO
ZOOLOGICO-->
          <region id="rghojas" height="33%" top="67%"
zIndex="4"/>
          <region id="rgtxt_desczoo" width="52%" height="22%"
left="30%" top="32%" zIndex="4"/>
          <region id="rgfondo_bot_zoo" width="25%" height="41%"
left="2.5%" top="24.5%" zIndex="4"/>
          <region id="rgbotredzoo" width="13%" height="10%"
left="3%" top="25%" zIndex="5"/>
        </region>
      </region>
    </regionBase>
  </head>
</ncl>
```

```

        <region id="rgbotgreenzoo" width="19%" height="10%"
left="3%" top="35%" zIndex="5"/>
        <region id="rgbotyellowzoo" width="16%" height="10%"
left="3%" top="45%" zIndex="5"/>
        <region id="rgbotbluezoo" width="24%" height="10%"
left="3%" top="55%" zIndex="5"/>
        <region id="rgtxt_info_zoo" width="82%"
height="16.5%" left="3%" top="81.5%" zIndex="5"/>
        <region id="rgdir_derecha1" width="6%" height="13%"
left="77%" top="42%" zIndex="4"/>
        <region id="rgdir_izquierda1" width="6%" height="13%"
left="28%" top="42%" zIndex="4"/>
        <region id="rgvisimg1" width="25%" height="58%"
left="42%" top="18%" zIndex="4"/>
        <region id="rgvisimg2" width="35%" height="55%"
left="35%" top="18%" zIndex="4"/>
        <region id="rgvisimg2_" width="35%" height="55%"
left="37%" top="18%" zIndex="4"/>
        <region id="rgvisimg3" width="45%" height="50%"
left="33%" top="20%" zIndex="4"/>
        <region id="rgvisimg4" width="45%" height="45%"
left="33%" top="22.5%" zIndex="4"/>
        <region id="rgvisimg5" width="45%" height="38%"
left="33%" top="26%" zIndex="4"/>
        <region id="rgutpl_zoo" width="12%" height="12%"
left="74%" top="14%" zIndex="6"/>
    </region>
    <region id="rgApporq" zIndex="3"> <!--REGIONES BOTON VERDE
ORQUIDEARIO-->
        <region id="rgfondo_txt_orq" width="82.5%"
height="17%" left="2.75%" top="81.25%" zIndex="4"/>
        <region id="rgfondo_bot_orq" width="19%" height="49%"
left="11.5%" top="24.5%" zIndex="4"/>
        <region id="rgtxt_descorq" width="52%" height="22%"
left="35%" top="37%" zIndex="4"/>
        <region id="rgbotredorq" width="18%" height="12%"
left="12%" top="25%" zIndex="5"/>
        <region id="rgbotgreenorq" width="18%" height="12%"
left="12%" top="37%" zIndex="5"/>
        <region id="rgbotyelloworq" width="18%" height="12%"
left="12%" top="49%" zIndex="5"/>
        <region id="rgbotblueorq" width="18%" height="12%"
left="12%" top="61%" zIndex="5"/>
        <region id="rgdir_derecha2" width="6%" height="13%"
left="82%" top="42%" zIndex="4"/>
        <region id="rgdir_izquierda2" width="6%" height="13%"
left="34%" top="42%" zIndex="4"/>
        <region id="rgvisimg6" width="45%" height="55%"
left="38%" top="20%" zIndex="4"/>
        <region id="rgvisimg7" width="45%" height="59%"
left="38%" top="18%" zIndex="4"/>
        <region id="rgvisimg8" width="40%" height="61%"
left="40.5%" top="18%" zIndex="4"/>
        <region id="rgvisimg9" width="25%" height="58%"
left="47%" top="18%" zIndex="4"/>

```

```

        <region id="rgvisimg10" width="35%" height="62%"
left="43%" top="18%" zIndex="4"/>
        <region id="rgutpl_orq" width="12%" height="12%"
left="82%" top="16%" zIndex="6"/>
    </region>
    <region id="rgAppviv" zIndex="3"> <!--REGIONES BOTON
AMARILLO VIVERO-->
        <region id="rgtxt_descviv" width="52%" height="30%"
left="39%" top="37%" zIndex="4"/>
        <region id="rgfondo_bot_viv" width="32%" height="49%"
left="2.5%" top="23.5%" zIndex="4"/>
        <region id="rgbotredviv" width="20%" height="12%"
left="3%" top="24%" zIndex="5"/>
        <region id="rgbotgreenviv" width="21%" height="12%"
left="3%" top="36%" zIndex="5"/>
        <region id="rgbotyellowviv" width="31%" height="12%"
left="3%" top="48%" zIndex="5"/>
        <region id="rgbotblueviv" width="29%" height="12%"
left="3%" top="60%" zIndex="5"/>
        <region id="rgdir_derecha3" width="6%" height="13%"
left="86%" top="46%" zIndex="4"/>
        <region id="rgdir_izquierda3" width="6%" height="13%"
left="36%" top="46%" zIndex="4"/>
        <region id="rgvisimg11" width="40%" height="50%"
left="44.5%" top="24%" zIndex="4"/>
        <region id="rgvisimg12" width="36%" height="50%"
left="46.5%" top="24%" zIndex="4"/>
        <region id="rgvisimg13" width="45%" height="50%"
left="42%" top="24%" zIndex="4"/>
        <region id="rgutpl_viv" width="12%" height="12%"
left="82%" top="4%" zIndex="6"/>
    </region>
    <region id="rgAppmapa" zIndex="3"> <!--REGIONES BOTON AZUL
MAPA-->
        <region id="rgimg_mapa" width="68.5%" height="82%"
left="3%" top="2.5%" zIndex="4"/>
        <region id="rgtxt_mapa" width="26.5%" height="53%"
left="72%" top="27%" zIndex="4"/>
        <region id="rgtxt_mapa1" width="67%" height="13%"
left="17%" top="85.5%" zIndex="4"/>
        <region id="rgutpl_mapa" width="12%" height="12%"
left="74%" top="9%" zIndex="6"/>
    </region>
</region>
</regionBase>
<descriptorBase>
    <descriptor id="dTv" region="rgTV"/>
    <descriptor id="dApp" region="rgApp"/>
    <descriptor id="dmusic"/>
    <!--DESCRIPTOR BOTON INICIAL-->
    <descriptor id="dreinicioinfo" region="rgreinicioinfo"/>

    <descriptor id="dinfo" region="rginfo" explicitDur="15s"/>
    <!--DESCRIPTORES PANTALLA INICIAL-->
    <descriptor id="dloading" region="rgloading" explicitDur="3s"/>

```

```

<descriptor id="dutpl" region="rgutpl"/>
<descriptor id="dloadingmap" region="rgloading"
explicitDur="1.5s"/>
<descriptor id="dtx_t_infoprin" region="rgtx_t_infoprin"/>
<descriptor id="dtx_t_descParque" region="rgtx_t_descParque"/>
<descriptor id="dimginicial" region="rgimginicial"/>
<descriptor id="dbotred" region="rgbotred"/>
<descriptor id="dbotgreen" region="rgbotgreen"/>
<descriptor id="dbotyellow" region="rgbotyellow"/>
<descriptor id="dbotblue" region="rgbotblue"/>
<descriptor id="dexit" region="rgexit"/>
<!--DESCRIPTORES OPCION BOTON ROJO-->
<descriptor id="dhojas" region="rghojas"/>
<descriptor id="dfondo_bot_zoo" region="rgfondo_bot_zoo"/>
<descriptor id="dtx_t_desczoo" region="rgtx_t_desczoo"/>
<descriptor id="dAppzoo" region="rgAppzoo"/>
<descriptor id="dbotredzoo" region="rgbotredzoo"/>
<descriptor id="dbotgreenzoo" region="rgbotgreenzoo"/>
<descriptor id="dbotyellowzoo" region="rgbotyellowzoo"/>
<descriptor id="dbotbluezoo" region="rgbotbluezoo"/>
<descriptor id="dtx_t_info_zoo" region="rgtx_t_info_zoo"/>
<descriptor id="ddir_derecha1" region="rgdir_derecha1"/>
<descriptor id="ddir_izquierda1" region="rgdir_izquierda1"/>
<descriptor id="dvistaimg1" region="rgvisimg1"/>
<descriptor id="dvistaimg2" region="rgvisimg2"/>
<descriptor id="dvistaimg2_" region="rgvisimg2_"/>
<descriptor id="dvistaimg3" region="rgvisimg3"/>
<descriptor id="dvistaimg4" region="rgvisimg4"/>
<descriptor id="dvistaimg5" region="rgvisimg5"/>
<descriptor id="dutpl_zoo" region="rgutpl_zoo"/>
<!--DESCRIPTORES OPCION BOTON VERDE-->
<descriptor id="dApporq" region="rgApporq"/>
<descriptor id="dfondo_bot_orq" region="rgfondo_bot_orq"/>
<descriptor id="dfondo_txt_orq" region="rgfondo_txt_orq"/>
<descriptor id="dtx_t_descorq" region="rgtx_t_descorq"/>
<descriptor id="dbotredorq" region="rgbotredorq"/>
<descriptor id="dbotgreenorq" region="rgbotgreenorq"/>
<descriptor id="dbotyelloworq" region="rgbotyelloworq"/>
<descriptor id="dbotblueorq" region="rgbotblueorq"/>
<descriptor id="ddir_derecha2" region="rgdir_derecha2"/>
<descriptor id="ddir_izquierda2" region="rgdir_izquierda2"/>
<descriptor id="dvistaimg6" region="rgvisimg6"/>
<descriptor id="dvistaimg7" region="rgvisimg7"/>
<descriptor id="dvistaimg8" region="rgvisimg8"/>
<descriptor id="dvistaimg9" region="rgvisimg9"/>
<descriptor id="dvistaimg10" region="rgvisimg10"/>
<descriptor id="dutpl_orq" region="rgutpl_orq"/>
<!--DESCRIPTORES OPCION BOTON AMARILLO-->
<descriptor id="dAppviv" region="rgAppviv"/>
<descriptor id="dfondo_bot_viv" region="rgfondo_bot_viv"/>
<descriptor id="dtx_t_descviv" region="rgtx_t_descviv"/>
<descriptor id="dbotredviv" region="rgbotredviv"/>
<descriptor id="dbotgreenviv" region="rgbotgreenviv"/>
<descriptor id="dbotyellowviv" region="rgbotyellowviv"/>
<descriptor id="dbotblueviv" region="rgbotblueviv"/>

```

```

        <descriptor id="ddir_derecha3" region="rgdir_derecha3"/>
        <descriptor id="ddir_izquierda3" region="rgdir_izquierda3"/>
        <descriptor id="dvistaimg11" region="rgvisimg11"/>
        <descriptor id="dvistaimg12" region="rgvisimg12"/>
        <descriptor id="dvistaimg13" region="rgvisimg13"/>
        <descriptor id="dutpl_viv" region="rgutpl_viv"/>
        <!--DESCRIPTORES OPCION BOTON AZUL-->
        <descriptor id="dAppmapa" region="rgAppmapa"/>
        <descriptor id="dimg_mapa" region="rgimg_mapa"/>
        <descriptor id="dtxt_mapa" region="rgtxt_mapa"/>
        <descriptor id="dtxt_mapa1" region="rgtxt_mapa1"/>
        <descriptor id="dutpl_mapa" region="rgutpl_mapa"/>
    </descriptorBase>
    <!--IMPORTACIÓN DEL DOCUMENTO DE CONECTORES-->
    <connectorBase>
        <importBase documentURI="ConnectorBase.ncl" alias="conector"/>
    </connectorBase>
</head>

<body>
    <!--PUERTA-->
    <port id="puerta" component="video"/>
    <!--ARCHIVOS MEDIA-->
    <!--<media id="video" src="sbtvd-ts://" descriptor="dTv">
        <property name="soundLevel" value="1"/>
    </media>-->
    <media id="video" src="media/video.mp4" descriptor="dTv">
        <property name="soundLevel" value="1"/>
    </media>
    <media id="music" src="media/relax.mp3" descriptor="dmusic"/>
    <!--MEDIA BOTON INICIAL-->
    <media id="reinicio_info" src="media/reinicio_info.png"
descriptor="dreinicioinfo"/>
    <media id="info" src="media/info.png" descriptor="dinfo"/>
    <!--MEDIAS PANTALLA INICIAL-->
    <media id="loading" src="media/fondo_load.png" descriptor="dloading"/>
    <media id="utpl_prin" src="media/utpl.png" descriptor="dutpl"/>
    <media id="fondo" src="media/fondo_prin.png" descriptor="dApp"/>
    <media id="txt_infoprin" src="media/info_txt_prin.png"
descriptor="dtxt_infoprin"/>
    <media id="txt_descParque" src="media/txt_descParque.png"
descriptor="dtxt_descParque"/>
    <media id="img_inicial" src="media/img_prin.png"
descriptor="dimginicial"/>
    <media id="exit" src="media/exit.png" descriptor="dexit"/>
    <media id="bot_redprin" src="media/bot_red.png" descriptor="dbotred"/>
    <media id="bot_greenprin" src="media/bot_green.png"
descriptor="dbotgreen"/>
    <media id="bot_yellowprin" src="media/bot_yellow.png"
descriptor="dbotyellow"/>
    <media id="bot_blueprin" src="media/bot_blue.png"
descriptor="dbotblue"/>
    <!--MEDIAS DE AMBIENTE DE BOTON ROJO ZOOLOGICO-->
    <media id="fondo_bot_zoo" src="media/fondo.png"
descriptor="dfondo_bot_zoo"/>

```



```

        <media id="flech_der1" src="media/f_derecha.png"
descriptor="ddir_derecha1"/>
        <media id="flech_izq1" src="media/f_izquierda.png"
descriptor="ddir_izquierda1"/>
        <media id="hojas" src="media/Zoologico/hojas_zoo.png"
descriptor="dhojas"/>
        <media id="txt_desczoo" src="media/Zoologico/intro_zoologico.png"
descriptor="dtxt_desczoo"/>
        <media id="fondo_zoo" src="media/Zoologico/fondo_zoo.png"
descriptor="dAppzoo"/>
        <media id="back1" src="media/back.png" descriptor="dexit"/>
        <media id="utpl_zoo" src="media/utpl1.png" descriptor="dutpl_zoo"/>
        <media id="botredzoo" src="media/Zoologico/bot_red_zoo.png"
descriptor="dbotredzoo"/>
        <media id="botgreenzoo" src="media/Zoologico/bot_green_zoo.png"
descriptor="dbotgreenzoo"/>
        <media id="botyellowzoo" src="media/Zoologico/bot_yellow_zoo.png"
descriptor="dbotyellowzoo"/>
        <media id="botbluezoo" src="media/Zoologico/bot_blue_zoo.png"
descriptor="dbotbluezoo"/>
        <media id="vistaimg1_1_1" src="media/Zoologico/Aves/amazonafestiva.png"
descriptor="dvistaimg1"/>
        <media id="vistaimg1_1_2"
src="media/Zoologico/Aves/guacamayoazuliamarillo.png" descriptor="dvistaimg2"/>
        <media id="vistaimg1_1_3"
src="media/Zoologico/Aves/guacamayoescarlata.png" descriptor="dvistaimg2"/>
        <media id="vistaimg1_1_4" src="media/Zoologico/Aves/loromacareno.png"
descriptor="dvistaimg2"/>
        <media id="vistaimg1_1_5" src="media/Zoologico/Aves/pavoreal.png"
descriptor="dvistaimg3"/>
        <media id="txt_zoo1_1_1" src="media/Zoologico/Aves/amazonafestivat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_1_2"
src="media/Zoologico/Aves/guacamayoazuliamarillot.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_1_3"
src="media/Zoologico/Aves/guacamayoescarlata.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_1_4" src="media/Zoologico/Aves/loromacarenot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_1_5" src="media/Zoologico/Aves/pavorealt.png"
descriptor="dtxt_info_zoo"/>
        <media id="vistaimg1_2_1"
src="media/Zoologico/Mamiferos/ardillasabanera.png" descriptor="dvistaimg2"/>
        <media id="vistaimg1_2_2"
src="media/Zoologico/Mamiferos/cabezademat.png" descriptor="dvistaimg3"/>
        <media id="vistaimg1_2_3"
src="media/Zoologico/Mamiferos/leonafricano.png" descriptor="dvistaimg4"/>
        <media id="vistaimg1_2_4"
src="media/Zoologico/Mamiferos/monocapuchinoblanco.png" descriptor="dvistaimg2"/>
        <media id="vistaimg1_2_5"
src="media/Zoologico/Mamiferos/osodeanteojos.png" descriptor="dvistaimg4"/>
        <media id="txt_zoo1_2_1"
src="media/Zoologico/Mamiferos/ardillasabanerat.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_2_2"
src="media/Zoologico/Mamiferos/cabezadematet.png" descriptor="dtxt_info_zoo"/>

```



```

        <media id="txt_zoo1_2_3"
src="media/Zoologico/Mamiferos/leonafricanot.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_2_4"
src="media/Zoologico/Mamiferos/monocapuchinoblancot.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_2_5"
src="media/Zoologico/Mamiferos/osodeanteojost.png" descriptor="dtxt_info_zoo"/>
        <media id="vistaimg1_3_1"
src="media/Zoologico/Reptiles/boaconstrictor.png" descriptor="dvistaimg3"/>
        <media id="vistaimg1_3_2"
src="media/Zoologico/Reptiles/boamatacaballo.png" descriptor="dvistaimg5"/>
        <media id="vistaimg1_3_3"
src="media/Zoologico/Reptiles/tortugalapagos.png" descriptor="dvistaimg2"/>
        <media id="vistaimg1_3_4"
src="media/Zoologico/Reptiles/tortugamordedora.png" descriptor="dvistaimg5"/>
        <media id="vistaimg1_3_5"
src="media/Zoologico/Reptiles/tortugamotelot.png" descriptor="dvistaimg2"/>
        <media id="txt_zoo1_3_1"
src="media/Zoologico/Reptiles/boaconstrictort.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_3_2"
src="media/Zoologico/Reptiles/boamatacaballot.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_3_3"
src="media/Zoologico/Reptiles/tortugalapagost.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_3_4"
src="media/Zoologico/Reptiles/tortugamordedorat.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_3_5"
src="media/Zoologico/Reptiles/tortugamotelot.png" descriptor="dtxt_info_zoo"/>
        <media id="vistaimg1_4_1"
src="media/Zoologico/Nuevo_Especimen/coatiamazonico.png" descriptor="dvistaimg2"/>
        <media id="vistaimg1_4_2"
src="media/Zoologico/Nuevo_Especimen/colambodecolanegra.png"
descriptor="dvistaimg5"/>
        <media id="vistaimg1_4_3"
src="media/Zoologico/Nuevo_Especimen/espatulasrosadas.png" descriptor="dvistaimg3"/>
        <media id="vistaimg1_4_4"
src="media/Zoologico/Nuevo_Especimen/tucanpiquiacanalado.png"
descriptor="dvistaimg2"/>
        <media id="vistaimg1_4_5"
src="media/Zoologico/Nuevo_Especimen/tucanswainson.png" descriptor="dvistaimg1"/>
        <media id="txt_zoo1_4_1"
src="media/Zoologico/Nuevo_Especimen/coatiamazonicot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_4_2"
src="media/Zoologico/Nuevo_Especimen/colambodecolanegrat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_4_3"
src="media/Zoologico/Nuevo_Especimen/espatulasrosadast.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_4_4"
src="media/Zoologico/Nuevo_Especimen/tucanpiquiacanaladot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_zoo1_4_5"
src="media/Zoologico/Nuevo_Especimen/tucanswainsont.png" descriptor="dtxt_info_zoo"/>
        <!-- MEDIAS DE AMBIENTE DE BOTON VERDE ORQUIDEARIO -->
        <media id="fondo_bot_orq" src="media/fondo.png"
descriptor="dfondo_bot_orq"/>

```

