



# UNIVERSIDAD DEL PAPALOAPAN

---

Campus Loma Bonita

INGENIERÍA EN COMPUTACIÓN

## TESIS

**Desarrollo de una aplicación móvil basada en Android  
para la gestión remota de presentaciones multimedia  
mediante Raspberry Pi**

presentada por  
**Zuleima Villa Martínez**

para obtener el título de  
**Ingeniero en Computación**

Director de tesis  
**M.C. José Domingo Juárez Hernández**



# UNIVERSIDAD DEL PAPALOAPAN

Campus Loma Bonita

INGENIERÍA EN COMPUTACIÓN

La presente tesis titulada **“DESARROLLO DE UNA APLICACIÓN MÓVIL BASADA EN ANDROID PARA LA GESTIÓN REMOTA DE PRESENTACIONES MULTIMEDIA MEDIANTE RASPBERRY PI”**, presentada por la pasante **ZULEIMA VILLA MARTINEZ**, bajo la dirección del M.C. José Domingo Juárez Hernández, ha sido aprobada por el jurado examinador y aceptada como requisito parcial para obtener el título de Ingeniero en Computación.

## JURADO EXAMINADOR

M.C. JOSÉ DOMINGO JUÁREZ HERNÁNDEZ  
DIRECTOR DE TESIS

DR. SERGIO FABIAN RUIZ PAZ  
PRESIDENTE

M.C. ARIEL LÓPEZ RODRÍGUEZ  
SECRETARIO

Loma Bonita, Oaxaca, México. 2019



# Universidad del Papaloapan

FECHA:	11 de Septiembre del 2019
ÁREA:	Vice-Rectoría Académica
OFICIO NÚMERO:	UNPA/VRA/230/2019
ASUNTO:	Autorización de Impresión de tesis.

**C. Zulcima Villa Martínez**  
**PRESENTE:**

En base al artículo 120 del reglamento de alumnos, por medio de la presente se aprueba la impresión de la tesis titulada "Desarrollo de una aplicación móvil basada en Android para la gestión remota de presentaciones multimedia mediante Raspberry Pi" así como la programación del examen profesional bajo la dirección del M.C. José Domingo Juárez Hernández.

Sin más por el momento aprovecho la ocasión para enviarle un cordial saludo.

Atentamente.  
terra ubérrima, mens aperta  
Bou Lo-tama, chi jí jú

  
MC. HÉCTOR LÓPEZ ARJONA  
Vice-Rector Académico.



C.c.p. M.C. Ariel López Rodríguez Jefe de Carrera de la Ing. en Computación  
C.c.p. L.P. Yesenia Barrientos Arenal. Jefa del Departamento de Servicios Escolares  
C.c.p. M.C. José Domingo Juárez Hernández. Director de Tesis.  
C.c.p. Archivo.

# UNIVERSIDAD DEL PAPALOAPAN

Campus Loma Bonita

Oficio: ICOM/09/003/19

Loma Bonita, Oaxaca; a 12 de septiembre de 2019.

**M.E. Yesenia Barrientos Arenal**  
**Jefa del Departamento de Servicios Escolares**  
**PRESENTE**

Mediante la presente, le informo que la jefatura de carrera a mi cargo, con el visto bueno de la Vicerrectoría Académica, ha designado a los siguientes profesores como sinodales del examen profesional de la alumna **C. Zuleima Villa Martínez**, quien defenderá su trabajo de tesis titulado "**Desarrollo de una aplicación móvil basada en Android para la gestión remota de presentaciones multimedia mediante Raspberry Pi**", para obtener el título de Licenciada en Ingeniería en Computación.

## Titulares:

Presidente: **Dr. Sergio Fabian Ruiz Paz.**  
Secretario: **M.C. Ariel López Rodríguez.**  
Vocal: **M.C. José Domingo Juárez Hernández.**

## Suplentes:

**M.C. Sandro Geovani Vázquez Díaz**  
**Dra. Nancy Pérez Castro.**

Sin más por el momento, le envío un cordial saludo.

**A t e n t a m e n t e**  
*terrauberrima, mens aperta*  
*Bou lo tama, chi, jí, jú*



**M.C. Ariel López Rodríguez**  
Carrera de Ingeniería en  
Computación

**M.C. Héctor López Arjona**  
Vicerrector Académico

C.c.p.: M.C. Héctor López Arjona. Vicerrector académico. Para su conocimiento  
C.c.p.: Archivo

Dedicado a mis padres y a mis hermanos.

## Agradecimientos

Agradezco a la Universidad del Papaloapan (Campus Loma Bonita), por brindarme la oportunidad de cursar mis estudios y superarme profesionalmente al formar parte de una de sus generaciones.

Agradezco a mi director de tesis, el M.C. José Domingo Juárez Hernández por haber aceptado ser el director de mi tesis, además por la paciencia que me tuvo durante todo este tiempo, por sus consejos, por sus enseñanzas, por todo el tiempo invertido en revisar mis avances de tesis y por su valiosa ayuda al guiarme en todo para terminar exitosamente este proyecto. ¡Gracias profe!

Agradezco a mis revisores de tesis, el M.C. Ariel López Rodríguez y el Dr. Sergio Fabián Ruiz Paz por el tiempo dedicado en revisar detalladamente este documento de tesis y por sus excelentes comentarios y observaciones.

Agradezco a los profesores: el M.C. Carlos Copto del Puerto, el M.C. Eduardo Ortiz Hernández y el M.C. Luis Alfredo Juárez Blanco, por haberme apoyado al inicio de este trabajo de tesis. Su ayuda fue fundamental.

Agradezco a mis padres, por apoyarme en todo momento y por estar conmigo siempre. Gracias mami y papi porque siempre me motivaron a terminar este proyecto, y lo he logrado. Quiero que siempre se sientan orgullosos de mí, así como yo me siento orgullosa de ustedes. Agradezco a mis dos hermanos, Zaira y Erwin porque también me apoyaron en todo, sé que cuento con ustedes y así será siempre.

Y al más importante, a Dios, por haberme dado la sabiduría necesaria para lograr una meta más en mi vida, sin tu ayuda oh Dios no lo hubiera logrado, gracias.

## Resumen

Actualmente, el uso de presentaciones multimedia en reuniones de negocios o en la enseñanza en una institución educativa se ha convertido en un método trascendente y necesario para presentar ciertos temas sobre un contexto en específico. Para esto, se requiere el equipo necesario como son una computadora portátil conectada a un proyector digital. Sin embargo, estar conectado a un proyector de la manera tradicional limita en gran medida la movilidad del ponente, ya que no puede moverse libremente y controlar su presentación al mismo tiempo.

Teniendo en cuenta lo anterior, este trabajo de tesis busca permitir el uso de tecnologías móviles, destacando en este trabajo, el uso de teléfonos inteligentes (Smartphone) que permitan la eliminación o el reemplazo de computadoras para realizar presentaciones. Para que esto suceda, se ha desarrollado un sistema basado en la arquitectura cliente-servidor, que tiene como objetivo gestionar remotamente las presentaciones multimedia de forma inalámbrica utilizando la tecnología Wi-Fi a través de un teléfono inteligente.

El sistema está formado por dos aplicaciones: una aplicación cliente y una aplicación servidor. La aplicación cliente, es una aplicación móvil basada en Android, la cual permite conectarse a una red hotspot Wi-Fi, seleccionar y enviar presentaciones multimedia y comandos de control en formato JSON al servidor. La aplicación servidor, es una aplicación de escritorio ejecutada en una tarjeta Raspberry Pi, la cual recibe las presentaciones multimedia y los comandos de control para generar acciones específicas como abrir una presentación, cerrar una presentación, entre otras.

Ambas aplicaciones utilizan el protocolo de transferencia de datos TCP/IP, que proporciona un canal de comunicación asíncrona confiable que las dos aplicaciones utilizan para comunicarse entre sí a través de sockets.

## **Abstract**

Currently, the use of multimedia presentations in business meetings or teaching in an educational institution has become a transcendent and necessary method to present certain topics on a specific context. For this, the necessary equipment such as a laptop computer which is connected to a digital projector are required. However, being connected to a projector in the traditional manner greatly limits the moderator's mobility, since he is unable to move around freely and control his presentation at the same time.

Taking the aforementioned into consideration, this thesis work seeks to enable the use of mobile technologies, highlighting in this work, the use of smartphones (Smartphone) which allow for the elimination or the replacement of computers for performing presentations. In order for this to take place, a system based on client-server architecture has been developed, which aims to remotely manage multimedia presentations wirelessly using Wi-Fi technology through a smartphone.

The system consists of two applications: a client application and a server application. The client application is a mobile application based on Android, which allows you to connect to a Wi-Fi hotspot network, select and send multimedia presentations and control commands in JSON format to the server. The server application is a desktop application executed on a Raspberry Pi card, which receives multimedia presentations and control commands to generate specific actions such as opening a presentation, closing a presentation, among others.

Both applications use the TCP/IP data transfer protocol, which provides a reliable asynchronous communication channel that the two applications use to communicate with each other via sockets.

## Contenido

	Página
Lista de Figuras .....	XIV
Lista de Tablas .....	XVIII
Lista de Acrónimos .....	XIX
Capítulo 1. Introducción .....	1
1.1 Introducción.....	2
1.2 Planteamiento del problema.....	2
1.3 Propuesta de solución .....	3
1.4 Objetivo general .....	3
1.4.1 Objetivos específicos .....	3
1.5 Justificación.....	4
1.6 Beneficios.....	4
1.7 Organización del documento de tesis.....	4
Capítulo 2. Marco teórico .....	6
2.1 Hardware utilizado.....	7
2.1.1 Teléfono inteligente.....	7
2.1.2 Raspberry Pi.....	7
2.1.3 Proyector digital .....	8
2.2 Sistemas operativos .....	8
2.2.1 Sistema operativo Android.....	8
2.2.2 Sistema operativo Raspbian .....	9
2.3 Entornos de desarrollo .....	9
2.3.1 IDE Android Studio 3.2.1.....	10
2.3.2 NetBeans IDE 8.1 .....	10
2.4 Paquetes utilizados .....	10
2.4.1 SDK de Android .....	10
2.4.2 SDK de Java.....	10
2.4.3 SDK de LibreOffice .....	11
2.5 Java .....	11
2.6 Presentación Multimedia .....	11
2.7 LibreOffice.....	12

2.8 LibreOffice Impress .....	12
2.9 Sockets .....	12
2.10 Formato JSON .....	13
2.11 Hotspot.....	13
2.12 Hotspot móvil.....	13
2.13 Punto de acceso inalámbrico.....	14
2.14 Wi-Fi.....	14
2.15 Sistema de presentación inalámbrica .....	14
Capítulo 3. Estado del arte.....	15
3.1 Aplicación Epson iProjection .....	16
3.2 Aplicación Panasonic Wireless Projector.....	17
3.3 Servicio de proyección de material docente basado en Raspberry Pi.....	18
3.4 Control remoto de presentaciones mediante Raspberry Pi y Android .....	19
3.5 Tabla de Comparación .....	20
Capítulo 4. Metodología de solución.....	22
4.1 Procedimiento de Instalación del Sistema Operativo Raspbian.....	25
4.1.1 Descargar el Sistema Operativo Raspbian.....	26
4.1.2 Tarjeta SD de arranque Raspbian.....	27
4.1.3 Transferir el Sistema Operativo a la tarjeta SD .....	29
4.1.4 Configuración de arranque de Raspbian.....	30
4.2 Proceso de instalación de LibreOffice Impress .....	32
4.3 Configuración de la red hotspot Wi-Fi.....	34
4.3.1 Actualización de repositorios y programas en el S.O. ....	36
4.3.2 Detección del adaptador USB Wi-Fi en el S.O. ....	37
4.3.3 Instalación de los paquetes.....	37
4.3.4 Configuración de IP estática .....	38
4.3.5 Configuración Dnsmasq.....	39
4.3.6 Configuración Hostapd.....	39
4.4 Creación de la aplicación cliente .....	41
4.4.1 Conectar a la red Wi-Fi RBPI.....	43
4.4.2 Seleccionar presentación.....	43
4.4.3 Enviar información básica .....	44
4.4.4 Enviar presentación .....	45

4.4.5 Enviar comandos de acceso rápido .....	45
4.4.5.1 Envío de comandos por medio de la API de LibreOffice.....	47
4.4.5.2 Activar control remoto en LibreOffice Impress .....	47
4.4.5.3 Conexión del equipo con el dispositivo móvil.....	48
4.4.5.4 Controlar la presentación de diapositivas .....	49
4.4.5.5 Comandos de Impress Remote .....	50
4.5 Creación de la aplicación servidor.....	50
4.6 Comunicación mediante Sockets .....	53
4.6.1 Arquitectura cliente – servidor.....	53
4.6.2 Comunicación cliente-servidor a través de sockets.....	53
4.6.2.1 Aplicación cliente.....	57
4.6.2.1.1 Crear socket .....	57
4.6.2.1.2 Solicitar conexión.....	58
4.6.2.1.3 Obtener flujos de E/S .....	59
4.6.2.1.4 Cerrar conexión .....	59
4.6.2.2 Aplicación servidor .....	60
4.6.2.2.1 Crear socket .....	61
4.6.2.2.2 Esperar conexión.....	61
4.6.2.2.3 Establecer conexión.....	62
4.6.2.2.4 Obtener flujos de E/S .....	62
4.6.2.2.5 Cerrar conexión .....	63
4.7 Proyectar presentación.....	64
Capítulo 5. Diseño de la aplicación cliente y aplicación servidor.....	65
5.1 Diseño de la aplicación servidor.....	66
5.1.1 Descripción de la interfaz de la aplicación servidor .....	66
5.1.1.1 Interfaz principal .....	66
5.1.1.2 Iniciar aplicación servidor .....	67
5.2 Diseño de la aplicación cliente .....	68
5.2.1 Descripción de las ventanas de la aplicación cliente.....	68
5.2.1.1 Ventana principal.....	68
5.2.1.2 Ventana buscar red Wi-Fi.....	71
5.2.1.3 Ventana seleccionar presentación.....	73
5.2.1.4 Ventana mostrar información.....	75

5.2.1.5 Ventana control remoto .....	78
Capítulo 6. Pruebas y resultados .....	83
6.1 Pruebas del sistema “Raspdroid-Projection” .....	84
6.1.1 Introducción .....	84
6.1.2 Objetivos .....	84
6.1.2.1 Objetivo general .....	84
6.1.2.2 Objetivos específicos .....	84
6.1.3 Modelo del sistema .....	85
6.2 Plan de pruebas de la aplicación cliente (APC-01-01) .....	85
6.2.1 Especificación del Plan de Pruebas de la aplicación cliente .....	86
6.2.1.1 Nomenclatura .....	86
6.2.2 Especificación de los Casos de Prueba de la aplicación cliente (APC-01-01) .....	88
6.2.2.1 Especificación APC-01-03.1 .....	88
6.2.2.2 Especificación APC-01-03.2 .....	89
6.2.2.3 Especificación APC-01-03.3 .....	91
6.2.2.4 Especificación APC-01-03.4 .....	93
6.2.2.5 Especificación APC-01-03.5 .....	95
6.2.2.6 Especificación APC-01-03.6 .....	96
6.2.2.7 Especificación APC-01-03.7 .....	97
6.2.2.8 Especificación APC-01-03.8 .....	99
6.2.2.9 Especificación APC-01-03.9 .....	101
6.2.2.10 Especificación APC-01-03.10 .....	102
6.2.2.11 Especificación APC-01-03.11 .....	103
6.2.2.12 Especificación APC-01-03.12 .....	105
6.2.2.13 Especificación APC-01-03.13 .....	105
6.2.2.14 Especificación APC-01-03.14 .....	108
6.2.2.15 Especificación APC-01-03.15 .....	109
6.2.2.16 Especificación APC-01-03.16 .....	111
6.3 Plan de pruebas de la aplicación servidor (APS-01-01) .....	113
6.3.1 Especificación del Plan de Pruebas de la aplicación servidor .....	113
6.3.1.1 Nomenclatura .....	113
6.3.2 Especificación de los Casos de Prueba de la aplicación servidor (APS-01-01) ..	115
6.3.2.1 Especificación APS-01-03.1 .....	115

6.3.2.2 Especificación APS-01-03.2 .....	116
6.3.2.3 Especificación APS-01-03.3 .....	117
6.3.2.4 Especificación APS-01-03.4 .....	118
6.3.2.5 Especificación APS-01-03.5 .....	120
6.3.2.6 Especificación APS-01-03.6 .....	121
6.3.2.7 Especificación APS-01-03.7 .....	123
6.4 Resultados del sistema “Raspdroid-Projection” .....	125
6.4.1 Resultados de los casos de prueba de la aplicación cliente .....	125
6.4.2 Resultados de los casos de prueba de la aplicación servidor .....	130
6.5 Conclusiones.....	132
Capítulo 7. Conclusiones y trabajos futuros .....	133
7.1 Conclusiones.....	134
7.2 Trabajos futuros .....	134
Anexos .....	135
Anexo A. Diseño de pruebas de la aplicación cliente .....	136
Anexo B. Diseño de pruebas de la aplicación servidor.....	146
8. Bibliografía.....	151

## Lista de Figuras

	Página
Figura 2.1: Porcentaje de distribución de versiones de la plataforma Android. ....	9
Figura 3.1: Proyección inalámbrica mediante iProjection. ....	16
Figura 3.2: Proyección inalámbrica mediante Wireless Projector. ....	17
Figura 3.3: Diseño global de la implementación. ....	18
Figura 3.4: Diagrama de arquitectura del sistema definido. ....	19
Figura 4.1: Arquitectura general del sistema "Raspdroid-Projection".....	23
Figura 4.2: Esquema general de las etapas de desarrollo. ....	24
Figura 4.3: Esquema de la instalación de Raspbian en Raspberry Pi. ....	25
Figura 4.4: Actividades de la instalación del Sistema Operativo Raspbian.....	26
Figura 4.5: Pantalla de descarga del S.O. Raspbian.....	27
Figura 4.6: SDFormatter V4.0.....	28
Figura 4.7: Memory Card Format complete.....	28
Figura 4.8: Win32 Disk Imager.....	29
Figura 4.9: Write Successful. ....	30
Figura 4.10: Ventana principal de configuración Raspbian. ....	31
Figura 4.11: Ventana de configuración de actualización de software. ....	32
Figura 4.12: Proceso de instalación de LibreOffice Impress en Raspberry Pi. ....	33
Figura 4.13: Comando para la instalación de LibreOffice Impress. ....	33
Figura 4.14: Ventana inicial de LibreOffice Impress. ....	34
Figura 4.15: Esquema del funcionamiento de la red hotspot Wi-Fi.....	35
Figura 4.16: Actividades de la configuración de la red hotspot Wi-Fi. ....	36
Figura 4.17: Comandos de actualización de repositorios y programas. ....	36
Figura 4.18: Adaptador USB Wi-Pi.....	37
Figura 4.19: Comando para detección del adaptador USB Wi-Fi. ....	37
Figura 4.20: Comando de instalación de dnsmasq y hostapd. ....	37
Figura 4.21: Comandos para detener los servicios dnsmasq y hostapd.....	38
Figura 4.22: Comando para abrir el archivo de configuración dhcpd.....	38
Figura 4.23: Comando para reiniciar dhcpd. ....	38
Figura 4.24: Comandos de configuración Dnsmasq.....	39
Figura 4.25: Comando para abrir el archivo de configuración hostapd.....	39
Figura 4.26: Comando para abrir el archivo hostapd.....	40

Figura 4.27: Comandos para iniciar los servicios hostapd y dnsmasq. ....	41
Figura 4.28: Instalación de la App cliente.....	41
Figura 4.29: Esquema de funcionamiento de la aplicación cliente. ....	42
Figura 4.30: Actividades de la creación de la aplicación cliente. ....	42
Figura 4.31: Conexión a la red Wi-Fi RBPI. ....	43
Figura 4.32: Diagrama de navegación de directorios. ....	44
Figura 4.33: Diagrama del envío de la información básica. ....	44
Figura 4.34: Envío de la presentación multimedia.....	45
Figura 4.35: Diagrama del envío de comandos. ....	46
Figura 4.36: Pantalla para activar el control remoto de Impress.....	48
Figura 4.37: Pantalla para activar funcionalidades experimentales de Impress.....	49
Figura 4.38: Aplicación servidor creada en la Raspberry Pi. ....	51
Figura 4.39: Esquema de funcionamiento de la aplicación servidor. ....	52
Figura 4.40: Esquema del establecimiento de conexiones. ....	52
Figura 4.41: Arquitectura cliente-servidor.....	53
Figura 4.42: Interfaz de conexión cliente-servidor. ....	54
Figura 4.43: Esquema general de comunicación. ....	55
Figura 4.44: Diagrama general del proceso de comunicación.....	55
Figura 4.45: Diagrama de los métodos cliente-servidor. ....	56
Figura 4.46: Proceso para establecer la comunicación (aplicación cliente).....	57
Figura 4.47: Creación del clientSocket.....	57
Figura 4.48: Solicitud de conexión a la aplicación servidor. ....	58
Figura 4.49: Envío del flujo de datos de la aplicación cliente. ....	59
Figura 4.50: Cierre de conexión del socket cliente.....	60
Figura 4.51: Proceso para establecer la comunicación (aplicación servidor). ....	60
Figura 4.52: Creación del serverSocket. ....	61
Figura 4.53: Esperar conexión. ....	62
Figura 4.54: Conexión bidireccional establecida. ....	62
Figura 4.55: Envío del flujo de datos de la aplicación servidor. ....	63
Figura 4.56: Cierre de conexión del socket servidor.....	63
Figura 4.57: Proceso de proyección.....	64
Figura 5.1: Interfaz principal de la aplicación servidor.....	66
Figura 5.2: Aplicación servidor iniciada. ....	67

Figura 5.3: Ventana principal de la App cliente. ....	69
Figura 5.4: Diagrama de clase de PrincipalActivity. ....	70
Figura 5.5: Código XML de activity_principal. ....	70
Figura 5.6: Redes Wi-Fi disponibles. ....	71
Figura 5.7: Diagrama de clase de WifiScanAdapter. ....	72
Figura 5.8: Código XML de network_list. ....	73
Figura 5.9: Seleccionar presentación. ....	74
Figura 5.10: Diagrama de clase de MainActivity. ....	75
Figura 5.11: Código del método showChooser de la clase MainActivity. ....	75
Figura 5.12: Información de la presentación multimedia. ....	76
Figura 5.13: Diagrama de clase de Informacion. ....	77
Figura 5.14: Código XML de activity_informacion. ....	77
Figura 5.15: Control remoto (1). ....	78
Figura 5.16: Control remoto (2). ....	79
Figura 5.17: Control remoto (3). ....	80
Figura 5.18: Diagrama de clase de SlidesPagerFragment. ....	81
Figura 5.19: Código XML de fragment_slides_pager. ....	82
Figura 6.1: Modelo del sistema “Raspdroid-Projection”. ....	85
Figura 6.2: Botón “BUSCAR RED WI-FI” habilitado. ....	89
Figura 6.3: Botón “SELECCIONAR PRESENTACIÓN” habilitado. ....	90
Figura 6.4: Botón “CONTROL REMOTO” habilitado. ....	92
Figura 6.5: SSID del punto de acceso Wi-Fi. ....	94
Figura 6.6: Puntos de acceso Wi-Fi. ....	95
Figura 6.7: Actualizar lista de puntos de acceso Wi-Fi. ....	97
Figura 6.8: Conexión a un punto de acceso Wi-Fi. ....	98
Figura 6.9: Explorador de archivos. ....	100
Figura 6.10: Información básica del archivo. ....	101
Figura 6.11: Buscar otro archivo de presentación multimedia. ....	103
Figura 6.12: Validación de la extensión del archivo. ....	104
Figura 6.13: Servidor de LibreOffice. ....	106
Figura 6.14: Servidor de LibreOffice sin ejecutarse. ....	107
Figura 6.15: Emparejamiento y ejecución de una presentación multimedia. ....	109
Figura 6.16: Visualización de diapositivas en forma de maya. ....	110

Figura 6.17: Parar presentación.....	112
Figura 6.18: Red hotspot Wi-Fi emitiendo señal.....	115
Figura 6.19: Ejecución de la aplicación servidor. ....	116
Figura 6.20: Servidor iniciado. ....	118
Figura 6.21: Recepción de un archivo de presentación multimedia.....	119
Figura 6.22: Servidor detenido.....	121
Figura 6.23: LibreOffice Impress iniciado. ....	122
Figura 6.24: Emparejamiento (1).....	123
Figura 6.25: Emparejamiento (2).....	124
Figura 6.26: Gráfica de resultados del dispositivo Verykool. ....	126
Figura 6.27: Gráfica de resultados del dispositivo Motorola. ....	128
Figura 6.28: Gráfica de resultados del dispositivo Huawei. ....	130
Figura 6.29: Gráfica de resultados de la Raspberry Pi.....	131

## Lista de Tablas

	Página
Tabla 3.1: Tabla de comparación.....	21
Tabla 4.1: Comandos enviados en formato JSON.....	47
Tabla 4.2: Comandos de Impress Remote.....	50
Tabla 6.1: Requerimientos de prueba de la App cliente.....	86
Tabla 6.2: Concepto del plan de pruebas de la aplicación cliente.....	87
Tabla 6.3: Documentación de pruebas de la aplicación cliente.....	87
Tabla 6.4: Módulos de la aplicación cliente.....	87
Tabla 6.5: Funciones de la aplicación cliente.....	87
Tabla 6.6: Requerimientos de prueba de la App servidor.....	113
Tabla 6.7: Concepto del plan de pruebas de la aplicación servidor.....	114
Tabla 6.8: Documentación de pruebas de la aplicación servidor.....	114
Tabla 6.9: Módulos de la aplicación servidor.....	114
Tabla 6.10: Funciones de la aplicación servidor.....	114
Tabla 6.11: Análisis de resultados del plan de pruebas con el dispositivo Verykool.....	125
Tabla 6.12: Análisis de resultados del plan de pruebas con el dispositivo Motorola.....	127
Tabla 6.13: Análisis de resultados del plan de pruebas con el dispositivo Huawei.....	128
Tabla 6.14: Análisis de resultados del plan de pruebas con la Raspberry Pi.....	130

## Lista de Acrónimos

<b>App:</b>	(Acortamiento del inglés <i>Application</i> ). Aplicación móvil.
<b>APC:</b>	( <i>SlideFiles-Projection</i> ). Nombre de la aplicación cliente.
<b>APS:</b>	( <i>ServerSlide-Projection</i> ). Nombre de la aplicación servidor.
<b>RBPI:</b>	Nombre de la red.
<b>S.O:</b>	( <i>Operating System</i> ). Sistema operativo.
<b>WI-FI:</b>	( <i>Wireless Fidelity</i> ). Fidelidad inalámbrica.
<b>HDMI:</b>	( <i>High Definition Multimedia Interface</i> ). Interfaz multimedia de alta definición.
<b>USB:</b>	( <i>Universal Serial Bus</i> ). Bus universal en serie.
<b>SDK:</b>	( <i>Software Development Kit</i> ). Kit de desarrollo de software.
<b>IDE:</b>	( <i>Integrated Development Environment</i> ). Entorno de desarrollo integrado.
<b>API:</b>	( <i>Application Programming Interface</i> ). Interfaz de programación de aplicaciones.
<b>PPT:</b>	( <i>PowerPoint File</i> ). Fichero PowerPoint.
<b>JSON:</b>	( <i>JavaScript Object Notation</i> ). Notación de objetos de JavaScript.
<b>TCP:</b>	( <i>Transmission Control Protocol</i> ). Protocolo de control de transmisión.
<b>IP:</b>	( <i>Internet Protocol</i> ). Protocolo de Internet.
<b>WLAN:</b>	( <i>Wireless Local Area Network</i> ). Red de área local inalámbrica.
<b>WAP/AP:</b>	( <i>Wireless Access Point</i> ). Punto de acceso inalámbrico.
<b>LAN:</b>	( <i>Local Area Network</i> ). Red de área local.
<b>WPA:</b>	( <i>Wi-Fi Protected Access</i> ). Acceso Wi-Fi protegido.
<b>DHCP:</b>	( <i>Dynamic Host Configuration Protocol</i> ). Protocolo de configuración de host dinámico.
<b>HOSTAPD:</b>	( <i>Host access point daemon</i> ). Demonio de punto de acceso de host.
<b>DHCPD:</b>	( <i>Dynamic Host Configuration Protocol Client Daemon</i> ). Protocolo de configuración dinámica de host de demonio cliente.
<b>SSID:</b>	( <i>Service Set Identifier</i> ). Identificador del conjunto de servicios.
<b>URI:</b>	( <i>Uniform Resource Identifier</i> ). Identificador de recursos uniformes.
<b>VGA:</b>	( <i>Video Graphics Array</i> ). Arreglo de vídeo gráfico.