```

        <media id="fondo_txt_orq" src="media/Orquideario/fondo_txt_orq.png"
descriptor="dfondo_txt_orq"/>
        <media id="flech_der2" src="media/f_derecha.png"
descriptor="ddir_derecha2"/>
        <media id="flech_izq2" src="media/f_izquierda.png"
descriptor="ddir_izquierda2"/>
        <media id="txt_descorq" src="media/Orquideario/intro_orquideario.png"
descriptor="dtxt_descorq"/>
        <media id="fondo_orq" src="media/Orquideario/fondo_orq.png"
descriptor="dApporq"/>
        <media id="back2" src="media/back.png" descriptor="dexit"/>
        <media id="utpl_orq" src="media/utpl1.png" descriptor="dutpl_orq"/>
        <media id="botredorq" src="media/Orquideario/bot_red_orq.png"
descriptor="dbotredorq"/>
        <media id="botgreenorq" src="media/Orquideario/bot_green_orq.png"
descriptor="dbotgreenorq"/>
        <media id="botyelloworq" src="media/Orquideario/bot_yellow_orq.png"
descriptor="dbotyelloworq"/>
        <media id="botblueorq" src="media/Orquideario/bot_blue_orq.png"
descriptor="dbotblueorq"/>
        <media id="vistaimg2_1_1"
src="media/Orquideario/Nave1/anguloavirginalis.png" descriptor="dvistaimg6"/>
        <media id="vistaimg2_1_2" src="media/Orquideario/Nave1/brassia.png"
descriptor="dvistaimg6"/>
        <media id="vistaimg2_1_3" src="media/Orquideario/Nave1/cattleyalba.png"
descriptor="dvistaimg7"/>
        <media id="vistaimg2_1_4" src="media/Orquideario/Nave1/cymbidium.png"
descriptor="dvistaimg6"/>
        <media id="vistaimg2_1_5" src="media/Orquideario/Nave1/dracula.png"
descriptor="dvistaimg7"/>
        <media id="txt_orq2_1_1"
src="media/Orquideario/Nave1/anguloavirginalist.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_1_2" src="media/Orquideario/Nave1/brassiat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_1_3" src="media/Orquideario/Nave1/cattleyalbat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_1_4" src="media/Orquideario/Nave1/cymbidiumt.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_1_5" src="media/Orquideario/Nave1/draculat.png"
descriptor="dtxt_info_zoo"/>
        <media id="vistaimg2_2_1" src="media/Orquideario/Nave2/epidendrum.png"
descriptor="dvistaimg8"/>
        <media id="vistaimg2_2_2"
src="media/Orquideario/Nave2/helciasanguinolenta.png" descriptor="dvistaimg7"/>
        <media id="vistaimg2_2_3" src="media/Orquideario/Nave2/huntleya.png"
descriptor="dvistaimg7"/>
        <media id="vistaimg2_2_4" src="media/Orquideario/Nave2/ida.png"
descriptor="dvistaimg6"/>
        <media id="vistaimg2_2_5"
src="media/Orquideario/Nave2/maxillarialuteoalba.png" descriptor="dvistaimg6"/>
        <media id="txt_orq2_2_1" src="media/Orquideario/Nave2/epidendrumt.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_2_2"
src="media/Orquideario/Nave2/helciasanguinolentat.png" descriptor="dtxt_info_zoo"/>

```

```

        <media id="txt_orq2_2_3" src="media/Orquideario/Nave2/huntleyat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_2_4" src="media/Orquideario/Nave2/idad.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_2_5"
src="media/Orquideario/Nave2/maxillarialuteoalbat.png" descriptor="dtxt_info_zoo"/>
        <media id="vistaimg2_3_1"
src="media/Orquideario/Nave3/maxillariasplendens.png" descriptor="dvistaimg7"/>
        <media id="vistaimg2_3_2"
src="media/Orquideario/Nave3/odontoglossum.png" descriptor="dvistaimg7"/>
        <media id="vistaimg2_3_3"
src="media/Orquideario/Nave3/ontidiumexcavatum.png" descriptor="dvistaimg7"/>
        <media id="vistaimg2_3_4"
src="media/Orquideario/Nave3/phragmipediumcaudatum.png" descriptor="dvistaimg7"/>
        <media id="vistaimg2_3_5"
src="media/Orquideario/Nave3/pleurothallis.png" descriptor="dvistaimg7"/>
        <media id="txt_orq2_3_1"
src="media/Orquideario/Nave3/maxillariasplendenst.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_3_2"
src="media/Orquideario/Nave3/odontoglossumt.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_3_3"
src="media/Orquideario/Nave3/ontidiumexcavatumt.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_3_4"
src="media/Orquideario/Nave3/phragmipediumcaudatumt.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_3_5"
src="media/Orquideario/Nave3/pleurothallist.png" descriptor="dtxt_info_zoo"/>
        <media id="vistaimg2_4_1"
src="media/Orquideario/Nave4/rodrigueziasecunda.png" descriptor="dvistaimg7"/>
        <media id="vistaimg2_4_2"
src="media/Orquideario/Nave4/sobraliafimbriata.png" descriptor="dvistaimg9"/>
        <media id="vistaimg2_4_3"
src="media/Orquideario/Nave4/stanhopeaconnata.png" descriptor="dvistaimg10"/>
        <media id="vistaimg2_4_4" src="media/Orquideario/Nave4/stelis.png"
descriptor="dvistaimg7"/>
        <media id="vistaimg2_4_5" src="media/Orquideario/Nave4/zootrophion.png"
descriptor="dvistaimg7"/>
        <media id="txt_orq2_4_1"
src="media/Orquideario/Nave4/rodrigueziasecundat.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_4_2"
src="media/Orquideario/Nave4/sobraliafimbriatat.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_4_3"
src="media/Orquideario/Nave4/stanhopeaconnatat.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_4_4" src="media/Orquideario/Nave4/stelist.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_orq2_4_5" src="media/Orquideario/Nave4/zootrophiont.png"
descriptor="dtxt_info_zoo"/>
        <!-- MEDIAS DE AMBIENTE DE BOTON AMARILLO VIVERO -->
        <media id="fondo_txt_viv" src="media/Vivero/fondo_txt_viv.png"
descriptor="dfondo_txt_orq"/>
        <media id="fondo_bot_viv" src="media/fondo.png"
descriptor="dfondo_bot_viv"/>
        <media id="flech_der3" src="media/f_derecha.png"
descriptor="ddir_derecha3"/>
        <media id="flech_izq3" src="media/f_izquierda.png"
descriptor="ddir_izquierda3"/>

```

```

        <media id="txt_descviv" src="media/Vivero/intro_vivero.png"
descriptor="dtxt_descviv"/>
        <media id="fondo_viv" src="media/Vivero/fondo_viv.png"
descriptor="dAppviv"/>
        <media id="back3" src="media/back.png" descriptor="dexit"/>
        <media id="utpl_viv" src="media/utpl1.png" descriptor="dutpl_viv"/>
        <media id="botredviv" src="media/Vivero/bot_red_viv.png"
descriptor="dbotredviv"/>
        <media id="botgreenviv" src="media/Vivero//bot_green_viv.png"
descriptor="dbotgreenviv"/>
        <media id="botyellowviv" src="media/Vivero//bot_yellow_viv.png"
descriptor="dbotyellowviv"/>
        <media id="botblueviv" src="media/Vivero//bot_blue_viv.png"
descriptor="dbotblueviv"/>
        <media id="vistaimg3_1_1" src="media/Vivero/Arboles/buganvilla.png"
descriptor="dvistaimg13"/>
        <media id="vistaimg3_1_2" src="media/Vivero/Arboles/calistemo.png"
descriptor="dvistaimg13"/>
        <media id="vistaimg3_1_3" src="media/Vivero/Arboles/ceibo.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_1_4" src="media/Vivero/Arboles/farolchino.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_1_5" src="media/Vivero/Arboles/tulipanafricano.png"
descriptor="dvistaimg11"/>
        <media id="txt_viv3_1_1" src="media/Vivero/Arboles/buganvillat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_1_2" src="media/Vivero/Arboles/calistemot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_1_3" src="media/Vivero/Arboles/ceibot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_1_4" src="media/Vivero/Arboles/farolchinot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_1_5" src="media/Vivero/Arboles/tulipanafricanot.png"
descriptor="dtxt_info_zoo"/>
        <media id="vistaimg3_2_1" src="media/Vivero/Frutales/capuli.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_2_2" src="media/Vivero/Frutales/guabo.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_2_3" src="media/Vivero/Frutales/naranjagria.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_2_4" src="media/Vivero/Frutales/nispero.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_2_5" src="media/Vivero/Frutales/uva.png"
descriptor="dvistaimg11"/>
        <media id="txt_viv3_2_1" src="media/Vivero/Frutales/capulit.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_2_2" src="media/Vivero/Frutales/guabot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_2_3" src="media/Vivero/Frutales/naranjagriat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_2_4" src="media/Vivero/Frutales/nisperot.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_2_5" src="media/Vivero/Frutales/uvat.png"
descriptor="dtxt_info_zoo"/>

```

```

        <media id="vistaimg3_3_1"
src="media/Vivero/Ornamentales/achiraornamental.png" descriptor="dvistaimg13"/>
        <media id="vistaimg3_3_2" src="media/Vivero/Ornamentales/amarilis.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_3_3" src="media/Vivero/Ornamentales/dalia.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_3_4"
src="media/Vivero/Ornamentales/lagrimademaria.png" descriptor="dvistaimg11"/>
        <media id="vistaimg3_3_5" src="media/Vivero/Ornamentales/lirio.png"
descriptor="dvistaimg11"/>
        <media id="txt_viv3_3_1"
src="media/Vivero/Ornamentales/achiraornamental.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_3_2" src="media/Vivero/Ornamentales/amarilist.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_3_3" src="media/Vivero/Ornamentales/daliat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_3_4"
src="media/Vivero/Ornamentales/lagrimademariat.png" descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_3_5" src="media/Vivero/Ornamentales/liriot.png"
descriptor="dtxt_info_zoo"/>
        <media id="vistaimg3_4_1" src="media/Vivero/Otros/boxus.png"
descriptor="dvistaimg12"/>
        <media id="vistaimg3_4_2" src="media/Vivero/Otros/cactus.png"
descriptor="dvistaimg13"/>
        <media id="vistaimg3_4_3" src="media/Vivero/Otros/costilladeadan.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_4_4" src="media/Vivero/Otros/hiedra.png"
descriptor="dvistaimg11"/>
        <media id="vistaimg3_4_5" src="media/Vivero/Otros/siempreviva.png"
descriptor="dvistaimg11"/>
        <media id="txt_viv3_4_1" src="media/Vivero/Otros/boxust.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_4_2" src="media/Vivero/Otros/cactust.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_4_3" src="media/Vivero/Otros/costilladeadant.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_4_4" src="media/Vivero/Otros/hiedrat.png"
descriptor="dtxt_info_zoo"/>
        <media id="txt_viv3_4_5" src="media/Vivero/Otros/siemprevivat.png"
descriptor="dtxt_info_zoo"/>
        <!-- MEDIAS DE AMBIENTE DE BOTON AZUL MAPA -->
        <media id="back4" src="media/back.png" descriptor="dexit"/>
        <media id="utpl_mapa" src="media/utpl1.png" descriptor="dutpl_mapa"/>
        <media id="img_mapa" src="media/Comollegar/mapazoo.png"
descriptor="ding_mapa"/>
        <media id="fondo_mapa" src="media/Comollegar/fondo_mapa.png"
descriptor="dAppmapa"/>
        <media id="txt_mapa" src="media/Comollegar/txt_mapa.png"
descriptor="dtxt_mapa"/>
        <media id="txt_mapa1" src="media/Comollegar/txt_mapa1.png"
descriptor="dtxt_mapa1"/>
        <!-- ENLACE -->
        <!-- Inicio aplicación -->
        <link xconnector="conector#onBeginStartNDelay">
            <bind role="onBegin" component="video"/>

```

```

        <linkParam name="delay" value="5s"/>
        <bind role="start" component="info"/>
    </link>
<!-- Reinicio FONDO MUSICAL -->
<link xconnector="conector#onEndStart">
    <bind role="onEnd" component="music" />
    <bind role="start" component="music" />
</link>
<!-- Reinicio INFO -->
<link xconnector="conector#onEndStartNDelay">
    <bind role="onEnd" component="reinicio_info" />
    <bind role="start" component="info" />
    <linkParam name="delay" value="300s"/>
</link>
<!-- Presentación INFO -->
<!-- Cambio de audio video a fondo musical -->
<link xconnector="conector#onKeySelectionSetN">
    <bind role="onSelection" component="info" />
    <linkParam name="keyCode" value="INFO"/>
    <bind role="set" component="video" interface="soundLevel" >
        <bindParam name="var" value="0"/>
    </bind>
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="info" />
    <linkParam name="keyCode" value="INFO"/>
    <bind role="stop" component="info" />
    <bind role="start" component="reinicio_info" />
    <bind role="start" component="loading" />
    <bind role="start" component="txt_infoprin" />
    <bind role="start" component="utpl_prin" />
    <bind role="start" component="txt_descParque" />
    <bind role="start" component="img_inicial" />
    <bind role="start" component="bot_redprin" />
    <bind role="start" component="bot_greenprin" />
    <bind role="start" component="bot_yellowprin" />
    <bind role="start" component="bot_blueprin" />
    <bind role="start" component="exit" />
    <bind role="start" component="fondo" />
    <bind role="start" component="music" />
</link>
<!-- FIN Presentación INFO -->
<!-- Cambio de audio fondo musical a video -->
<link xconnector="conector#onKeySelectionSetN">
    <bind role="onSelection" component="exit" />
    <linkParam name="keyCode" value="EXIT"/>
    <bind role="set" component="video" interface="soundLevel" >
        <bindParam name="var" value="1"/>
    </bind>
</link>
<link xconnector="conector#onKeySelectionAbort">
    <bind role="onSelection" component="exit" />
    <linkParam name="keyCode" value="EXIT"/>
    <bind role="abort" component="music" />
</link>

```



```

<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="exit" />
  <linkParam name="keyCode" value="EXIT"/>
  <bind role="stop" component="reinicio_info" />
  <bind role="stop" component="fondo" />
  <bind role="stop" component="utpl_prin" />
  <bind role="stop" component="loading" />
  <bind role="stop" component="txt_infoprin" />
  <bind role="stop" component="txt_descParque" />
  <bind role="stop" component="img_inicial" />
  <bind role="stop" component="bot_redprin" />
  <bind role="stop" component="bot_greenprin" />
  <bind role="stop" component="bot_yellowprin" />
  <bind role="stop" component="bot_blueprin" />
  <bind role="stop" component="exit" />
</link>
<!-- Presentación BOTON ROJO -->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="bot_redprin" />
  <linkParam name="keyCode" value="RED"/>
  <bind role="stop" component="exit" />
  <bind role="stop" component="fondo" />
  <bind role="stop" component="utpl_prin" />
  <bind role="stop" component="txt_descParque" />
  <bind role="stop" component="img_inicial" />
  <bind role="stop" component="txt_infoprin" />
  <bind role="stop" component="bot_redprin" />
  <bind role="stop" component="bot_greenprin" />
  <bind role="stop" component="bot_yellowprin" />
  <bind role="stop" component="bot_blueprin" />
</link>
<link xconnector="conector#onKeySelectionStartN">
  <bind role="onSelection" component="bot_redprin" />
  <linkParam name="keyCode" value="RED"/>
  <bind role="start" component="loading" />
  <bind role="start" component="fondo_bot_zoo" />
  <bind role="start" component="hojas" />
  <bind role="start" component="utpl_zoo" />
  <bind role="start" component="txt_desczoo" />
  <bind role="start" component="botredzoo" />
  <bind role="start" component="botgreenzoo" />
  <bind role="start" component="botyellowzoo" />
  <bind role="start" component="botbluezoo" />
  <bind role="start" component="fondo_zoo" />
  <bind role="start" component="back1" />
</link>
<!-- FIN Presentación BOTON ROJO -->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="back1" />
  <linkParam name="keyCode" value="BACK"/>
  <bind role="stop" component="fondo_bot_zoo" />
  <bind role="stop" component="botredzoo" />
  <bind role="stop" component="botgreenzoo" />
  <bind role="stop" component="botyellowzoo" />
  <bind role="stop" component="botbluezoo" />

```

```

    <bind role="stop" component="txt_desczoo" />
    <bind role="stop" component="back1" />
    <bind role="stop" component="hojas" />
    <bind role="stop" component="utpl_zoo" />
    <bind role="stop" component="fondo_zoo" />
    <bind role="stop" component="flech_der1" />
    <bind role="stop" component="flech_izq1" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="back1" />
    <linkParam name="keyCode" value="BACK"/>
    <bind role="stop" component="vistaimg1_1_1" />
    <bind role="stop" component="txt_zoo1_1_1" />
    <bind role="stop" component="vistaimg1_1_2" />
    <bind role="stop" component="txt_zoo1_1_2" />
    <bind role="stop" component="vistaimg1_1_3" />
    <bind role="stop" component="txt_zoo1_1_3" />
    <bind role="stop" component="vistaimg1_1_4" />
    <bind role="stop" component="txt_zoo1_1_4" />
    <bind role="stop" component="vistaimg1_1_5" />
    <bind role="stop" component="txt_zoo1_1_5" />
    <bind role="stop" component="vistaimg1_2_1" />
    <bind role="stop" component="txt_zoo1_2_1" />
    <bind role="stop" component="vistaimg1_2_2" />
    <bind role="stop" component="txt_zoo1_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="back1" />
    <linkParam name="keyCode" value="BACK"/>
    <bind role="stop" component="vistaimg1_2_3" />
    <bind role="stop" component="txt_zoo1_2_3" />
    <bind role="stop" component="vistaimg1_2_4" />
    <bind role="stop" component="txt_zoo1_2_4" />
    <bind role="stop" component="vistaimg1_2_5" />
    <bind role="stop" component="txt_zoo1_2_5" />
    <bind role="stop" component="vistaimg1_3_1" />
    <bind role="stop" component="txt_zoo1_3_1" />
    <bind role="stop" component="vistaimg1_3_2" />
    <bind role="stop" component="txt_zoo1_3_2" />
    <bind role="stop" component="vistaimg1_3_3" />
    <bind role="stop" component="txt_zoo1_3_3" />
    <bind role="stop" component="vistaimg1_3_4" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="back1" />
    <linkParam name="keyCode" value="BACK"/>
    <bind role="stop" component="txt_zoo1_3_4" />
    <bind role="stop" component="vistaimg1_3_5" />
    <bind role="stop" component="txt_zoo1_3_5" />
    <bind role="stop" component="vistaimg1_4_1" />
    <bind role="stop" component="txt_zoo1_4_1" />
    <bind role="stop" component="vistaimg1_4_2" />
    <bind role="stop" component="txt_zoo1_4_2" />
    <bind role="stop" component="vistaimg1_4_3" />
    <bind role="stop" component="txt_zoo1_4_3" />

```