## Capítulo 1. Introducción

---

En este capítulo, a manera de introducción, se da una breve descripción del contexto presentado en un ambiente de presentación multimedia para posteriormente describir el problema que dio pie al desarrollo de esta tesis. Se expone también una breve descripción de la propuesta de solución a la problemática presentada y se describen los objetivos a cumplir junto con la justificación y los beneficios esperados. Como parte final de este capítulo, se presenta la organización general del documento de tesis.

## 1.1 Introducción

Cuando se habla de sistemas de presentación inalámbrica [30], uno de los componentes importantes es el dispositivo desde el cual se quiere proyectar los contenidos multimedia. Actualmente, existen sistemas de presentación inalámbrica que utilizan nuevas tecnologías como son los dispositivos móviles.

Con respecto a este tipo de tecnología, hoy en día, los dispositivos móviles han experimentado una gran evolución, ya que por su pequeño tamaño y sus capacidades de funcionamiento, su aceptación ha ido creciendo a un ritmo asombroso, además, de que el mercado poco a poco va generando una producción masiva de aplicaciones para estos dispositivos. Actualmente, uno de los dispositivos móviles más utilizado es el teléfono inteligente (Smartphone en inglés) [2], el cual puede utilizar el sistema operativo Android.

Debido al gran potencial con el que cuentan los teléfonos inteligentes para el desarrollo de actividades, como: laborales, educativas, procesamiento de información, correo electrónico, redes sociales, etc., similares al de una computadora, hacen de éste una herramienta ideal para utilizarse como parte esencial en la proyección de contenidos multimedia en clases o en reuniones de trabajo, pudiendo así, cambiar la forma tradicional de cómo se realiza la proyección por medio del proyector apoyado de una computadora [1].

## 1.2 Planteamiento del problema

Actualmente, en la mayoría de las clases en las instituciones educativas se cuenta con un sistema tradicional de proyección que los profesores y alumnos utilizan para exponer temas sobre un contexto en específico. Pero con este sistema tradicional actual, es necesario que una computadora portátil esté conectada en todo momento a un proyector digital mediante cables de energía y comunicación necesarios, impidiendo al ponente disponer de la movilidad, ya que es necesario estar cerca de la computadora para poder gestionar las diapositivas. Además, de que no siempre se cuenta con una computadora portátil para exponer.

## 1.3 Propuesta de solución

La propuesta de solución busca aprovechar el potencial de los dispositivos móviles, en este caso, los teléfonos inteligentes, mediante el desarrollo de una aplicación móvil basada en Android para gestionar remotamente las presentaciones multimedia de manera inalámbrica utilizando la tecnología Wi-Fi. El sistema está basado en una tarjeta Raspberry Pi conectada a un proyector digital, la cual es controlada por medio de una aplicación servidor, ofreciendo con ello funcionalidades de un servidor a dispositivos para presentaciones multimedia.

Esta propuesta de solución busca mejorar la experiencia de usuario, ya que brinda una comodidad al no tener que transportar una computadora portátil, además, de no tener que conectar todo el equipo para iniciar una exposición.

## 1.4 Objetivo general

Mejorar la experiencia de usuario en cuanto a practicidad, simplicidad, usabilidad y movilidad al momento de presentar o exponer presentaciones multimedia, mediante el desarrollo de una aplicación móvil basada en Android para gestionar remotamente este tipo de contenido multimedia.

### 1.4.1 Objetivos específicos

- Mejorar la experiencia de usuario en cuanto a practicidad, simplicidad, usabilidad y movilidad, al facilitar la conectividad de dispositivos para la realización de una presentación multimedia.
- Hacer uso eficiente de los teléfonos inteligentes para la realización de presentaciones multimedia.

## 1.5 Justificación

La mayoría de las instituciones educativas cuenta con un proyector digital de bajo costo y sin conexión inalámbrica. Con la propuesta de solución, se proporciona a los proyectores digitales (que normalmente requieren una conexión por cable) una nueva capacidad de conexión inalámbrica, esto utilizando una Raspberry Pi de bajo costo, comparado con el costo de un proyector nuevo con conexión inalámbrica, el cual supera en un porcentaje alto el costo que se debería invertir.

## 1.6 Beneficios

Este trabajo de tesis ofrece los siguientes beneficios:

- Aprovechar la funcionalidad de los dispositivos móviles (teléfonos inteligentes) para gestionar remotamente presentaciones multimedia en lugar de hacerlo con una computadora portátil.
- Evitar el uso de cables para conectar el proyector con el dispositivo móvil.
- Permite la movilidad al usuario al momento de exponer o presentar este tipo de contenido multimedia.
- La aplicación desarrollada es de código abierto (open source) y se puede obtener de forma gratuita.
- Para proyectar las presentaciones multimedia se puede utilizar cualquier tipo de proyector digital conectado a la Raspberry Pi.

## 1.7 Organización del documento de tesis

El presente documento de tesis se compone de siete capítulos, dos anexos y una bibliografía. A continuación, se describen de forma general los capítulos que componen el contenido de la tesis:

**Capítulo 1: Introducción.** En este capítulo, a manera de introducción, se da una breve descripción del contexto presentado en un ambiente de presentación multimedia para

posteriormente describir el problema que dio pie al desarrollo de esta tesis. Se expone también una breve descripción de la propuesta de solución a la problemática presentada y se describen los objetivos a cumplir junto con la justificación y los beneficios esperados. Como parte final de este capítulo, se presenta la organización general del documento de tesis.

**Capítulo 2. Marco teórico.** En este capítulo, se presentan para una mejor comprensión de la lectura, los conceptos empleados en esta tesis que darán sustento al desarrollo de la misma.

**Capítulo 3. Estado del arte.** En este capítulo, se presentan los resultados del proceso de búsqueda de información relevante con respecto a trabajos de investigación relacionados que sirven como punto de referencia y comparación con el trabajo propuesto.

**Capítulo 4. Metodología de solución.** En este capítulo, se describe de manera general la metodología de solución del sistema denominado “Raspdroid-Projection”. Además, se presentan las etapas de desarrollo, actividades y componentes para el desarrollo de esta tesis.

**Capítulo 5. Diseño de la aplicación cliente y aplicación servidor.** En este capítulo, se muestra el diseño de las ventanas de la aplicación cliente y el diseño de la interfaz principal de la aplicación servidor correspondientes al sistema “Raspdroid-Projection”, además se describe la funcionalidad de cada una de éstas.

**Capítulo 6. Pruebas y resultados.** En este capítulo, se describe de manera general las pruebas realizadas a la aplicación cliente y a la aplicación servidor del sistema “Raspdroid-Projection”, además, se presentan los resultados obtenidos de cada una de las pruebas de cada aplicación.

**Capítulo 7. Conclusiones y trabajos futuros.** En este capítulo, se presentan las conclusiones obtenidas en base al proceso de investigación y a la metodología de solución llevados a cabo en este trabajo de tesis, y además, se listan los trabajos futuros.

## Capítulo 2. Marco teórico

---

En este capítulo, se presentan para una mejor comprensión de la lectura, los conceptos empleados en esta tesis que darán sustento al desarrollo de la misma.

## 2.1 Hardware utilizado

A continuación, se describen conceptos sobre diferentes dispositivos físicos utilizados para el desarrollo e implementación del trabajo de tesis.

### 2.1.1 Teléfono inteligente

Un teléfono inteligente (Smartphone en inglés) [2], es un dispositivo electrónico de dimensiones pequeñas, y con una plataforma informática móvil que cumple funciones similares a las de una computadora como, por ejemplo: navegar por Internet, reproducir datos multimedia, realizar transacciones bancarias, orientarnos vía GPS, descargar juegos, conectarnos a las redes sociales, mayor capacidad de almacenar datos, etc.

Los teléfonos inteligentes han evolucionado desde su primera aparición en cuanto a tamaño, función y aplicaciones; y su uso se ha incrementado progresivamente en los últimos años. Hoy un factor importante que hace atractivo a un teléfono inteligente es su sistema operativo, que junto con el hardware logran desarrollar operaciones que permiten correr cierta cantidad de procesos que son traducidos en importantes y llamativas aplicaciones, convirtiéndolos en dispositivos más competitivos que otros.

### 2.1.2 Raspberry Pi

Raspberry Pi [3], es una computadora del tamaño de una tarjeta de crédito, desarrollada por la Fundación Raspberry Pi (Reino Unido) con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas. El diseño de la placa incluye un procesador central a 700 MHz ARM1176JZF-S y tiene un máximo de 512 MB de RAM. Cuenta, además, con una unidad de procesamiento gráfico (GPU), conexiones para USB 2.0, salida de video, audio, controlador Ethernet, gráficos de alta definición, E/S para fines generales, una tarjeta SD de almacenamiento a largo plazo. Además, la Raspberry Pi tiene unas dimensiones suficientemente pequeñas, lo que la hace fácil de transportar sin ningún tipo de problema. Otra ventaja es su bajo costo, aproximadamente de \$1500 pesos, convirtiendo a la Raspberry Pi en un hardware muy importante que aporta una buena solución en el desarrollo de este sistema.

### 2.1.3 Proyector digital

Un proyector digital [4], es un dispositivo encargado de recibir por medio de un puerto, las señales de video procedentes de una computadora. Éste procesa la señal digital y la decodifica para poder ser enviada por medio de luz a unos micros-espejos encargados de la proyección digital en alguna superficie clara, por ejemplo, una pantalla de proyección.

## 2.2 Sistemas operativos

A continuación, se describen conceptos sobre los sistemas operativos que fueron elegidos y utilizados en el trabajo de tesis.

### 2.2.1 Sistema operativo Android

Android [5], es un sistema operativo desarrollado por Google sobre el núcleo del sistema operativo Linux. Está diseñado para que los desarrolladores puedan crear aplicaciones en las que se aproveche al máximo los recursos y las funciones de los dispositivos móviles.

Actualmente, Google ofrece de forma gratuita el SDK (Software Development Kit) oficial, con el cual se pueden construir aplicaciones en el lenguaje de programación Java, esto se ha hecho con el fin de permitirles a los desarrolladores diseñar aplicaciones tomando ventaja de todas las características que los dispositivos pueden ofrecer.

El sistema operativo Android, al igual que los propios dispositivos móviles, ha evolucionado rápidamente, acumulando una gran cantidad de versiones. Al día de hoy, existe una cantidad relativa de dispositivos con una versión de Android. Según la página oficial de Android Developer [6], Google ha lanzado periódicamente nuevas versiones para dispositivos móviles actuales. En la figura 2.1, se presenta la cantidad relativa de dispositivos que usan una versión determinada de la plataforma Android.

Los datos presentados en la gráfica de la figura 2.1 fueron recopilados en un periodo de 7 días hasta el 7 de mayo de 2019, y no se muestran versiones con una distribución inferior a 0.1%.

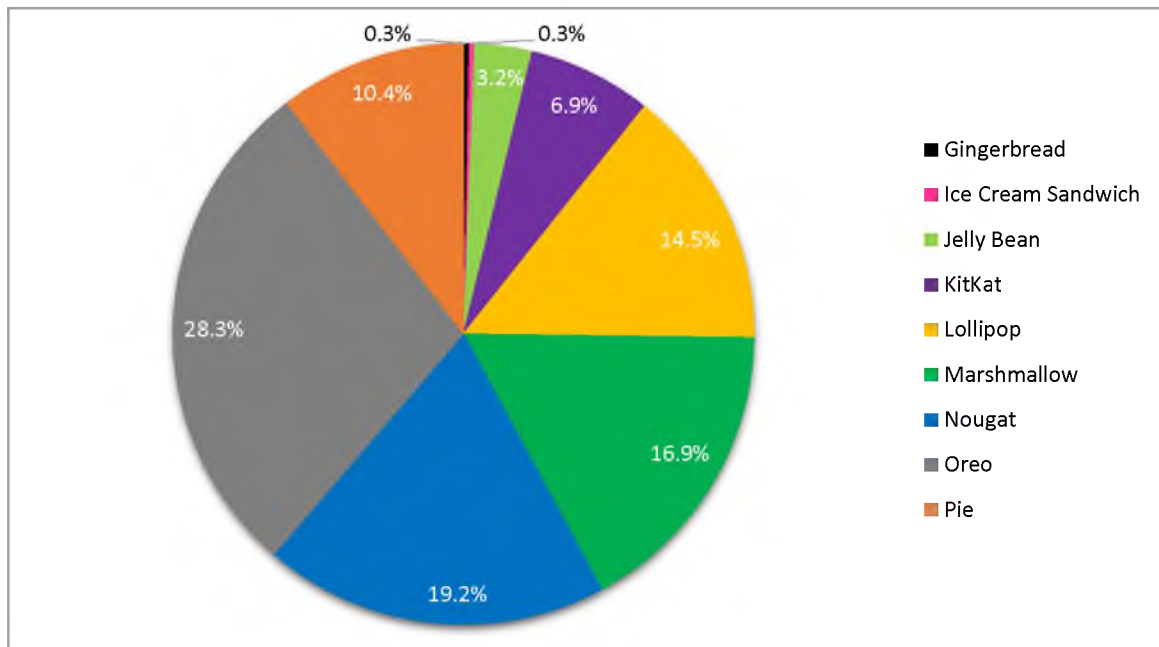


Figura 2.1: Porcentaje de distribución de versiones de la plataforma Android.

### 2.2.2 Sistema operativo Raspbian

Raspbian es un sistema operativo basado en Debian optimizado para el hardware de la Raspberry Pi. Este sistema operativo es el conjunto de programas básicos y utilidades que hacen funcionar la Raspberry Pi, el cual, además proporciona más de 35,000 paquetes y software pre-compilado e incluido en un formato agradable para una fácil instalación [7].

## 2.3 Entornos de desarrollo

Para llevar a cabo el desarrollo de la aplicación cliente y la aplicación servidor, se utilizaron dos entornos de desarrollo que se describen a continuación.

### 2.3.1 IDE Android Studio 3.2.1

Android Studio es el entorno de desarrollo integrado (IDE, por sus siglas en inglés) oficial para el desarrollo de aplicaciones en Android, el cual ofrece funciones que aumentan la productividad durante la compilación de Apps para Android basándose en IntelliJ IDEA [8].

### 2.3.2 NetBeans IDE 8.1

NetBeans IDE es un entorno de desarrollo integrado de código abierto escrito en el lenguaje de programación Java, pero puede servir para escribir, compilar, depurar y ejecutar otras aplicaciones escritas en otros lenguajes de programación como C/C++, PHP, HTML5, entre otros [9].

## 2.4 Paquetes utilizados

A continuación, se describen conceptos sobre diferentes paquetes utilizados que proporcionan herramientas para el desarrollo de este trabajo de tesis.

### 2.4.1 SDK de Android

El SDK (Software Development Kit) es el kit de desarrollo de software de Android [10], que proporciona APIs y un conjunto completo de herramientas necesarias para desarrollar aplicaciones en la plataforma Android utilizando el lenguaje de programación Java [11]. Estas herramientas incluyen un depurador, bibliotecas, un emulador y documentación [12].

### 2.4.2 SDK de Java

El SDK de Java es un paquete instalable que contiene la interfaz de programación de aplicaciones (API) de Java, la cual es una colección grande de clases agrupadas en bibliotecas y listas para ser utilizadas. Además, el SDK de Java también contiene herramientas de desarrollo para supervisar, desarrollar, depurar, y documentar aplicaciones hechas en el lenguaje Java [13].

### 2.4.3 SDK de LibreOffice

El SDK de LibreOffice proporciona un conjunto de herramientas de desarrollo, bibliotecas base, archivos de encabezado y archivos de clase auxiliar que se pueden usar para desarrollar componentes Java / C ++, o para control remoto de office [14].

## 2.5 Java

Java es un lenguaje de programación derivado del lenguaje C, el cual tiene su propia estructura, reglas de sintaxis y paradigma de programación. El paradigma de programación del lenguaje Java se basa en el concepto de programación orientada a objetos (OOP) [15]. Actualmente, Java es la base para prácticamente todos los tipos de aplicaciones de red, además, del estándar global para desarrollar y distribuir aplicaciones móviles y embebidas, juegos, contenido basado en web y software de empresa [16].

## 2.6 Presentación Multimedia

Una presentación multimedia es un recurso didáctico que permite integrar imágenes, gráficos, audio, video, texto, efectos, entre otros, en páginas llamadas “diapositivas”, al mismo tiempo que admite interactuar con otros programas, los cuales logran que los diferentes estilos de aprendizaje de las personas (alumnos) se cubran perfectamente y les sea más fácil entender la información proporcionada [17].

Una presentación multimedia es compatible y puede ser guardada con los siguientes formatos de presentaciones [18]:

**PPTX:** Presentación Office Open XML. Es un formato de archivo comprimido, basado en XML desarrollado por Microsoft para representación de hojas de cálculo, gráficos, presentaciones, y documentos de procesamiento de texto.

**PPT:** Formato de archivo usado por Microsoft PowerPoint.

**ODP:** Presentación OpenDocument. Es el formato que representa un documento de presentación creado por la aplicación Impress, que es parte de las suites de oficinas basadas en OpenOffice.

**PDF:** Formato de documento portátil. Es un formato de archivo usado para la representación de documentos en una manera independiente de la aplicación de software, hardware, y sistemas operativos.

## 2.7 LibreOffice

LibreOffice [19], es una poderosa suite de oficina, la cual incluye potentes herramientas como son: el procesador de textos Writer, la hoja de cálculos Calc, el editor de presentaciones Impress, la aplicación de dibujo y diagramas de flujo Draw, la base de datos e interfaz con otras bases de datos Base, y el editor de fórmulas matemáticas Math.

## 2.8 LibreOffice Impress

Impress [20], es un programa de presentación del paquete de oficina de LibreOffice, es una herramienta multiplataforma, gratuita y de libre distribución. Esta herramienta permite crear diapositivas que pueden incluir diagramas, objetos de dibujo, texto, multimedia y muchos otros elementos, además, de hacer transiciones entre diapositivas, animaciones de los elementos de una diapositiva y efectos multimedia.

## 2.9 Sockets

Un socket es un punto final de un enlace de comunicación bidireccional entre dos programas que se ejecutan en la red. Un socket está vinculado a un número de puerto para que la capa TCP pueda identificar la aplicación a la que están destinados los datos.

Un punto final es una combinación de una dirección IP y un número de puerto. Cada conexión TCP puede ser identificada de manera única por sus dos puntos finales. De esta manera, se pueden tener múltiples conexiones entre el host y el servidor [21].

## 2.10 Formato JSON

JSON es un formato ligero de intercambio de datos. Está basado en un subconjunto del lenguaje de programación JavaScript, y el Standard ECMA-262. JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, entre otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

JSON está constituido por dos estructuras: Una colección de pares de nombre/valor, que en varios lenguajes es conocido como un objeto, registro, estructura, diccionario, tabla hash, lista de claves o un arreglo asociativo, y una lista ordenada de valores, que en la mayoría de los lenguajes, esto se implementa como arreglos, vectores, listas o secuencias [22].

## 2.11 Hotspot

Un hotspot [23], es una ubicación física donde las personas pueden acceder a Internet generalmente usando Wi-Fi, a través de una red de área local inalámbrica (WLAN) con un enrutador conectado a un proveedor de servicios de Internet.

La mayoría de las personas se refieren a estas ubicaciones como "zonas Wi-Fi" o "conexiones Wi-Fi". En pocas palabras, las zonas activas son los lugares físicos donde los usuarios pueden conectar de forma inalámbrica sus dispositivos móviles, como teléfonos inteligentes y tabletas, a Internet.

Un hotspot puede estar en un lugar privado o público, como en una cafetería, un hotel, un aeropuerto o incluso un avión. Mientras que muchos hotspots públicos ofrecen acceso inalámbrico gratuito a una red abierta, otros requieren pago.

## 2.12 Hotspot móvil

Un hotspot móvil (a veces llamado hotspot portátil) es un hotspot que es móvil. Mientras que un hotspot Wi-Fi "normal" está vinculado a una ubicación física, se puede crear un hotspot

móvil utilizando la conexión de datos de un teléfono inteligente para conectar una computadora portátil a Internet [23].

### **2.13 Punto de acceso inalámbrico**

Un punto de acceso inalámbrico (WAP) es un dispositivo de red que permite que un dispositivo compatible con Wi-Fi se conecte a una red cableada. El punto de acceso inalámbrico puede estar físicamente conectado a un enrutador o estar integrado en el mismo enrutador [23].

### **2.14 Wi-Fi**

Wi-Fi es la tecnología que permite que un teléfono inteligente o una computadora acceda a Internet a través de una conexión inalámbrica. Wi-Fi usa señales de radio para enviar y recibir datos entre el dispositivo habilitado y el punto de acceso inalámbrico (WAP) [23].

### **2.15 Sistema de presentación inalámbrica**

Un sistema de presentación inalámbrica [30] es un dispositivo de transmisión multimedia que permite que uno o más usuarios muestren los contenidos de una computadora, Tablet o Smartphone en una pantalla de mayores dimensiones, utilizando para ello tecnología de espejo de pantalla. Esta tecnología es apropiada en los entornos de colaboración, como son salas de reuniones, salas de conferencias, espacios de congregación y aulas.

## Capítulo 3. Estado del arte

---

En este capítulo, se presentan los resultados del proceso de búsqueda de información relevante con respecto a trabajos de investigación relacionados que sirven como punto de referencia y comparación con el trabajo propuesto.

En la actualidad existen aplicaciones que son utilizadas para proyectar contenidos multimedia de manera inalámbrica utilizando nuevas tecnologías como son los dispositivos móviles; en este caso los teléfonos inteligentes. A continuación, se describen dos aplicaciones y dos trabajos de tesis en donde se lleva a cabo la proyección inalámbrica de archivos utilizando un teléfono inteligente.

### 3.1 Aplicación Epson iProjection

El corporativo Epson (Epson Corporate) desarrolló la aplicación iProjection para dispositivos iOS (iPhone, iPad o iPod Touch) y Android (teléfonos inteligentes y tabletas). Esta aplicación permite seleccionar y controlar imágenes, fotos, presentaciones y archivos almacenados en los dispositivos móviles, proyectándolos de forma inalámbrica Wi-Fi a través de un proyector Epson con conectividad de red [24], [25], tal como se presenta en la figura 3.1.



Figura 3.1: Proyección inalámbrica mediante iProjection.

## 3.2 Aplicación Panasonic Wireless Projector

La corporación Panasonic (Panasonic Corporation) desarrolló la aplicación Wireless Projector para dispositivos iOS (iPhone, iPad o iPod Touch) y Android (teléfonos inteligentes y tabletas) [26]. Esta aplicación permite enviar capturas de pantalla, imágenes con formato JPEG / PNG y archivos de documentos PDF y PowerPoint guardados localmente en los dispositivos móviles, además, permite proyectar páginas web con el navegador web incorporado y fotografías tomadas con la cámara incorporada en dichos dispositivos móviles a través de una red LAN inalámbrica Wi-Fi a un proyector inalámbrico Panasonic [27], como se presenta en la figura 3.2.



Figura 3.2: Proyección inalámbrica mediante Wireless Projector.

### 3.3 Servicio de proyección de material docente basado en Raspberry Pi

En este trabajo de tesis se desarrolló un sistema de proyección de material docente, el cual proporciona un servicio de proyección de material docente usando una Raspberry Pi y algunos servidores correspondientes para su correcto funcionamiento. El sistema permite visualizar a través de la Raspberry Pi el material de enseñanza haciendo uso del proyector, además, permite controlar el material docente por medio de comandos de control a través de dispositivos inalámbricos (computadora portátil, Tablet, Smartphone) [28], en la figura 3.3 se muestra el diseño global de la implementación.

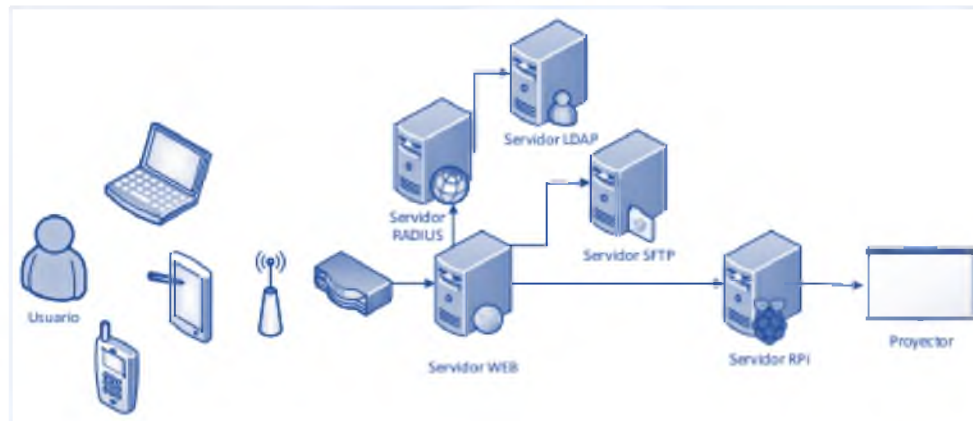


Figura 3.3: Diseño global de la implementación.

### 3.4 Control remoto de presentaciones mediante Raspberry Pi y Android

En este trabajo de tesis se desarrolló un sistema para el control remoto de las presentaciones usando un servidor formado por una Raspberry Pi y un cliente formado por una aplicación móvil Android. Este sistema permite el control de las presentaciones a través de comandos de control de la presentación tales como abrir una presentación, avanzar y retroceder diapositiva, entre otros; que son enviados desde la aplicación móvil al servidor. El protocolo de comunicación inalámbrica utilizado es por vía Bluetooth permitiendo a la aplicación cliente conectarse al servidor [29], en la figura 3.4 se muestra el diagrama de la arquitectura correspondiente al sistema definido.

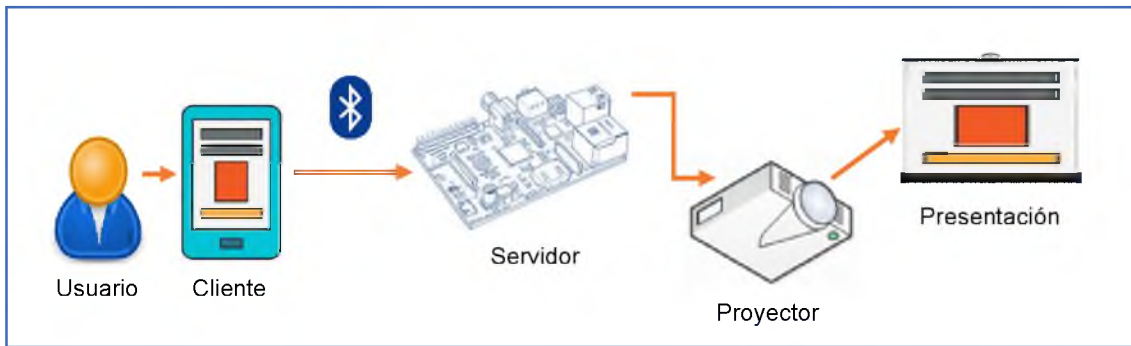


Figura 3.4: Diagrama de arquitectura del sistema definido.

### 3.5 Tabla de Comparación

Finalmente, en la tabla 3.1 se presenta una comparativa de la propuesta de solución con los trabajos relacionados señalados en este capítulo. A continuación, se describen los símbolos y columnas de la tabla mencionada.

#### Descripción de los símbolos utilizados:

- El símbolo **✓**: Se utiliza cuando la característica que hace referencia a la columna es verdadera con respecto a la aplicación/trabajo relacionado.
- El símbolo **✗**: Se utiliza cuando la característica que hace referencia a la columna es falsa con respecto a la aplicación/trabajo relacionado.

#### Descripción de las columnas:

- Aplicación/Trabajo relacionado (1ra columna): Hace referencia a las aplicaciones y trabajos relacionados a la propuesta de solución.
- S.O. Android (2da columna): Hace referencia al sistema operativo utilizado por la aplicación/trabajo relacionado.
- Tecnología (3ra columna): Señala el tipo de comunicación en la que se basa la aplicación/trabajo relacionado.
- Raspberry Pi (4ta columna): Indica si la aplicación/trabajo relacionado utiliza una tarjeta Raspberry Pi como servidor.
- Proyector indefinido (5ta columna): Indica si se utiliza un proyector indefinido en cuanto a marca, precio, tamaño y color para la proyección de contenidos.
- Comandos de control (6ta columna): Indica si se utilizan comandos de control para la gestión remota de presentaciones multimedia.
- Visualización en tiempo real (7ma columna): Indica si en la aplicación/trabajo relacionado existe visualización en tiempo real de las presentaciones multimedia.
- LibreOffice (8va columna): Indica si la aplicación/trabajo relacionado utiliza el programa de ofimática LibreOffice Impress.

Tabla 3.1: Tabla de comparación.

Aplicación/ Trabajo relacionado	S.O. Android	Tecnología		Raspberry Pi	Proyector indefinido	Comandos de control	Visualización en tiempo real	Libre Office
		Wi-Fi	Bluetooth					
iProjection	✓	✓	✗	✗	✗	✗	✓	✗
Wireless Projector	✓	✓	✗	✗	✗	✗	✓	✗
Proyección de material docente	✓	✓	✗	✓	✓	✓	✗	✗
Control remoto de presenta- ciones	✓	✗	✓	✓	✓	✓	✗	✓
Propuesta de solución	✓	✓	✗	✓	✓	✓	✓	✓

Las aplicaciones y los trabajos relacionados presentados en la tabla 3.1, llevan a cabo la proyección inalámbrica de archivos utilizando teléfonos inteligentes. Sin embargo, aunque que estas aplicaciones y estos trabajos relacionados se enfocan a la proyección inalámbrica, existen ciertas desventajas de estas soluciones con respecto a la solución propuesta en esta tesis; entre las cuales se puede mencionar el uso de la tecnología Bluetooth, la cual tiene menos alcance y es menos rápida que la tecnología Wi-Fi utilizada en este trabajo de tesis.

También, se puede mencionar el uso de proyectores inalámbricos, los cuales tienen mayor costo que un proyector digital sin conectividad de red; se menciona esto, ya que la aplicación iProjection de Epson y la aplicación Wireless Projector de Panasonic brindan versatilidad en el manejo de presentaciones y contenidos únicamente cuando se muestran a través de proyectores inalámbricos de la propia marca comercial.

## Capítulo 4. Metodología de solución

---

En este capítulo, se describe de manera general la metodología de solución del sistema denominado "Raspdroid-Projection". Además, se presentan las etapas de desarrollo, actividades y componentes para el desarrollo de esta tesis.

El sistema “Raspdroid-Projection” está estructurado en dos partes: una aplicación cliente instalada en un teléfono inteligente con capacidad Wi-Fi desde la cual el usuario gestiona las presentaciones; y una aplicación servidor que atiende las peticiones del cliente, la cual es creada en una tarjeta Raspberry Pi conectada a través del puerto HDMI a un proyector digital, teniendo como única salida la visualización de las presentaciones multimedia.

Para realizar la comunicación entre la aplicación cliente y la aplicación servidor se lleva a cabo una conexión mediante sockets. En la figura 4.1 se observa la comunicación entre los diferentes componentes que conforman la arquitectura general del sistema “Raspdroid-Projection”.

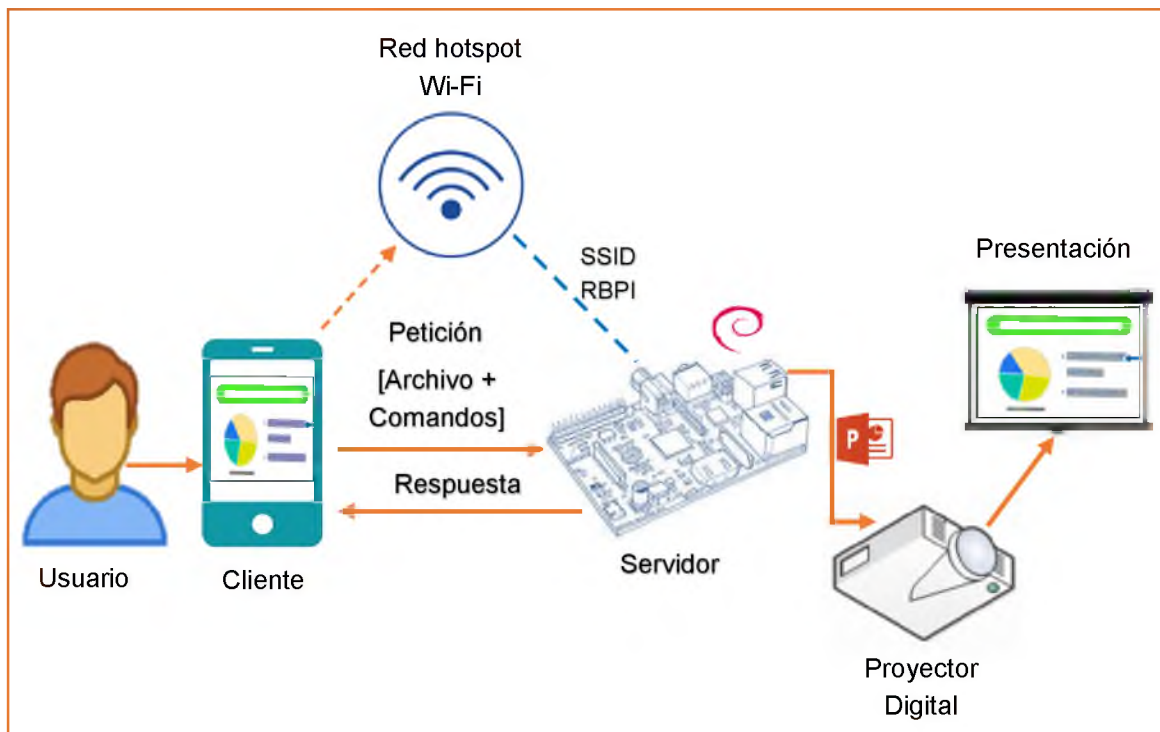


Figura 4.1: Arquitectura general del sistema “Raspdroid-Projection”.

El sistema implementado se divide en seis etapas de desarrollo que se describen de forma detallada en este capítulo. En la figura 4.2 se muestra el esquema de las seis etapas de desarrollo que intervienen en el desarrollo del trabajo de tesis.

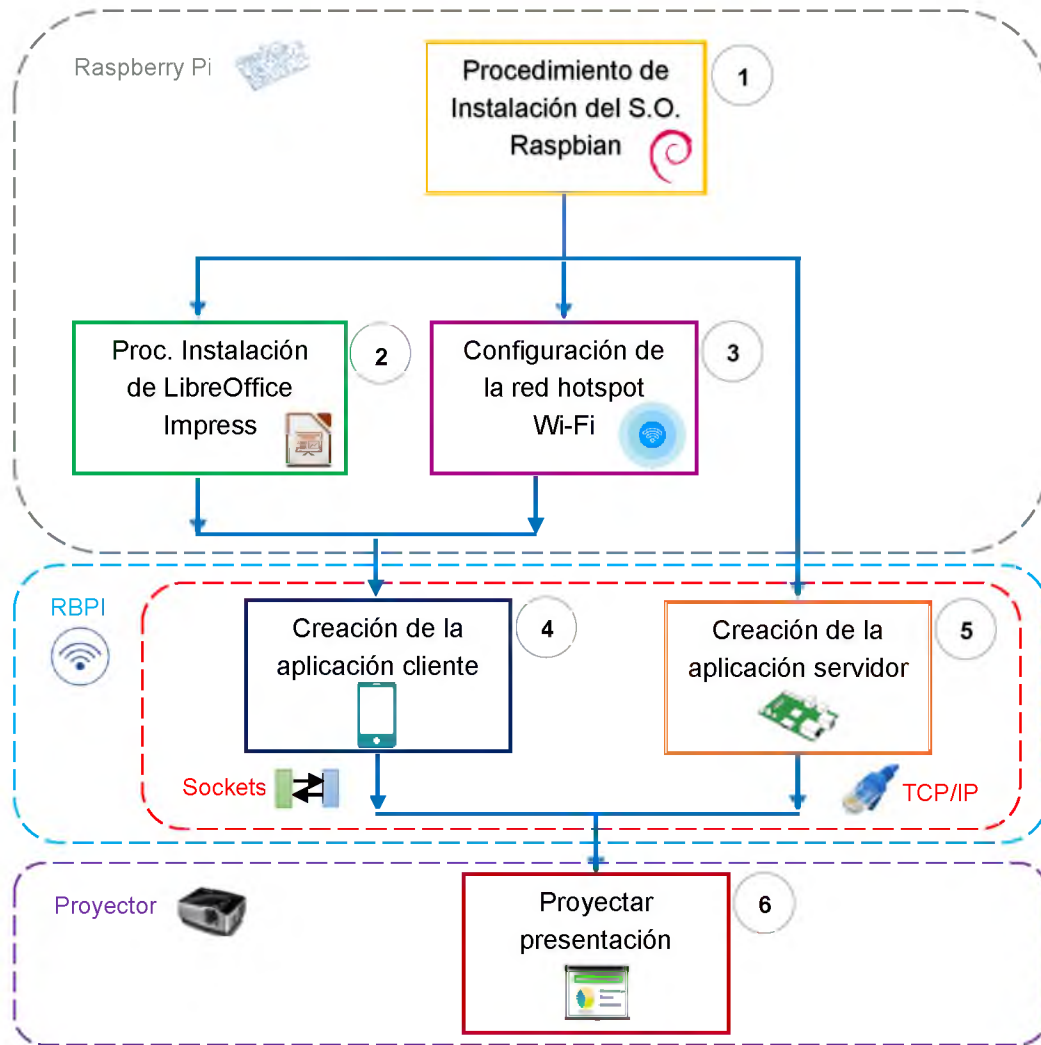


Figura 4.2: Esquema general de las etapas de desarrollo.

A continuación, se describe de forma más detallada cada una de las etapas de desarrollo:

## 4.1 Procedimiento de Instalación del Sistema Operativo Raspbian

El Sistema Operativo elegido para instalar es Raspbian [ver figura 4.3], el cual está optimizado para el hardware de la Raspberry Pi. Este S.O. es una distribución del Sistema Operativo GNU/Linux basado en Debian recomendada por la propia fundación creadora de Raspberry Pi por su facilidad de configuración. Raspbian se encuentra disponible en la página oficial de Raspberry Pi, en la cual se presentan varias distribuciones para descargar e instalar dependiendo de las necesidades del usuario.

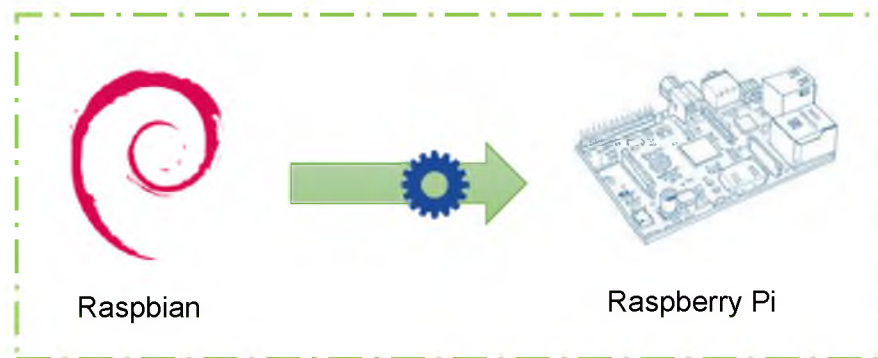


Figura 4.3: Esquema de la instalación de Raspbian en Raspberry Pi.

En la figura 4.4 se muestran y posteriormente se describen las actividades que se llevan a cabo en la primera etapa de desarrollo.

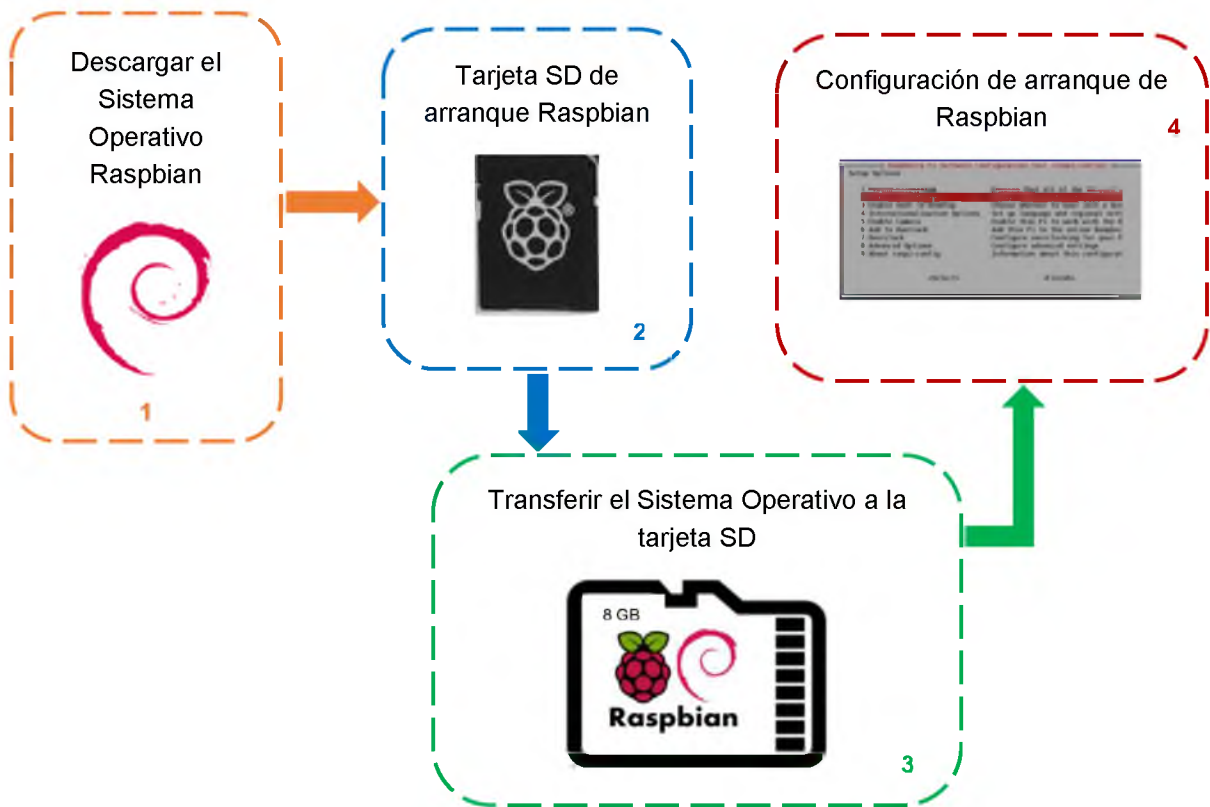


Figura 4.4: Actividades de la instalación del Sistema Operativo Raspbian.

#### 4.1.1 Descargar el Sistema Operativo Raspbian

En esta actividad se lleva a cabo la descarga del sistema operativo Raspbian, el cual, se encuentra disponible en la página oficial de Raspberry Pi. En el archivo descargado se encuentra una imagen ISO. En la figura 4.5 se observa la pantalla de descarga de Raspbian desde la página oficial.

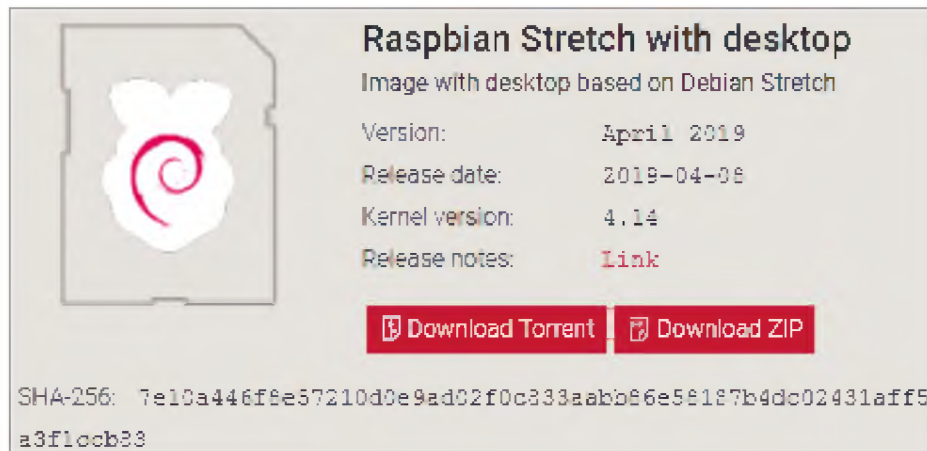


Figura 4.5: Pantalla de descarga del S.O. Raspbian.

#### 4.1.2 Tarjeta SD de arranque Raspbian

En esta actividad se lleva a cabo la preparación de la tarjeta SD para la transferencia de los archivos base del S.O. Raspbian, para ello se utiliza la aplicación SDFormatter para usuarios de Windows. En la figura 4.6 y figura 4.7 se observan las ventanas del proceso de configuración para la preparación de la tarjeta SD.

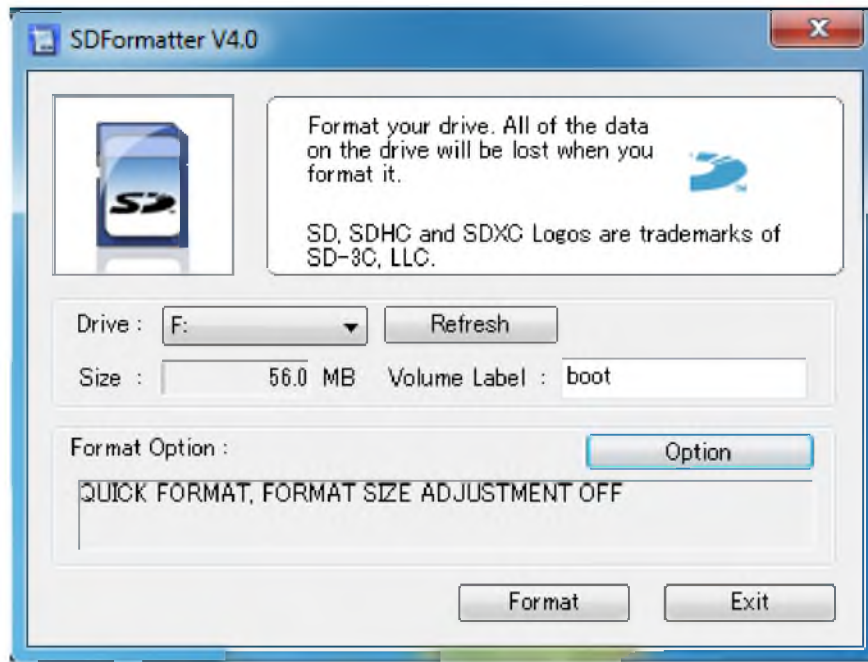


Figura 4.6: SDFormatter V4.0

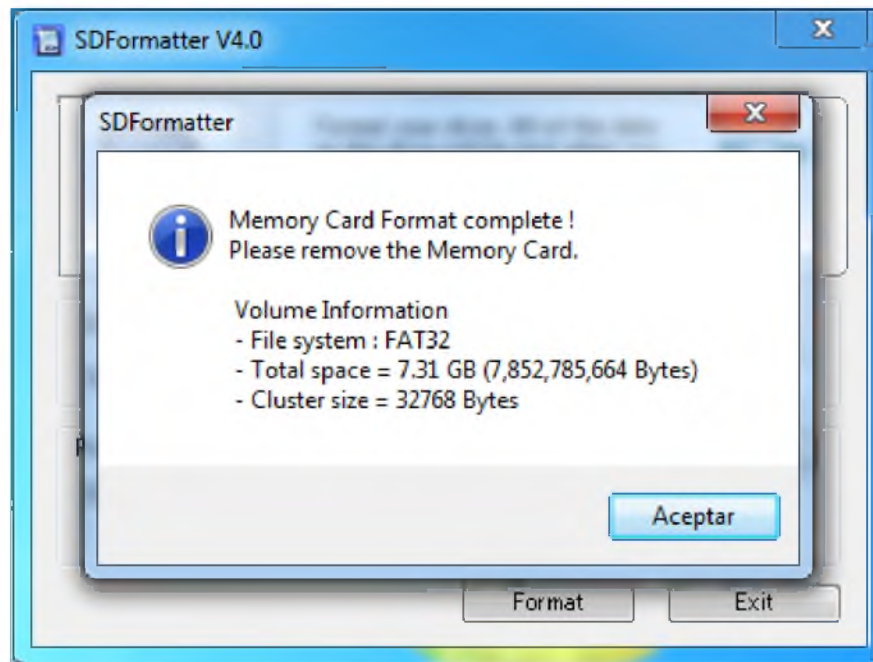


Figura 4.7: Memory Card Format complete.

### 4.1.3 Transferir el Sistema Operativo a la tarjeta SD

En esta actividad se lleva a cabo la transferencia de los archivos base de la imagen ISO en la tarjeta SD utilizando la aplicación Win32DiskImager para usuarios de Windows, la cual se descarga directamente desde su página oficial. Para realizar una correcta transferencia de Raspbian, se utiliza una tarjeta SD con almacenamiento adecuado (más de 2GB) que proporciona un mejor rendimiento de la Raspberry Pi. En la figura 4.8 y figura 4.9 se observan las ventanas del proceso de transferencia de archivos base del S.O. Raspbian.

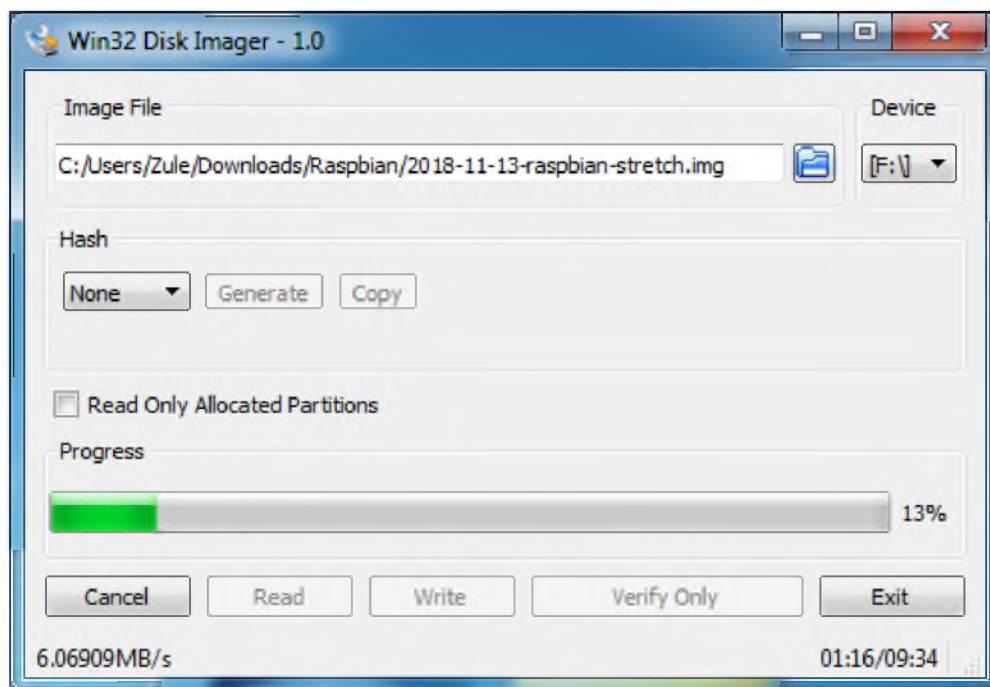


Figura 4.8: Win32 Disk Imager.

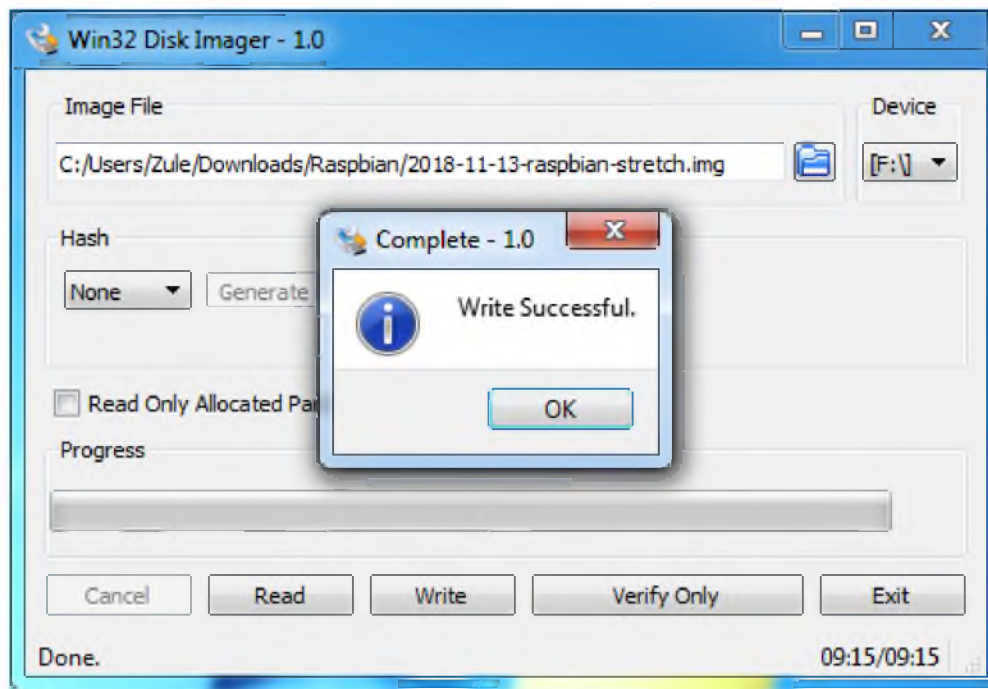


Figura 4.9: Write Successful.

#### 4.1.4 Configuración de arranque de Raspbian

Una vez que el sistema operativo Raspbian se inicia en la Raspberry Pi, se ejecuta automáticamente un asistente de configuración para cambiar la zona horaria y el tipo de teclado, cambiar contraseña, seleccionar una red Wi-Fi, buscar actualizaciones, entre otras configuraciones básicas. En la figura 4.10 y figura 4.11 se muestran dos de las ventanas de configuración básica de arranque de Raspbian.

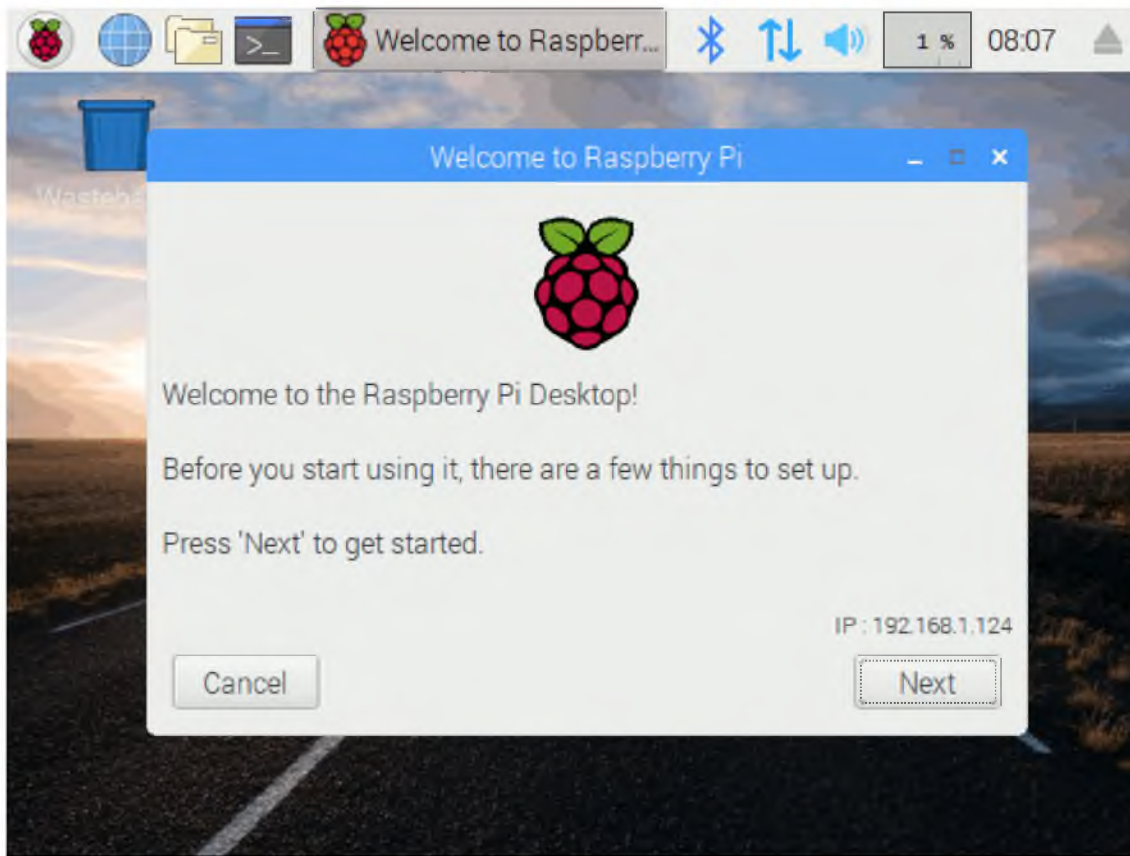


Figura 4.10: Ventana principal de configuración Raspbian.