```

        <bind role="stop" component="vistaimg1_4_4" />
        <bind role="stop" component="txt_zoo1_4_4" />
        <bind role="stop" component="vistaimg1_4_5" />
        <bind role="stop" component="txt_zoo1_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStartN">
        <bind role="onSelection" component="back1" />
        <linkParam name="keyCode" value="BACK"/>
        <bind role="start" component="loading" />
        <bind role="start" component="exit" />
        <bind role="start" component="utpl_prin" />
        <bind role="start" component="txt_infoprin" />
        <bind role="start" component="txt_descParque" />
        <bind role="start" component="img_inicial" />
        <bind role="start" component="bot_redprin" />
        <bind role="start" component="bot_greenprin" />
        <bind role="start" component="bot_yellowprin" />
        <bind role="start" component="bot_blueprin" />
        <bind role="start" component="fondo" />
    </link>
    <!-- Presentación BOTON ROJO ZOO-->
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botredzoo" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="vistaimg1_2_1" />
        <bind role="stop" component="txt_zoo1_2_1" />
        <bind role="stop" component="vistaimg1_2_2" />
        <bind role="stop" component="txt_zoo1_2_2" />
        <bind role="stop" component="vistaimg1_2_3" />
        <bind role="stop" component="txt_zoo1_2_3" />
        <bind role="stop" component="vistaimg1_2_4" />
        <bind role="stop" component="txt_zoo1_2_4" />
        <bind role="stop" component="vistaimg1_2_5" />
        <bind role="stop" component="txt_zoo1_2_5" />
        <bind role="stop" component="vistaimg1_3_1" />
        <bind role="stop" component="txt_zoo1_3_1" />
        <bind role="stop" component="vistaimg1_3_2" />
        <bind role="stop" component="txt_zoo1_3_2" />
        <bind role="stop" component="vistaimg1_3_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botredzoo" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="txt_zoo1_3_3" />
        <bind role="stop" component="vistaimg1_3_4" />
        <bind role="stop" component="txt_zoo1_3_4" />
        <bind role="stop" component="vistaimg1_3_5" />
        <bind role="stop" component="txt_zoo1_3_5" />
        <bind role="stop" component="vistaimg1_4_1" />
        <bind role="stop" component="txt_zoo1_4_1" />
        <bind role="stop" component="vistaimg1_4_2" />
        <bind role="stop" component="txt_zoo1_4_2" />
        <bind role="stop" component="vistaimg1_4_3" />
        <bind role="stop" component="txt_zoo1_4_3" />
        <bind role="stop" component="vistaimg1_4_4" />
    </link>

```

```

        <bind role="stop" component="txt_zoo1_4_4" />
        <bind role="stop" component="vistaimg1_4_5" />
        <bind role="stop" component="txt_zoo1_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="botredzoo" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="hojas" />
        <bind role="stop" component="txt_desczoo" />
        <bind role="stop" component="vistaimg1_1_2" />
        <bind role="stop" component="txt_zoo1_1_2" />
        <bind role="stop" component="vistaimg1_1_3" />
        <bind role="stop" component="txt_zoo1_1_3" />
        <bind role="stop" component="vistaimg1_1_4" />
        <bind role="stop" component="txt_zoo1_1_4" />
        <bind role="stop" component="vistaimg1_1_5" />
        <bind role="stop" component="txt_zoo1_1_5" />
        <bind role="start" component="vistaimg1_1_1" />
        <bind role="start" component="txt_zoo1_1_1" />
        <bind role="start" component="flech_der1" />
        <bind role="start" component="flech_izq1" />
    </link>
    <!--Navegacion imagenes de opcion ROJO en ZOO TECLEADO DERECHA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_1_1" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_1_1" />
        <bind role="stop" component="vistaimg1_1_1" />
        <bind role="start" component="vistaimg1_1_2" />
        <bind role="start" component="txt_zoo1_1_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_1_2" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>e
        <bind role="stop" component="txt_zoo1_1_2" />
        <bind role="stop" component="vistaimg1_1_2" />
        <bind role="start" component="vistaimg1_1_3" />
        <bind role="start" component="txt_zoo1_1_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_1_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_1_3" />
        <bind role="stop" component="vistaimg1_1_3" />
        <bind role="start" component="vistaimg1_1_4" />
        <bind role="start" component="txt_zoo1_1_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_1_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="start" component="vistaimg1_1_5" />
        <bind role="stop" component="txt_zoo1_1_4" />
        <bind role="stop" component="vistaimg1_1_4" />
        <bind role="start" component="vistaimg1_1_5" />
        <bind role="start" component="txt_zoo1_1_5" />
    </link>

```

```

</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_1_5" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_zoo1_1_5" />
  <bind role="stop" component="vistaimg1_1_5" />
  <bind role="start" component="vistaimg1_1_1" />
  <bind role="start" component="txt_zoo1_1_1" />
</link>
<!--Navegacion imagenes de opcion ROJO en ZOO TECLEADO IZQUIERDA-->
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_1_1" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_1_1" />
  <bind role="stop" component="vistaimg1_1_1" />
  <bind role="start" component="vistaimg1_1_5" />
  <bind role="start" component="txt_zoo1_1_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_1_2" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_1_2" />
  <bind role="stop" component="vistaimg1_1_2" />
  <bind role="start" component="vistaimg1_1_1" />
  <bind role="start" component="txt_zoo1_1_1" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_1_3" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_1_3" />
  <bind role="stop" component="vistaimg1_1_3" />
  <bind role="start" component="vistaimg1_1_2" />
  <bind role="start" component="txt_zoo1_1_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_1_4" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_1_4" />
  <bind role="stop" component="vistaimg1_1_4" />
  <bind role="start" component="vistaimg1_1_3" />
  <bind role="start" component="txt_zoo1_1_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_1_5" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_1_5" />
  <bind role="stop" component="vistaimg1_1_5" />
  <bind role="start" component="vistaimg1_1_4" />
  <bind role="start" component="txt_zoo1_1_4" />
</link>
<!-- Presentación BOTON VERDE ZOO-->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botgreenzoo" />
  <linkParam name="keyCode" value="GREEN"/>
  <bind role="stop" component="vistaimg1_1_1" />

```

```

<bind role="stop" component="txt_zoo1_1_1" />
<bind role="stop" component="vistaimg1_1_2" />
<bind role="stop" component="txt_zoo1_1_2" />
<bind role="stop" component="vistaimg1_1_3" />
<bind role="stop" component="txt_zoo1_1_3" />
<bind role="stop" component="vistaimg1_1_4" />
<bind role="stop" component="txt_zoo1_1_4" />
<bind role="stop" component="vistaimg1_1_5" />
<bind role="stop" component="txt_zoo1_1_5" />
<bind role="stop" component="vistaimg1_3_1" />
<bind role="stop" component="txt_zoo1_3_1" />
<bind role="stop" component="vistaimg1_3_2" />
<bind role="stop" component="txt_zoo1_3_2" />
<bind role="stop" component="vistaimg1_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botgreenzoo" />
  <linkParam name="keyCode" value="GREEN"/>
  <bind role="stop" component="txt_zoo1_3_3" />
  <bind role="stop" component="vistaimg1_3_4" />
  <bind role="stop" component="txt_zoo1_3_4" />
  <bind role="stop" component="vistaimg1_3_5" />
  <bind role="stop" component="txt_zoo1_3_5" />
  <bind role="stop" component="vistaimg1_4_1" />
  <bind role="stop" component="txt_zoo1_4_1" />
  <bind role="stop" component="vistaimg1_4_2" />
  <bind role="stop" component="txt_zoo1_4_2" />
  <bind role="stop" component="vistaimg1_4_3" />
  <bind role="stop" component="txt_zoo1_4_3" />
  <bind role="stop" component="vistaimg1_4_4" />
  <bind role="stop" component="txt_zoo1_4_4" />
  <bind role="stop" component="vistaimg1_4_5" />
  <bind role="stop" component="txt_zoo1_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="botgreenzoo" />
  <linkParam name="keyCode" value="GREEN"/>
  <bind role="stop" component="hojas" />
  <bind role="stop" component="txt_desczoo" />
  <bind role="stop" component="vistaimg1_2_2" />
  <bind role="stop" component="txt_zoo1_2_2" />
  <bind role="stop" component="vistaimg1_2_3" />
  <bind role="stop" component="txt_zoo1_2_3" />
  <bind role="stop" component="vistaimg1_2_4" />
  <bind role="stop" component="txt_zoo1_2_4" />
  <bind role="stop" component="vistaimg1_2_5" />
  <bind role="stop" component="txt_zoo1_2_5" />
  <bind role="start" component="vistaimg1_2_1" />
  <bind role="start" component="txt_zoo1_2_1" />
  <bind role="start" component="flech_der1" />
  <bind role="start" component="flech_izq1" />
</link>
<!--Navegacion imagenes de opcion VERDE en ZOO TECLEADO DERECHA-->
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_2_1" />

```

```

        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_2_1" />
        <bind role="stop" component="vistaimg1_2_1" />
        <bind role="start" component="vistaimg1_2_2" />
        <bind role="start" component="txt_zoo1_2_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_2_2" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_2_2" />
        <bind role="stop" component="vistaimg1_2_2" />
        <bind role="start" component="vistaimg1_2_3" />
        <bind role="start" component="txt_zoo1_2_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_2_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_2_3" />
        <bind role="stop" component="vistaimg1_2_3" />
        <bind role="start" component="vistaimg1_2_4" />
        <bind role="start" component="txt_zoo1_2_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_2_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_2_4" />
        <bind role="stop" component="vistaimg1_2_4" />
        <bind role="start" component="vistaimg1_2_5" />
        <bind role="start" component="txt_zoo1_2_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_2_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_2_5" />
        <bind role="stop" component="vistaimg1_2_5" />
        <bind role="start" component="vistaimg1_2_1" />
        <bind role="start" component="txt_zoo1_2_1" />
    </link>
    <!--Navegacion imagenes de opcion VERDE en ZOO TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_2_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_2_1" />
        <bind role="stop" component="vistaimg1_2_1" />
        <bind role="start" component="vistaimg1_2_5" />
        <bind role="start" component="txt_zoo1_2_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_2_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_2_2" />
        <bind role="stop" component="vistaimg1_2_2" />
        <bind role="start" component="vistaimg1_2_1" />
        <bind role="start" component="txt_zoo1_2_1" />
    </link>

```

```

<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_2_3" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_2_3" />
  <bind role="stop" component="vistaimg1_2_3" />
  <bind role="start" component="vistaimg1_2_2" />
  <bind role="start" component="txt_zoo1_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_2_4" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_2_4" />
  <bind role="stop" component="vistaimg1_2_4" />
  <bind role="start" component="vistaimg1_2_3" />
  <bind role="start" component="txt_zoo1_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_2_5" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_2_5" />
  <bind role="stop" component="vistaimg1_2_5" />
  <bind role="start" component="vistaimg1_2_4" />
  <bind role="start" component="txt_zoo1_2_4" />
</link>
<!-- Presentación BOTON AMARILLO ZOO-->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botyellowzoo" />
  <linkParam name="keyCode" value="YELLOW"/>
  <bind role="stop" component="vistaimg1_1_1" />
  <bind role="stop" component="txt_zoo1_1_1" />
  <bind role="stop" component="vistaimg1_1_2" />
  <bind role="stop" component="txt_zoo1_1_2" />
  <bind role="stop" component="vistaimg1_1_3" />
  <bind role="stop" component="txt_zoo1_1_3" />
  <bind role="stop" component="vistaimg1_1_4" />
  <bind role="stop" component="txt_zoo1_1_4" />
  <bind role="stop" component="vistaimg1_1_5" />
  <bind role="stop" component="txt_zoo1_1_5" />
  <bind role="stop" component="vistaimg1_2_1" />
  <bind role="stop" component="txt_zoo1_2_1" />
  <bind role="stop" component="vistaimg1_2_2" />
  <bind role="stop" component="txt_zoo1_2_2" />
  <bind role="stop" component="vistaimg1_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botyellowzoo" />
  <linkParam name="keyCode" value="YELLOW"/>
  <bind role="stop" component="txt_zoo1_2_3" />
  <bind role="stop" component="vistaimg1_2_4" />
  <bind role="stop" component="txt_zoo1_2_4" />
  <bind role="stop" component="vistaimg1_2_5" />
  <bind role="stop" component="txt_zoo1_2_5" />
  <bind role="stop" component="vistaimg1_4_1" />
  <bind role="stop" component="txt_zoo1_4_1" />
  <bind role="stop" component="vistaimg1_4_2" />

```



```

        <bind role="stop" component="txt_zoo1_4_2" />
        <bind role="stop" component="vistaimg1_4_3" />
        <bind role="stop" component="txt_zoo1_4_3" />
        <bind role="stop" component="vistaimg1_4_4" />
        <bind role="stop" component="txt_zoo1_4_4" />
        <bind role="stop" component="vistaimg1_4_5" />
        <bind role="stop" component="txt_zoo1_4_5" />
</link><link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="botyellowzoo" />
    <linkParam name="keyCode" value="YELLOW"/>
    <bind role="stop" component="hojas" />
    <bind role="stop" component="txt_desczoo" />
    <bind role="stop" component="vistaimg1_3_2" />
    <bind role="stop" component="txt_zoo1_3_2" />
    <bind role="stop" component="vistaimg1_3_3" />
    <bind role="stop" component="txt_zoo1_3_3" />
    <bind role="stop" component="vistaimg1_3_4" />
    <bind role="stop" component="txt_zoo1_3_4" />
    <bind role="stop" component="vistaimg1_3_5" />
    <bind role="stop" component="txt_zoo1_3_5" />
    <bind role="start" component="vistaimg1_3_1" />
    <bind role="start" component="txt_zoo1_3_1" />
    <bind role="start" component="flech_der1" />
    <bind role="start" component="flech_izq1" />
</link>
<!--Navegacion imagenes de opcion AMARILLO en ZOO TECLEADO DERECHA-->
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_3_1" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_zoo1_3_1" />
    <bind role="stop" component="vistaimg1_3_1" />
    <bind role="start" component="vistaimg1_3_2" />
    <bind role="start" component="txt_zoo1_3_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_3_2" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_zoo1_3_2" />
    <bind role="stop" component="vistaimg1_3_2" />
    <bind role="start" component="vistaimg1_3_3" />
    <bind role="start" component="txt_zoo1_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_3_3" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_zoo1_3_3" />
    <bind role="stop" component="vistaimg1_3_3" />
    <bind role="start" component="vistaimg1_3_4" />
    <bind role="start" component="txt_zoo1_3_4" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_3_4" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_zoo1_3_4" />
    <bind role="stop" component="vistaimg1_3_4" />

```

```

        <bind role="start" component="vistaimg1_3_5" />
        <bind role="start" component="txt_zoo1_3_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_3_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_zoo1_3_5" />
        <bind role="stop" component="vistaimg1_3_5" />
        <bind role="start" component="vistaimg1_3_1" />
        <bind role="start" component="txt_zoo1_3_1" />
    </link>
    <!--Navegacion imagenes de opcion AMARILLO en ZOO TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_3_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_3_1" />
        <bind role="stop" component="vistaimg1_3_1" />
        <bind role="start" component="vistaimg1_3_5" />
        <bind role="start" component="txt_zoo1_3_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_3_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_3_2" />
        <bind role="stop" component="vistaimg1_3_2" />
        <bind role="start" component="vistaimg1_3_1" />
        <bind role="start" component="txt_zoo1_3_1" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_3_3" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_3_3" />
        <bind role="stop" component="vistaimg1_3_3" />
        <bind role="start" component="vistaimg1_3_2" />
        <bind role="start" component="txt_zoo1_3_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_3_4" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_3_4" />
        <bind role="stop" component="vistaimg1_3_4" />
        <bind role="start" component="vistaimg1_3_3" />
        <bind role="start" component="txt_zoo1_3_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg1_3_5" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_zoo1_3_5" />
        <bind role="stop" component="vistaimg1_3_5" />
        <bind role="start" component="vistaimg1_3_4" />
        <bind role="start" component="txt_zoo1_3_4" />
    </link>
    <!-- Presentación BOTON AZUL ZOO-->
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botbluezoo" />

```



```

<linkParam name="keyCode" value="BLUE"/>
<bind role="stop" component="vistaimg1_1_1" />
<bind role="stop" component="txt_zoo1_1_1" />
<bind role="stop" component="vistaimg1_1_2" />
<bind role="stop" component="txt_zoo1_1_2" />
<bind role="stop" component="vistaimg1_1_3" />
<bind role="stop" component="txt_zoo1_1_3" />
<bind role="stop" component="vistaimg1_1_4" />
<bind role="stop" component="txt_zoo1_1_4" />
<bind role="stop" component="vistaimg1_1_5" />
<bind role="stop" component="txt_zoo1_1_5" />
<bind role="stop" component="vistaimg1_2_1" />
<bind role="stop" component="txt_zoo1_2_1" />
<bind role="stop" component="vistaimg1_2_2" />
<bind role="stop" component="txt_zoo1_2_2" />
<bind role="stop" component="vistaimg1_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botbluezoo" />
  <linkParam name="keyCode" value="BLUE"/>
  <bind role="stop" component="txt_zoo1_2_3" />
  <bind role="stop" component="vistaimg1_2_4" />
  <bind role="stop" component="txt_zoo1_2_4" />
  <bind role="stop" component="vistaimg1_2_5" />
  <bind role="stop" component="txt_zoo1_2_5" />
  <bind role="stop" component="vistaimg1_3_1" />
  <bind role="stop" component="txt_zoo1_3_1" />
  <bind role="stop" component="vistaimg1_3_2" />
  <bind role="stop" component="txt_zoo1_3_2" />
  <bind role="stop" component="vistaimg1_3_3" />
  <bind role="stop" component="txt_zoo1_3_3" />
  <bind role="stop" component="vistaimg1_3_4" />
  <bind role="stop" component="txt_zoo1_3_4" />
  <bind role="stop" component="vistaimg1_3_5" />
  <bind role="stop" component="txt_zoo1_3_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="botbluezoo" />
  <linkParam name="keyCode" value="BLUE"/>
  <bind role="stop" component="hojas" />
  <bind role="stop" component="txt_desczoo" />
  <bind role="stop" component="vistaimg1_4_2" />
  <bind role="stop" component="txt_zoo1_4_2" />
  <bind role="stop" component="vistaimg1_4_3" />
  <bind role="stop" component="txt_zoo1_4_3" />
  <bind role="stop" component="vistaimg1_4_4" />
  <bind role="stop" component="txt_zoo1_4_4" />
  <bind role="stop" component="vistaimg1_4_5" />
  <bind role="stop" component="txt_zoo1_4_5" />
  <bind role="start" component="vistaimg1_4_1" />
  <bind role="start" component="txt_zoo1_4_1" />
  <bind role="start" component="flech_der1" />
  <bind role="start" component="flech_izq1" />
</link>
<!--Navegacion imagenes de opcion AZUL en ZOO TECLEADO DERECHA-->

```

```

<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_1" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_zoo1_4_1" />
  <bind role="stop" component="vistaimg1_4_1" />
  <bind role="start" component="vistaimg1_4_2" />
  <bind role="start" component="txt_zoo1_4_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_2" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_zoo1_4_2" />
  <bind role="stop" component="vistaimg1_4_2" />
  <bind role="start" component="vistaimg1_4_3" />
  <bind role="start" component="txt_zoo1_4_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_3" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_zoo1_4_3" />
  <bind role="stop" component="vistaimg1_4_3" />
  <bind role="start" component="vistaimg1_4_4" />
  <bind role="start" component="txt_zoo1_4_4" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_4" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_zoo1_4_4" />
  <bind role="stop" component="vistaimg1_4_4" />
  <bind role="start" component="vistaimg1_4_5" />
  <bind role="start" component="txt_zoo1_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_5" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_zoo1_4_5" />
  <bind role="stop" component="vistaimg1_4_5" />
  <bind role="start" component="vistaimg1_4_1" />
  <bind role="start" component="txt_zoo1_4_1" />
</link>
<!--Navegacion imagenes de opcion AZUL en ZOO TECLEADO IZQUIERDA-->
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_1" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_4_1" />
  <bind role="stop" component="vistaimg1_4_1" />
  <bind role="start" component="vistaimg1_4_5" />
  <bind role="start" component="txt_zoo1_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg1_4_2" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_zoo1_4_2" />
  <bind role="stop" component="vistaimg1_4_2" />
  <bind role="start" component="vistaimg1_4_1" />

```

```

        <bind role="start" component="txt_zoo1_4_1" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_4_3" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_zoo1_4_3" />
    <bind role="stop" component="vistaimg1_4_3" />
    <bind role="start" component="vistaimg1_4_2" />
    <bind role="start" component="txt_zoo1_4_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_4_4" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_zoo1_4_4" />
    <bind role="stop" component="vistaimg1_4_4" />
    <bind role="start" component="vistaimg1_4_3" />
    <bind role="start" component="txt_zoo1_4_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg1_4_5" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_zoo1_4_5" />
    <bind role="stop" component="vistaimg1_4_5" />
    <bind role="start" component="vistaimg1_4_4" />
    <bind role="start" component="txt_zoo1_4_4" />
</link>
<!-- Presentación BOTON VERDE -->
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="bot_greenprin" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="stop" component="fondo" />
    <bind role="stop" component="txt_descParque" />
    <bind role="stop" component="txt_infoprin" />
    <bind role="stop" component="img_inicial" />
    <bind role="stop" component="exit" />
    <bind role="stop" component="utpl_prin" />
    <bind role="stop" component="bot_redprin" />
    <bind role="stop" component="bot_greenprin" />
    <bind role="stop" component="bot_yellowprin" />
    <bind role="stop" component="bot_blueprin" />
</link>
<link xconnector="conector#onKeySelectionStartN">
    <bind role="onSelection" component="bot_greenprin" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="start" component="fondo_bot_orq" />
    <bind role="start" component="loading" />
    <bind role="start" component="fondo_botones" />
    <bind role="start" component="txt_descorq" />
    <bind role="start" component="botredorq" />
    <bind role="start" component="botgreenorq" />
    <bind role="start" component="botyelloworq" />
    <bind role="start" component="botblueorq" />
    <bind role="start" component="back2" />
    <bind role="start" component="utpl_orq" />
    <bind role="start" component="fondo_orq" />

```

```

</link>
<!-- FIN Presentación BOTON VERDE -->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="back2" />
  <linkParam name="keyCode" value="BACK"/>
  <bind role="stop" component="fondo_bot_orq" />
  <bind role="stop" component="back2" />
  <bind role="stop" component="utpl_orq" />
  <bind role="stop" component="txt_descorq" />
  <bind role="stop" component="flech_der2" />
  <bind role="stop" component="flech_izq2" />
  <bind role="stop" component="fondo_orq" />
  <bind role="stop" component="botredorq" />
  <bind role="stop" component="botgreenorq" />
  <bind role="stop" component="botyelloworq" />
  <bind role="stop" component="botblueorq" />
  <bind role="stop" component="fondo_txt_orq" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="back2" />
  <linkParam name="keyCode" value="BACK"/>
  <bind role="stop" component="vistaimg2_1_1" />
  <bind role="stop" component="txt_orq2_1_1" />
  <bind role="stop" component="vistaimg2_1_2" />
  <bind role="stop" component="txt_orq2_1_2" />
  <bind role="stop" component="vistaimg2_1_3" />
  <bind role="stop" component="txt_orq2_1_3" />
  <bind role="stop" component="vistaimg2_1_4" />
  <bind role="stop" component="txt_orq2_1_4" />
  <bind role="stop" component="vistaimg2_1_5" />
  <bind role="stop" component="txt_orq2_1_5" />
  <bind role="stop" component="vistaimg2_2_1" />
  <bind role="stop" component="txt_orq2_2_1" />
  <bind role="stop" component="vistaimg2_2_2" />
  <bind role="stop" component="txt_orq2_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="back2" />
  <linkParam name="keyCode" value="BACK"/>
  <bind role="stop" component="vistaimg2_2_3" />
  <bind role="stop" component="txt_orq2_2_3" />
  <bind role="stop" component="vistaimg2_2_4" />
  <bind role="stop" component="txt_orq2_2_4" />
  <bind role="stop" component="vistaimg2_2_5" />
  <bind role="stop" component="txt_orq2_2_5" />
  <bind role="stop" component="vistaimg2_3_1" />
  <bind role="stop" component="txt_orq2_3_1" />
  <bind role="stop" component="vistaimg2_3_2" />
  <bind role="stop" component="txt_orq2_3_2" />
  <bind role="stop" component="vistaimg2_3_3" />
  <bind role="stop" component="txt_orq2_3_3" />
  <bind role="stop" component="vistaimg2_3_4" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="back2" />

```

```

        <linkParam name="keyCode" value="BACK"/>
        <bind role="stop" component="txt_orq2_3_4" />
        <bind role="stop" component="vistaimg2_3_5" />
        <bind role="stop" component="txt_orq2_3_5" />
        <bind role="stop" component="vistaimg2_4_1" />
        <bind role="stop" component="txt_orq2_4_1" />
        <bind role="stop" component="vistaimg2_4_2" />
        <bind role="stop" component="txt_orq2_4_2" />
        <bind role="stop" component="vistaimg2_4_3" />
        <bind role="stop" component="txt_orq2_4_3" />
        <bind role="stop" component="vistaimg2_4_4" />
        <bind role="stop" component="txt_orq2_4_4" />
        <bind role="stop" component="vistaimg2_4_5" />
        <bind role="stop" component="txt_orq2_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStartN">
        <bind role="onSelection" component="back2" />
        <linkParam name="keyCode" value="BACK"/>
        <bind role="start" component="loading" />
        <bind role="start" component="txt_infoprin" />
        <bind role="start" component="txt_descParque" />
        <bind role="start" component="img_inicial" />
        <bind role="start" component="bot_redprin" />
        <bind role="start" component="bot_greenprin" />
        <bind role="start" component="bot_yellowprin" />
        <bind role="start" component="bot_blueprin" />
        <bind role="start" component="exit" />
        <bind role="start" component="utpl_prin" />
        <bind role="start" component="fondo" />
    </link>
    <!-- Presentación BOTON ROJO ORQ-->
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botredorq" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="vistaimg2_2_1" />
        <bind role="stop" component="txt_orq2_2_1" />
        <bind role="stop" component="vistaimg2_2_2" />
        <bind role="stop" component="txt_orq2_2_2" />
        <bind role="stop" component="vistaimg2_2_3" />
        <bind role="stop" component="txt_orq2_2_3" />
        <bind role="stop" component="vistaimg2_2_4" />
        <bind role="stop" component="txt_orq2_2_4" />
        <bind role="stop" component="vistaimg2_2_5" />
        <bind role="stop" component="txt_orq2_2_5" />
        <bind role="stop" component="vistaimg2_3_1" />
        <bind role="stop" component="txt_orq2_3_1" />
        <bind role="stop" component="vistaimg2_3_2" />
        <bind role="stop" component="txt_orq2_3_2" />
        <bind role="stop" component="vistaimg2_3_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botredorq" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="txt_orq2_3_3" />
        <bind role="stop" component="vistaimg2_3_4" />

```

```

        <bind role="stop" component="txt_orq2_3_4" />
        <bind role="stop" component="vistaimg2_3_5" />
        <bind role="stop" component="txt_orq2_3_5" />
        <bind role="stop" component="vistaimg2_4_1" />
        <bind role="stop" component="txt_orq2_4_1" />
        <bind role="stop" component="vistaimg2_4_2" />
        <bind role="stop" component="txt_orq2_4_2" />
        <bind role="stop" component="vistaimg2_4_3" />
        <bind role="stop" component="txt_orq2_4_3" />
        <bind role="stop" component="vistaimg2_4_4" />
        <bind role="stop" component="txt_orq2_4_4" />
        <bind role="stop" component="vistaimg2_4_5" />
        <bind role="stop" component="txt_orq2_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="botredorq" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="txt_descorq" />
        <bind role="stop" component="vistaimg2_1_2" />
        <bind role="stop" component="txt_orq2_1_2" />
        <bind role="stop" component="vistaimg2_1_3" />
        <bind role="stop" component="txt_orq2_1_3" />
        <bind role="stop" component="vistaimg2_1_4" />
        <bind role="stop" component="txt_orq2_1_4" />
        <bind role="stop" component="vistaimg2_1_5" />
        <bind role="stop" component="txt_orq2_1_5" />
        <bind role="start" component="vistaimg2_1_1" />
        <bind role="start" component="txt_orq2_1_1" />
        <bind role="start" component="flech_der2" />
        <bind role="start" component="flech_izq2" />
        <bind role="start" component="fondo_txt_orq" />
    </link>
    <!--Navegacion imagenes de opcion ROJO en ORQ TECLEADO DERECHA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_1_1" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_1_1" />
        <bind role="stop" component="vistaimg2_1_1" />
        <bind role="start" component="vistaimg2_1_2" />
        <bind role="start" component="txt_orq2_1_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_1_2" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_1_2" />
        <bind role="stop" component="vistaimg2_1_2" />
        <bind role="start" component="vistaimg2_1_3" />
        <bind role="start" component="txt_orq2_1_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_1_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_1_3" />
        <bind role="stop" component="vistaimg2_1_3" />
        <bind role="start" component="vistaimg2_1_4" />
    </link>

```



```

        <bind role="start" component="txt_orq2_1_4" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_4" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_orq2_1_4" />
    <bind role="stop" component="vistaimg2_1_4" />
    <bind role="start" component="vistaimg2_1_5" />
    <bind role="start" component="txt_orq2_1_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_5" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_orq2_1_5" />
    <bind role="stop" component="vistaimg2_1_5" />
    <bind role="start" component="vistaimg2_1_1" />
    <bind role="start" component="txt_orq2_1_1" />
</link>
<!--Navegacion imagenes de opcion ROJO en ORQ TECLEADO IZQUIERDA-->
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_1" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_orq2_1_1" />
    <bind role="stop" component="vistaimg2_1_1" />
    <bind role="start" component="vistaimg2_1_5" />
    <bind role="start" component="txt_orq2_1_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_2" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_orq2_1_2" />
    <bind role="stop" component="vistaimg2_1_2" />
    <bind role="start" component="vistaimg2_1_1" />
    <bind role="start" component="txt_orq2_1_1" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_3" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_orq2_1_3" />
    <bind role="stop" component="vistaimg2_1_3" />
    <bind role="start" component="vistaimg2_1_2" />
    <bind role="start" component="txt_orq2_1_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_4" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_orq2_1_4" />
    <bind role="stop" component="vistaimg2_1_4" />
    <bind role="start" component="vistaimg2_1_3" />
    <bind role="start" component="txt_orq2_1_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg2_1_5" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop" component="txt_orq2_1_5" />

```

```

        <bind role="stop" component="vistaimg2_1_5" />
        <bind role="start" component="vistaimg2_1_4" />
        <bind role="start" component="txt_orq2_1_4" />
</link>
<!-- Presentación BOTON VERDE ORQ-->
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botgreenorq" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="stop" component="vistaimg2_1_1" />
    <bind role="stop" component="txt_orq2_1_1" />
    <bind role="stop" component="vistaimg2_1_2" />
    <bind role="stop" component="txt_orq2_1_2" />
    <bind role="stop" component="vistaimg2_1_3" />
    <bind role="stop" component="txt_orq2_1_3" />
    <bind role="stop" component="vistaimg2_1_4" />
    <bind role="stop" component="txt_orq2_1_4" />
    <bind role="stop" component="vistaimg2_1_5" />
    <bind role="stop" component="txt_orq2_1_5" />
    <bind role="stop" component="vistaimg2_3_1" />
    <bind role="stop" component="txt_orq2_3_1" />
    <bind role="stop" component="vistaimg2_3_2" />
    <bind role="stop" component="txt_orq2_3_2" />
    <bind role="stop" component="vistaimg2_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botgreenorq" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="stop" component="txt_orq2_3_3" />
    <bind role="stop" component="vistaimg2_3_4" />
    <bind role="stop" component="txt_orq2_3_4" />
    <bind role="stop" component="vistaimg2_3_5" />
    <bind role="stop" component="txt_orq2_3_5" />
    <bind role="stop" component="vistaimg2_4_1" />
    <bind role="stop" component="txt_orq2_4_1" />
    <bind role="stop" component="vistaimg2_4_2" />
    <bind role="stop" component="txt_orq2_4_2" />
    <bind role="stop" component="vistaimg2_4_3" />
    <bind role="stop" component="txt_orq2_4_3" />
    <bind role="stop" component="vistaimg2_4_4" />
    <bind role="stop" component="txt_orq2_4_4" />
    <bind role="stop" component="vistaimg2_4_5" />
    <bind role="stop" component="txt_orq2_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="botgreenorq" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="stop" component="txt_descorq" />
    <bind role="stop" component="vistaimg2_2_2" />
    <bind role="stop" component="txt_orq2_2_2" />
    <bind role="stop" component="vistaimg2_2_3" />
    <bind role="stop" component="txt_orq2_2_3" />
    <bind role="stop" component="vistaimg2_2_4" />
    <bind role="stop" component="txt_orq2_2_4" />
    <bind role="stop" component="vistaimg2_2_5" />
    <bind role="stop" component="txt_orq2_2_5" />

```



```

        <bind role="start" component="vistaimg2_2_1" />
        <bind role="start" component="txt_orq2_2_1" />
        <bind role="start" component="flech_der2" />
        <bind role="start" component="flech_izq2" />
        <bind role="start" component="fondo_txt_orq" />
    </link>
    <!--Navegacion imagenes de opcion VERDE en ORQ TECLEADO DERECHA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_2_1" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_2_1" />
        <bind role="stop" component="vistaimg2_2_1" />
        <bind role="start" component="vistaimg2_2_2" />
        <bind role="start" component="txt_orq2_2_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_2_2" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_2_2" />
        <bind role="stop" component="vistaimg2_2_2" />
        <bind role="start" component="vistaimg2_2_3" />
        <bind role="start" component="txt_orq2_2_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_2_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_2_3" />
        <bind role="stop" component="vistaimg2_2_3" />
        <bind role="start" component="vistaimg2_2_4" />
        <bind role="start" component="txt_orq2_2_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_2_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_2_4" />
        <bind role="stop" component="vistaimg2_2_4" />
        <bind role="start" component="vistaimg2_2_5" />
        <bind role="start" component="txt_orq2_2_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_2_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_2_5" />
        <bind role="stop" component="vistaimg2_2_5" />
        <bind role="start" component="vistaimg2_2_1" />
        <bind role="start" component="txt_orq2_2_1" />
    </link>
    <!--Navegacion imagenes de opcion VERDE en ORQ TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_2_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_2_1" />
        <bind role="stop" component="vistaimg2_2_1" />
        <bind role="start" component="vistaimg2_2_5" />
        <bind role="start" component="txt_orq2_2_5" />
    </link>

```

```

</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_2_2" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_orq2_2_2" />
  <bind role="stop" component="vistaimg2_2_2" />
  <bind role="start" component="vistaimg2_2_1" />
  <bind role="start" component="txt_orq2_2_1" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_2_3" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_orq2_2_3" />
  <bind role="stop" component="vistaimg2_2_3" />
  <bind role="start" component="vistaimg2_2_2" />
  <bind role="start" component="txt_orq2_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_2_4" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_orq2_2_4" />
  <bind role="stop" component="vistaimg2_2_4" />
  <bind role="start" component="vistaimg2_2_3" />
  <bind role="start" component="txt_orq2_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_2_5" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_orq2_2_5" />
  <bind role="stop" component="vistaimg2_2_5" />
  <bind role="start" component="vistaimg2_2_4" />
  <bind role="start" component="txt_orq2_2_4" />
</link>
<!-- Presentación BOTON AMARILLO ORQ-->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botyelloworq" />
  <linkParam name="keyCode" value="YELLOW"/>
  <bind role="stop" component="vistaimg2_1_1" />
  <bind role="stop" component="txt_orq2_1_1" />
  <bind role="stop" component="vistaimg2_1_2" />
  <bind role="stop" component="txt_orq2_1_2" />
  <bind role="stop" component="vistaimg2_1_3" />
  <bind role="stop" component="txt_orq2_1_3" />
  <bind role="stop" component="vistaimg2_1_4" />
  <bind role="stop" component="txt_orq2_1_4" />
  <bind role="stop" component="vistaimg2_1_5" />
  <bind role="stop" component="txt_orq2_1_5" />
  <bind role="stop" component="vistaimg2_2_1" />
  <bind role="stop" component="txt_orq2_2_1" />
  <bind role="stop" component="vistaimg2_2_2" />
  <bind role="stop" component="txt_orq2_2_2" />
  <bind role="stop" component="vistaimg2_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botyelloworq" />

```

```

<linkParam name="keyCode" value="YELLOW"/>
<bind role="stop" component="txt_orq2_2_3" />
<bind role="stop" component="vistaimg2_2_4" />
<bind role="stop" component="txt_orq2_2_4" />
<bind role="stop" component="vistaimg2_2_5" />
<bind role="stop" component="txt_orq2_2_5" />
<bind role="stop" component="vistaimg2_4_1" />
<bind role="stop" component="txt_orq2_4_1" />
<bind role="stop" component="vistaimg2_4_2" />
<bind role="stop" component="txt_orq2_4_2" />
<bind role="stop" component="vistaimg2_4_3" />
<bind role="stop" component="txt_orq2_4_3" />
<bind role="stop" component="vistaimg2_4_4" />
<bind role="stop" component="txt_orq2_4_4" />
<bind role="stop" component="vistaimg2_4_5" />
<bind role="stop" component="txt_orq2_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="botyelloworq" />
  <linkParam name="keyCode" value="YELLOW"/>
  <bind role="stop" component="txt_descorq" />
  <bind role="stop" component="vistaimg2_3_2" />
  <bind role="stop" component="txt_orq2_3_2" />
  <bind role="stop" component="vistaimg2_3_3" />
  <bind role="stop" component="txt_orq2_3_3" />
  <bind role="stop" component="vistaimg2_3_4" />
  <bind role="stop" component="txt_orq2_3_4" />
  <bind role="stop" component="vistaimg2_3_5" />
  <bind role="stop" component="txt_orq2_3_5" />
  <bind role="start" component="vistaimg2_3_1" />
  <bind role="start" component="txt_orq2_3_1" />
  <bind role="start" component="flech_der2" />
  <bind role="start" component="flech_izq2" />
  <bind role="start" component="fondo_txt_orq" />
</link>
<!--Navegacion imagenes de opcion AMARILLO en ORQ TECLEADO DERECHA-->
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_3_1" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_orq2_3_1" />
  <bind role="stop" component="vistaimg2_3_1" />
  <bind role="start" component="vistaimg2_3_2" />
  <bind role="start" component="txt_orq2_3_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_3_2" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_orq2_3_2" />
  <bind role="stop" component="vistaimg2_3_2" />
  <bind role="start" component="vistaimg2_3_3" />
  <bind role="start" component="txt_orq2_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg2_3_3" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>

```