Figura 4.11: Ventana de configuración de actualización de software.

## 4.2 Proceso de instalación de LibreOffice Impress

En esta etapa de desarrollo se lleva a cabo la instalación del programa LibreOffice Impress del paquete de oficina de LibreOffice como se muestra en la figura 4.12, el cual es utilizado para visualizar las presentaciones multimedia, [ver capítulo 2, sección 2.8 “LibreOffice Impress”].

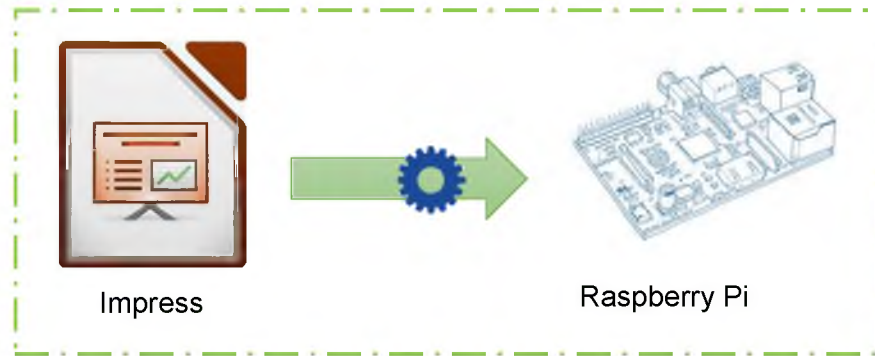


Figura 4.12: Proceso de instalación de LibreOffice Impress en Raspberry Pi.

La instalación de LibreOffice Impress se realiza con la siguiente línea de comando, la cual se ejecuta en la terminal del S.O. Raspbian, como se muestra en la figura 4.13.

```
pi@raspberrypi ~$ sudo apt-get install libreoffice-impress
```

Figura 4.13: Comando para la instalación de LibreOffice Impress.

Al finalizar la instalación, se observa la ventana principal de una presentación con la herramienta Impress de LibreOffice, como se muestra en la figura 4.14.

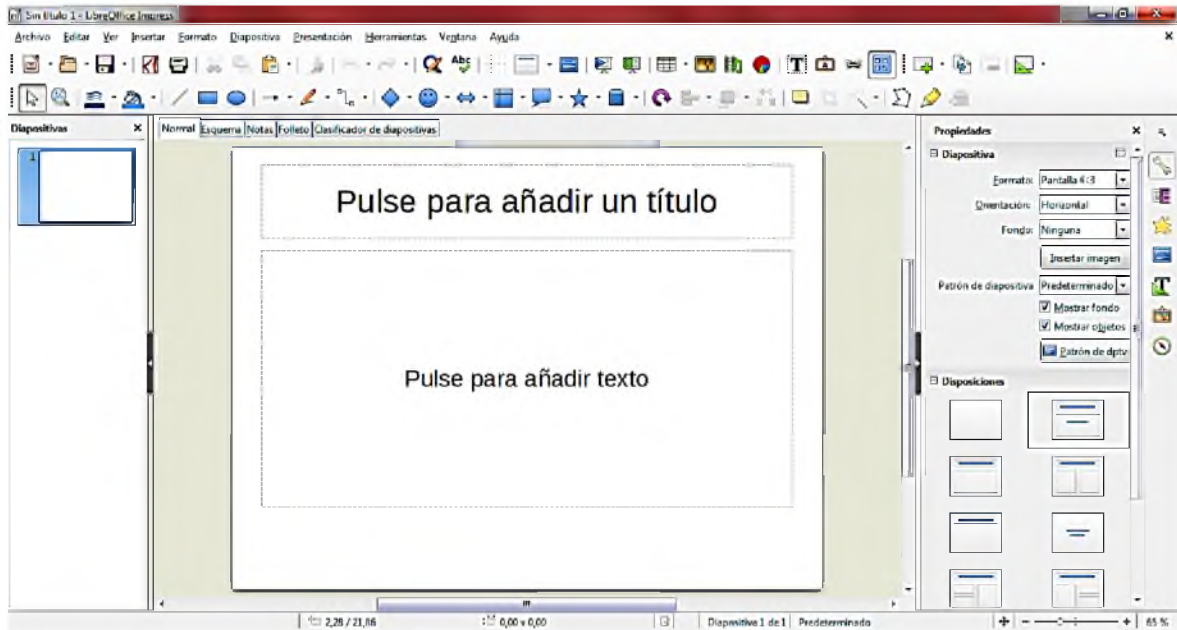


Figura 4.14: Ventana inicial de LibreOffice Impress.

### 4.3 Configuración de la red hotspot Wi-Fi

En esta etapa de desarrollo se lleva a cabo el proceso de configuración de la red *hotspot Wi-Fi* RBPI en la Raspberry Pi. A continuación, se describe el funcionamiento y las actividades de configuración:

Cada dispositivo móvil se conecta a una red de área local (LAN) por medio de la tecnología Wi-Fi, por lo que, se lleva a cabo un proceso de configuración para que la Raspberry Pi funcione como punto de acceso sin conexión a internet, y así, los usuarios puedan conectar de forma inalámbrica sus teléfonos inteligentes. Para llevar a cabo este proceso, se utilizó un adaptador de red USB (Wi-Pi) como interfaz wlan0. Además, se configuró la seguridad de la red RBPI mediante el cifrado WPA2.

Como parte de la configuración e instalación que se realiza en la Raspberry Pi, la red *hotspot Wi-Fi* actúa como un servidor DHCP asignando una dirección IP a los teléfonos inteligentes

que se conecten a ella. En la figura 4.15 se muestra el esquema del funcionamiento de la red *hotspot Wi-Fi*.

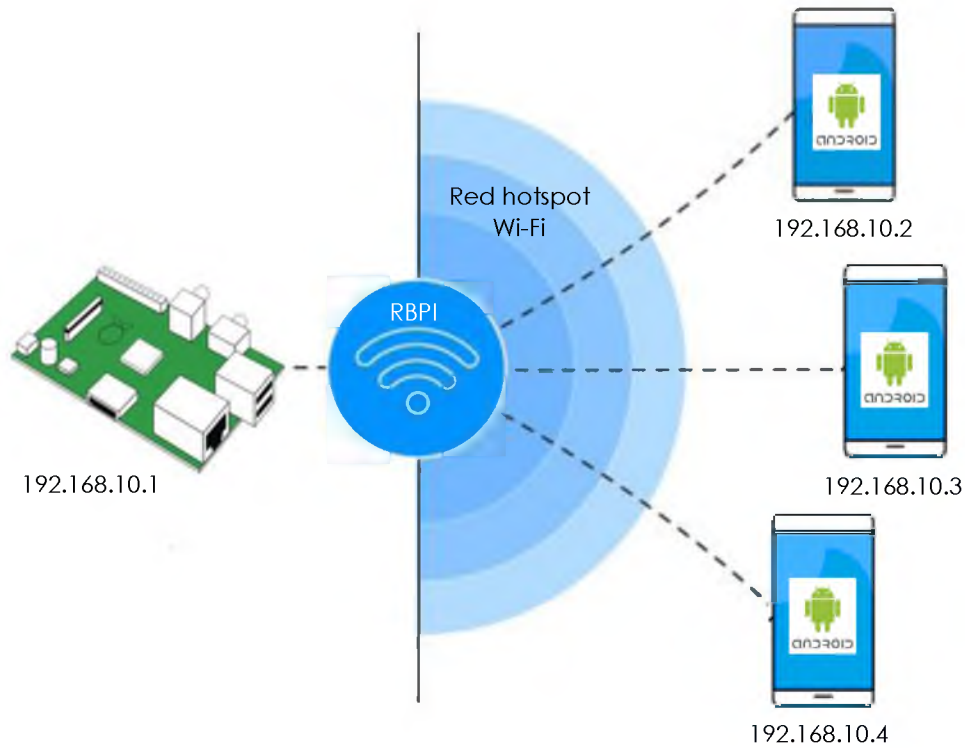


Figura 4.15: Esquema del funcionamiento de la red hotspot Wi-Fi.

En la figura 4.16 se muestran y posteriormente se describen las actividades que se llevan a cabo en esta etapa de desarrollo.

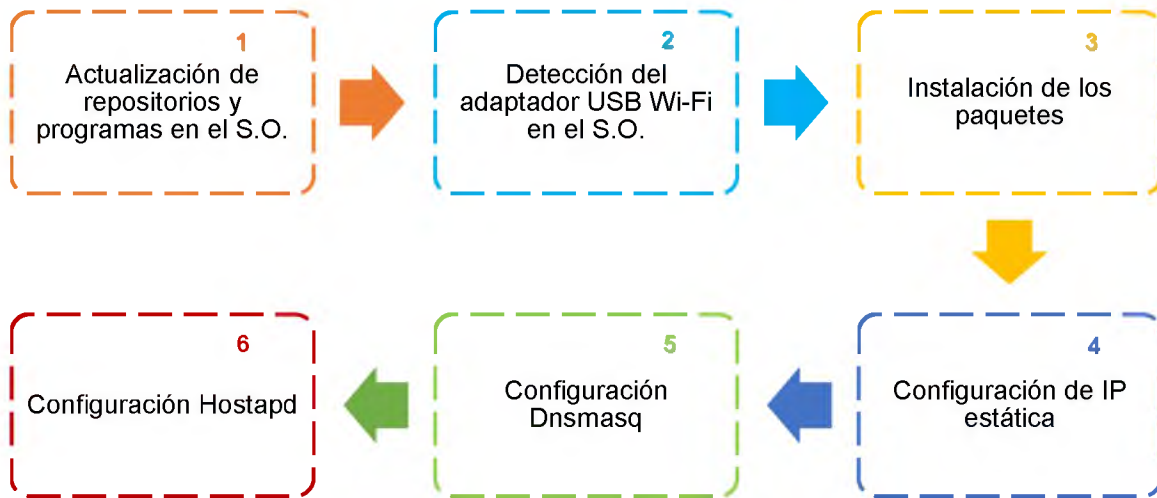


Figura 4.16: Actividades de la configuración de la red hotspot Wi-Fi.

### 4.3.1 Actualización de repositorios y programas en el S.O.

En esta actividad se lleva a cabo la actualización de los repositorios, la actualización de Raspbian y las aplicaciones que están contenidas en los repositorios. Esta actualización se realiza con los siguientes comandos, los cuales se ejecutan en la terminal del S.O. Raspbian, como se muestra en la figura 4.17.

```
pi@raspberrypi ~$ sudo apt-get update  
pi@raspberrypi ~$ sudo apt-get upgrade
```

Figura 4.17: Comandos de actualización de repositorios y programas.

### 4.3.2 Detección del adaptador USB Wi-Fi en el S.O.

En esta actividad se lleva a cabo la detección del adaptador siguiendo el proceso que se describe a continuación: Al conectar el dispositivo, el núcleo del sistema comprueba que el adaptador USB Wi-Fi (Wi-Fi), [ver figura 4.18], es detectado y que está activado en modo AP.



Figura 4.18: Adaptador USB Wi-Fi.

En la figura 4.19 se muestra el comando para detectar en el S.O. el adaptador USB Wi-Fi:

```
pi@raspberrypi ~$ lsusb
```

Figura 4.19: Comando para detección del adaptador USB Wi-Fi.

### 4.3.3 Instalación de los paquetes

En esta actividad se realiza la instalación de los paquetes o servicios requeridos: *hostapd* y *dnsmasq* con el comando que se muestra en la figura 4.20. A continuación, se describen de forma breve éstos paquetes o servicios:

- Hostapd: Es un demonio (servicio) o paquete que permite utilizar el Wi-Fi incorporado en una Raspberry Pi como un punto de acceso inalámbrico.
- Dnsmasq: Actúa como un servidor DHCP y DNS combinado para asignar direcciones IP y procesar las solicitudes de DNS a través de la propia Raspberry Pi.

```
pi@raspberrypi ~$ sudo apt-get install dnsmasq hostapd
```

Figura 4.20: Comando de instalación de dnsmasq y hostapd.

Para realizar la configuración del punto de acceso inalámbrico se recomienda detener los servicios *dnsmasq* y *hostapd* como se muestra en la figura 4.21.

```
pi@raspberrypi ~$ sudo systemctl stop dnsmasq
pi@raspberrypi ~$ sudo systemctl stop hostapd
```

Figura 4.21: Comandos para detener los servicios dnsmasq y hostapd.

### 4.3.4 Configuración de IP estática

En esta actividad se realiza la configuración de la interfaz wlan0 con la IP estática 192.168.10.1, la cual se asigna a la Raspberry Pi. Para esto, se edita el archivo de configuración *dhcpcd* con el comando que se muestra en la figura 4.22.

```
pi@raspberrypi ~$ sudo nano /etc/dhcpcd.conf
```

Figura 4.22: Comando para abrir el archivo de configuración dhcpcd.

Y se agregan las siguientes líneas al final del archivo:

```
interface wlan0
static ip_address 192.168.10.1/24
```

Después de configurar el archivo se reinicia el servicio *dhcpcd* para aplicar los cambios, ejecutando el comando que muestra la figura 4.23.

```
pi@raspberrypi ~$ sudo service dhcpcd restart
```

Figura 4.23: Comando para reiniciar dhcpcd.

### 4.3.5 Configuración Dnsmasq

En esta actividad se lleva a cabo la configuración del servidor DHCP proporcionado por *dnsmasq* para configurar el rango de direcciones IP que el servidor emitirá. Para llevar a cabo esta configuración, se recomienda cambiar el nombre del archivo de configuración “dnsmasq.conf” y crear uno nuevo con los comandos que se muestran en la figura 4.24.

```
pi@raspberrypi ~$ sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.old
pi@raspberrypi ~$ sudo nano /etc/dnsmasq.conf
```

Figura 4.24: Comandos de configuración Dnsmasq.

En el archivo de configuración se agregan las siguientes líneas:

```
interface=wlan0
dhcp-range=192.168.10.2,192.168.10.255,255.255.255.0,12h
```

### 4.3.6 Configuración Hostapd

En esta actividad se configura el punto de acceso Wi-Fi con *hostapd* a través del archivo de configuración “hostapd.conf”. En este archivo se agregan diversos parámetros para la configuración de la red inalámbrica RBPI. La figura 4.25 muestra el comando que abre tal archivo.

```
pi@raspberrypi ~$ sudo nano /etc/hostapd/hostapd.conf
```

Figura 4.25: Comando para abrir el archivo de configuración hostapd.

Posteriormente, se agregan en el archivo de configuración las siguientes líneas:

```

interface=wlan0 #Nombre de la tarjeta de red Wi-Fi
driver=nl80211 #Driver utilizado por el servicio hostapd, normalmente es
nl80211, que soporta muchas tarjetas de red
ssid=RBPI #Nombre de la red
hw_mode=g #Modo de operación (a = IEEE 802.1a (11mbps), b = IEEE 802.11b
(22mbps), g = IEEE 802.11g (54mbps), Default: IEEE 802.11b (banda 2.4 GHz
channel=6 #Número de canal (IEEE 802.11) utilizado por la red Wi-Fi creada
ieee80211n=1 #ieee80211n si está en 1, el modo del Wi-Fi es n (104mbps)
wmm_enabled=1 #Si está a 1, se activa el modo WMM para funcionalidad HT
high throughput (alto rendimiento) completa. Esto activa sistemas más
potentes de modulación de la señal y uso de frecuencias que permite alcanzar
velocidades más altas
#ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40] #Habilita canales de 40MHz con
un intervalo de protección de 20ns. Aquí se muestran los diferentes
indicadores que quieren activar al utilizar HT. Esto depende fuertemente
de la tarjeta Wi-Fi utilizada y su driver
macaddr_acl=0 #Acepta todas las direcciones MAC
auth_algs=1 #Utiliza la autenticación WPA
ignore_broadcast_ssid=0 #Exige a los clientes que conozcan el nombre de la
red
wpa=2 #Utiliza WPA2
wpa_key_mgmt=WPA-PSK #Utiliza una clave pre-compartida
wpa_passphrase=raspberry #Contraseña de la red
rsn_pairwise=CCMP #Utiliza AES, en lugar de TKIP

```

A continuación, se indica a *hostapd* la ubicación del archivo de configuración “hostapd.conf” para vincularlo como predeterminado. En la figura 4.26 se muestra el comando para abrir el archivo en el cual se vincula el archivo de configuración “hostapd.conf”.

```

pi@raspberrypi ~$ sudo nano /etc/default/hostapd

```

Figura 4.26: Comando para abrir el archivo hostapd.

Y se asigna a la línea # DAEMON\_CONF = "" lo siguiente:

```

DAEMON_CONF="/etc/hostapd/hostapd.conf"

```

Por último, se deben iniciar los servicios después de la configuración para efectuar los cambios, ejecutando los comandos que se muestran en la figura 4.27.

```
pi@raspberrypi ~$ sudo systemctl start hostapd
pi@raspberrypi ~$ sudo systemctl start dnsmasq
```

Figura 4.27: Comandos para iniciar los servicios hostapd y dnsmasq.

### 4.4 Creación de la aplicación cliente

En esta etapa de desarrollo se lleva a cabo la creación de la aplicación cliente. Ésta corresponde a la aplicación instalada en un teléfono inteligente con sistema operativo Android, como se muestra en la figura 4.28. Para la codificación de la aplicación Android se ha utilizado el lenguaje de programación Java + XML bajo el entorno de desarrollo Android Studio versión 3.2.1, siendo el entorno oficial que recomienda Google para la realización de las aplicaciones Android.

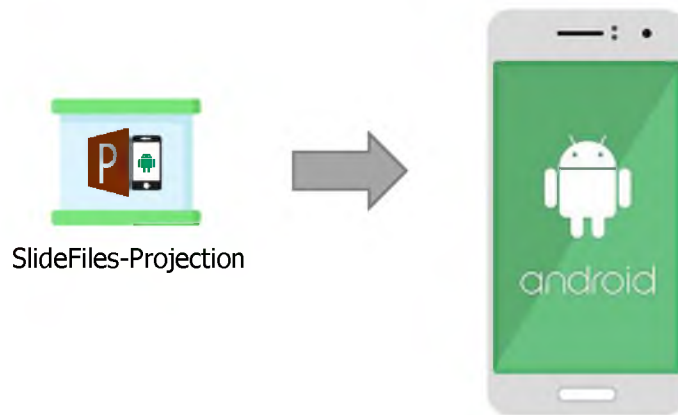


Figura 4.28: Instalación de la App cliente.

En la figura 4.29 se muestra el funcionamiento de la aplicación cliente.

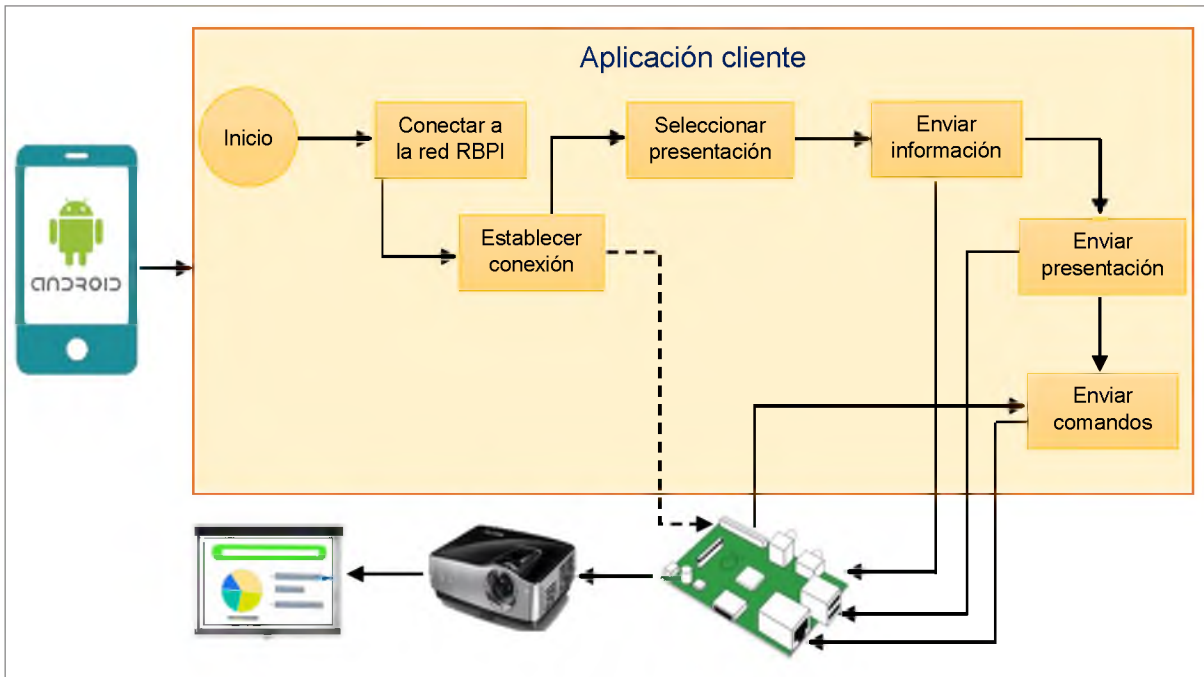


Figura 4.29: Esquema de funcionamiento de la aplicación cliente.

En la figura 4.30 se muestran y se describen las actividades que se llevan a cabo en esta etapa de desarrollo.

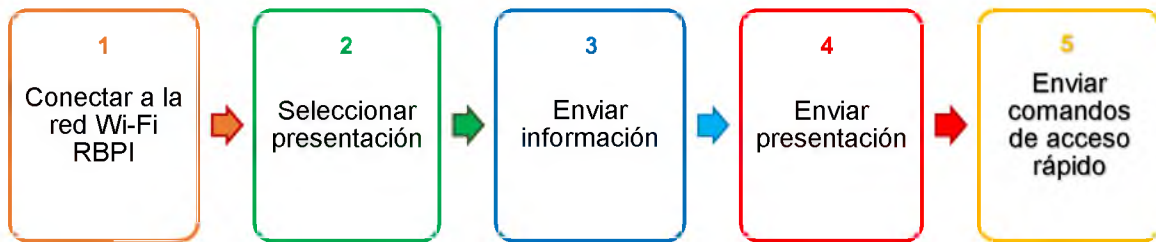


Figura 4.30: Actividades de la creación de la aplicación cliente.

#### 4.4.1 Conectar a la red Wi-Fi RBPI

En esta actividad la aplicación cliente se conecta a la red *hotspot Wi-Fi* RBPI [ver figura 4.31] a través de un método para la conexión a puntos de acceso inalámbricos. Este método realiza una búsqueda entre los SSID que están en el rango de acceso en los cuales se encuentra la red RBPI, nombre que posee el SSID de la red que emite la Raspberry Pi.

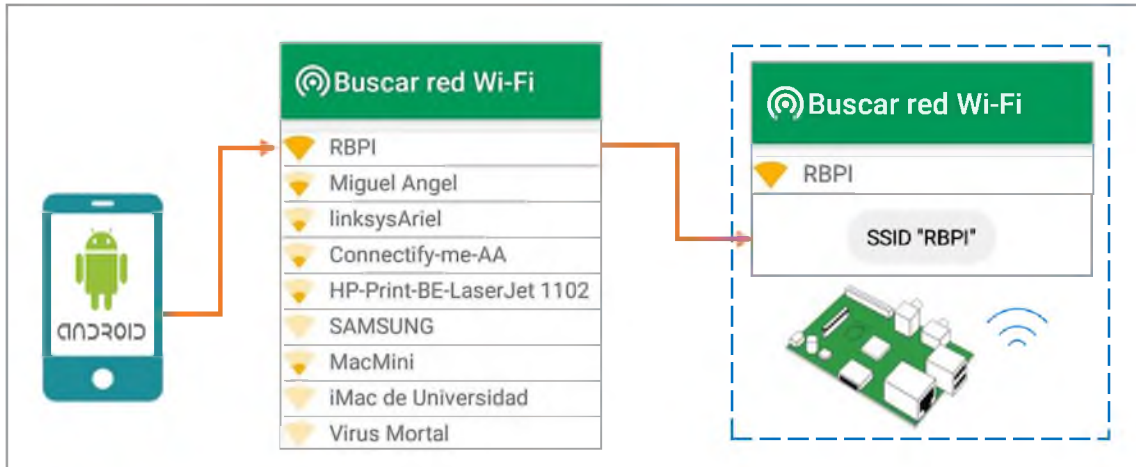


Figura 4.31: Conexión a la red Wi-Fi RBPI.

#### 4.4.2 Seleccionar presentación

Para examinar documentos, imágenes u otros tipos de archivos en los directorios, se utilizó el método ACTION\_GET\_CONTENT, el cual es utilizado para abrir el proveedor de archivos predeterminado en teléfonos inteligentes Android mediante el método `intent.setType("**/*")`, permitiendo seleccionar un archivo con cualquier extensión disponible, obteniendo así, el nombre del archivo y ruta del identificador de recursos uniforme (URI), a través de un proveedor de contenido como se visualiza en la figura 4.32.



Figura 4.32: Diagrama de navegación de directorios.

### 4.4.3 Enviar información básica

Una vez establecida la conexión entre la aplicación cliente y la aplicación servidor a través de la red RBPI [ver sección 4.4.1 “Conectar a la red Wi-Fi RBPI”], el primer mensaje de comunicación que envía la aplicación cliente a la aplicación servidor es una trama definida de cadena de caracteres en formato JSON, llevado a cabo en un hilo que se ejecuta de forma aleatoria conforme a las necesidades de la aplicación cliente y la aplicación servidor.

Este JSON contiene la información básica necesaria del archivo de presentación multimedia [nombre del archivo, tamaño (en bytes) y la extensión]; el cual se envía para reservar un espacio en memoria optimizando con ello el rendimiento de la tarjeta Raspberry Pi, como se muestra en la figura 4.33.

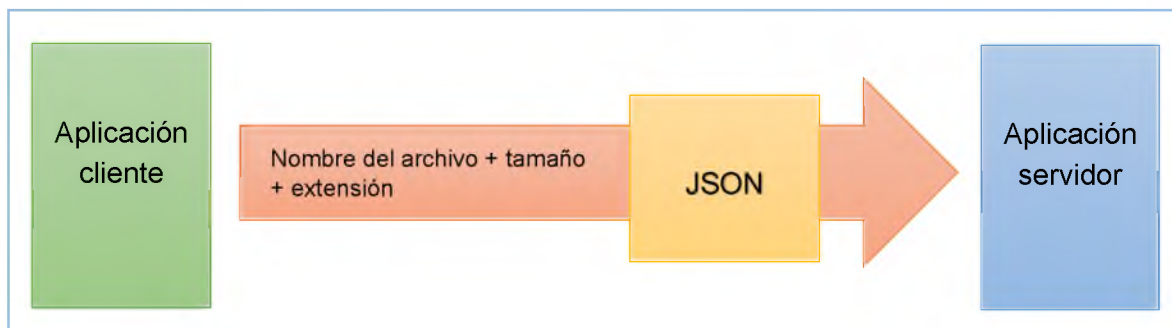


Figura 4.33: Diagrama del envío de la información básica.

A continuación, se muestra la estructura del formato JSON que permite enviar la información básica:

```
{“nombre”: “Presentación1”, “tamaño”: “10000”, “extensión”: “pptx”}
```

### 4.4.4 Enviar presentación

La aplicación cliente envía a la aplicación servidor la presentación multimedia que se desea proyectar como se observa en la figura 4.34, a través del método “sendFile()” en segundo plano, el cual es gestionado en un hilo que se ejecuta de forma aleatoria, cuando el hilo principal de la aplicación cliente deja de ejecutarse, la aplicación servidor continua activa (ejecutándose).

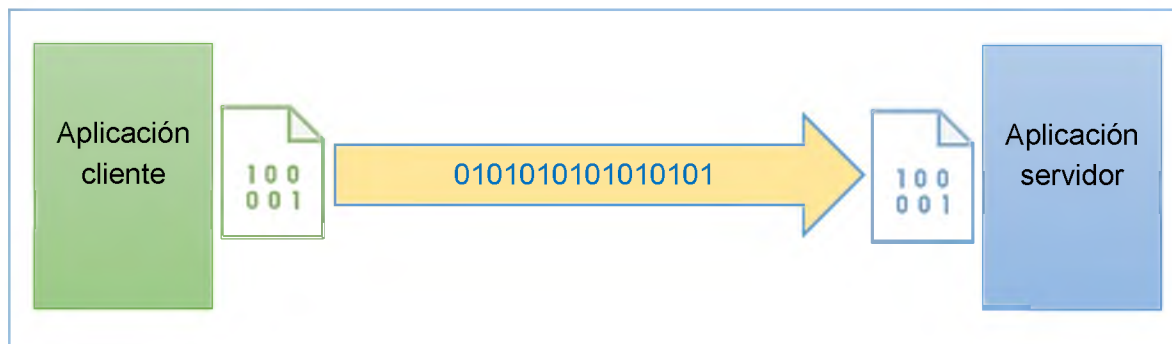


Figura 4.34: Envío de la presentación multimedia.