```

        <bind role="stop"    component="txt_orq2_3_3" />
        <bind role="stop"    component="vistaimg2_3_3" />
        <bind role="start"   component="vistaimg2_3_4" />
        <bind role="start"   component="txt_orq2_3_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_3_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop"    component="txt_orq2_3_4" />
        <bind role="stop"    component="vistaimg2_3_4" />
        <bind role="start"   component="vistaimg2_3_5" />
        <bind role="start"   component="txt_orq2_3_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_3_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop"    component="txt_orq2_3_5" />
        <bind role="stop"    component="vistaimg2_3_5" />
        <bind role="start"   component="vistaimg2_3_1" />
        <bind role="start"   component="txt_orq2_3_1" />
    </link>
    <!--Navegacion imagenes de opcion AMARILLO en ORQ TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_3_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop"    component="txt_orq2_3_1" />
        <bind role="stop"    component="vistaimg2_3_1" />
        <bind role="start"   component="vistaimg2_3_5" />
        <bind role="start"   component="txt_orq2_3_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_3_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop"    component="txt_orq2_3_2" />
        <bind role="stop"    component="vistaimg2_3_2" />
        <bind role="start"   component="vistaimg2_3_1" />
        <bind role="start"   component="txt_orq2_3_1" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_3_3" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop"    component="txt_orq2_3_3" />
        <bind role="stop"    component="vistaimg2_3_3" />
        <bind role="start"   component="vistaimg2_3_2" />
        <bind role="start"   component="txt_orq2_3_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_3_4" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop"    component="txt_orq2_3_4" />
        <bind role="stop"    component="vistaimg2_3_4" />
        <bind role="start"   component="vistaimg2_3_3" />
        <bind role="start"   component="txt_orq2_3_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">

```

```

        <bind role="onSelection" component="vistaimg2_3_5" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_3_5" />
        <bind role="stop" component="vistaimg2_3_5" />
        <bind role="start" component="vistaimg2_3_4" />
        <bind role="start" component="txt_orq2_3_4" />
</link>
<!-- Presentación BOTON AZUL ORQ-->
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botblueorq" />
    <linkParam name="keyCode" value="BLUE"/>
    <bind role="stop" component="vistaimg2_1_1" />
    <bind role="stop" component="txt_orq2_1_1" />
    <bind role="stop" component="vistaimg2_1_2" />
    <bind role="stop" component="txt_orq2_1_2" />
    <bind role="stop" component="vistaimg2_1_3" />
    <bind role="stop" component="txt_orq2_1_3" />
    <bind role="stop" component="vistaimg2_1_4" />
    <bind role="stop" component="txt_orq2_1_4" />
    <bind role="stop" component="vistaimg2_1_5" />
    <bind role="stop" component="txt_orq2_1_5" />
    <bind role="stop" component="vistaimg2_2_1" />
    <bind role="stop" component="txt_orq2_2_1" />
    <bind role="stop" component="vistaimg2_2_2" />
    <bind role="stop" component="txt_orq2_2_2" />
    <bind role="stop" component="vistaimg2_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botblueorq" />
    <linkParam name="keyCode" value="BLUE"/>
    <bind role="stop" component="txt_orq2_2_3" />
    <bind role="stop" component="vistaimg2_2_4" />
    <bind role="stop" component="txt_orq2_2_4" />
    <bind role="stop" component="vistaimg2_2_5" />
    <bind role="stop" component="txt_orq2_2_5" />
    <bind role="stop" component="vistaimg2_3_1" />
    <bind role="stop" component="txt_orq2_3_1" />
    <bind role="stop" component="vistaimg2_3_2" />
    <bind role="stop" component="txt_orq2_3_2" />
    <bind role="stop" component="vistaimg2_3_3" />
    <bind role="stop" component="txt_orq2_3_3" />
    <bind role="stop" component="vistaimg2_3_4" />
    <bind role="stop" component="txt_orq2_3_4" />
    <bind role="stop" component="vistaimg2_3_5" />
    <bind role="stop" component="txt_orq2_3_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="botblueorq" />
    <linkParam name="keyCode" value="BLUE"/>
    <bind role="stop" component="txt_descorq" />
    <bind role="stop" component="vistaimg2_4_2" />
    <bind role="stop" component="txt_orq2_4_2" />
    <bind role="stop" component="vistaimg2_4_3" />
    <bind role="stop" component="txt_orq2_4_3" />
    <bind role="stop" component="vistaimg2_4_4" />

```

```

        <bind role="stop" component="txt_orq2_4_4" />
        <bind role="stop" component="vistaimg2_4_5" />
        <bind role="stop" component="txt_orq2_4_5" />
        <bind role="start" component="vistaimg2_4_1" />
        <bind role="start" component="txt_orq2_4_1" />
        <bind role="start" component="flech_der2" />
        <bind role="start" component="flech_izq2" />
        <bind role="start" component="fondo_txt_orq" />
    </link>
    <!--Navegacion imagenes de opcion AZUL en ORQ TECLEADO DERECHA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_1" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_4_1" />
        <bind role="stop" component="vistaimg2_4_1" />
        <bind role="start" component="vistaimg2_4_2" />
        <bind role="start" component="txt_orq2_4_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_2" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_4_2" />
        <bind role="stop" component="vistaimg2_4_2" />
        <bind role="start" component="vistaimg2_4_3" />
        <bind role="start" component="txt_orq2_4_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_4_3" />
        <bind role="stop" component="vistaimg2_4_3" />
        <bind role="start" component="vistaimg2_4_4" />
        <bind role="start" component="txt_orq2_4_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_4_4" />
        <bind role="stop" component="vistaimg2_4_4" />
        <bind role="start" component="vistaimg2_4_5" />
        <bind role="start" component="txt_orq2_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_orq2_4_5" />
        <bind role="stop" component="vistaimg2_4_5" />
        <bind role="start" component="vistaimg2_4_1" />
        <bind role="start" component="txt_orq2_4_1" />
    </link>
    <!--Navegacion imagenes de opcion AZUL en ORQ TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_4_1" />

```



```

        <bind role="stop" component="vistaimg2_4_1" />
        <bind role="start" component="vistaimg2_4_5" />
        <bind role="start" component="txt_orq2_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_4_2" />
        <bind role="stop" component="vistaimg2_4_2" />
        <bind role="start" component="vistaimg2_4_1" />
        <bind role="start" component="txt_orq2_4_1" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_3" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_4_3" />
        <bind role="stop" component="vistaimg2_4_3" />
        <bind role="start" component="vistaimg2_4_2" />
        <bind role="start" component="txt_orq2_4_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_4" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_4_4" />
        <bind role="stop" component="vistaimg2_4_4" />
        <bind role="start" component="vistaimg2_4_3" />
        <bind role="start" component="txt_orq2_4_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg2_4_5" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_orq2_4_5" />
        <bind role="stop" component="vistaimg2_4_5" />
        <bind role="start" component="vistaimg2_4_4" />
        <bind role="start" component="txt_orq2_4_4" />
    </link>
    <!-- Presentación BOTON AMARILLO -->
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="bot_yellowprin" />
        <linkParam name="keyCode" value="YELLOW"/>
        <bind role="stop" component="fondo" />
        <bind role="stop" component="exit" />
        <bind role="stop" component="utpl_prin" />
        <bind role="stop" component="txt_descParque" />
        <bind role="stop" component="img_inicial" />
        <bind role="stop" component="txt_infoprin" />
        <bind role="stop" component="bot_redprin" />
        <bind role="stop" component="bot_greenprin" />
        <bind role="stop" component="bot_yellowprin" />
        <bind role="stop" component="bot_blueprin" />
    </link>
    <link xconnector="conector#onKeySelectionStartN">
        <bind role="onSelection" component="bot_yellowprin" />
        <linkParam name="keyCode" value="YELLOW"/>
        <bind role="start" component="fondo_bot_viv" />

```

```

    <bind role="start" component="loading" />
    <bind role="start" component="utpl_viv" />
    <bind role="start" component="txt_descviv" />
    <bind role="start" component="botredviv" />
    <bind role="start" component="botgreenviv" />
    <bind role="start" component="botyellowviv" />
    <bind role="start" component="botblueviv" />
    <bind role="start" component="fondo_viv" />
    <bind role="start" component="back3" />
</link>
<!-- FIN Presentación BOTON AMARILLO -->
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="back3" />
    <linkParam name="keyCode" value="BACK"/>
    <bind role="stop" component="fondo_bot_viv" />
    <bind role="stop" component="back3" />
    <bind role="stop" component="utpl_viv" />
    <bind role="stop" component="flech_der3" />
    <bind role="stop" component="flech_izq3" />
    <bind role="stop" component="fondo_viv" />
    <bind role="stop" component="txt_descviv" />
    <bind role="stop" component="botredviv" />
    <bind role="stop" component="botgreenviv" />
    <bind role="stop" component="botyellowviv" />
    <bind role="stop" component="botblueviv" />
    <bind role="stop" component="fondo_txt_viv" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="back3" />
    <linkParam name="keyCode" value="BACK"/>
    <bind role="stop" component="vistaimg3_1_1" />
    <bind role="stop" component="txt_viv3_1_1" />
    <bind role="stop" component="vistaimg3_1_2" />
    <bind role="stop" component="txt_viv3_1_2" />
    <bind role="stop" component="vistaimg3_1_3" />
    <bind role="stop" component="txt_viv3_1_3" />
    <bind role="stop" component="vistaimg3_1_4" />
    <bind role="stop" component="txt_viv3_1_4" />
    <bind role="stop" component="vistaimg3_1_5" />
    <bind role="stop" component="txt_viv3_1_5" />
    <bind role="stop" component="vistaimg3_2_1" />
    <bind role="stop" component="txt_viv3_2_1" />
    <bind role="stop" component="vistaimg3_2_2" />
    <bind role="stop" component="txt_viv3_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="back3" />
    <linkParam name="keyCode" value="BACK"/>
    <bind role="stop" component="vistaimg3_2_3" />
    <bind role="stop" component="txt_viv3_2_3" />
    <bind role="stop" component="vistaimg3_2_4" />
    <bind role="stop" component="txt_viv3_2_4" />
    <bind role="stop" component="vistaimg3_2_5" />
    <bind role="stop" component="txt_viv3_2_5" />
    <bind role="stop" component="vistaimg3_3_1" />

```



```

        <bind role="stop" component="txt_viv3_3_1" />
        <bind role="stop" component="vistaimg3_3_2" />
        <bind role="stop" component="txt_viv3_3_2" />
        <bind role="stop" component="vistaimg3_3_3" />
        <bind role="stop" component="txt_viv3_3_3" />
        <bind role="stop" component="vistaimg3_3_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="back3" />
        <linkParam name="keyCode" value="BACK"/>
        <bind role="stop" component="txt_viv3_3_4" />
        <bind role="stop" component="vistaimg3_3_5" />
        <bind role="stop" component="txt_viv3_3_5" />
        <bind role="stop" component="vistaimg3_4_1" />
        <bind role="stop" component="txt_viv3_4_1" />
        <bind role="stop" component="vistaimg3_4_2" />
        <bind role="stop" component="txt_viv3_4_2" />
        <bind role="stop" component="vistaimg3_4_3" />
        <bind role="stop" component="txt_viv3_4_3" />
        <bind role="stop" component="vistaimg3_4_4" />
        <bind role="stop" component="txt_viv3_4_4" />
        <bind role="stop" component="vistaimg3_4_5" />
        <bind role="stop" component="txt_viv3_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStartN">
        <bind role="onSelection" component="back3" />
        <linkParam name="keyCode" value="BACK"/>
        <bind role="start" component="loading" />
        <bind role="start" component="txt_infoprin" />
        <bind role="start" component="txt_descParque" />
        <bind role="start" component="img_inicial" />
        <bind role="start" component="bot_redprin" />
        <bind role="start" component="bot_greenprin" />
        <bind role="start" component="bot_yellowprin" />
        <bind role="start" component="bot_blueprin" />
        <bind role="start" component="fondo" />
        <bind role="start" component="exit" />
        <bind role="start" component="utpl_prin" />
    </link>
    <!-- Presentación BOTON ROJO VIV-->
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botredviv" />
        <linkParam name="keyCode" value="RED"/>
        <bind role="stop" component="vistaimg3_2_1" />
        <bind role="stop" component="txt_viv3_2_1" />
        <bind role="stop" component="vistaimg3_2_2" />
        <bind role="stop" component="txt_viv3_2_2" />
        <bind role="stop" component="vistaimg3_2_3" />
        <bind role="stop" component="txt_viv3_2_3" />
        <bind role="stop" component="vistaimg3_2_4" />
        <bind role="stop" component="txt_viv3_2_4" />
        <bind role="stop" component="vistaimg3_2_5" />
        <bind role="stop" component="txt_viv3_2_5" />
        <bind role="stop" component="vistaimg3_3_1" />
        <bind role="stop" component="txt_viv3_3_1" />
    </link>

```

```

        <bind role="stop" component="vistaimg3_3_2" />
        <bind role="stop" component="txt_viv3_3_2" />
        <bind role="stop" component="vistaimg3_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botredviv" />
    <linkParam name="keyCode" value="RED"/>
    <bind role="stop" component="txt_viv3_3_3" />
    <bind role="stop" component="vistaimg3_3_4" />
    <bind role="stop" component="txt_viv3_3_4" />
    <bind role="stop" component="vistaimg3_3_5" />
    <bind role="stop" component="txt_viv3_3_5" />
    <bind role="stop" component="vistaimg3_4_1" />
    <bind role="stop" component="txt_viv3_4_1" />
    <bind role="stop" component="vistaimg3_4_2" />
    <bind role="stop" component="txt_viv3_4_2" />
    <bind role="stop" component="vistaimg3_4_3" />
    <bind role="stop" component="txt_viv3_4_3" />
    <bind role="stop" component="vistaimg3_4_4" />
    <bind role="stop" component="txt_viv3_4_4" />
    <bind role="stop" component="vistaimg3_4_5" />
    <bind role="stop" component="txt_viv3_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="botredviv" />
    <linkParam name="keyCode" value="RED"/>
    <bind role="stop" component="txt_descviv" />
    <bind role="stop" component="vistaimg3_1_2" />
    <bind role="stop" component="txt_viv3_1_2" />
    <bind role="stop" component="vistaimg3_1_3" />
    <bind role="stop" component="txt_viv3_1_3" />
    <bind role="stop" component="vistaimg3_1_4" />
    <bind role="stop" component="txt_viv3_1_4" />
    <bind role="stop" component="vistaimg3_1_5" />
    <bind role="stop" component="txt_viv3_1_5" />
    <bind role="start" component="vistaimg3_1_1" />
    <bind role="start" component="txt_viv3_1_1" />
    <bind role="start" component="flech_der3" />
    <bind role="start" component="flech_izq3" />
    <bind role="start" component="fondo_txt_viv" />
</link>
<!--Navegacion imagenes de opcion ROJO en VIV TECLEADO DERECHA-->
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_1_1" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_1_1" />
    <bind role="stop" component="vistaimg3_1_1" />
    <bind role="start" component="vistaimg3_1_2" />
    <bind role="start" component="txt_viv3_1_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_1_2" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_1_2" />
    <bind role="stop" component="vistaimg3_1_2" />

```

```

        <bind role="start" component="vistaimg3_1_3" />
        <bind role="start" component="txt_viv3_1_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_1_3" />
        <bind role="stop" component="vistaimg3_1_3" />
        <bind role="start" component="vistaimg3_1_4" />
        <bind role="start" component="txt_viv3_1_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_1_4" />
        <bind role="stop" component="vistaimg3_1_4" />
        <bind role="start" component="vistaimg3_1_5" />
        <bind role="start" component="txt_viv3_1_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_1_5" />
        <bind role="stop" component="vistaimg3_1_5" />
        <bind role="start" component="vistaimg3_1_1" />
        <bind role="start" component="txt_viv3_1_1" />
    </link>
    <!--Navegacion imagenes de opcion ROJO en VIV TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_1_1" />
        <bind role="stop" component="vistaimg3_1_1" />
        <bind role="start" component="vistaimg3_1_5" />
        <bind role="start" component="txt_viv3_1_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_1_2" />
        <bind role="stop" component="vistaimg3_1_2" />
        <bind role="start" component="vistaimg3_1_1" />
        <bind role="start" component="txt_viv3_1_1" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_3" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_1_3" />
        <bind role="stop" component="vistaimg3_1_3" />
        <bind role="start" component="vistaimg3_1_2" />
        <bind role="start" component="txt_viv3_1_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_1_4" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>

```

```

        <bind role="stop"    component="txt_viv3_1_4" />
        <bind role="stop"    component="vistaimg3_1_4" />
        <bind role="start"   component="vistaimg3_1_3" />
        <bind role="start"   component="txt_viv3_1_3" />
    </link>
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_1_5" />
    <linkParam name="keyCode" value="CURSOR_LEFT"/>
    <bind role="stop"    component="txt_viv3_1_5" />
    <bind role="stop"    component="vistaimg3_1_5" />
    <bind role="start"   component="vistaimg3_1_4" />
    <bind role="start"   component="txt_viv3_1_4" />
</link>
<!-- Presentación BOTON VERDE VIV-->
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botgreenviv" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="stop" component="vistaimg3_1_1" />
    <bind role="stop" component="txt_viv3_1_1" />
    <bind role="stop" component="vistaimg3_1_2" />
    <bind role="stop" component="txt_viv3_1_2" />
    <bind role="stop" component="vistaimg3_1_3" />
    <bind role="stop" component="txt_viv3_1_3" />
    <bind role="stop" component="vistaimg3_1_4" />
    <bind role="stop" component="txt_viv3_1_4" />
    <bind role="stop" component="vistaimg3_1_5" />
    <bind role="stop" component="txt_viv3_1_5" />
    <bind role="stop" component="vistaimg3_3_1" />
    <bind role="stop" component="txt_viv3_3_1" />
    <bind role="stop" component="vistaimg3_3_2" />
    <bind role="stop" component="txt_viv3_3_2" />
    <bind role="stop" component="vistaimg3_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
    <bind role="onSelection" component="botgreenviv" />
    <linkParam name="keyCode" value="GREEN"/>
    <bind role="stop" component="txt_viv3_3_3" />
    <bind role="stop" component="vistaimg3_3_4" />
    <bind role="stop" component="txt_viv3_3_4" />
    <bind role="stop" component="vistaimg3_3_5" />
    <bind role="stop" component="txt_viv3_3_5" />
    <bind role="stop" component="vistaimg3_4_1" />
    <bind role="stop" component="txt_viv3_4_1" />
    <bind role="stop" component="vistaimg3_4_2" />
    <bind role="stop" component="txt_viv3_4_2" />
    <bind role="stop" component="vistaimg3_4_3" />
    <bind role="stop" component="txt_viv3_4_3" />
    <bind role="stop" component="vistaimg3_4_4" />
    <bind role="stop" component="txt_viv3_4_4" />
    <bind role="stop" component="vistaimg3_4_5" />
    <bind role="stop" component="txt_viv3_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="botgreenviv" />
    <linkParam name="keyCode" value="GREEN"/>

```

```

        <bind role="stop" component="txt_descviv" />
        <bind role="stop" component="vistaimg3_2_2" />
        <bind role="stop" component="txt_viv3_2_2" />
        <bind role="stop" component="vistaimg3_2_3" />
        <bind role="stop" component="txt_viv3_2_3" />
        <bind role="stop" component="vistaimg3_2_4" />
        <bind role="stop" component="txt_viv3_2_4" />
        <bind role="stop" component="vistaimg3_2_5" />
        <bind role="stop" component="txt_viv3_2_5" />
        <bind role="start" component="vistaimg3_2_1" />
        <bind role="start" component="txt_viv3_2_1" />
        <bind role="start" component="flech_der3" />
        <bind role="start" component="flech_izq3" />
        <bind role="start" component="fondo_txt_viv" />
</link>
<!--Navegacion imagenes de opcion VERDE en VIV TECLEADO DERECHA-->
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_2_1" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_2_1" />
    <bind role="stop" component="vistaimg3_2_1" />
    <bind role="start" component="vistaimg3_2_2" />
    <bind role="start" component="txt_viv3_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_2_2" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_2_2" />
    <bind role="stop" component="vistaimg3_2_2" />
    <bind role="start" component="vistaimg3_2_3" />
    <bind role="start" component="txt_viv3_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_2_3" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_2_3" />
    <bind role="stop" component="vistaimg3_2_3" />
    <bind role="start" component="vistaimg3_2_4" />
    <bind role="start" component="txt_viv3_2_4" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_2_4" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_2_4" />
    <bind role="stop" component="vistaimg3_2_4" />
    <bind role="start" component="vistaimg3_2_5" />
    <bind role="start" component="txt_viv3_2_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
    <bind role="onSelection" component="vistaimg3_2_5" />
    <linkParam name="keyCode" value="CURSOR_RIGHT"/>
    <bind role="stop" component="txt_viv3_2_5" />
    <bind role="stop" component="vistaimg3_2_5" />
    <bind role="start" component="vistaimg3_2_1" />
    <bind role="start" component="txt_viv3_2_1" />

```

```

</link>
<!--Navegacion imagenes de opcion VERDE en VIV TECLEADO IZQUIERDA-->
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_2_1" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_2_1" />
  <bind role="stop" component="vistaimg3_2_1" />
  <bind role="start" component="vistaimg3_2_5" />
  <bind role="start" component="txt_viv3_2_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_2_2" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_2_2" />
  <bind role="stop" component="vistaimg3_2_2" />
  <bind role="start" component="vistaimg3_2_1" />
  <bind role="start" component="txt_viv3_2_1" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_2_3" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_2_3" />
  <bind role="stop" component="vistaimg3_2_3" />
  <bind role="start" component="vistaimg3_2_2" />
  <bind role="start" component="txt_viv3_2_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_2_4" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_2_4" />
  <bind role="stop" component="vistaimg3_2_4" />
  <bind role="start" component="vistaimg3_2_3" />
  <bind role="start" component="txt_viv3_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_2_5" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_2_5" />
  <bind role="stop" component="vistaimg3_2_5" />
  <bind role="start" component="vistaimg3_2_4" />
  <bind role="start" component="txt_viv3_2_4" />
</link>
<!-- Presentación BOTON AMARILLO VIV-->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botyellowviv" />
  <linkParam name="keyCode" value="YELLOW"/>
  <bind role="stop" component="vistaimg3_1_1" />
  <bind role="stop" component="txt_viv3_1_1" />
  <bind role="stop" component="vistaimg3_1_2" />
  <bind role="stop" component="txt_viv3_1_2" />
  <bind role="stop" component="vistaimg3_1_3" />
  <bind role="stop" component="txt_viv3_1_3" />
  <bind role="stop" component="vistaimg3_1_4" />
  <bind role="stop" component="txt_viv3_1_4" />
  <bind role="stop" component="vistaimg3_1_5" />

```



```

        <bind role="stop" component="txt_viv3_1_5" />
        <bind role="stop" component="vistaimg3_2_1" />
        <bind role="stop" component="txt_viv3_2_1" />
        <bind role="stop" component="vistaimg3_2_2" />
        <bind role="stop" component="txt_viv3_2_2" />
        <bind role="stop" component="vistaimg3_2_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopN">
        <bind role="onSelection" component="botyellowviv" />
        <linkParam name="keyCode" value="YELLOW"/>
        <bind role="stop" component="txt_viv3_2_3" />
        <bind role="stop" component="vistaimg3_2_4" />
        <bind role="stop" component="txt_viv3_2_4" />
        <bind role="stop" component="vistaimg3_2_5" />
        <bind role="stop" component="txt_viv3_2_5" />
        <bind role="stop" component="vistaimg3_4_1" />
        <bind role="stop" component="txt_viv3_4_1" />
        <bind role="stop" component="vistaimg3_4_2" />
        <bind role="stop" component="txt_viv3_4_2" />
        <bind role="stop" component="vistaimg3_4_3" />
        <bind role="stop" component="txt_viv3_4_3" />
        <bind role="stop" component="vistaimg3_4_4" />
        <bind role="stop" component="txt_viv3_4_4" />
        <bind role="stop" component="vistaimg3_4_5" />
        <bind role="stop" component="txt_viv3_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="botyellowviv" />
        <linkParam name="keyCode" value="YELLOW"/>
        <bind role="stop" component="txt_descviv" />
        <bind role="stop" component="vistaimg3_3_2" />
        <bind role="stop" component="txt_viv3_3_2" />
        <bind role="stop" component="vistaimg3_3_3" />
        <bind role="stop" component="txt_viv3_3_3" />
        <bind role="stop" component="vistaimg3_3_4" />
        <bind role="stop" component="txt_viv3_3_4" />
        <bind role="stop" component="vistaimg3_3_5" />
        <bind role="stop" component="txt_viv3_3_5" />
        <bind role="start" component="vistaimg3_3_1" />
        <bind role="start" component="txt_viv3_3_1" />
        <bind role="start" component="flech_der3" />
        <bind role="start" component="flech_izq3" />
        <bind role="start" component="fondo_txt_viv" />
    </link>
    <!--Navegacion imagenes de opcion AMARILLO en VIV TECLEADO DERECHA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_1" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_3_1" />
        <bind role="stop" component="vistaimg3_3_1" />
        <bind role="start" component="vistaimg3_3_2" />
        <bind role="start" component="txt_viv3_3_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_2" />

```

```

        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_3_2" />
        <bind role="stop" component="vistaimg3_3_2" />
        <bind role="start" component="vistaimg3_3_3" />
        <bind role="start" component="txt_viv3_3_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_3" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_3_3" />
        <bind role="stop" component="vistaimg3_3_3" />
        <bind role="start" component="vistaimg3_3_4" />
        <bind role="start" component="txt_viv3_3_4" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_4" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_3_4" />
        <bind role="stop" component="vistaimg3_3_4" />
        <bind role="start" component="vistaimg3_3_5" />
        <bind role="start" component="txt_viv3_3_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_5" />
        <linkParam name="keyCode" value="CURSOR_RIGHT"/>
        <bind role="stop" component="txt_viv3_3_5" />
        <bind role="stop" component="vistaimg3_3_5" />
        <bind role="start" component="vistaimg3_3_1" />
        <bind role="start" component="txt_viv3_3_1" />
    </link>
    <!--Navegacion imagenes de opcion AMARILLO en ORQ TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_3_1" />
        <bind role="stop" component="vistaimg3_3_1" />
        <bind role="start" component="vistaimg3_3_5" />
        <bind role="start" component="txt_viv3_3_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_3_2" />
        <bind role="stop" component="vistaimg3_3_2" />
        <bind role="start" component="vistaimg3_3_1" />
        <bind role="start" component="txt_viv3_3_1" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_3_3" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_3_3" />
        <bind role="stop" component="vistaimg3_3_3" />
        <bind role="start" component="vistaimg3_3_2" />
        <bind role="start" component="txt_viv3_3_2" />
    </link>

```



```

<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_3_4" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_3_4" />
  <bind role="stop" component="vistaimg3_3_4" />
  <bind role="start" component="vistaimg3_3_3" />
  <bind role="start" component="txt_viv3_3_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_3_5" />
  <linkParam name="keyCode" value="CURSOR_LEFT"/>
  <bind role="stop" component="txt_viv3_3_5" />
  <bind role="stop" component="vistaimg3_3_5" />
  <bind role="start" component="vistaimg3_3_4" />
  <bind role="start" component="txt_viv3_3_4" />
</link>
<!-- Presentación BOTON AZUL VIV-->
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botblueviv" />
  <linkParam name="keyCode" value="BLUE"/>
  <bind role="stop" component="vistaimg3_1_1" />
  <bind role="stop" component="txt_viv3_1_1" />
  <bind role="stop" component="vistaimg3_1_2" />
  <bind role="stop" component="txt_viv3_1_2" />
  <bind role="stop" component="vistaimg3_1_3" />
  <bind role="stop" component="txt_viv3_1_3" />
  <bind role="stop" component="vistaimg3_1_4" />
  <bind role="stop" component="txt_viv3_1_4" />
  <bind role="stop" component="vistaimg3_1_5" />
  <bind role="stop" component="txt_viv3_1_5" />
  <bind role="stop" component="vistaimg3_2_1" />
  <bind role="stop" component="txt_viv3_2_1" />
  <bind role="stop" component="vistaimg3_2_2" />
  <bind role="stop" component="txt_viv3_2_2" />
  <bind role="stop" component="vistaimg3_2_3" />
</link>
<link xconnector="conector#onKeySelectionStopN">
  <bind role="onSelection" component="botblueviv" />
  <linkParam name="keyCode" value="BLUE"/>
  <bind role="stop" component="txt_viv3_2_3" />
  <bind role="stop" component="vistaimg3_2_4" />
  <bind role="stop" component="txt_viv3_2_4" />
  <bind role="stop" component="vistaimg3_2_5" />
  <bind role="stop" component="txt_viv3_2_5" />
  <bind role="stop" component="vistaimg3_3_1" />
  <bind role="stop" component="txt_viv3_3_1" />
  <bind role="stop" component="vistaimg3_3_2" />
  <bind role="stop" component="txt_viv3_3_2" />
  <bind role="stop" component="vistaimg3_3_3" />
  <bind role="stop" component="txt_viv3_3_3" />
  <bind role="stop" component="vistaimg3_3_4" />
  <bind role="stop" component="txt_viv3_3_4" />
  <bind role="stop" component="vistaimg3_3_5" />
  <bind role="stop" component="txt_viv3_3_5" />
</link>

```

```

<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="botblueviv" />
  <linkParam name="keyCode" value="BLUE"/>
  <bind role="stop" component="txt_descviv" />
  <bind role="stop" component="vistaimg3_4_2" />
  <bind role="stop" component="txt_viv3_4_2" />
  <bind role="stop" component="vistaimg3_4_3" />
  <bind role="stop" component="txt_viv3_4_3" />
  <bind role="stop" component="vistaimg3_4_4" />
  <bind role="stop" component="txt_viv3_4_4" />
  <bind role="stop" component="vistaimg3_4_5" />
  <bind role="stop" component="txt_viv3_4_5" />
  <bind role="start" component="vistaimg3_4_1" />
  <bind role="start" component="txt_viv3_4_1" />
  <bind role="start" component="flech_der3" />
  <bind role="start" component="flech_izq3" />
  <bind role="start" component="fondo_txt_viv" />
</link>
<!--Navegacion imagenes de opcion AZUL en VIV TECLEADO DERECHA-->
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_4_1" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_viv3_4_1" />
  <bind role="stop" component="vistaimg3_4_1" />
  <bind role="start" component="vistaimg3_4_2" />
  <bind role="start" component="txt_viv3_4_2" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_4_2" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_viv3_4_2" />
  <bind role="stop" component="vistaimg3_4_2" />
  <bind role="start" component="vistaimg3_4_3" />
  <bind role="start" component="txt_viv3_4_3" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_4_3" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_viv3_4_3" />
  <bind role="stop" component="vistaimg3_4_3" />
  <bind role="start" component="vistaimg3_4_4" />
  <bind role="start" component="txt_viv3_4_4" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_4_4" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_viv3_4_4" />
  <bind role="stop" component="vistaimg3_4_4" />
  <bind role="start" component="vistaimg3_4_5" />
  <bind role="start" component="txt_viv3_4_5" />
</link>
<link xconnector="conector#onKeySelectionStopNStartN">
  <bind role="onSelection" component="vistaimg3_4_5" />
  <linkParam name="keyCode" value="CURSOR_RIGHT"/>
  <bind role="stop" component="txt_viv3_4_5" />

```

```

        <bind role="stop" component="vistaimg3_4_5" />
        <bind role="start" component="vistaimg3_4_1" />
        <bind role="start" component="txt_viv3_4_1" />
    </link>
    <!--Navegacion imagenes de opcion AZUL en VIV TECLEADO IZQUIERDA-->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_4_1" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_4_1" />
        <bind role="stop" component="vistaimg3_4_1" />
        <bind role="start" component="vistaimg3_4_5" />
        <bind role="start" component="txt_viv3_4_5" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_4_2" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_4_2" />
        <bind role="stop" component="vistaimg3_4_2" />
        <bind role="start" component="vistaimg3_4_1" />
        <bind role="start" component="txt_viv3_4_1" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_4_3" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_4_3" />
        <bind role="stop" component="vistaimg3_4_3" />
        <bind role="start" component="vistaimg3_4_2" />
        <bind role="start" component="txt_viv3_4_2" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_4_4" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_4_4" />
        <bind role="stop" component="vistaimg3_4_4" />
        <bind role="start" component="vistaimg3_4_3" />
        <bind role="start" component="txt_viv3_4_3" />
    </link>
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="vistaimg3_4_5" />
        <linkParam name="keyCode" value="CURSOR_LEFT"/>
        <bind role="stop" component="txt_viv3_4_5" />
        <bind role="stop" component="vistaimg3_4_5" />
        <bind role="start" component="vistaimg3_4_4" />
        <bind role="start" component="txt_viv3_4_4" />
    </link>
    <!-- Presentación BOTON AZUL -->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="bot_blueprin" />
        <linkParam name="keyCode" value="BLUE"/>
        <bind role="stop" component="fondo" />
        <bind role="stop" component="txt_infoprin" />
        <bind role="stop" component="exit" />
        <bind role="stop" component="utpl_prin" />
        <bind role="stop" component="txt_descParque" />
        <bind role="stop" component="img_inicial" />

```