### 4.4.5 Enviar comandos de acceso rápido

Cuando el archivo de presentación multimedia es recibido completamente por la aplicación servidor, la aplicación cliente manda la petición “abrir el archivo” mediante el comando “OPN”. En ese momento, la aplicación servidor abre el archivo y se inicia el proceso de enviar peticiones de tipo cliente-servidor, donde cada petición es una acción que se realiza por medio de comandos sobre el documento recibido. Para llevar a cabo este proceso, la aplicación

cliente envía un segundo mensaje en formato JSON, el cual contiene la información necesaria [nombre del archivo y el comando] utilizada para realizar la gestión del archivo de presentación multimedia, como se muestra en la figura 4.35.

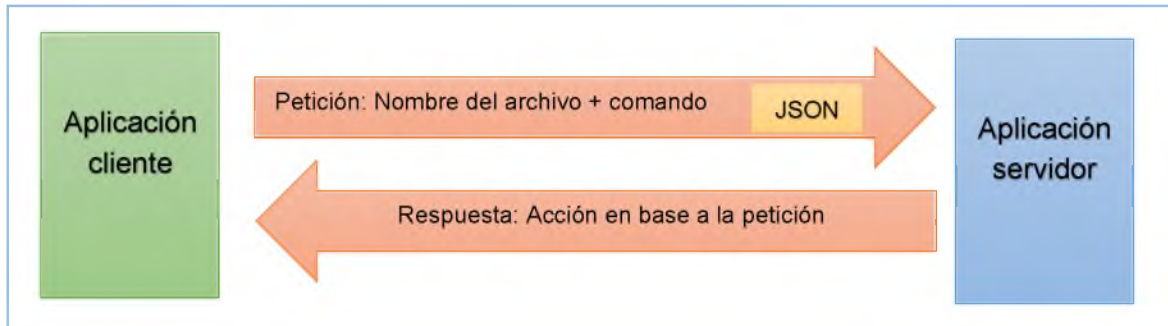


Figura 4.35: Diagrama del envío de comandos.

Este envío de la trama definida de cadena de caracteres en formato JSON, es llevado a cabo en un hilo que se ejecuta de forma aleatoria al primer mensaje en formato JSON [ver sección 4.4.3 “Enviar información básica”].

La cadena en formato JSON se enviará utilizando el mismo formato de mensaje anterior [ver sección 4.4.3 “Enviar información básica”].

```
{“nombre”: “Presentación1”, “nemónico”: “OPN”}
```

Donde:

- “nombre”: Es el nombre del archivo de presentación multimedia.
- “nemónico”: Corresponde al comando, el cual es elegido y creado, utilizando una combinación de tres letras mayúsculas abreviadas en base al nombre de la petición en inglés.

El nemónico (comando) enviado en cada formato JSON variará en función del tipo de petición que se quiera realizar sobre la presentación multimedia recibida por la aplicación servidor. En la tabla 4.1, se muestran las peticiones con sus respectivos comandos/nemónicos de acceso rápido permitidos y la acción correspondiente al comando.

Tabla 4.1: Comandos enviados en formato JSON.

Petición	Comando (Nemónico)	Acción
Abrir	OPN	Abre la presentación
Cerrar	CLS	Cierra la presentación
Ejecutar	EXE	Pone en modo presentación
Maximizar	MAX	Maximiza la presentación
Minimizar	MIN	Minimiza la presentación

### 4.4.5.1 Envío de comandos por medio de la API de LibreOffice

Impress Remote es una aplicación de código abierto, disponible para los sistemas operativos Android e iOS, la cual permite controlar las presentaciones de diapositivas de LibreOffice Impress a través de un dispositivo móvil.

Para hacer uso de la aplicación Impress Remote, se especifica una serie de requisitos que se deben cumplir para su correcto funcionamiento:

Equipo:

- GNU/Linux, Windows o macOS.
- Conectividad de red inalámbrica.
- LibreOffice versión 4.1 (o superior).

Dispositivo móvil:

- Teléfono inteligente con una versión de Android 2.3 (o superior) o iOS, con conectividad de red.
- La aplicación Impress Remote instalada en el dispositivo móvil.

### 4.4.5.2 Activar control remoto en LibreOffice Impress

Para ejecutar una presentación de diapositivas de Impress, se debe permitir que Impress sea controlado por el dispositivo móvil, para ello se debe activar el control remoto de Impress de la siguiente manera:

Abrir LibreOffice Impress, dirigirse a *Herramientas > Opciones > LibreOffice Impress > General*, a continuación, se debe ver la pantalla mostrada en la figura 4.36, y en la opción de *Presentación*, seleccionar la casilla **Activar el control remoto**, pulsando en Aceptar.

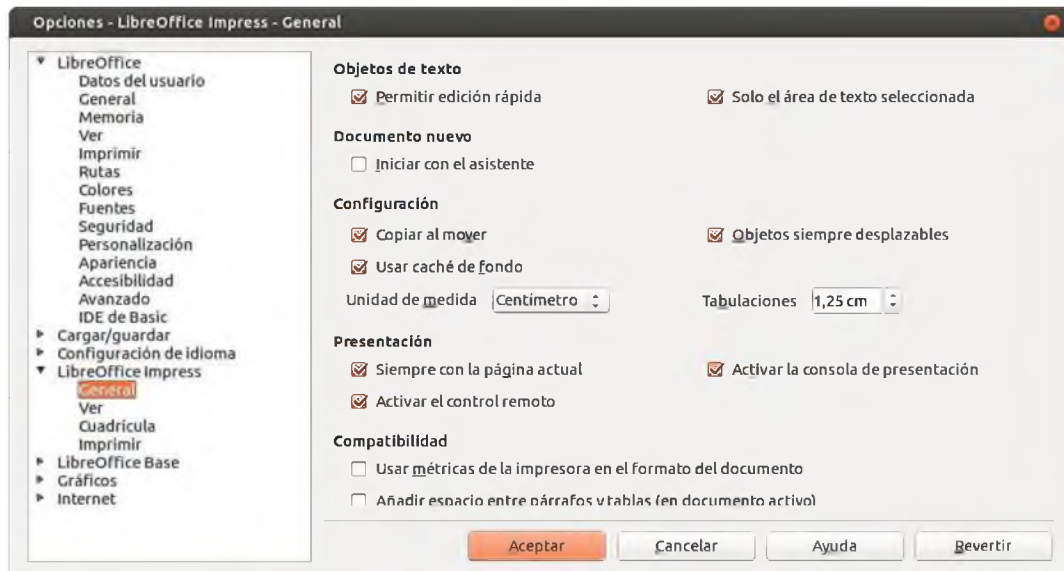


Figura 4.36: Pantalla para activar el control remoto de Impress.

### 4.4.5.3 Conexión del equipo con el dispositivo móvil

La conexión que se establece entre el equipo que ejecuta la presentación de LibreOffice Impress y el dispositivo móvil se realiza a través de un enlace de una red LAN, para ello, es necesario que tanto el equipo como el dispositivo estén conectados a la red *hotspot Wi-Fi RBPI* [ver sección 4.4.1 “Conectar a la red Wi-Fi RBPI”], cuando se haya establecido el emparejamiento, el dispositivo móvil estará preparado para controlar la presentación.

Para conectar la aplicación Android y el equipo a través de la red Wi-Fi, se necesita activar las funcionalidades experimentales de LibreOffice, para ello es necesario realizar la siguiente acción: Abrir LibreOffice Impress, dirigirse a *Herramientas > Opciones > LibreOffice > Avanzado*, tal como se presenta en la figura 4.37, y en la opción de *Funcionalidades opcionales*, seleccionar la casilla **Activar funcionalidades experimentales**, pulsando en Aceptar.

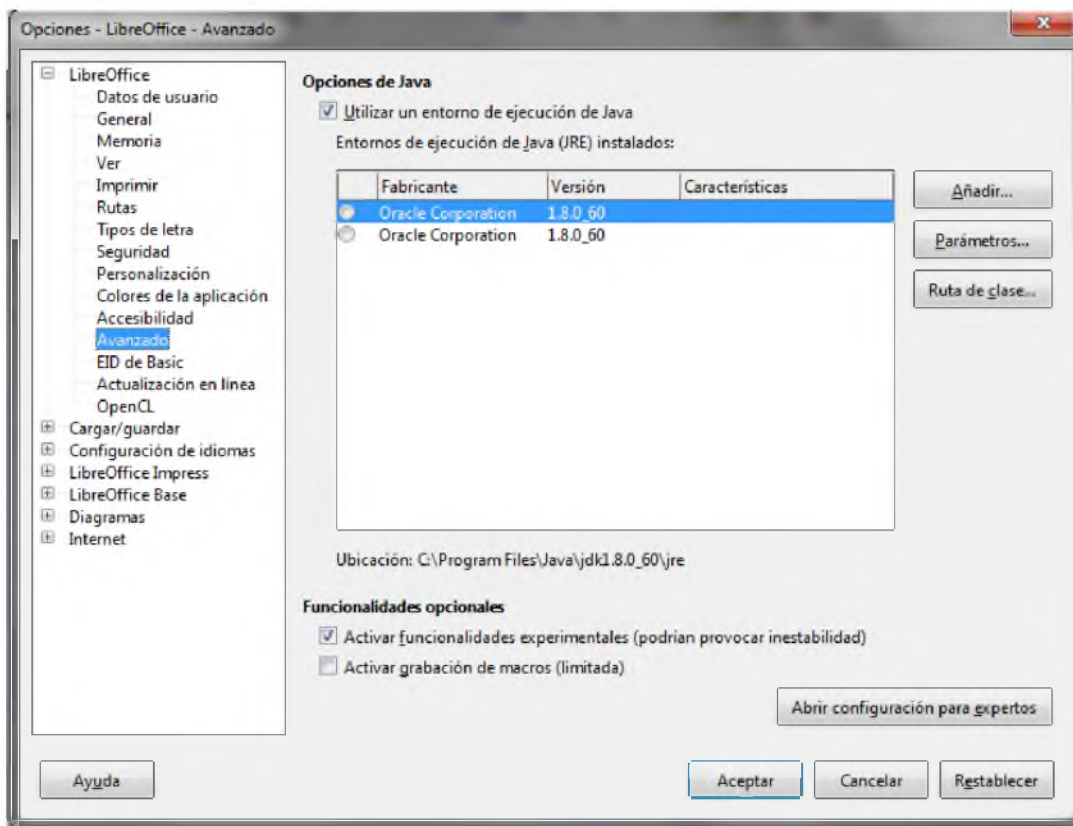


Figura 4.37: Pantalla para activar funcionalidades experimentales de Impress.

#### 4.4.5.4 Controlar la presentación de diapositivas

- Asegurar que tanto el teléfono móvil como el equipo que ejecuta la presentación de LibreOffice Impress estén emparejados a través de la conexión de la red *hotspot Wi-Fi* RBPI [ver sección 4.4.1 “Conectar a la red Wi-Fi RBPI”].
- Abrir la presentación multimedia de LibreOffice Impress que se desea mostrar.
- Una vez que se haya seleccionado el nombre del servidor de LibreOffice, se cargan automáticamente las diapositivas de la presentación en Impress Remote con la sección de notas.
- Deslizar hacia la izquierda o hacia la derecha en la pantalla del dispositivo móvil para cambiar las diapositivas. Las notas de las diapositivas se muestran en la parte inferior de la pantalla del teléfono.

### 4.4.5.5 Comandos de Impress Remote

La API de LibreOffice proporciona comandos que permiten gestionar remotamente las presentaciones multimedia, estos comandos son ejecutados cuando se abre la presentación.

Utilizando el siguiente formato de mensaje:

"MESSAGE\n\n" or "MESSAGE\nDATA\nDATA2...\n\n"

Un mensaje consiste en una o más líneas. La primera línea es la descripción del mensaje, las otras líneas pueden agregar cualquier dato necesario. Se debe seguir leyendo un mensaje hasta que se llegue a una línea vacía con la cual concluye el mensaje.

A continuación, en la tabla 4.2 se muestran los comandos utilizados y la acción que corresponde a cada uno:

Tabla 4.2: Comandos de Impress Remote.

Comando	Acción
transition_next	Avanzar presentación
transition_previous	Retroceder presentación
presentation_start	Iniciar presentación
presentation_stop	Detener presentación

## 4.5 Creación de la aplicación servidor

En esta etapa de desarrollo se lleva a cabo la creación de la aplicación servidor. Esta aplicación es creada en la tarjeta Raspberry Pi, su diseño se muestra en la figura 4.38. Se ha decidido crear en la Raspberry Pi debido a su reducido tamaño. Para el desarrollo de la aplicación servidor se ha utilizado el lenguaje de programación Java bajo el entorno de desarrollo NetBeans versión 8.1 y la versión 8 del JDK de Java.

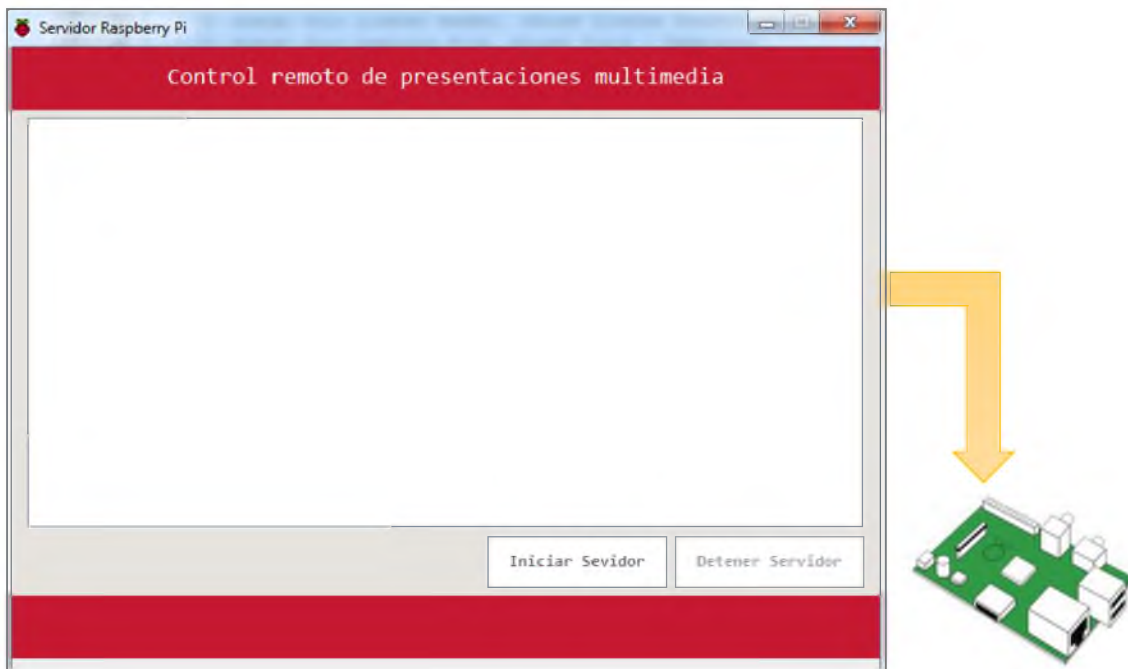


Figura 4.38: Aplicación servidor creada en la Raspberry Pi.

La aplicación servidor es la encargada del establecimiento de conexiones con los clientes que estén conectados a la red *hotspot Wi-Fi* RBPI [ver sección 4.4.1 “Conectar a la red Wi-Fi RBPI”] y de atender sus peticiones cuando se conecten a ella [figura 4.40]. Se basa en un `ServerSocket` que está permanentemente esperando nuevas conexiones. Cada conexión es tratada en un hilo independiente, permitiendo hasta N conexiones de manera simultánea, teniendo en cuenta que el límite se define en el rango que ofrece la red *hotspot Wi-Fi*, pero solo puede servir a un único cliente. Cuando llega una nueva conexión, la aplicación servidor espera hasta que recibe la presentación procedente de la aplicación móvil, además, de la cadena de comandos de control, donde se debe especificar el tipo de petición para realizar la tarea solicitada. Dependiendo del tipo de petición, el servidor ejecutará la tarea correspondiente; en ese momento comienza el periodo donde se puede gestionar la presentación multimedia.

En la figura 4.39 se muestra el esquema de funcionamiento de la aplicación servidor y en la figura 4.40 se muestra el esquema del establecimiento de conexiones, respectivamente.

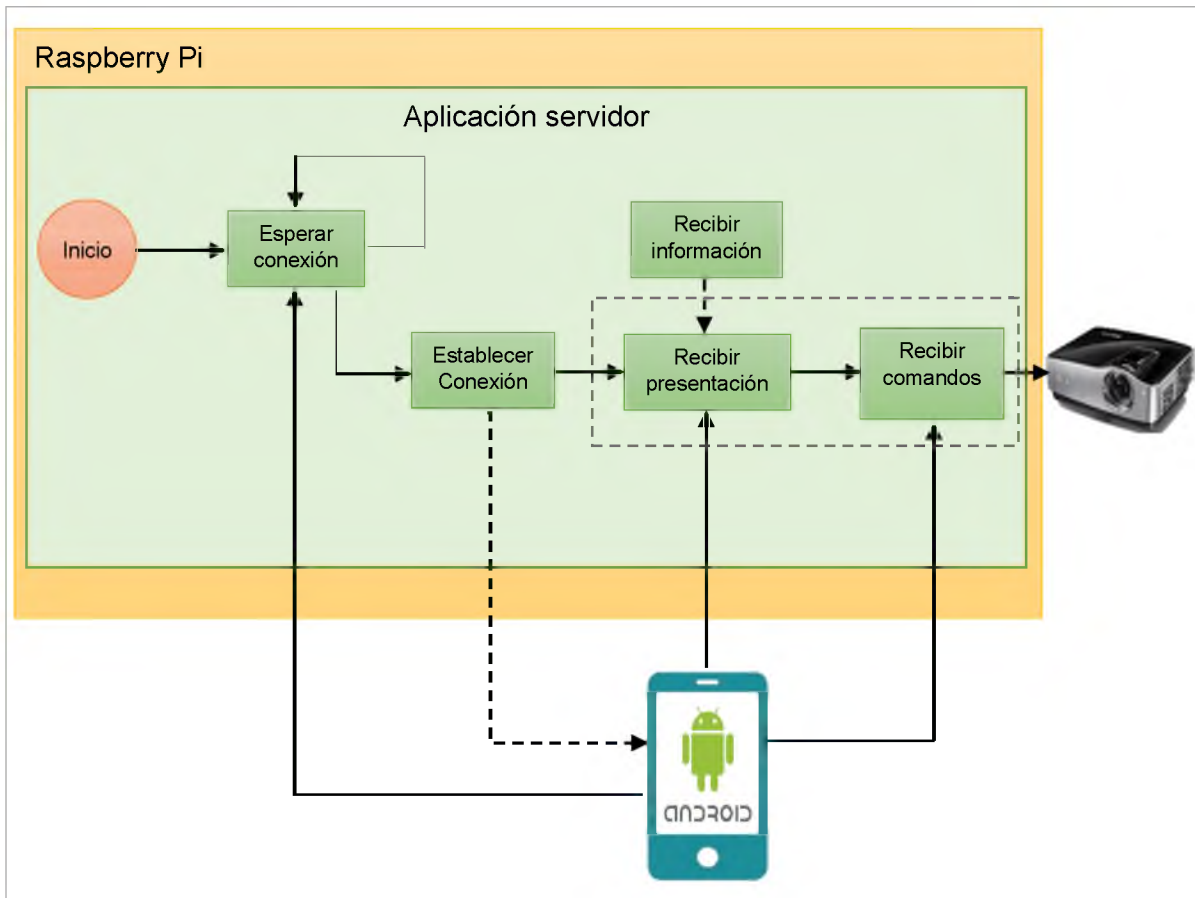


Figura 4.39: Esquema de funcionamiento de la aplicación servidor.

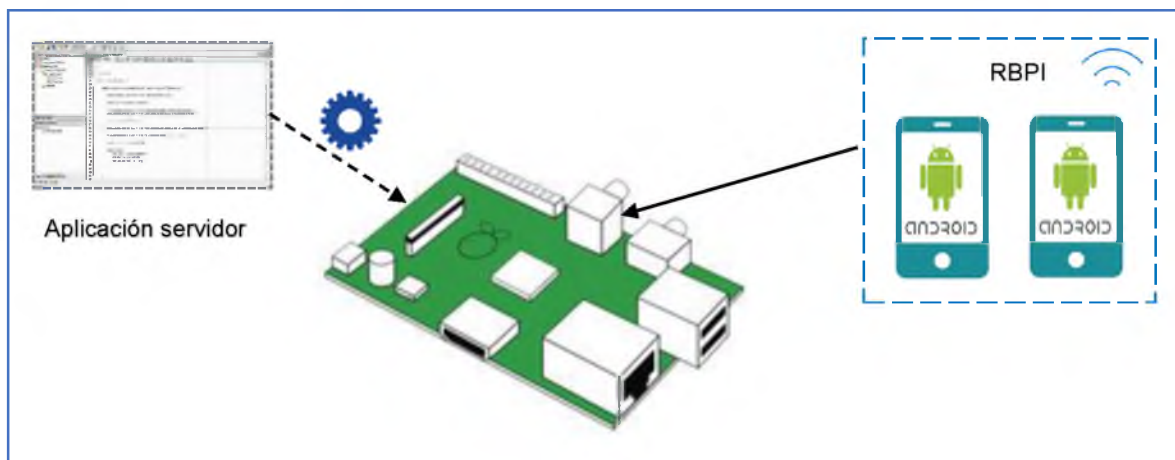


Figura 4.40: Esquema del establecimiento de conexiones.

## 4.6 Comunicación mediante Sockets

### 4.6.1 Arquitectura cliente – servidor

En un nivel básico, los sistemas basados en la red constan de un servidor, un cliente y un medio para la comunicación, como se muestra en la figura 4.41. Una computadora o un dispositivo que ejecuta un programa que realiza una solicitud de servicios se denomina cliente. Una computadora o un dispositivo que ejecuta un programa que ofrece servicios solicitados de uno o más clientes se llama servidor.

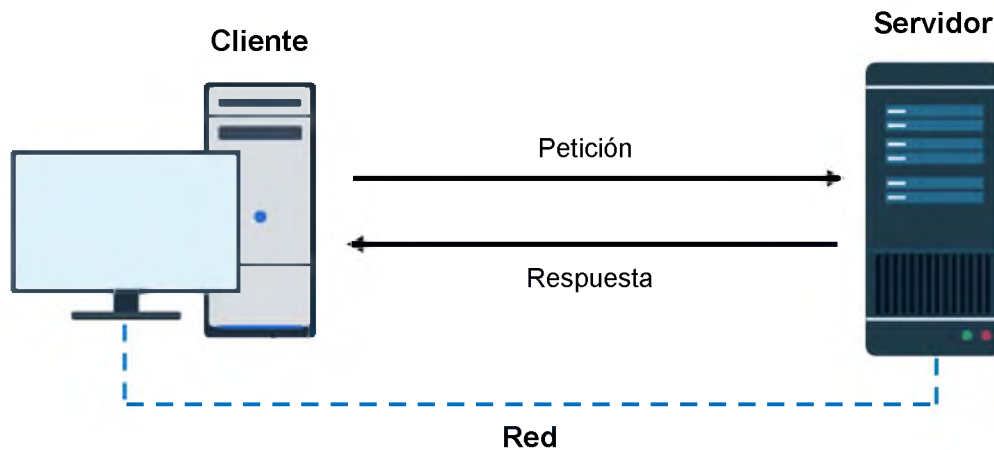


Figura 4.41: Arquitectura cliente-servidor.

Para llevar a cabo la comunicación de este tipo de arquitectura se utilizan sockets para mandar y recibir flujo de datos.

### 4.6.2 Comunicación cliente-servidor a través de sockets

La forma de establecer la comunicación es por medio de sockets, los cuales son mecanismos que permiten la comunicación entre procesos que pueden estar o no en la misma computadora o dispositivo. Los sockets empleados en este trabajo de tesis son del tipo orientado a conexión. En la figura 4.42 se muestra la interfaz de conexión entre la aplicación cliente y la aplicación servidor.

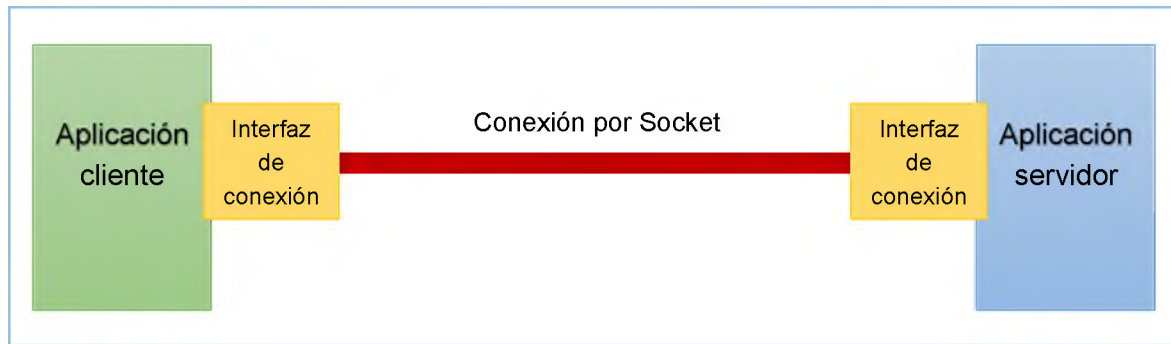


Figura 4.42: Interfaz de conexión cliente-servidor.

La conexión típica de red está constituida por dos sockets, un socket cliente y un socket servidor. Para lograr la conexión, ambas aplicaciones deben estar en el mismo rango de red, de esta manera, la aplicación cliente se debe conectar a la red *hotspot Wi-Fi* RBPI que es emitida por la Raspberry Pi.

Asimismo, estas aplicaciones utilizan el protocolo de transferencia de datos TCP/IP, el cual, proporciona un canal confiable de comunicación asíncrona que las dos aplicaciones utilizan para comunicarse entre sí, además, el protocolo TCP utiliza puertos para asignar datos entrantes a un proceso particular que se ejecuta en una computadora. Como se visualiza en la figura 4.43.

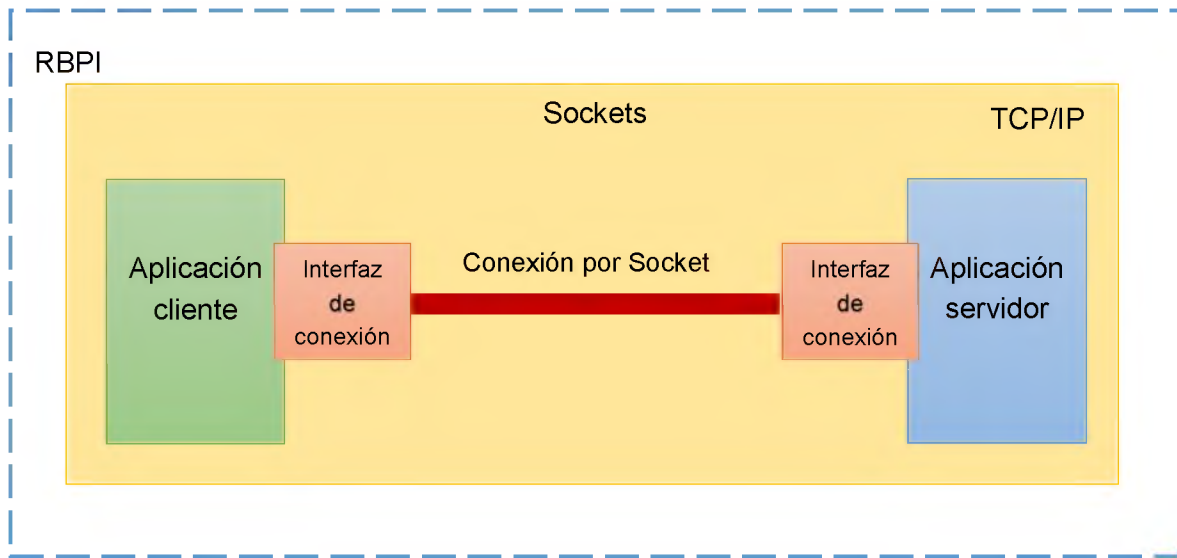


Figura 4.43: Esquema general de comunicación.

En la figura 4.44 se muestra el diagrama general del proceso de comunicación.