```

        <bind role="stop" component="bot_redprin" />
        <bind role="stop" component="bot_greenprin" />
        <bind role="stop" component="bot_yellowprin" />
        <bind role="stop" component="bot_blueprin" />
        <bind role="start" component="loadingmap" />
        <bind role="start" component="img_mapa" />
        <bind role="start" component="txt_mapa" />
        <bind role="start" component="txt_mapa1" />
        <bind role="start" component="fondo_mapa" />
        <bind role="start" component="back4" />
        <bind role="start" component="utpl_mapa" />
    </link>
    <!-- FIN Presentación BOTON BLUE -->
    <link xconnector="conector#onKeySelectionStopNStartN">
        <bind role="onSelection" component="back4" />
        <linkParam name="keyCode" value="BACK"/>
        <bind role="stop" component="back4" />
        <bind role="stop" component="img_mapa" />
        <bind role="stop" component="fondo_mapa" />
        <bind role="stop" component="txt_mapa" />
        <bind role="stop" component="txt_mapa1" />
        <bind role="stop" component="utpl_mapa" />
        <bind role="start" component="utpl_prin" />
        <bind role="start" component="loading" />
        <bind role="start" component="txt_infoprin" />
        <bind role="start" component="txt_descParque" />
        <bind role="start" component="img_inicial" />
        <bind role="start" component="bot_redprin" />
        <bind role="start" component="bot_greenprin" />
        <bind role="start" component="bot_yellowprin" />
        <bind role="start" component="bot_blueprin" />
        <bind role="start" component="fondo" />
        <bind role="start" component="exit" />
    </link>
</body>
</ncl>

```

ANEXO C

***DISEÑO DEL LAYOUT DE PANTALLA Y
VISUALIZACIÓN GRÁFICA DEL CÓDIGO COMPLETO
DE LA APLICACIÓN INTERACTIVA MEDIANTE NCL
COMPOSER***

En este anexo se detalla el código en forma gráfica utilizando NCL COMPOSER.

Hay que tener en cuenta que el código es bastante extenso por lo que existirán varias imágenes que representan cada media y cada enlace del programa principal y no será posible obtener una buena visibilidad del mismo, es por esto que el código está dividido por partes para obtener una mejor visión de lo que hace la aplicación interactiva. Además de esto se presentan las imágenes del uso principal de esta herramienta, esto es el diseño del layout de pantalla, es decir, donde se presentarán cada imagen y cada botón de la aplicación interactiva.

Los diseños de cada layout de pantalla y cada parte de código programado en NCL Eclipse y reflejado en NCL Composer, se presenta a continuación:

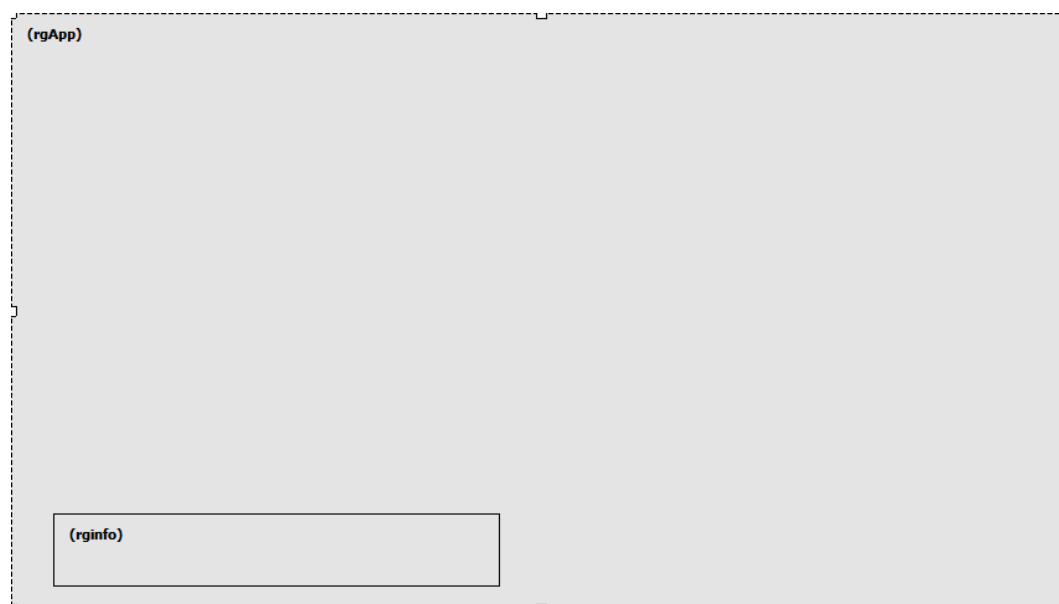


Figura 1. Layout de pantalla de presentación del botón INFO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

En la figura anterior se presenta el layout de pantalla en donde se especifica la región donde se presentará el botón INFO para acceso a la aplicación interactiva, ésta solo se activará si el usuario interactúa mediante su control remoto.

A continuación se presenta mediante la visualización estructural de NCL Composer el código gráfico de lo que realiza el botón INFO si es que éste es presionado.

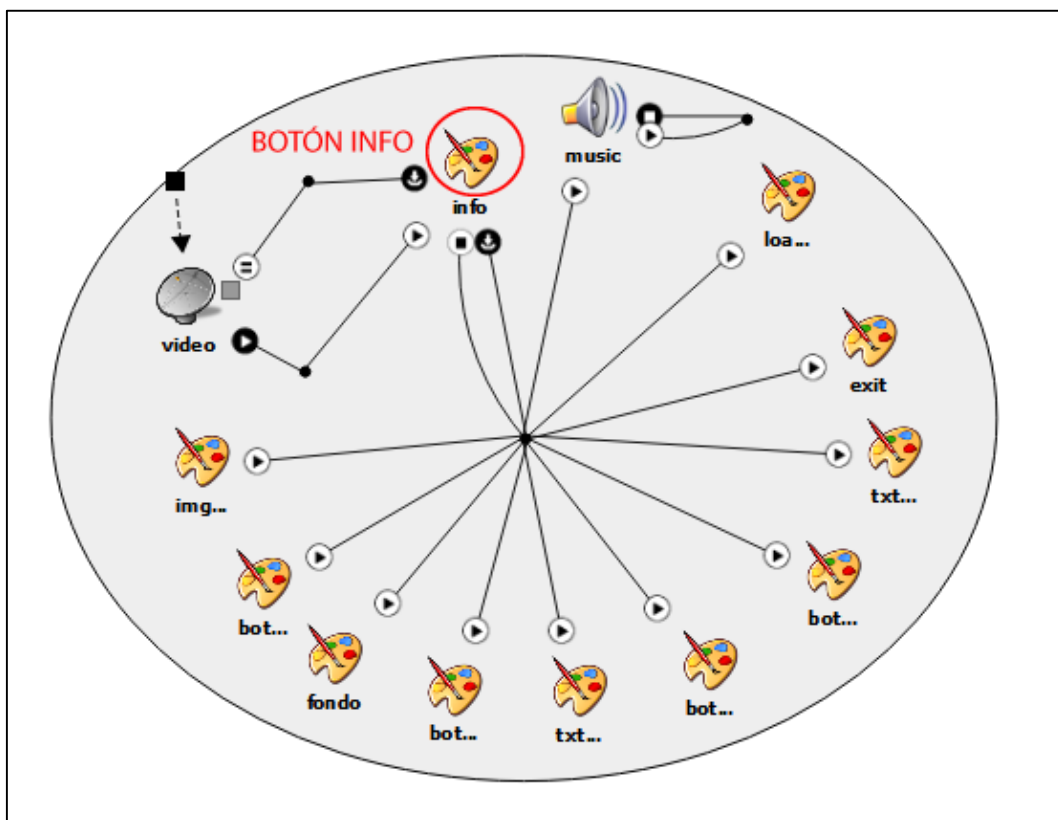


Figura 2. Programación gráfica de las acciones que realiza el botón INFO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

Los iconos que se encuentran de color negro indican que si se presionan esos botones, inmediatamente se activan las acciones que se encuentran especificadas en cada icono de color blanco para cada media.

Una vez que inicia el vídeo (señal de televisión digital) inicia luego de cinco segundos el botón INFO y si éste se presiona, se envía a silenciar el audio del vídeo, se detiene el botón INFO y se presentan todos los archivos medias que se encuentran especificados en el código, el archivo de audio indica que cada vez que éste termina se vuelve a reiniciar.

En la figura 3 se muestra la visión de diseño de layout del acceso del botón INFO, en este se presentan las diferentes regiones utilizadas para presentar el fondo de la aplicación, algunas imágenes, una descripción del Parque y la ubicación de los diferentes botones de acceso y de salida de la aplicación.

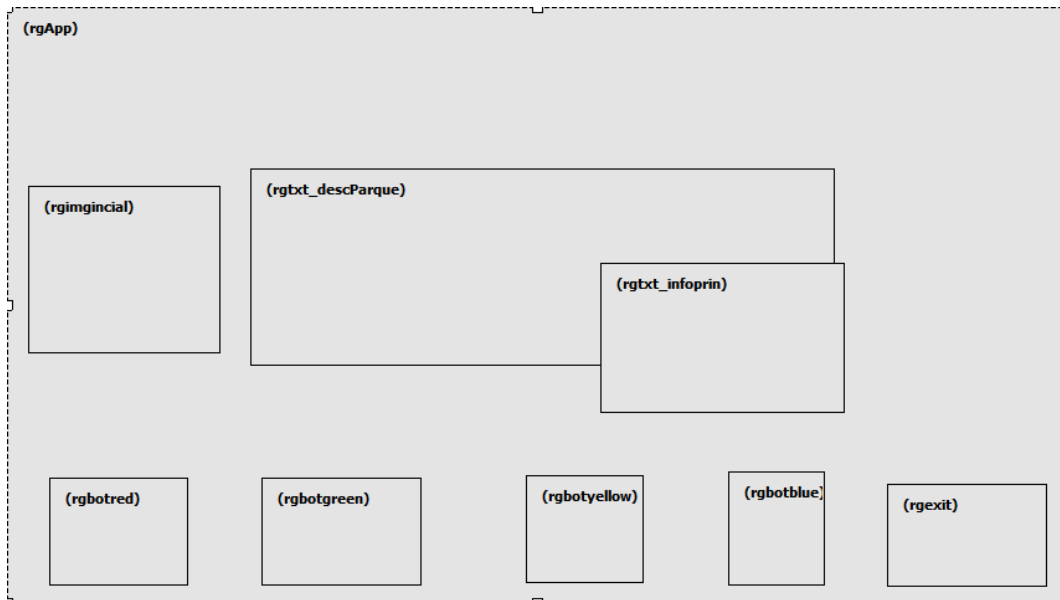


Figura 3. Layout de pantalla de la aplicación cuando se ha presionado el botón INFO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

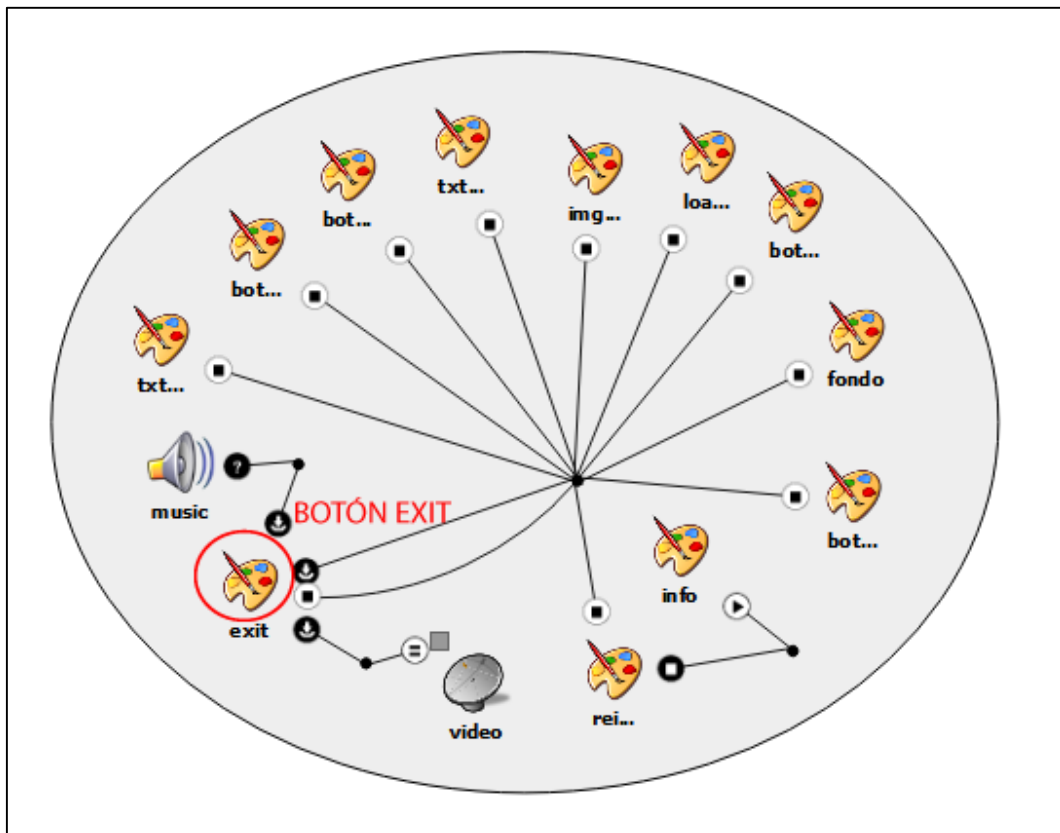


Figura 4. Activación del botón EXIT para salir de la aplicación y reiniciar el botón INFO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

En la figura 4 se presenta el código generado por NCL Composer para salir de la aplicación y luego reiniciar la misma, aquí se puede observar que si se presiona el botón EXIT el audio se detiene, el audio del vídeo vuelve a reproducirse, el botón EXIT y todos los demás medias se detienen. Una vez que esto se haya realizado en la imagen “reinicio_info” existe un enlace que especifica que si éste media se detiene, luego de cinco minutos el botón INFO vuelve a aparecer para darnos un nuevo acceso a la aplicación interactiva.

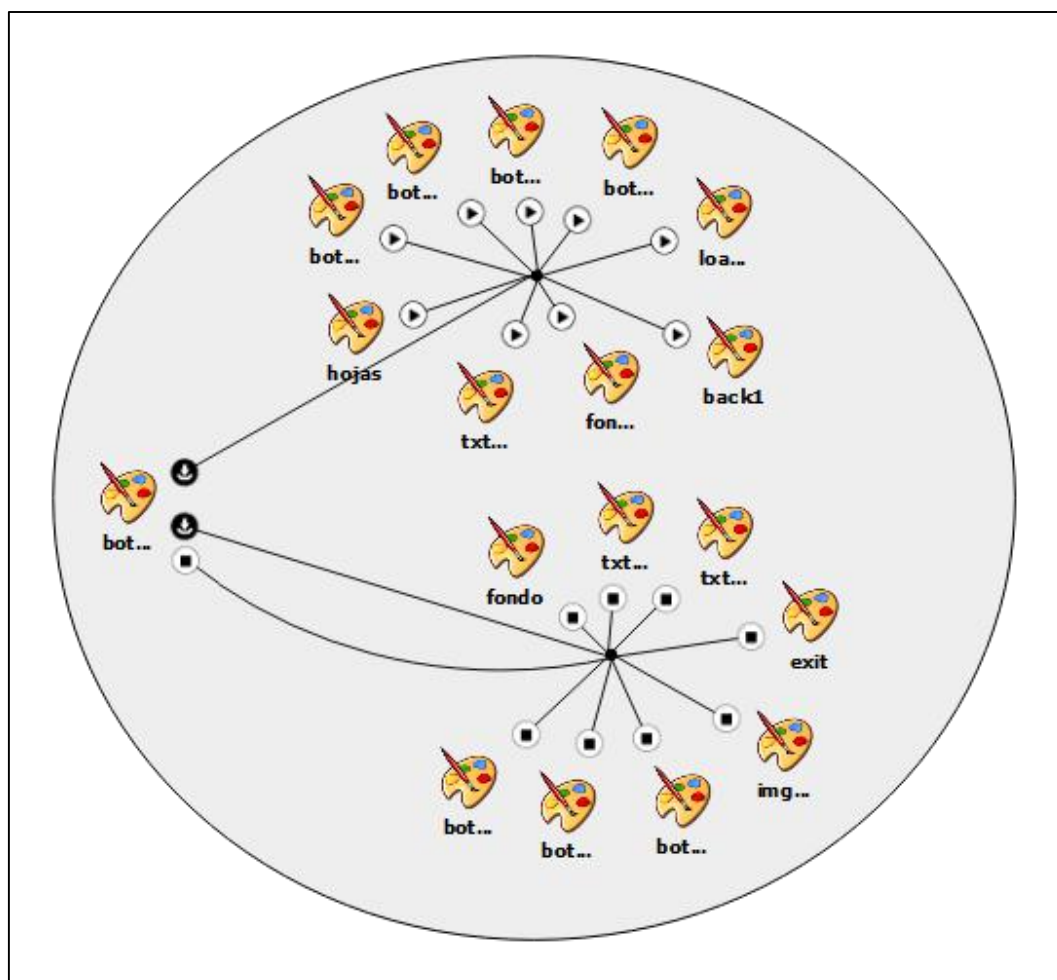


Figura 5. Activación del botón ZOOLOGICO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

En la figura 5 se presenta el acceso al botón ROJO - ZOOLOGICO, los medias que se detienen que generalmente son los del menú anterior y los medias que se presentan son nuevas imágenes y botones de acceso y retorno. Este código es similar para acceder al botón VERDE – ORQUIDEARIO y AMARILLO – VIVERO.

En las siguientes imágenes se presentan los layout de diseño de los tres primeros botones cuando estos son presionados.

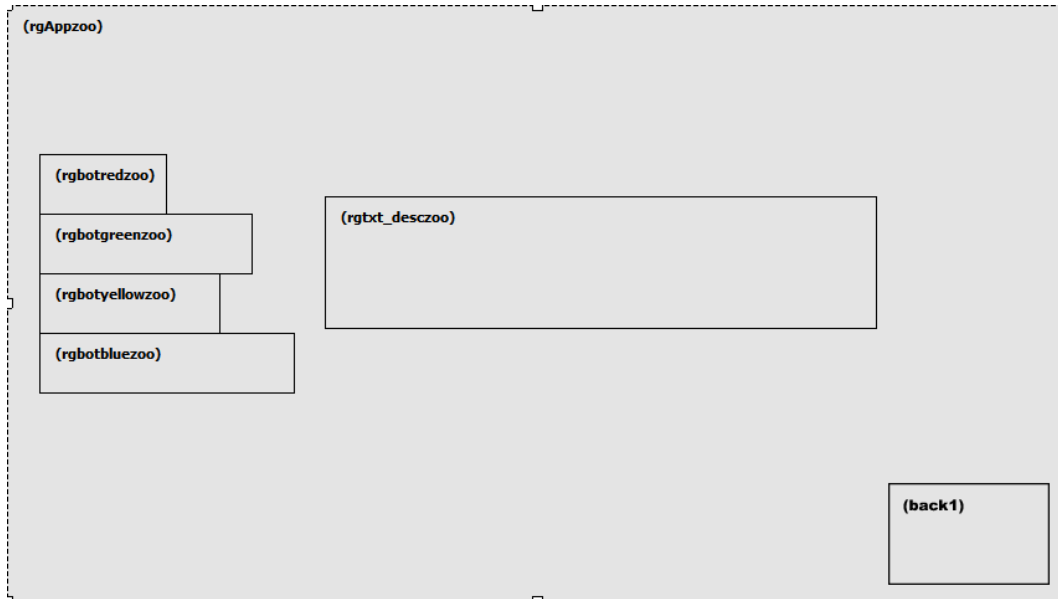


Figura 6. Layout de pantalla de presentación del botón ROJO – ZOOLOGICO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

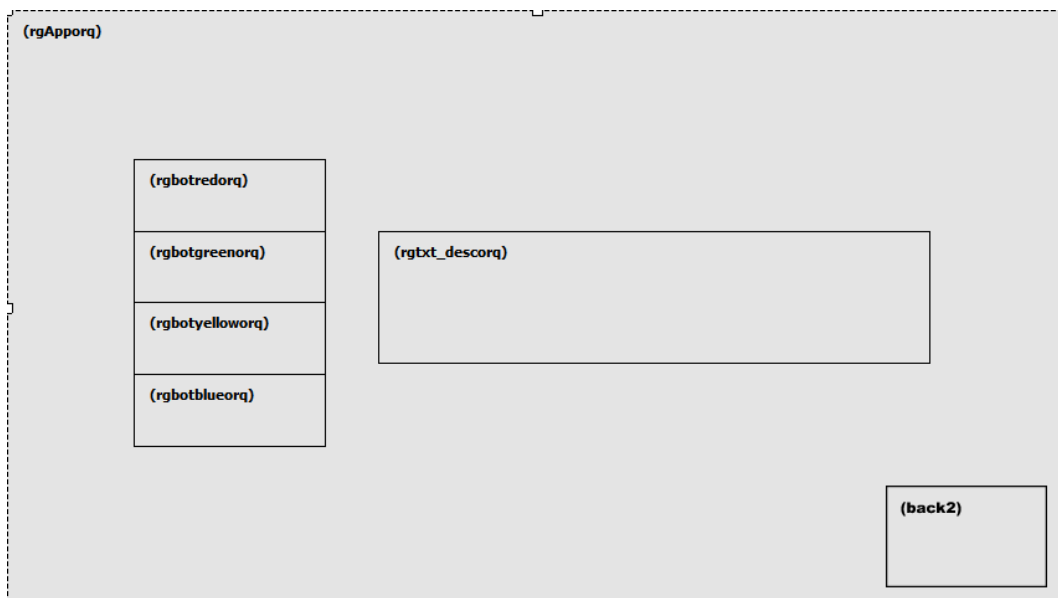


Figura 7. Layout de pantalla de presentación del botón VERDE – ORQUIDEARIO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

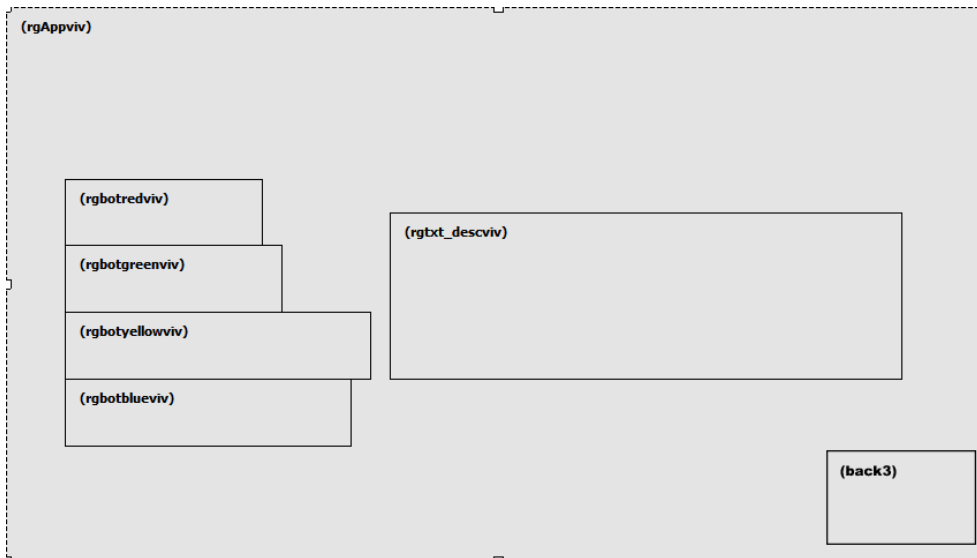


Figura 8. Layout de pantalla de presentación del botón AMARILLO – VIVERO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

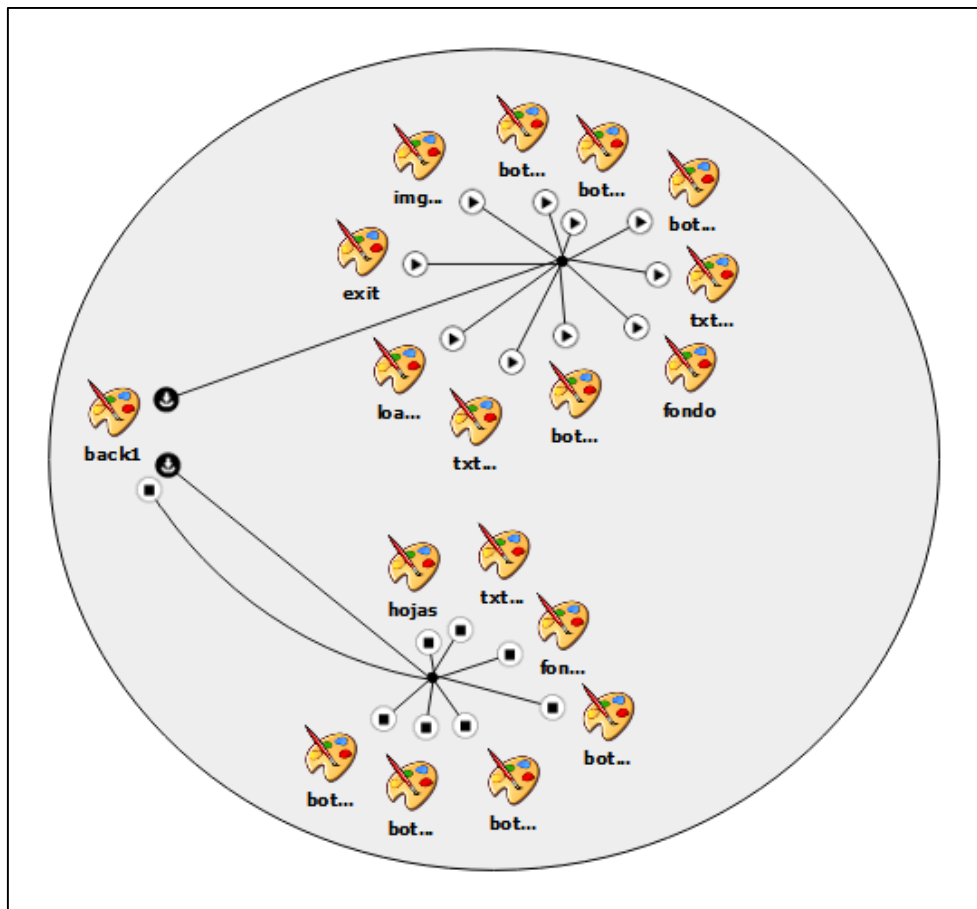


Figura 9. Retorno al menú principal desde la opción BACK del ZOOLOGICO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

En la figura 9 se presenta el retorno al menú principal mediante el botón BACK, los medias que se detienen son los que fueron presentados cuando se ingresa al menú del ZOOLOGÍCO y los medias que se presentan son los medias que dan forma al menú principal. Este código es similar para el retorno al menú principal cuando el usuario se encuentra en el menú del ORQUIDEARIO o el menú del VIVERO.

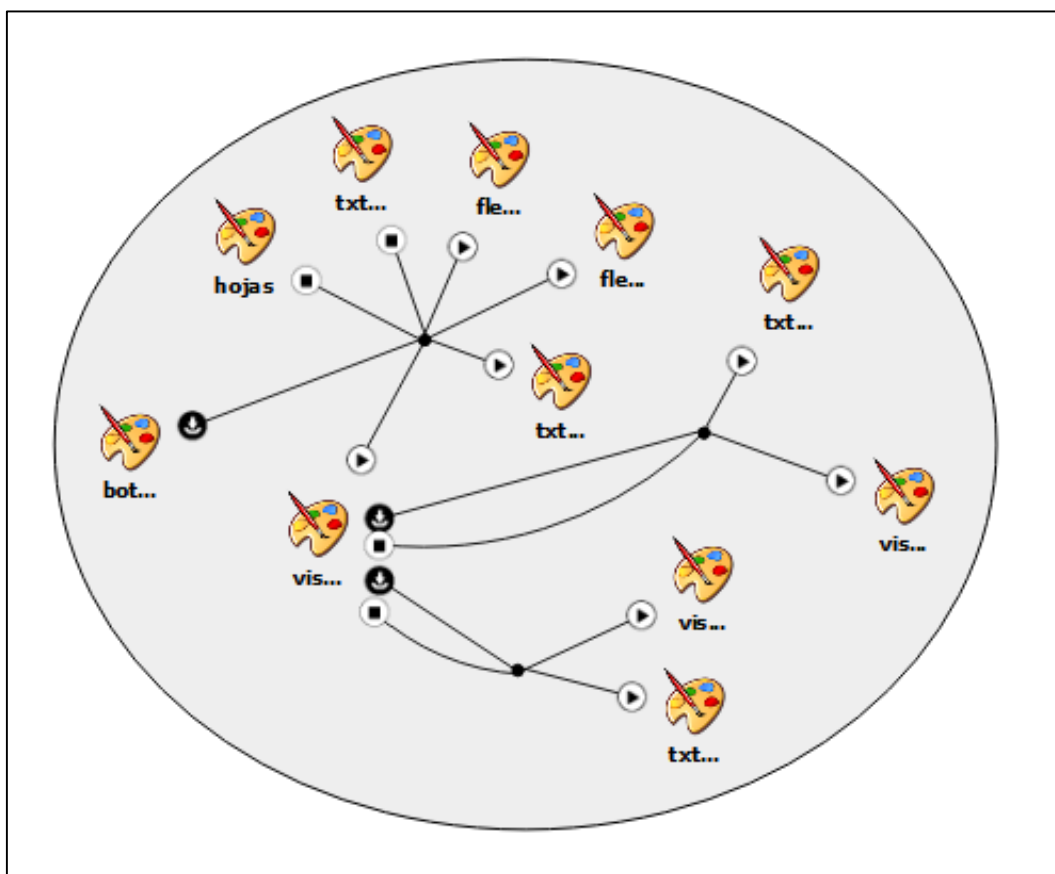


Figura 10. Navegación de las imágenes que se encuentre en la opción ROJO “Aves” del menú ZOOLOGÍCO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

En la figura anterior podemos observar el ingreso a la navegación de las imágenes que muestran las especies de aves que existen en el menú del Zoológico, ésta se activa al presionar el botón ROJO. Éste código es similar para ingresar a cualquier ambiente de navegación de imágenes de cualquier menú en la que el usuario se encuentre. A continuación se presentan los diferentes layout de pantalla en donde se visualizan las diferentes especies animales y vegetales.

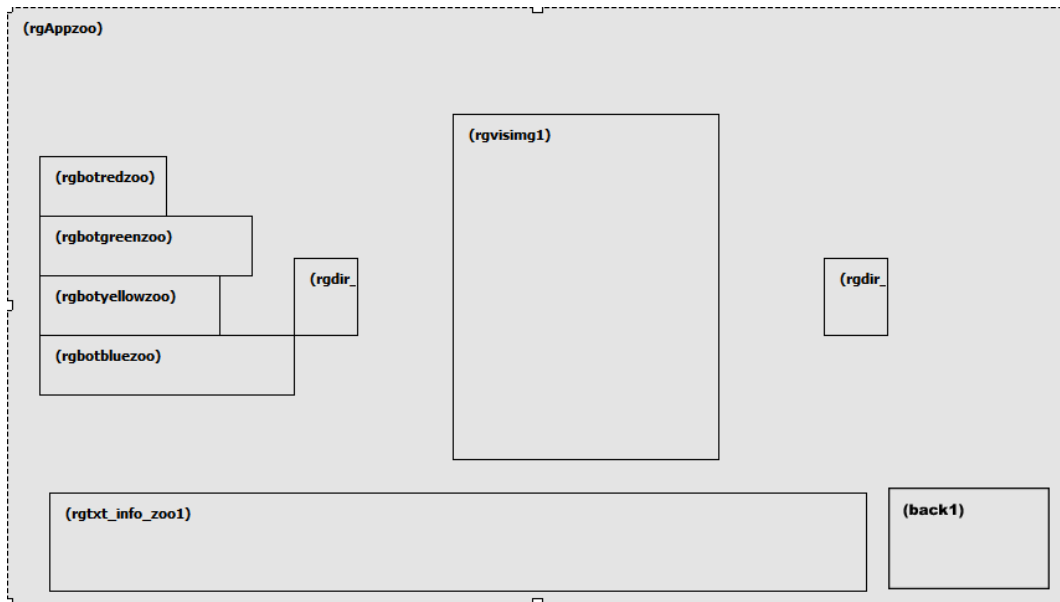


Figura 11. Layout de pantalla de visualización de imágenes dentro de la opción ZOOLOGICO.
 Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

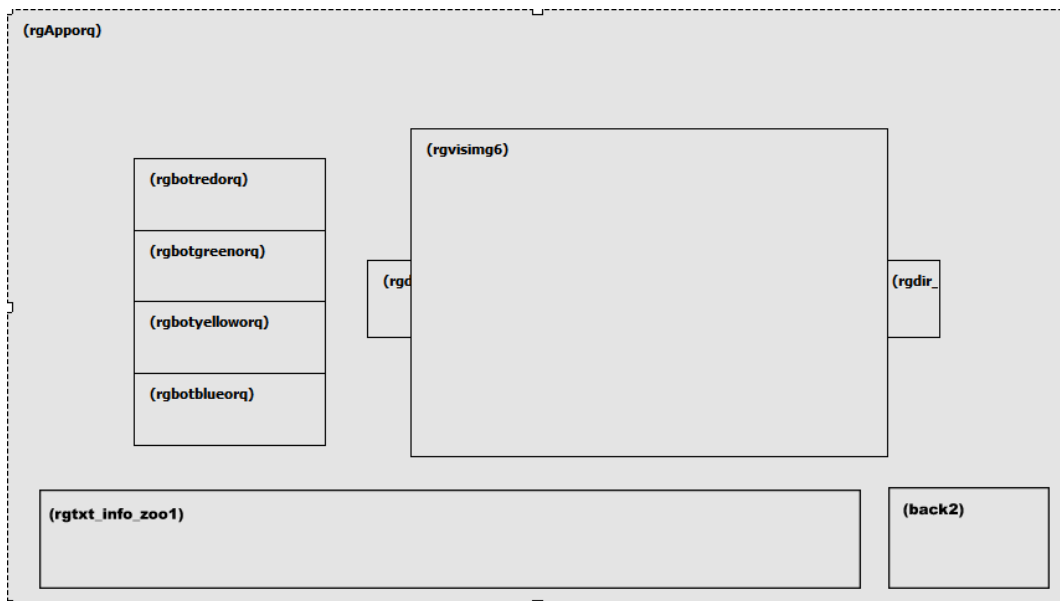


Figura 12. Layout de pantalla de visualización de imágenes dentro de la opción ORQUIDEARIO.
 Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

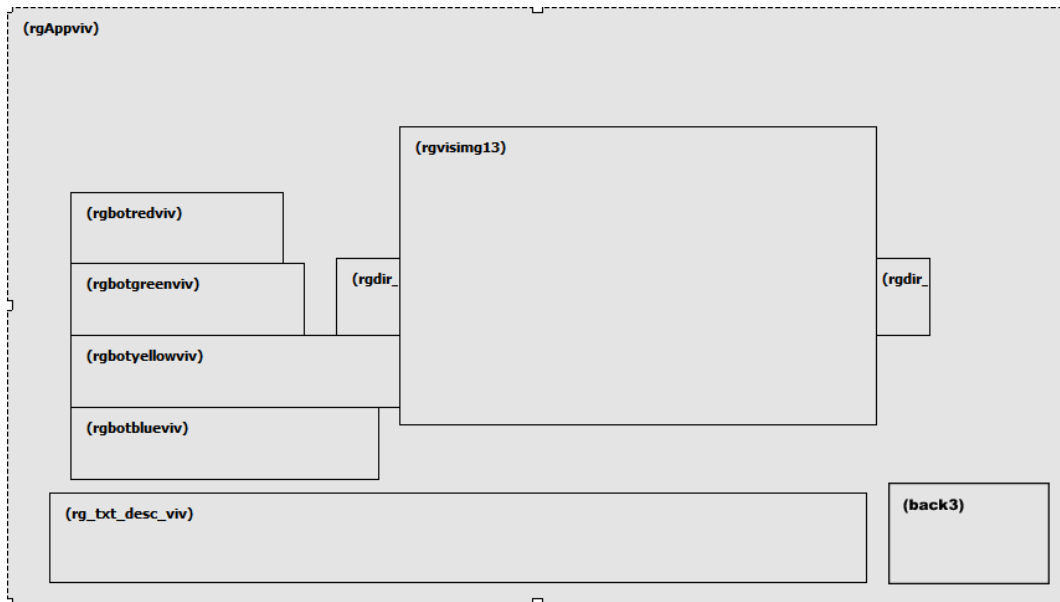


Figura 13. Layout de pantalla de visualización de imágenes dentro de la opción VIVERO.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

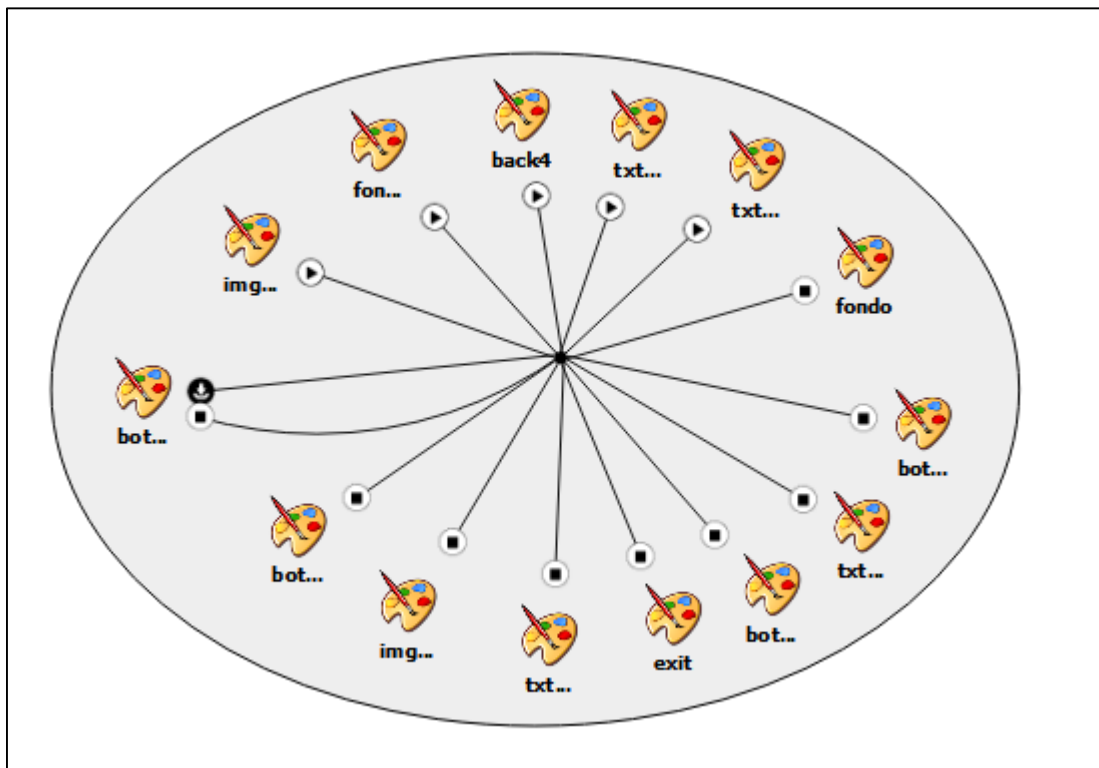


Figura 14. Activación del botón AZUL.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

En la figura anterior se muestra los medias que serán activados cuando se presione el botón AZUL del menú principal y en la siguiente figura se muestra el layout de pantalla cuando se ingrese a éste botón.

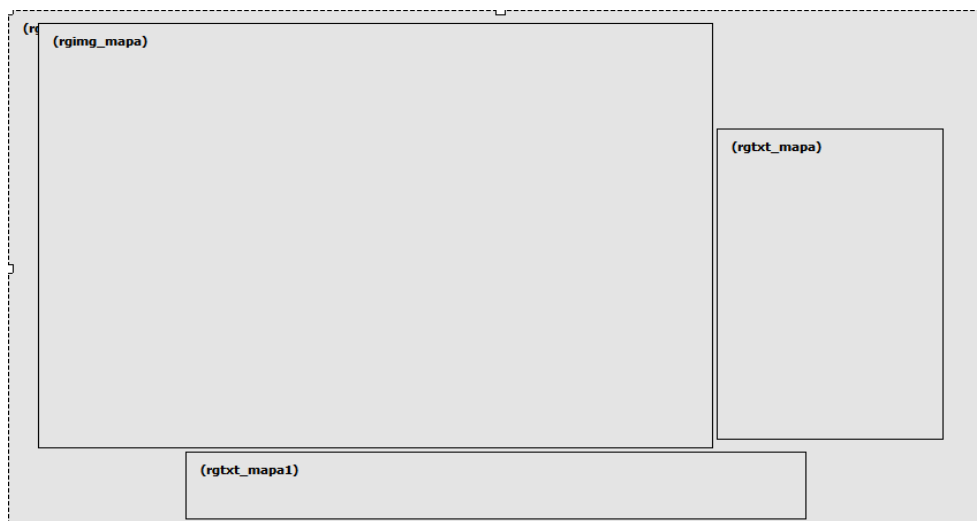


Figura 15. Layout de pantalla de presentación del botón AZUL – CÓMO LLEGAR.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión de diseño.

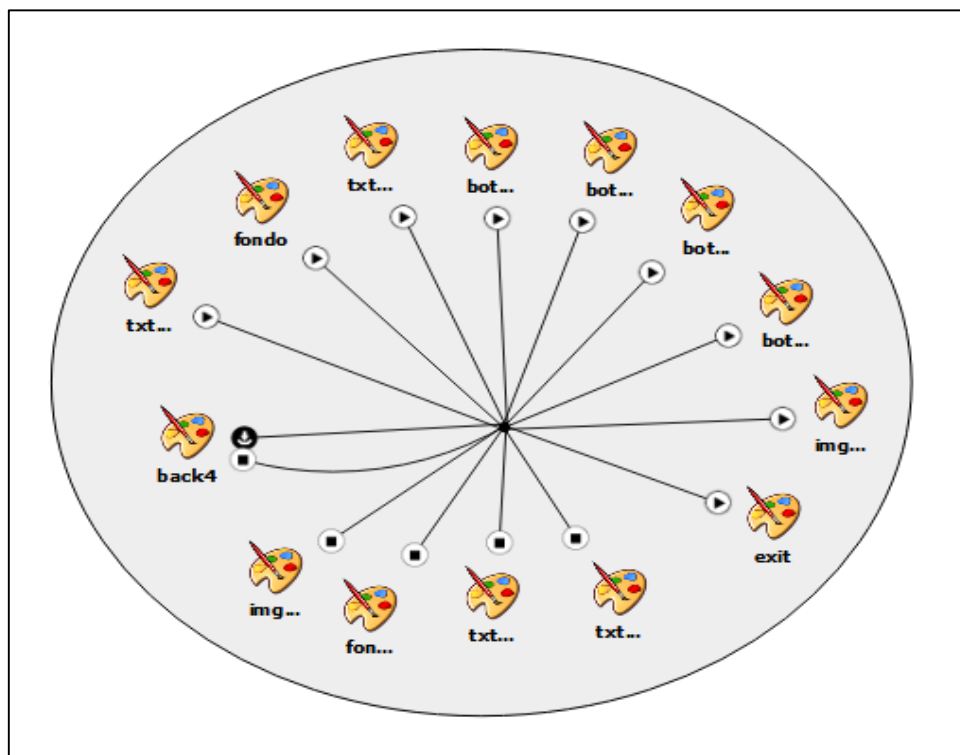


Figura 16. Retorno al menú principal desde la opción BACK de la opción CÓMO LLEGAR.

Fuente: Creado por el autor mediante captura de pantalla en el software NCL Composer, visión estructural.

En la figura anterior se observa el proceso que se realiza luego de presionar el botón BACK desde el ambiente de visualización de la opción CÓMO LLEGAR, se detienen los textos y el mapa de ubicación del parque y se inician todos los archivos media del menú principal.