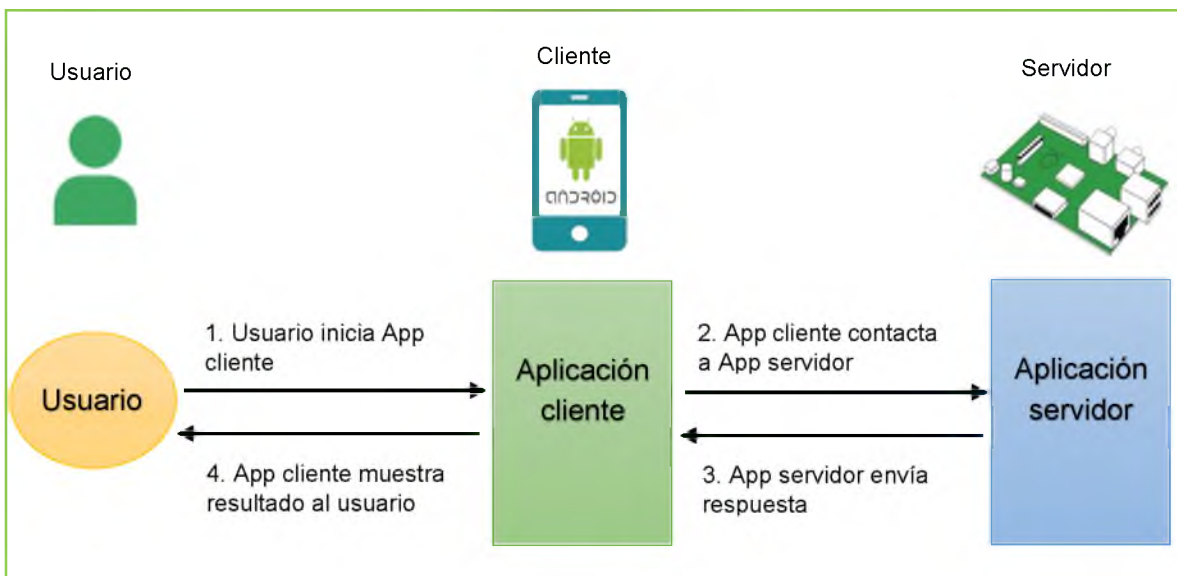


Figura 4.44: Diagrama general del proceso de comunicación.

En la figura 4.45 se muestra el diagrama de los métodos utilizados y su interacción.

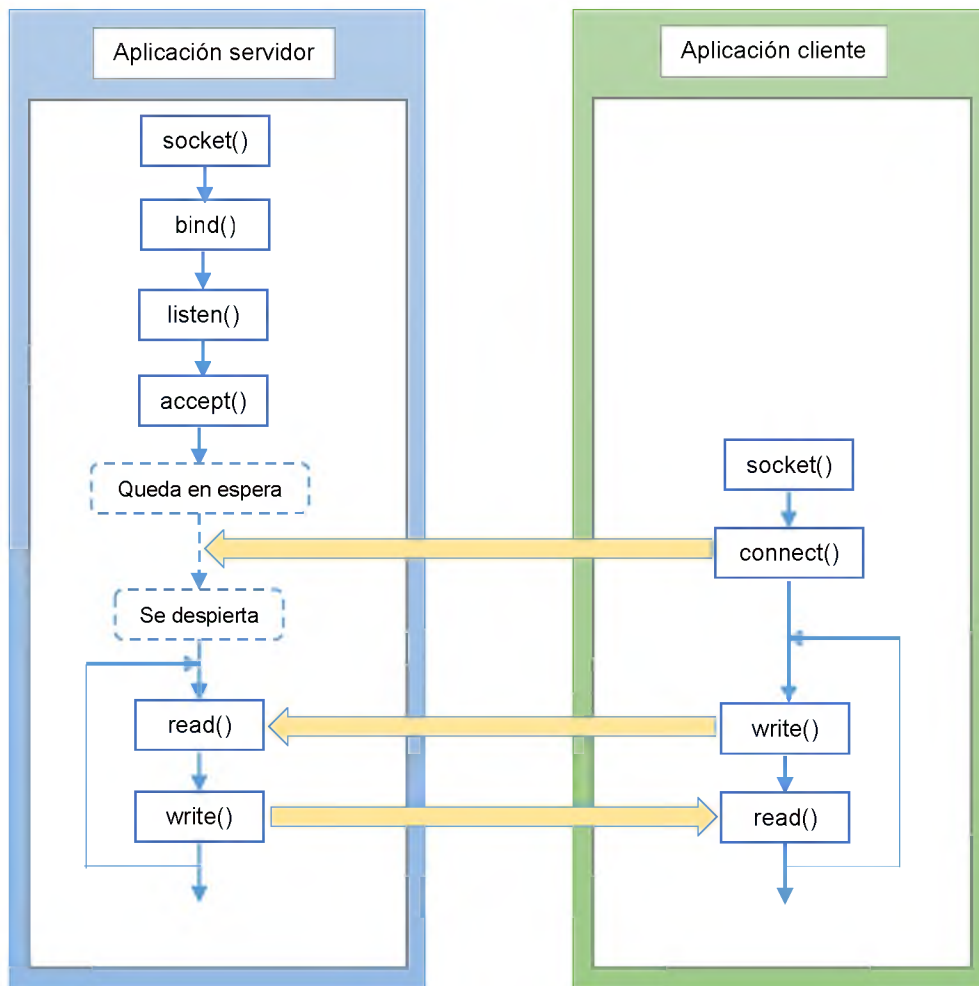


Figura 4.45: Diagrama de los métodos cliente-servidor.

Las dos aplicaciones utilizan sockets de manera diferente en ciertos pasos de la comunicación. En la figura 4.46 se describe el proceso que lleva a cabo la aplicación cliente para establecer la comunicación.

### 4.6.2.1 Aplicación cliente

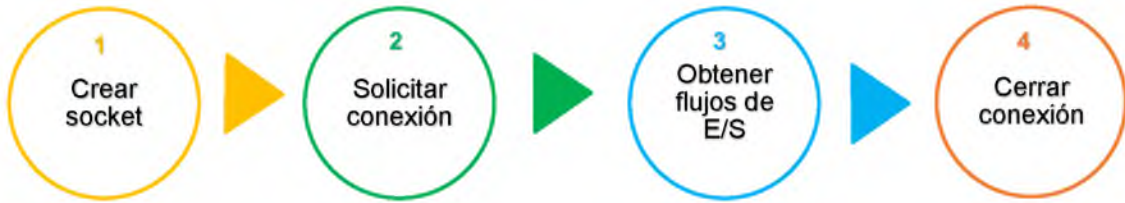


Figura 4.46: Proceso para establecer la comunicación (aplicación cliente).

#### 4.6.2.1.1 Crear socket

Para crear el socket, la aplicación cliente debe conocer la dirección IP de la Raspberry Pi en la que se ejecuta la aplicación servidor y el número de puerto en el que la aplicación servidor está escuchando, como se muestra en la figura 4.47 y en la línea de código siguiente, respectivamente.

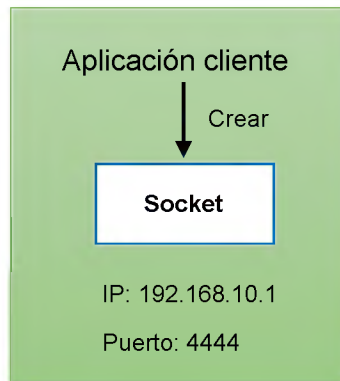


Figura 4.47: Creación del clientSocket.

```
Socket clientSocket = new Socket("192.168.10.1", 4444);
```

Donde:

- Primer argumento: La dirección IP 192.168.10.1 estática de la Raspberry Pi.
- Segundo argumento: El número de puerto TCP que se establece, en el cual la aplicación servidor comenzará a escuchar peticiones, por defecto es 4444.

El puerto está representado por un valor entero positivo (16 bits). Algunos puertos se han reservado para permitir servicios que son comunes o conocidos, por esta razón los procesos o servicios a nivel de usuario generalmente usan un valor de número de puerto en el rango de 1024 a 65535 ya que son los puertos dinámicos y éstos están disponibles para conectar servicios.

### 4.6.2.1.2 Solicitar conexión

Para solicitar una conexión con la aplicación servidor, la aplicación cliente utiliza la dirección IP y el número de puerto del socket que ha sido creado, como se aprecia en la figura 4.48.

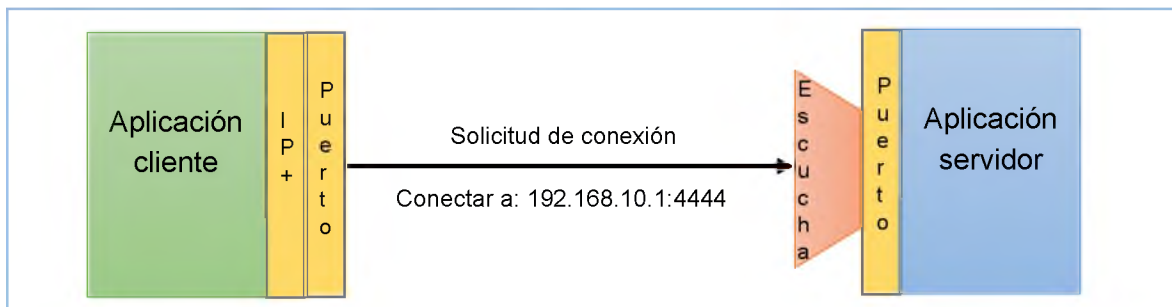


Figura 4.48: Solicitud de conexión a la aplicación servidor.

Cabe mencionar que al mismo tiempo, el método `accept()` de la aplicación servidor retorna con un parámetro que es un nuevo objeto `Socket`, el cual se utiliza para realizar el envío de datos con la aplicación cliente.

### 4.6.2.1.3 Obtener flujos de E/S

Si se acepta la conexión, se crea un socket con éxito en la aplicación servidor; y así, la aplicación cliente puede usar el socket para obtener flujos de entrada y salida para comunicarse con la aplicación servidor escribiendo o leyendo desde sus sockets, tal como se muestra en las siguientes líneas de código y en la figura 4.49, respectivamente.

```
DataInputStream is = new DataInputStream(client.getInputStream());  
DataOutputStream os = new DataOutputStream(client.getOutputStream());
```

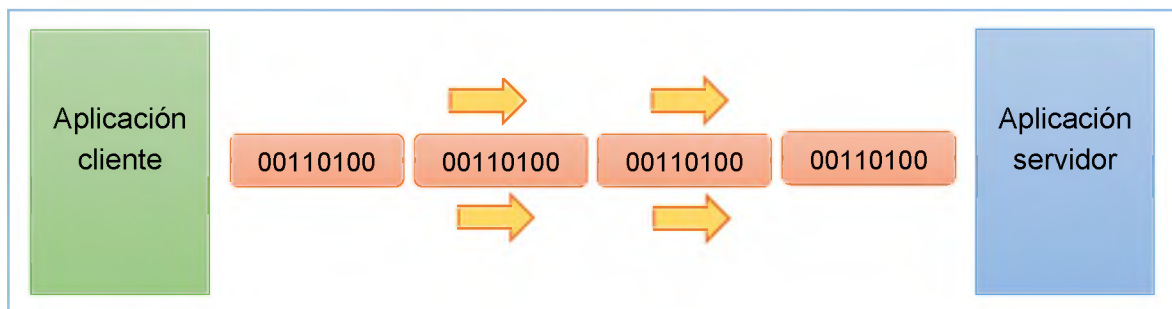


Figura 4.49: Envío del flujo de datos de la aplicación cliente.

El flujo de entrada de la aplicación cliente está conectado al flujo de salida de la aplicación servidor, al igual que el flujo de entrada de la aplicación servidor está conectado al flujo de salida de la aplicación cliente.

### 4.6.2.1.4 Cerrar conexión

La conexión del socket se cierra cuando el cliente se desconecta, de esta manera el puerto se libera y la aplicación servidor puede entregarlo a otro cliente, en la figura 4.50 se muestra un esquema del cierre de conexión del socket cliente.

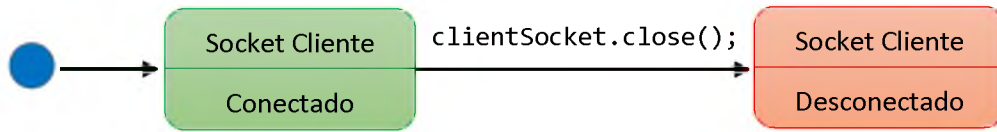


Figura 4.50: Cierre de conexión del socket cliente.

En la figura 4.51 se describe el proceso que lleva a cabo la aplicación servidor para establecer la comunicación.

#### 4.6.2.2 Aplicación servidor

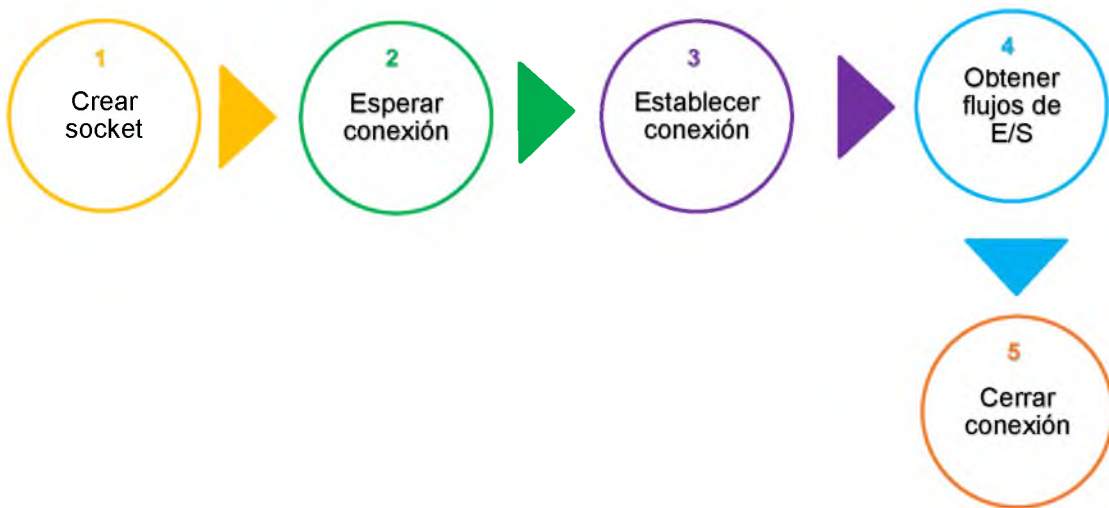


Figura 4.51: Proceso para establecer la comunicación (aplicación servidor).

### 4.6.2.2.1 Crear socket

La aplicación servidor se ejecuta continuamente en la Raspberry Pi y tiene un `ServerSocket`, cuando el `ServerSocket` recibe la solicitud crea un socket en un puerto que no esté ocupado; para este caso es el puerto 4444, como se visualiza en la figura 4.52 y en la línea de código siguiente, respectivamente.

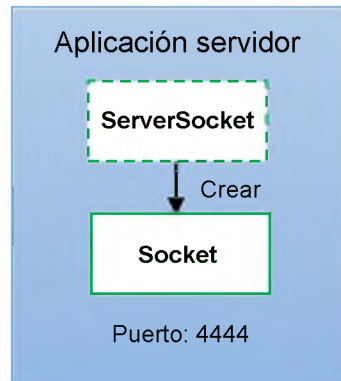


Figura 4.52: Creación del `serverSocket`.

```
ServerSocket serverSocket = new ServerSocket(4444);
```

### 4.6.2.2.2 Esperar conexión

La aplicación servidor solo espera, escuchando el socket de un cliente para realizar una solicitud de conexión, tal y como se aprecia en la siguiente línea de código y en la figura 4.53, respectivamente.

```
Socket clientSocket = serverSocket.accept();
```

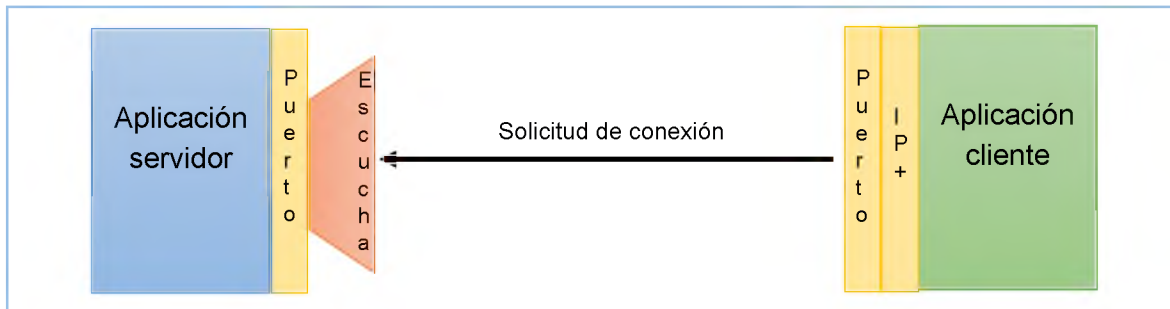


Figura 4.53: Esperar conexión.

#### 4.6.2.2.3 Establecer conexión

Si el contexto de comunicación entre el cliente y el servidor es propicio, entonces la aplicación servidor acepta la conexión como se muestra en la figura 4.54, y obtiene un objeto Socket (clientSocket) como resultado de la conexión con una vinculación entre ambos a través del puerto 4444, teniendo también su extremo remoto configurado en la dirección y el puerto del cliente.

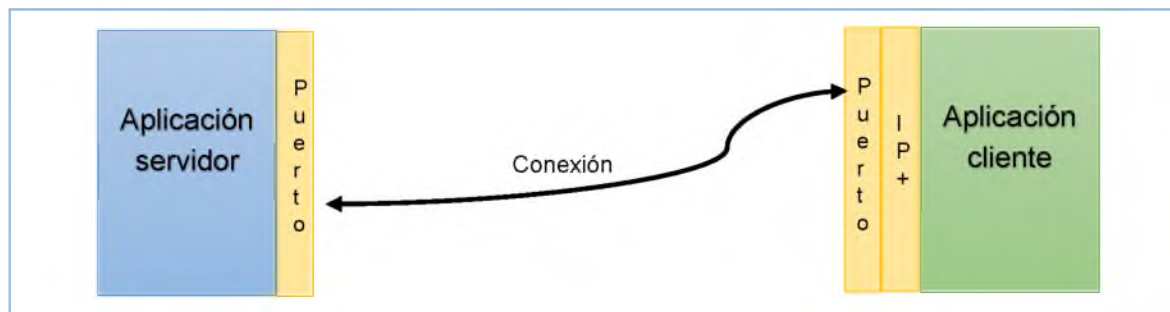


Figura 4.54: Conexión bidireccional establecida.

#### 4.6.2.2.4 Obtener flujos de E/S

El nuevo objeto Socket pone a la aplicación servidor en conexión directa con la aplicación cliente para obtener los flujos de entrada y salida asociados con el socket, logrando con ello

habilitar la escritura y lectura desde y hacia el cliente, tal como se muestra en las siguientes líneas de código y en la figura 4.55, respectivamente.

```
DataInputStream is = new DataInputStream(client.getInputStream());  
DataOutputStream os = new DataOutputStream(client.getOutputStream());
```

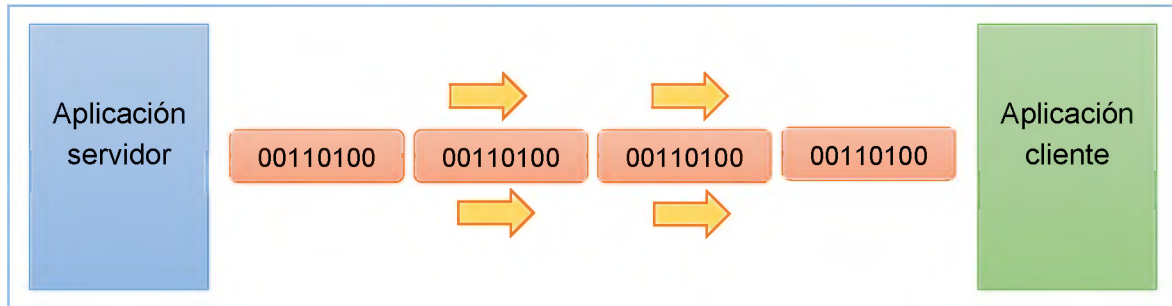


Figura 4.55: Envío del flujo de datos de la aplicación servidor.

Las secuencias de entrada se utilizan para leer datos del servidor, mientras que las secuencias de salida se utilizan para escribir datos en el servidor.

#### 4.6.2.2.5 Cerrar conexión

Finalmente, cuando se cierra la conexión existente con un cliente, se borra el documento temporal en la Raspberry Pi, permaneciendo el socket en espera de un nuevo cliente mediante una nueva llamada al servidor por medio del puerto de comunicación, en la figura 4.56 se muestra un esquema del cierre de conexión del socket servidor.



Figura 4.56: Cierre de conexión del socket servidor.

## 4.7 Proyectar presentación

En esta etapa de desarrollo se describe cómo se lleva a cabo la proyección de las presentaciones multimedia. El encargado de este proceso es el proyector digital, el cual se mantiene conectado a la tarjeta Raspberry Pi; o bien, se puede usar un cable convertidor de HDMI a VGA (si el proyector no cuenta con entrada HDMI). De esta manera las presentaciones multimedia que el usuario elige son proyectadas sobre una pantalla de proyección y gestionadas desde cualquier teléfono inteligente con sistema operativo Android.

Cabe mencionar que durante el proceso de proyección, sólo se puede mostrar un documento en el proyector. A continuación, en la figura 4.57 se muestra el proceso de proyección:

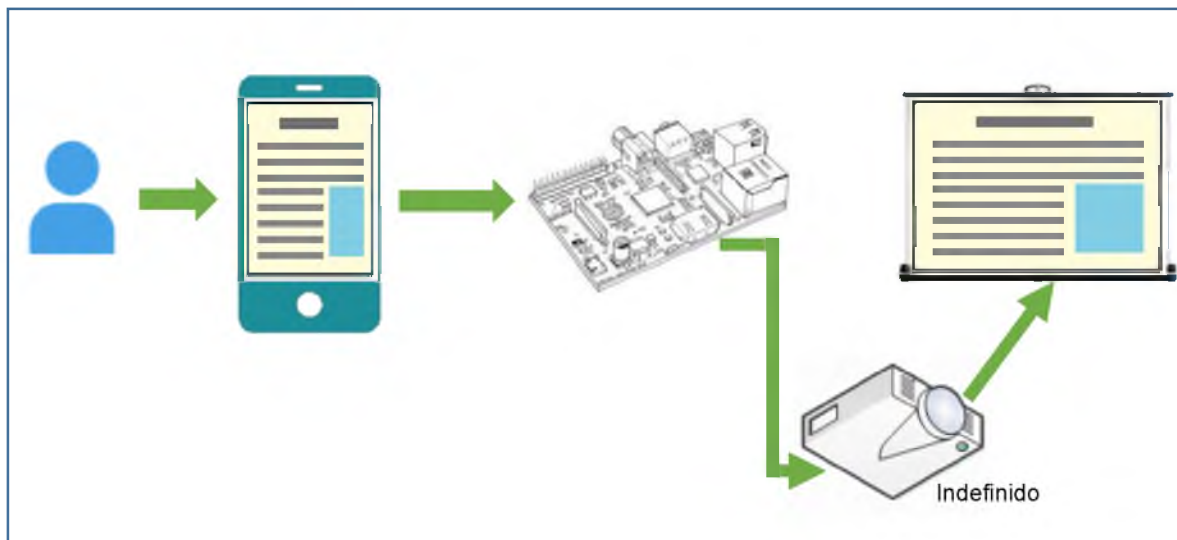


Figura 4.57: Proceso de proyección.

En este trabajo de tesis el sistema implementado es compatible y funciona correctamente con cualquier tipo de proyector (en cuanto a marca, precio, tamaño y color).

## Capítulo 5. Diseño de la aplicación cliente y aplicación servidor

---

En este capítulo, se muestra el diseño de las ventanas de la aplicación cliente y el diseño de la interfaz principal de la aplicación servidor correspondientes al sistema "Raspdroid-Projection", además se describe la funcionalidad de cada una de éstas.

## 5.1 Diseño de la aplicación servidor

En esta sección se muestra el diseño con respecto a la aplicación servidor.

### 5.1.1 Descripción de la interfaz de la aplicación servidor

La aplicación servidor está formada por una interfaz principal. A continuación, se describe su funcionalidad:

#### 5.1.1.1 Interfaz principal

Al ejecutar la aplicación servidor instalada en la tarjeta Raspberry Pi se visualiza la interfaz principal diseñada para esta aplicación. Ésta es la que permite de forma fácil al usuario la manipulación del servidor con respecto a su elección; la cual se observa en la figura 5.1.

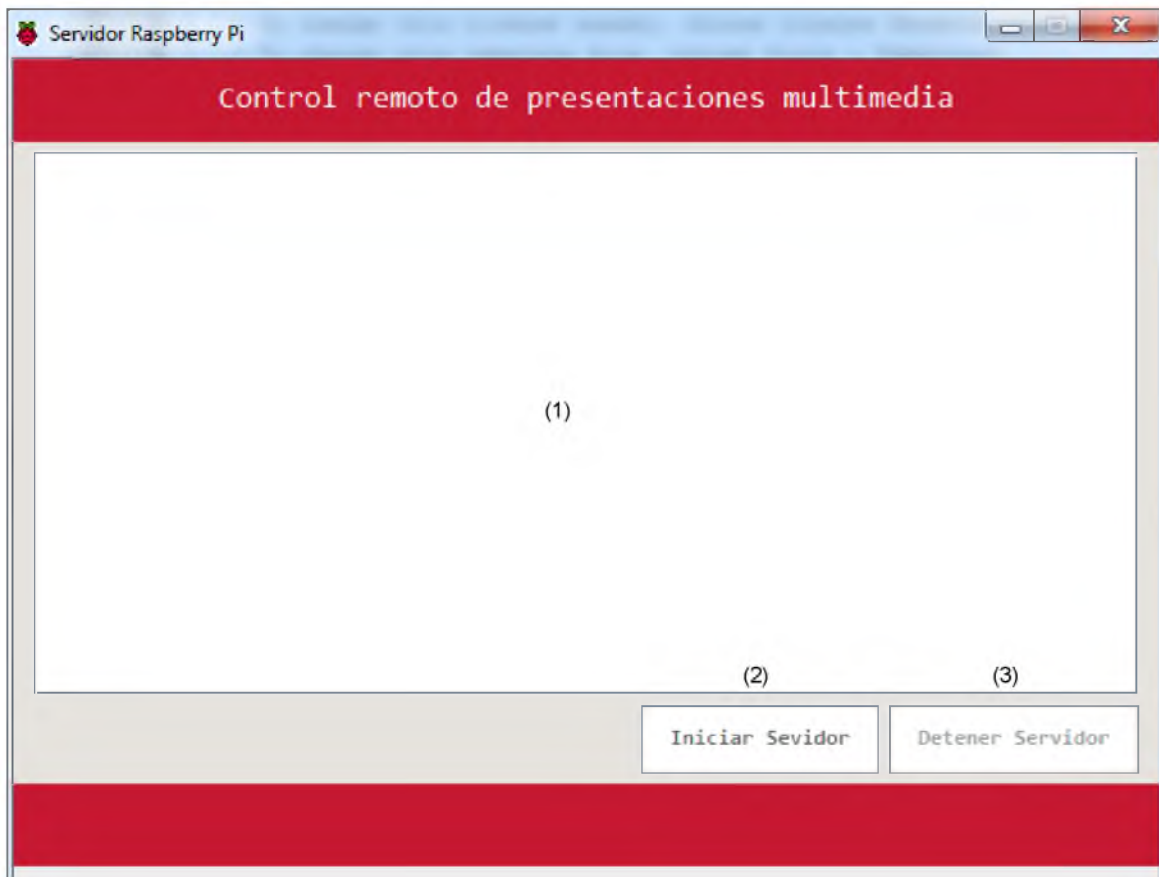


Figura 5.1: Interfaz principal de la aplicación servidor.

El diseño de la interfaz principal de la aplicación servidor incluye un área de texto y dos botones:

- El área de texto (1), muestra los mensajes de las acciones de la aplicación servidor.
- El botón (2) “Iniciar Servidor”, lleva a cabo la ejecución de la multitarea en la aplicación servidor.
- El botón (3) “Detener Servidor”, detiene la aplicación servidor.

### 5.1.1.2 Iniciar aplicación servidor

Cuando se inicia la aplicación servidor instalada en la tarjeta Raspberry Pi, el servidor espera una conexión entrante de un cliente, como se muestra en la figura 5.2.

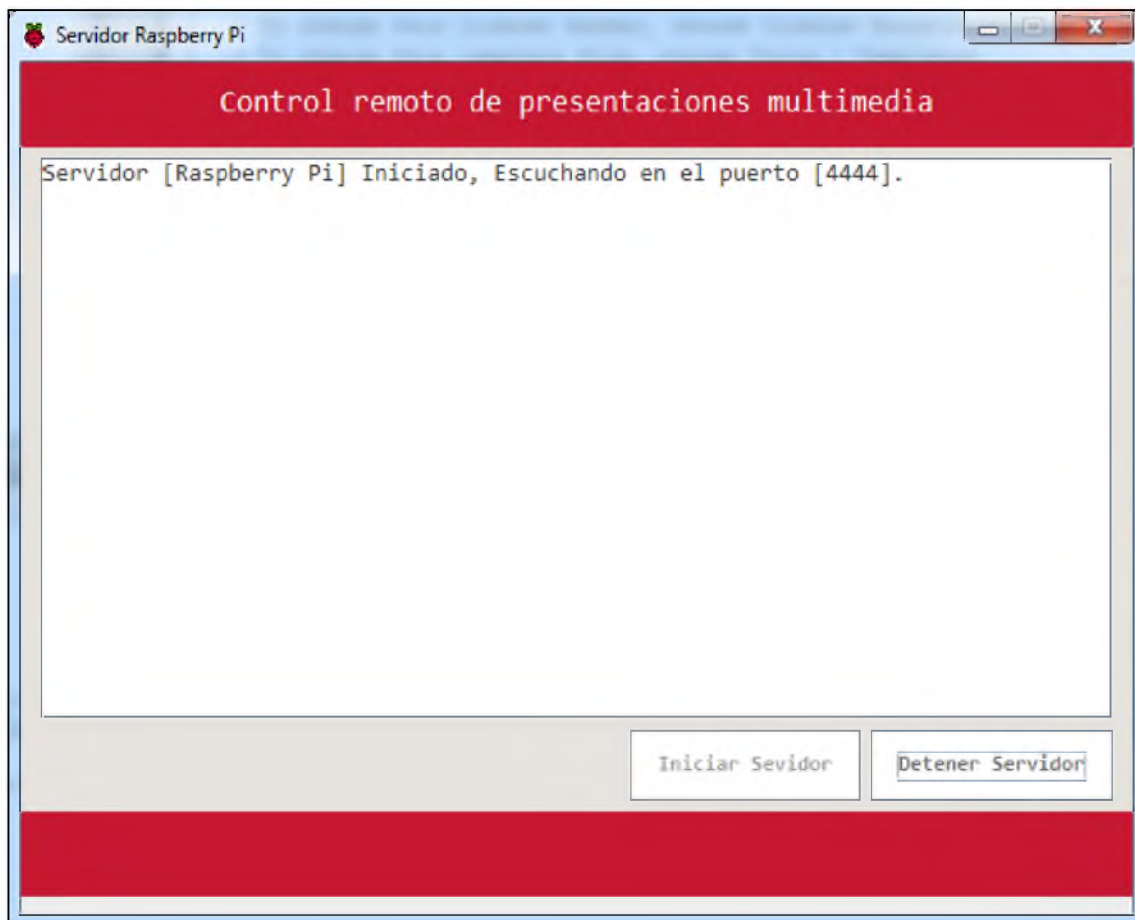


Figura 5.2: Aplicación servidor iniciada.

## 5.2 Diseño de la aplicación cliente

En esta sección se muestra el diseño con respecto a la aplicación cliente.

### 5.2.1 Descripción de las ventanas de la aplicación cliente

La aplicación cliente está formada por cinco ventanas:

1. Ventana principal
2. Ventana buscar red Wi-Fi
3. Ventana seleccionar presentación
4. Ventana mostrar información
5. Ventana control remoto

A continuación, se describen las funcionalidades de los elementos de cada ventana:

#### 5.2.1.1 Ventana principal

Al ejecutar la aplicación cliente, se visualiza la ventana principal diseñada para la propia aplicación, la cual ofrece al usuario una navegación fácil y sencilla. Su funcionalidad permite conectarse a la red *hotspot Wi-Fi* RBPI, la cual es generada por el servidor, también permite seleccionar y enviar presentaciones multimedia, las cuales son controladas remotamente. Su diseño se muestra a continuación en la figura 5.3.

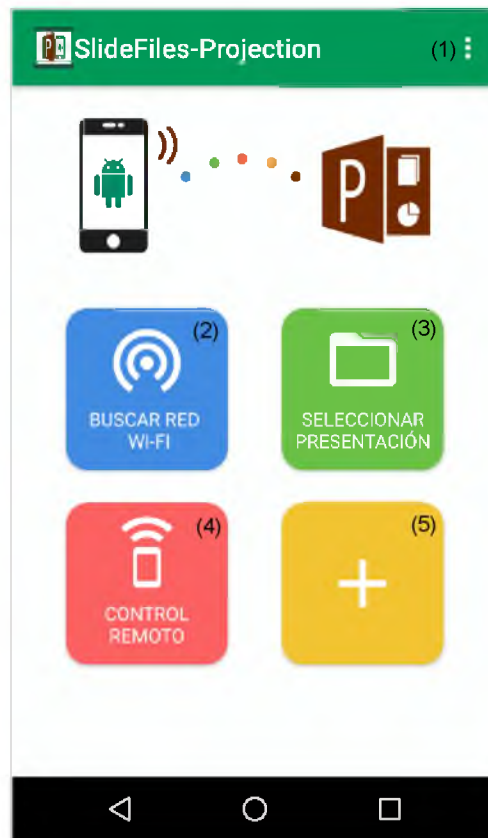


Figura 5.3: Ventana principal de la App cliente.

El diseño de la ventana *principal* de la aplicación cliente incluye un menú y cuatro botones:

- El menú (1) “Acerca de”, muestra la información sobre la aplicación.
- El botón (2) “BUSCAR RED WI-FI”, realiza el escaneo de las redes Wi-Fi disponibles, permitiendo la conexión a la red Wi-Fi deseada.
- El botón (3) “SELECCIONAR PRESENTACIÓN”, muestra el explorador con todos los archivos propios del teléfono, permitiendo al usuario la navegación entre archivos para seleccionar la presentación multimedia.
- El botón (4) “CONTROL REMOTO”, lleva a cabo el control remoto de las presentaciones multimedia.
- El botón (5) “+”, llevará a cabo nuevas funcionalidades.

Una vez definida la función de la ventana *principal* de la aplicación cliente, se procede a diseñar el diagrama de clase y la generación del código XML correspondientes a la actividad principal de esta ventana, como se muestra en la figura 5.4 y figura 5.5, respectivamente:

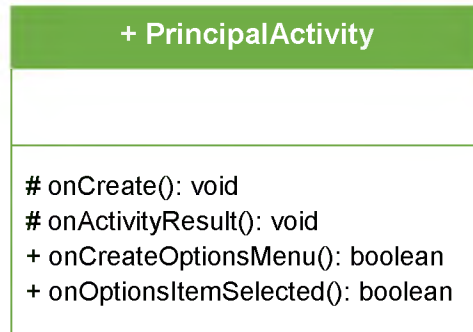


Figura 5.4: Diagrama de clase de `PrincipalActivity`.

```
<?xml version="1.0" encoding="utf-8" ?>
<android.support.constraint.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:gravity="center"
android:background="#fdfefe"
tools:context=".PrincipalActivity">

<Button...> <!--Botón buscar red Wi-Fi 138dp-->

<Button...> <!--Botón seleccionar presentación-->

<Button...> <!--Botón control remoto-->

<Button...> <!--Botón +-->

<TextView...>

</android.support.constraint.ConstraintLayout>
```

Figura 5.5: Código XML de `activity_principal`.

### 5.2.1.2 Ventana buscar red Wi-Fi

Esta ventana permite la conexión a la red *hotspot Wi-Fi* RBPI. En este proceso la aplicación realiza el escaneo de las redes Wi-Fi disponibles, mostrando una lista de los SSID de todas las redes Wi-Fi disponibles. En la figura 5.6 se muestra la lista generada por la aplicación y en ella se observa la red Wi-Fi identificada con el SSID RBPI.



Figura 5.6: Redes Wi-Fi disponibles.

Al seleccionar la red RBPI, se muestra un cuadro de diálogo de conexión (AlertDialog) que permite escribir la clave de seguridad de la red (password), al seleccionar <OK> el teléfono

inteligente se conecta a la red Wi-Fi y de esta manera puede seguir con la correcta navegación de la App.

Si el proceso de conexión a la red RBPI se realiza de forma exitosa, se mostrará la ventana *principal* de la aplicación con el botón “SELECCIONAR PRESENTACIÓN” habilitado.

Una vez definida la función de la ventana *buscar red Wi-Fi*, se procede a diseñar el diagrama de clase y la generación del código XML correspondientes a la actividad principal de esta ventana, como se muestra en la figura 5.7 y figura 5.8, respectivamente.



Figura 5.7: Diagrama de clase de WifiScanAdapter.

```
<?xml version="1.0" encoding="utf-8" ?>
<android.support.v7.widget.CardView
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/card_view"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    card_view:cardCornerRadius="1dp"
    android:padding="10dp"
    android:layout_margin="1dp">

    <LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal">

        <ImageView...> <!--ImageView Wi-Fi logo-->

        <TextView...> <!--TextView SSID_nombre-->

    </LinearLayout>
</android.support.v7.widget.CardView>
```

Figura 5.8: Código XML de network\_list.

### 5.2.1.3 Ventana seleccionar presentación

Esta ventana permite al usuario elegir y seleccionar una presentación multimedia del sistema de archivos del propio teléfono inteligente. En la ventana se muestra la estructura del explorador de archivos (un directorio con todos los archivos del teléfono) en el cual se puede acceder de manera más organizada, permitiendo así, la navegación entre archivos para seleccionar una presentación multimedia; como se observa en la figura 5.9.

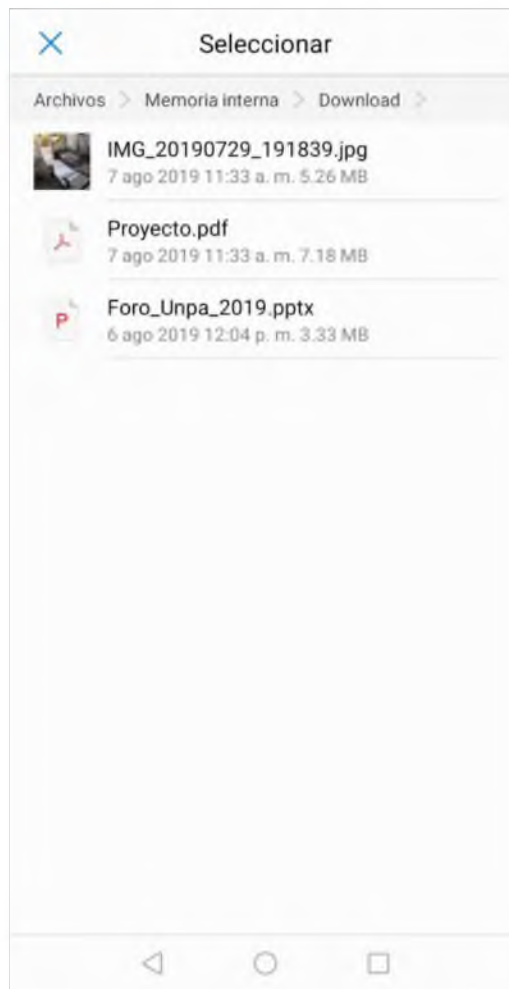


Figura 5.9: Seleccionar presentación.

Una vez definida la función de la ventana *seleccionar presentación*, se procede a diseñar el diagrama de clase y el código del método `showChooser`, el cual genera en tiempo de ejecución la ventana del explorador de archivos, estos, correspondientes a la actividad principal de esta ventana, como se muestra en la figura 5.10 y figura 5.11, respectivamente.

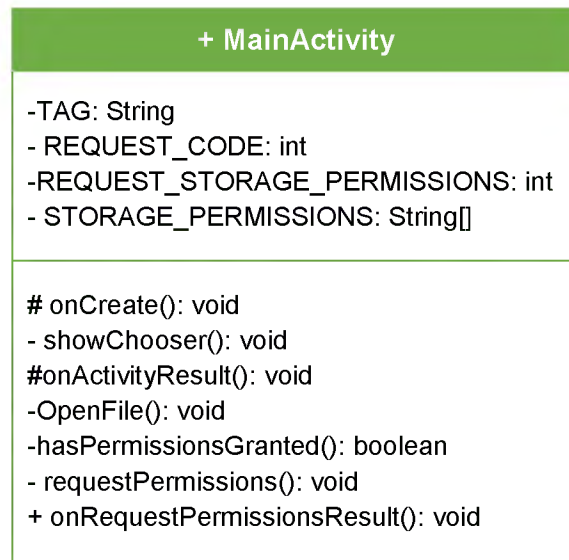


Figura 5.10: Diagrama de clase de MainActivity.

```
private void showChooser() {
    // Use the GET_CONTENT intent from the utility class
    Intent target = FileUtils.createGetContentIntent();
    // Create the chooser Intent
    Intent intent = Intent.createChooser(target, "SlideFiles");
    try {
        startActivityForResult(intent, REQUEST_CODE);
    } catch (ActivityNotFoundException e) {
        // The reason for the existence of aFileChooser
    }
}
```

Figura 5.11: Código del método showChooser de la clase MainActivity.

#### 5.2.1.4 Ventana mostrar información

Esta ventana muestra la información básica de la presentación multimedia seleccionada. La información mostrada es: nombre del archivo, tamaño del archivo (en bytes) y la extensión del archivo, como se muestra en la figura 5.12.



Figura 5.12: Información de la presentación multimedia.

El diseño de esta ventana consta de tres campos de texto y dos botones:

- El campo de texto (1), muestra el nombre del archivo.
- El campo de texto (2), muestra el tamaño del archivo en bytes.
- El campo de texto (3), muestra la extensión del archivo.
- El botón (4) “BUSCAR”, regresa a la ventana *seleccionar presentación*, para buscar y seleccionar otra presentación multimedia.
- El botón (5) “ENVIAR”, envía la información básica del archivo mediante un formato JSON, además, envía el archivo multimedia al servidor, permitiendo con esto, habilitar el botón “CONTROL REMOTO” para controlar la presentación multimedia.

Una vez definida la función de la ventana *mostrar información*, se procede a diseñar el diagrama de clase y la generación del código XML correspondientes a la actividad principal de esta ventana, como se muestra en la figura 5.13 y figura 5.14, respectivamente.

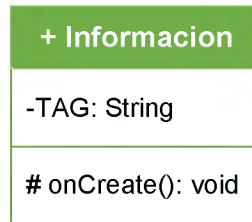


Figura 5.13: Diagrama de clase de Información.

```
<?xml version="1.0" encoding="utf-8" ?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:background="#fdfe"
    tools:context=".Informacion">

    <TextView...> <!--TextView extensión-->

    <TextView...> <!--TextView tamaño-->

    <TextView...> <!--TextView nombre-->

    <Button...> <!--Botón enviar presentación-->

    <Button...> <!--Botón buscar otra presentación-->

    <TextView...> <!--TextView imagen-->

    <TextView...> <!--TextView mostrar nombre-->

    <TextView...> <!--TextView mostrar tamaño-->

    <TextView...> <!--TextView mostrar extensión-->

</android.support.constraint.ConstraintLayout>
```

Figura 5.14: Código XML de activity\_informacion.

### 5.2.1.5 Ventana control remoto

Al seleccionar el botón "CONTROL REMOTO" se muestra la ventana *control remoto*. Esta ventana la cual se muestra en la figura 5.15, figura 5.16 y figura 5.17, controla remotamente la presentación multimedia por medio de comandos de control que la API de LibreOffice ofrece.

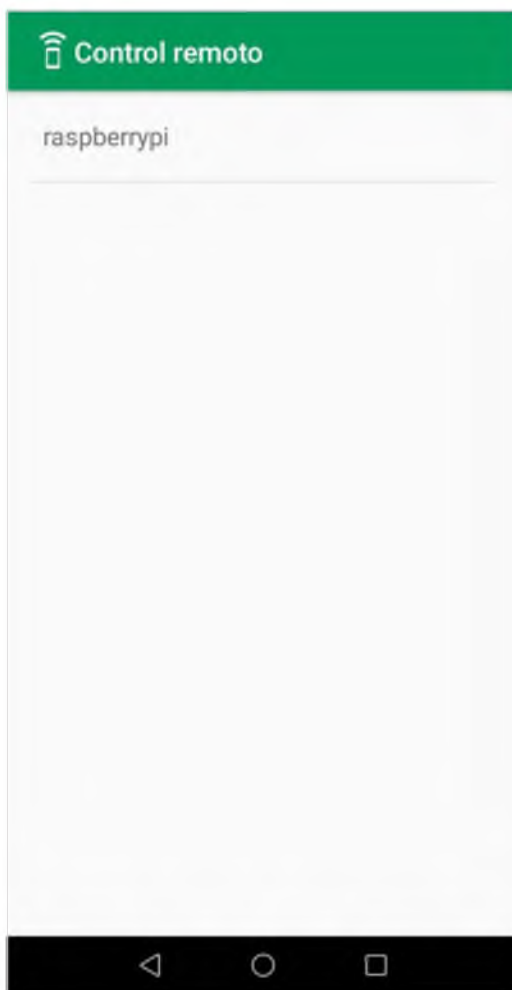


Figura 5.15: Control remoto (1).



Figura 5.16: Control remoto (2).



Figura 5.17: Control remoto (3).

Una vez definida la función de la ventana *control remoto*, se procede a diseñar el diagrama de clase y la generación del código XML correspondientes a la actividad principal de esta ventana como se muestra en la figura 5.18 y figura 5.19, respectivamente.

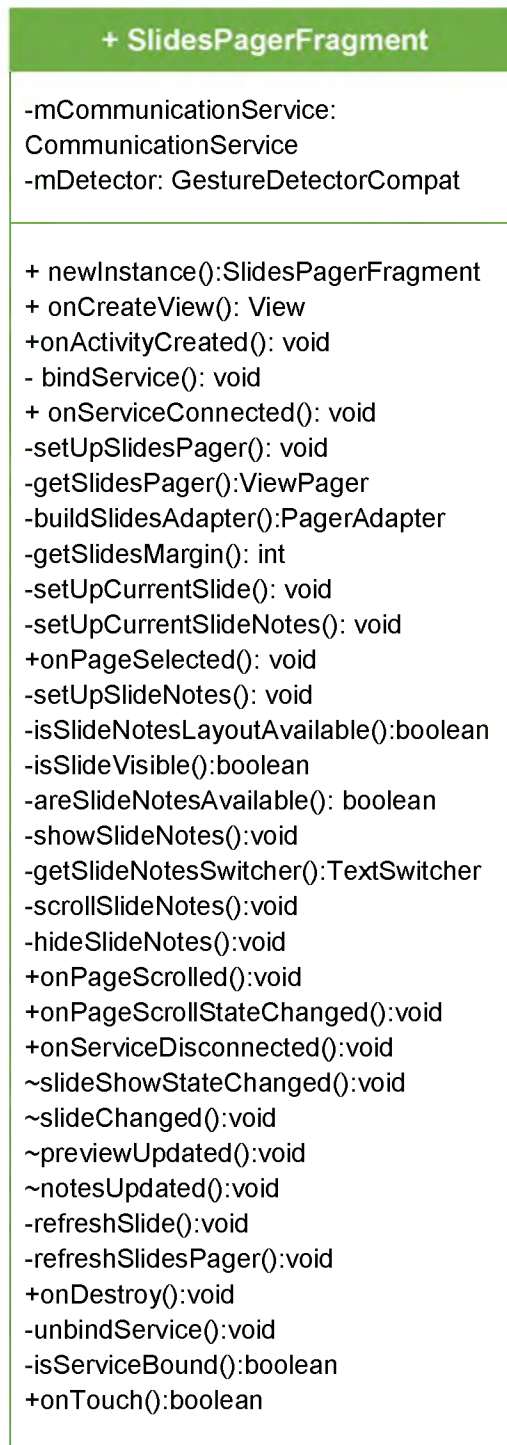


Figura 5.18: Diagrama de clase de SlidesPagerFragment.

```
<LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:orientation="vertical"
  android:gravity="center"
  android:paddingTop="@dimen/padding_slides_pager"
  android:paddingBottom="@dimen/padding_slides_pager"
  android:layout_width="match_parent"
  android:layout_height="match_parent">

  <android.support.v4.view.ViewPager...> <!--ver diapositiva-->

  <LinearLayout
    android:id="@+id/layout_notes"
    android:orientation="vertical"
    android:paddingLeft="@dimen/padding_slides_pager"
    android:paddingRight="@dimen/padding_slides_pager"
    android:layout_weight="3"
    android:layout_width="match_parent"
    android:layout_height="0dp">

    <TextView...> <!--TextView Notas-->

    <ScrollView...> <!--ScrollView Notas-->

  </LinearLayout>
</LinearLayout>
```

Figura 5.19: Código XML de fragment\_slides\_pager.

## Capítulo 6. Pruebas y resultados

---

En este capítulo, se describe de manera general las pruebas realizadas a la aplicación cliente y a la aplicación servidor del sistema "Raspdroid-Projection", además, se presentan los resultados obtenidos de cada una de las pruebas de cada aplicación.

## 6.1 Pruebas del sistema “Raspdroid-Projection”

### 6.1.1 Introducción

En el presente documento se especifican los resultados obtenidos del sistema de presentación multimedia “Raspdroid-Projection” en un ambiente controlado, que permiten comprobar el funcionamiento de los diferentes elementos que forman parte de la aplicación cliente y la aplicación servidor.

En el documento se lleva a cabo la especificación del diseño de pruebas [ver Anexo A y Anexo B] y la especificación de los casos de prueba que se realizaron a la aplicación cliente y a la aplicación servidor.

Este documento se apega a los lineamientos establecidos en el estándar IEEE 829-2008 [31] para la documentación de pruebas de software, el cual detalla las actividades necesarias para realizar las pruebas al sistema “Raspdroid-Projection” compuesto por la aplicación cliente y la aplicación servidor.

### 6.1.2 Objetivos

#### 6.1.2.1 Objetivo general

Probar y verificar que las funcionalidades de cada módulo de la aplicación cliente y de la aplicación servidor se realicen de forma exitosa en cada una de las pruebas realizadas al sistema de presentación multimedia “Raspdroid-Projection”.

#### 6.1.2.2 Objetivos específicos

- Probar la funcionalidad de la aplicación cliente a través de la navegación de cada una de sus ventanas, al enviar y gestionar presentaciones multimedia bajo un ambiente controlado.
- Probar la funcionalidad de la aplicación servidor a través de su interfaz principal, al recibir y proyectar archivos de presentación multimedia bajo un ambiente controlado.

### 6.1.3 Modelo del sistema

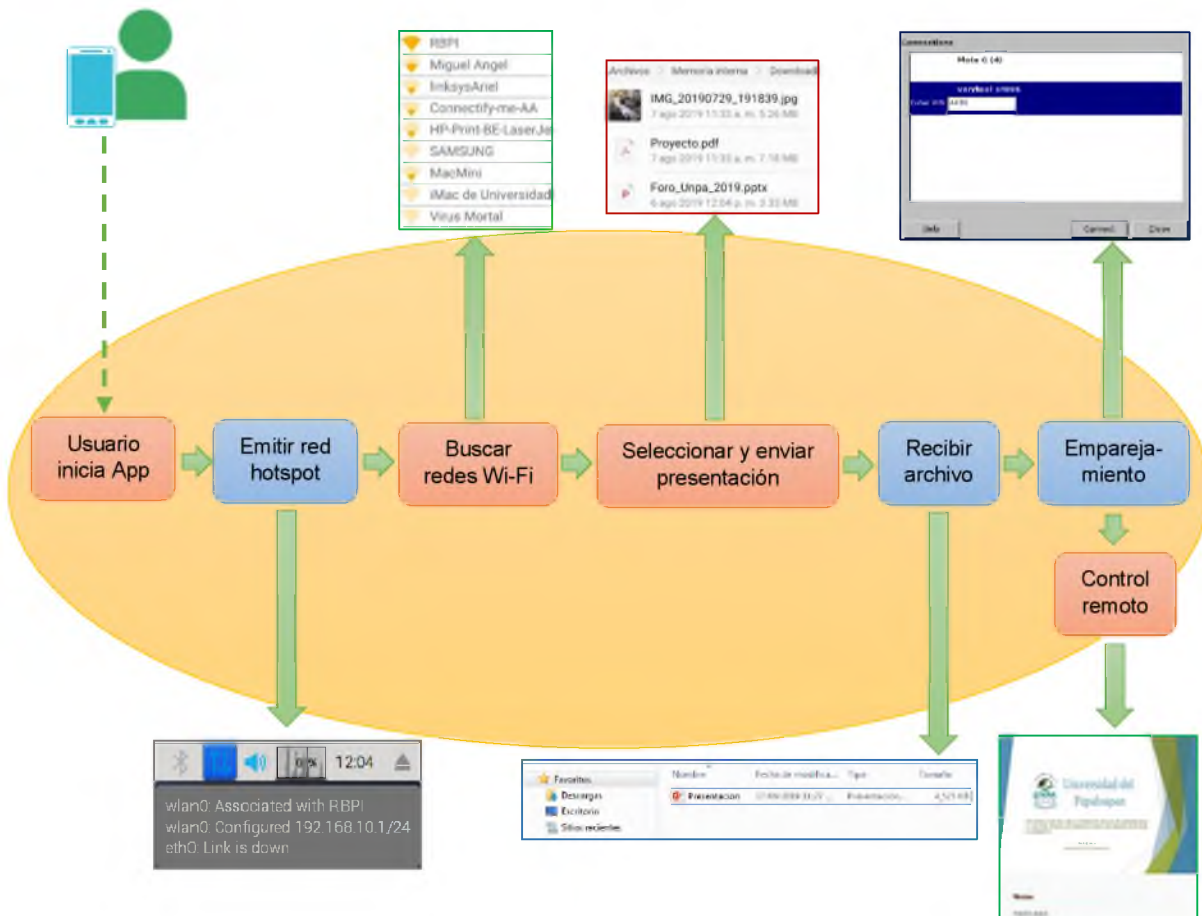


Figura 6.1: Modelo del sistema "Raspdroid-Projection".

## 6.2 Plan de pruebas de la aplicación cliente (APC-01-01)

En este plan de pruebas se verifica que la funcionalidad de la aplicación cliente sea exitosa al conectarse a un punto de acceso Wi-Fi, además de seleccionar y enviar una presentación multimedia a la aplicación servidor para posteriormente realizar la gestión del archivo.

El nombre de la aplicación cliente se abrevia Aplicación Cliente APC para referirlo de una forma más simple en este documento.

Las características de los requerimientos para realizar las pruebas a la aplicación cliente se muestran en la tabla 6.1.

Tabla 6.1: Requerimientos de prueba de la App cliente.

Teléfonos inteligentes	
Modelo	Sistema Operativo
Verykool Leo 3G s4006	Android 4.4.2
Motorola Moto G4	Android 7.0
Huawei Y6 2018 (ATU-LX3)	Android 8.0
Otros requerimientos	
Proyector digital	Indefinido

El plan de pruebas de la aplicación (APC-01-01) está formado por una especificación del plan de pruebas y una especificación de los casos de prueba. Tales especificaciones se describen a continuación:

## 6.2.1 Especificación del Plan de Pruebas de la aplicación cliente

### 6.2.1.1 Nomenclatura

Se presenta a continuación la nomenclatura que será utilizada a lo largo del documento referente al plan de pruebas para la aplicación cliente, la cual forma parte del sistema “Raspdroid-Projection” presentado en este documento de tesis. El formato establecido para referirse a cada apartado del plan de pruebas de la aplicación cliente es **APC-XX-YY**. Donde:

- **APC** es el prefijo utilizado para referirse a Aplicación Cliente, como referencia a la aplicación cliente.
- **XX** es el prefijo utilizado para representar el tipo de concepto a ser evaluado.
- **YY** es el prefijo utilizado para representar un valor consecutivo del concepto.

A continuación se definen cada uno de los conceptos y valores consecutivos para cada uno de ellos utilizados en el plan de prueba:

Tabla 6.2: Concepto del plan de pruebas de la aplicación cliente.

Valor	Tipo
01	Documentación de Pruebas
02	Módulos de la Aplicación Cliente
03	Funciones de la Aplicación Cliente

Tabla 6.3: Documentación de pruebas de la aplicación cliente.

Formato	Descripción
APC-01-01	Plan de Pruebas de la aplicación
APC-01-02	Especificación del Diseño de Pruebas
APC-01-03	Especificación de los Casos de Prueba

Tabla 6.4: Módulos de la aplicación cliente.

Formato	Descripción
APC-02-01	► Módulo “SlideFiles” de la ventana principal de la aplicación cliente.
APC-02-02	► Módulo “WifiConnect” para el escaneo de puntos de acceso Wi-Fi disponibles.
APC-02-03	► Módulo “FileChooser” para la selección y envío de presentaciones multimedia.
APC-02-04	► Módulo “ImpressRemote” para el control remoto de presentaciones multimedia.

Tabla 6.5: Funciones de la aplicación cliente.

Formato	Descripción
APC-03-01	► Mostrar la ventana principal de la aplicación cliente con los botones funcionales.
APC-03-02	► Mostrar la lista de puntos de acceso Wi-Fi disponibles.
APC-03-03	► Mostrar el explorador de archivos del propio teléfono inteligente para seleccionar y enviar una presentación multimedia.
APC-03-04	► Controlar remotamente una presentación multimedia desde un teléfono inteligente.

Las pruebas en la aplicación cliente no contemplan probar las siguientes características:

- Pruebas de la aplicación con Tabletas o algún otro dispositivo diferente a un teléfono inteligente.

- Pruebas de la aplicación con teléfonos inteligentes con versiones anteriores a la versión 4.4.2 con nivel de API 19.
- Pruebas de la aplicación donde más de dos usuarios envíen una presentación multimedia.
- No se probará la barra de opciones del explorador de archivos del módulo APC-02-03.

## 6.2.2 Especificación de los Casos de Prueba de la aplicación cliente (APC-01-01)

### 6.2.2.1 Especificación APC-01-03.1

“Caso de prueba para comprobar el estado del botón **“BUSCAR RED WI-FI”**”.

#### 6.2.2.1.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: **APC-01-02.1**

#### 6.2.2.1.2 Especificaciones de entrada

Al ejecutar la aplicación cliente las entradas esperadas se presentan a continuación, para este caso de prueba:

- La aplicación cliente se ejecuta de forma correcta.
- La aplicación cliente no se ejecuta de forma correcta.

#### 6.2.2.1.3 Especificaciones de salida

Si la aplicación cliente se ejecuta de forma correcta, se muestra la ventana principal de la aplicación como se visualiza en la figura 6.2.

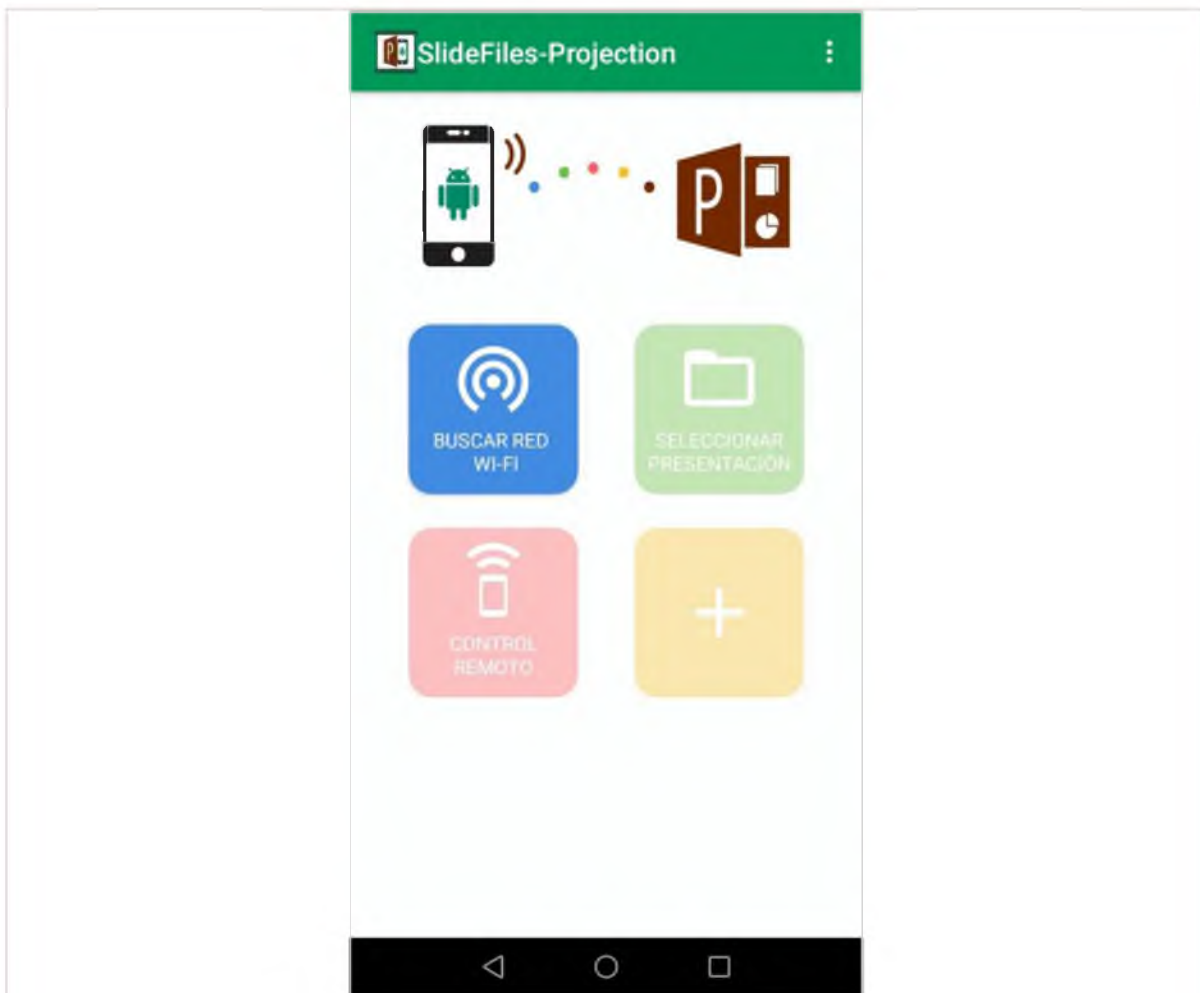


Figura 6.2: Botón “BUSCAR RED WI-FI” habilitado.

Para el caso de la entrada donde la aplicación cliente no se ejecuta de forma correcta, no se muestra la ventana principal de la aplicación.

#### 6.2.2.1.4 Análisis de Resultados

Para el caso de la entrada donde la aplicación cliente se ejecuta correctamente, se obtiene como resultado la ventana principal de la aplicación con el botón “BUSCAR RED WI-FI” habilitado. En el caso de la entrada donde la aplicación cliente no se ejecuta correctamente, no se muestra tal ventana.

#### 6.2.2.2 Especificación APC-01-03.2

“Caso de prueba para comprobar el estado del botón “SELECCIONAR PRESENTACIÓN””.

#### 6.2.2.2.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.2*

#### 6.2.2.2 Especificaciones de entrada

Al seleccionar el botón “SELECCIONAR PRESENTACIÓN” las entradas esperadas se presentan a continuación, para este caso de prueba:

- El teléfono inteligente se encuentra conectado a un punto de acceso Wi-Fi.
- El teléfono inteligente no se encuentra conectado a ningún punto de acceso Wi-Fi.

#### 6.2.2.3 Especificaciones de salida

Si el teléfono inteligente se encuentra conectado a un punto de acceso Wi-Fi, se muestra la ventana principal de la aplicación como se visualiza en la figura 6.3.



Figura 6.3: Botón “SELECCIONAR PRESENTACIÓN” habilitado.

Para el caso de la entrada donde el teléfono inteligente no se encuentra conectado a ningún punto de acceso Wi-Fi, no se habilita el botón “SELECCIONAR PRESENTACIÓN”.

#### 6.2.2.2.4 Análisis de Resultados

Para el caso de la entrada donde el teléfono inteligente se encuentra conectado a un punto de acceso Wi-Fi, se obtiene como resultado la ventana principal de la aplicación con el botón “SELECCIONAR PRESENTACIÓN” habilitado. En el caso de la entrada donde el teléfono inteligente no se encuentra conectado a un punto de acceso Wi-Fi, no se habilita el botón “SELECCIONAR PRESENTACIÓN” hasta que el usuario se conecte a un punto de acceso Wi-Fi.

#### 6.2.2.3 Especificación APC-01-03.3

“Caso de prueba para comprobar el estado del botón “CONTROL REMOTO””.

##### 6.2.2.3.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.3*

##### 6.2.2.3.2 Especificaciones de entrada

Al seleccionar el botón “CONTROL REMOTO” las entradas esperadas se presentan a continuación, para este caso de prueba:

- El archivo pptx fue enviado a la aplicación servidor.
- El archivo pptx no fue enviado a la aplicación servidor.

##### 6.2.2.3.3 Especificaciones de salida

Si el archivo pptx fue enviado a la aplicación servidor de forma correcta, se muestra la ventana principal de la aplicación como se visualiza en la figura 6.4.

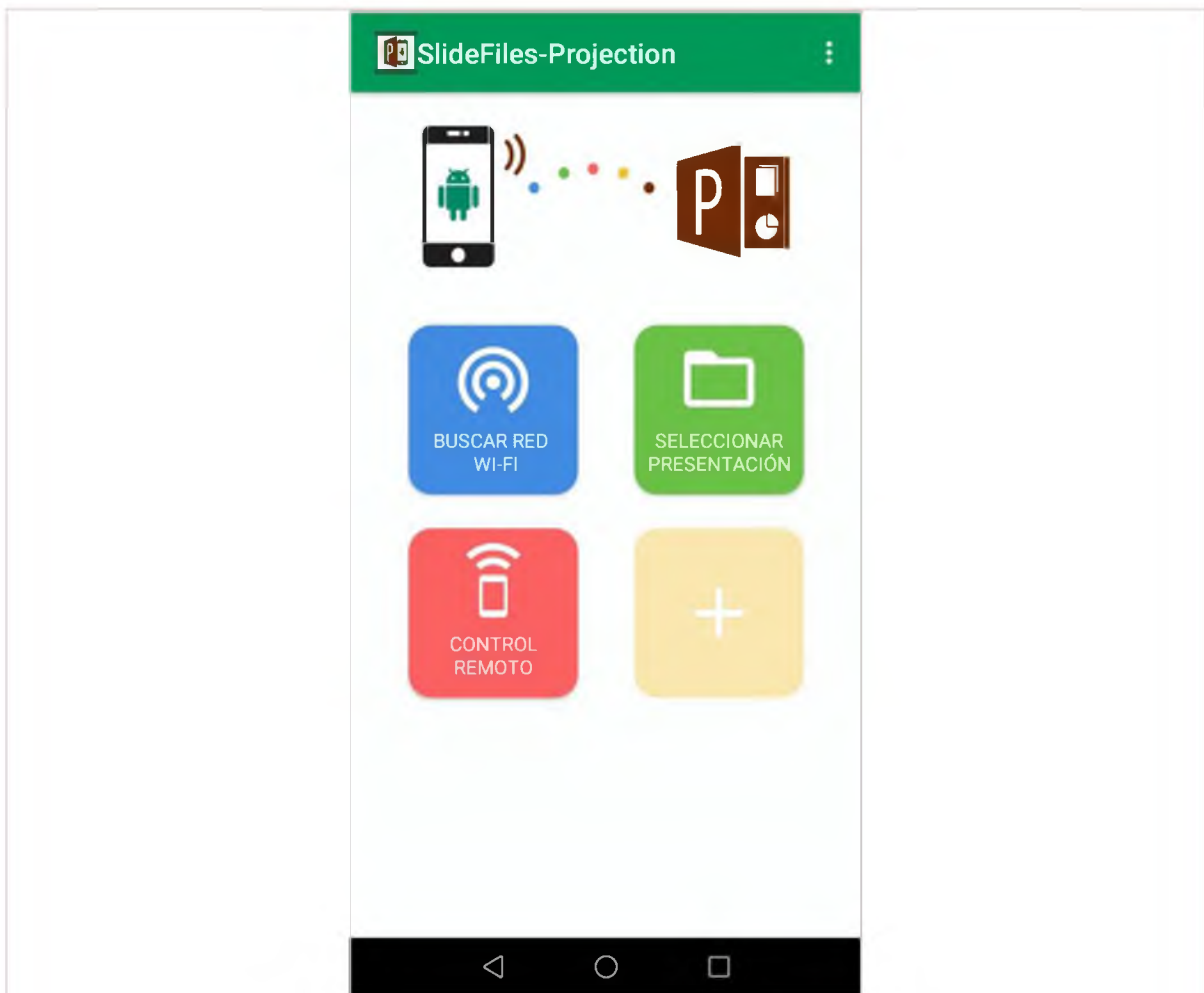


Figura 6.4: Botón “CONTROL REMOTO” habilitado.

Para el caso de la entrada donde el archivo pptx no fue enviado a la aplicación servidor, no se habilita el botón “CONTROL REMOTO”.

#### 6.2.2.3.4 Análisis de Resultados

Para el caso de la entrada donde el archivo pptx fue enviado a la aplicación servidor de forma exitosa, se obtiene como resultado la ventana principal de la aplicación con el botón “CONTROL REMOTO” habilitado. En el caso de la entrada donde el archivo pptx no fue enviado a la aplicación servidor, no se habilita el botón “CONTROL REMOTO” hasta que el usuario elija y envíe una presentación multimedia.

#### **6.2.2.4 Especificación APC-01-03.4**

“Caso de prueba para comprobar el estado del dispositivo Wi-Fi del teléfono inteligente”.

##### **6.2.2.4.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.4*

##### **6.2.2.4.2 Especificaciones de entrada**

Al seleccionar el botón “**BUSCAR RED WI-FI**” de la aplicación cliente las entradas esperadas se presentan a continuación, para este caso de prueba:

- El dispositivo Wi-Fi del teléfono inteligente se encuentra activado.
- El dispositivo Wi-Fi del teléfono inteligente se encuentra desactivado.
- La ubicación en el teléfono inteligente se encuentra activada (manualmente)..

##### **6.2.2.4.3 Especificaciones de salida**

La aplicación cliente muestra en un mensaje el SSID del punto de acceso Wi-Fi al cual se encuentra conectado, tal como se visualiza en la figura 6.5.

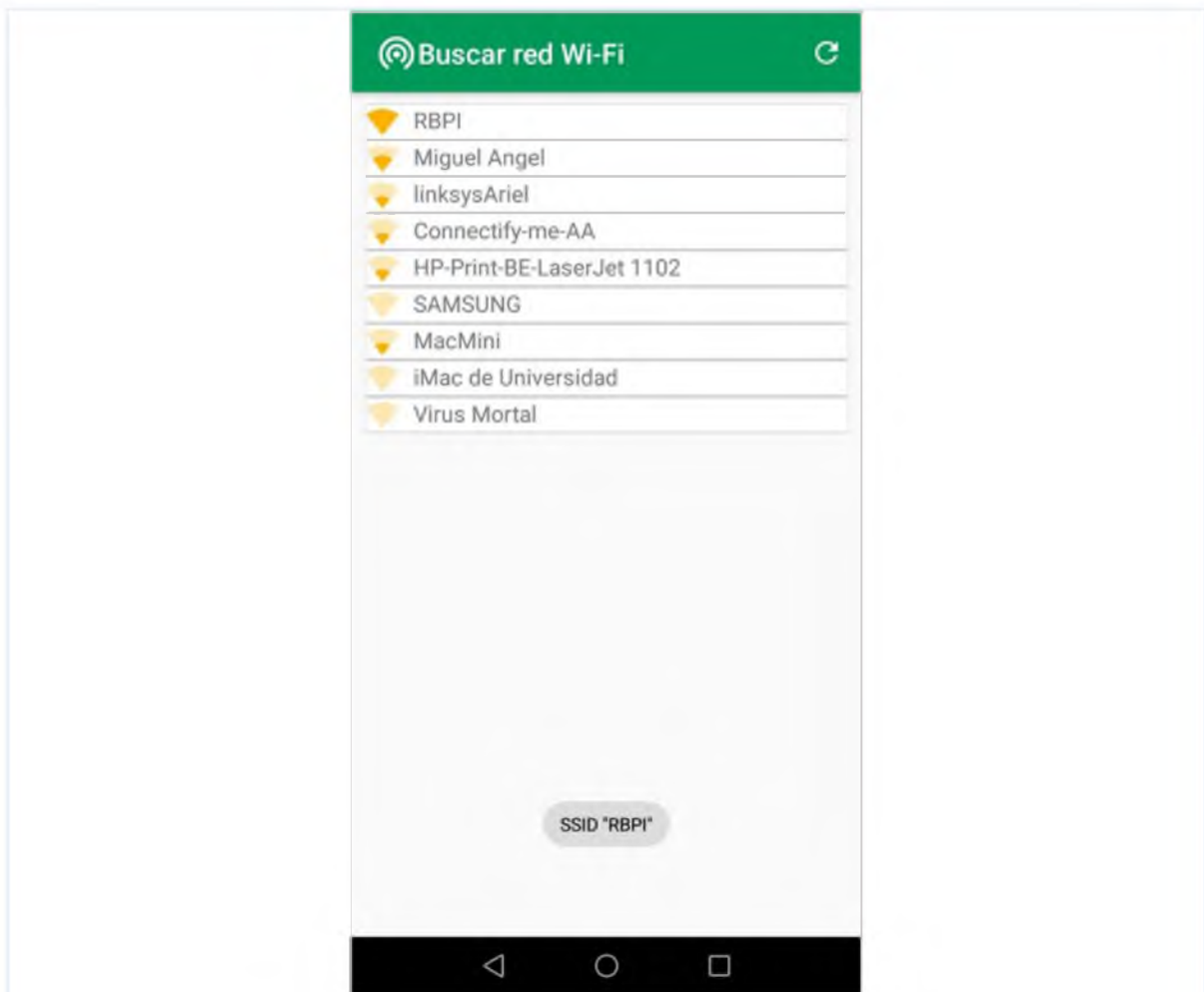


Figura 6.5: SSID del punto de acceso Wi-Fi.

Para el caso de la entrada donde el dispositivo Wi-Fi se encuentra desactivado, la aplicación lo activará de forma automática.

#### 6.2.2.4.4 Análisis de Resultados

Si el dispositivo Wi-Fi del teléfono inteligente se encuentra activado y la ubicación en el teléfono inteligente se ha activado manualmente (dependiendo de la versión de Android), se obtiene como resultado el SSID del punto de acceso Wi-Fi al cual se encuentra conectado y la lista de los puntos de acceso Wi-Fi.

Para el caso de la entrada donde el dispositivo Wi-Fi del teléfono inteligente se encuentra desactivado el resultado por parte de la aplicación cliente es la activación automática del dispositivo Wi-Fi del dispositivo.

### 6.2.2.5 Especificación APC-01-03.5

“Caso de prueba para el escaneo y listado de puntos de acceso Wi-Fi”.

#### 6.2.2.5.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.5*

#### 6.2.2.5.2 Especificaciones de entrada

En esta prueba se recibe como entrada:

- Se muestra la lista de puntos de acceso Wi-Fi.
- No se muestra la lista de puntos de acceso Wi-Fi.

#### 6.2.2.5.3 Especificaciones de salida

La aplicación cliente muestra la lista de puntos de acceso Wi-Fi disponibles, tal como se visualiza en la figura 6.6.



Figura 6.6: Puntos de acceso Wi-Fi.

Para el caso de la entrada donde no se muestra la lista de puntos de acceso Wi-Fi, es importante mencionar que es necesario activar previamente la ubicación en el teléfono inteligente; de esta forma se podrán ver los puntos de acceso Wi-Fi disponibles.

### **6.2.2.5.4 Análisis de Resultados**

Se muestran los SSID en la lista de puntos de acceso Wi-Fi visibles, en la cual el usuario podrá conectarse a uno de los puntos de acceso Wi-Fi, continuando así, con el proceso de navegación de forma correcta.

Si la lista de puntos de acceso Wi-Fi no es visible, el usuario debe activar la ubicación en el teléfono inteligente o en su caso podrá seleccionar el botón “Actualizar” con el cual se actualiza la lista de puntos de acceso Wi-Fi disponibles.

### **6.2.2.6 Especificación APC-01-03.6**

“Caso de prueba para actualizar la lista de puntos de acceso Wi-Fi”.

#### **6.2.2.6.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.6*

#### **6.2.2.6.2 Especificaciones de entrada**

En esta prueba se recibe como entrada:

- La lista de puntos de acceso Wi-Fi es actualizada.
- La lista de puntos de acceso Wi-Fi no es actualizada.

#### **6.2.2.6.3 Especificaciones de salida**

Si la lista de puntos de acceso Wi-Fi es actualizada, la aplicación cliente actualiza la lista de puntos de acceso Wi-Fi, tal como se visualiza en la figura 6.7.

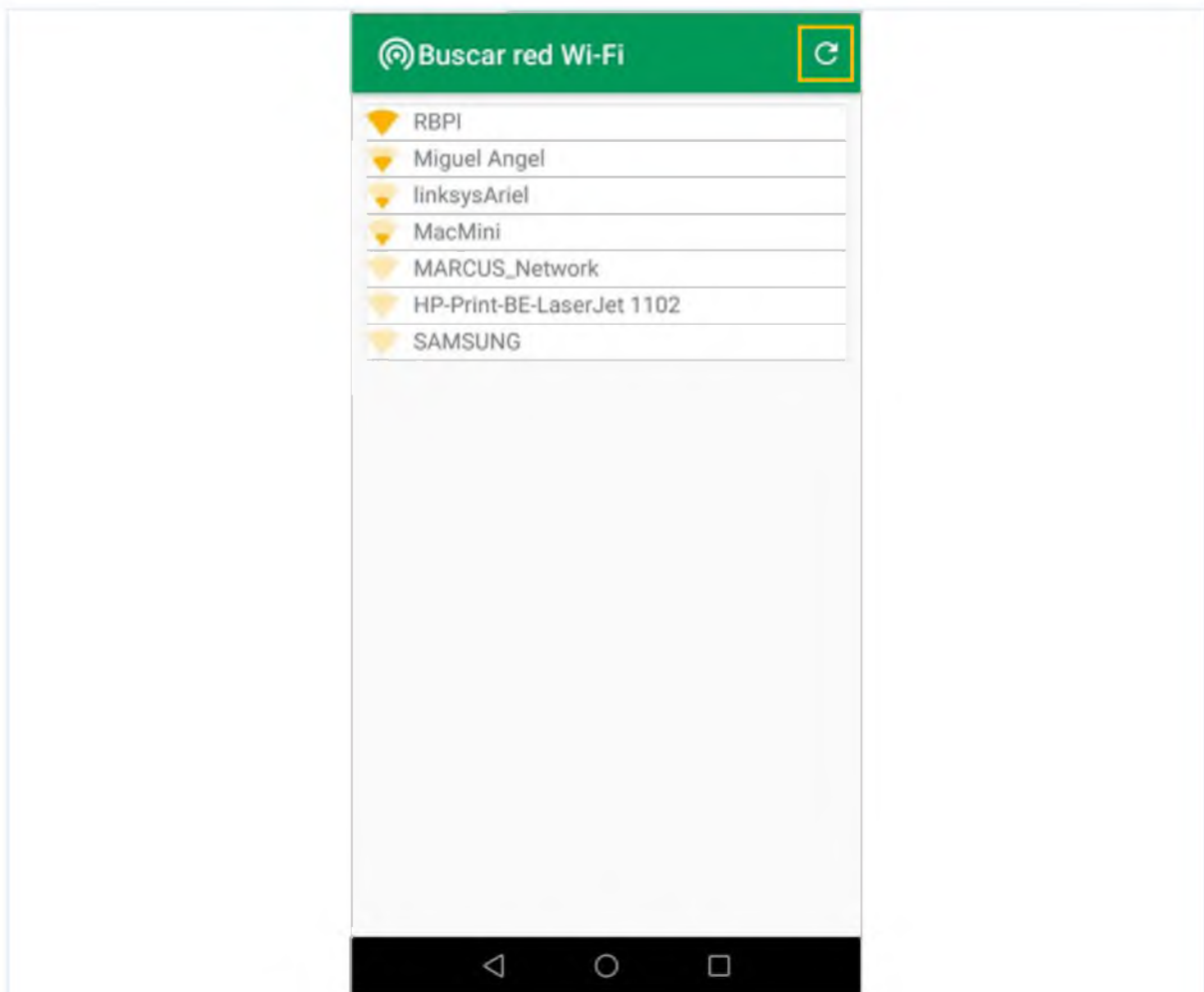


Figura 6.7: Actualizar lista de puntos de acceso Wi-Fi.

Para el caso de la entrada donde la lista de puntos de acceso Wi-Fi no es actualizada, simplemente la lista de puntos de acceso Wi-Fi se mantiene en el mismo estado.

#### 6.2.2.6.4 Análisis de Resultados

Para la entrada donde la lista de puntos de acceso Wi-Fi es actualizada, se obtiene como resultado la actualización de la lista de puntos de acceso, este proceso se puede realizar por ejemplo, cuando no se visualiza el punto de acceso al cual se desea conectar.

#### 6.2.2.7 Especificación APC-01-03.7

“Caso de prueba para la conexión a un punto de acceso Wi-Fi”.

##### 6.2.2.7.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.7*

#### 6.2.2.7.2 Especificaciones de entrada

En esta prueba se recibe como entrada:

- El teléfono inteligente se conecta a un punto de acceso Wi-Fi.
- El teléfono inteligente no se conecta a un punto de acceso Wi-Fi.

#### 6.2.2.7.3 Especificaciones de salida

Al intentar conectar el teléfono inteligente a un punto de acceso Wi-Fi, la aplicación muestra una ventana para ingresar la contraseña. En la figura 6.8 se muestra la ventana de conexión a un punto de acceso.

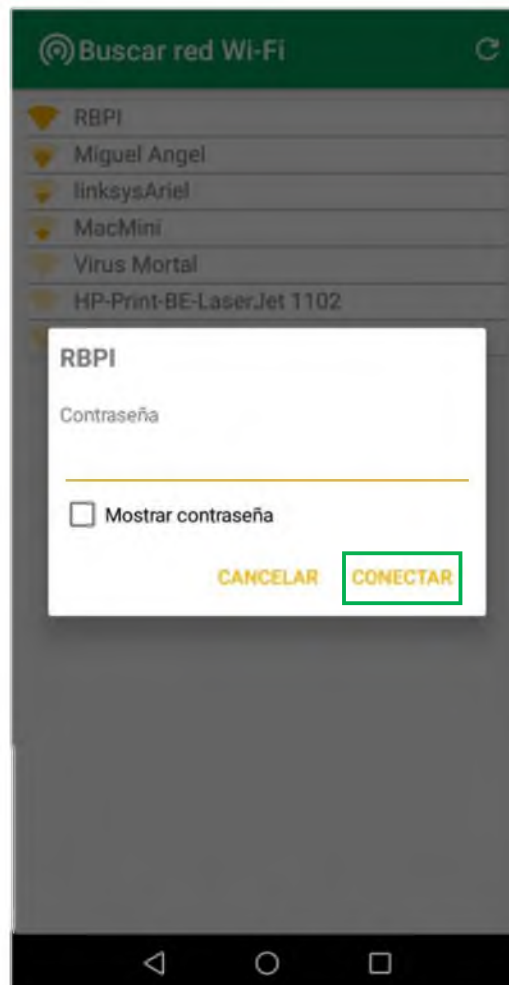


Figura 6.8: Conexión a un punto de acceso Wi-Fi.

Para el caso de la entrada donde el teléfono inteligente no se conecta a un punto de acceso Wi-Fi, la aplicación cliente no muestra la ventana principal con el botón “SELECCIONAR PRESENTACIÓN” habilitado hasta que se haya conectado.

#### 6.2.2.7.4 Análisis de Resultados

Si el teléfono inteligente se conecta de forma exitosa a un punto de acceso Wi-Fi, se muestra la pantalla principal de la aplicación cliente con el botón “SELECCIONAR PRESENTACIÓN” habilitado, con esto, el usuario podrá continuar con el proceso de navegación de forma correcta.

Para el caso donde el teléfono inteligente no se conecta a un punto de acceso Wi-Fi, el usuario debe ingresar la contraseña correcta para ese punto de acceso Wi-Fi.

#### 6.2.2.8 Especificación APC-01-03.8

“Caso de prueba para el explorador de archivos del teléfono inteligente”.

##### 6.2.2.8.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.8*

##### 6.2.2.8.2 Especificaciones de entrada

Al elegir el botón “SELECCIONAR PRESENTACIÓN” las entradas esperadas se presentan a continuación, para este caso de prueba:

- El explorador de archivos existe.
- El explorador de archivos no existe.

##### 6.2.2.8.3 Especificaciones de salida

La aplicación cliente muestra el explorador de archivos propio del teléfono inteligente, tal como se muestra en la figura 6.9.

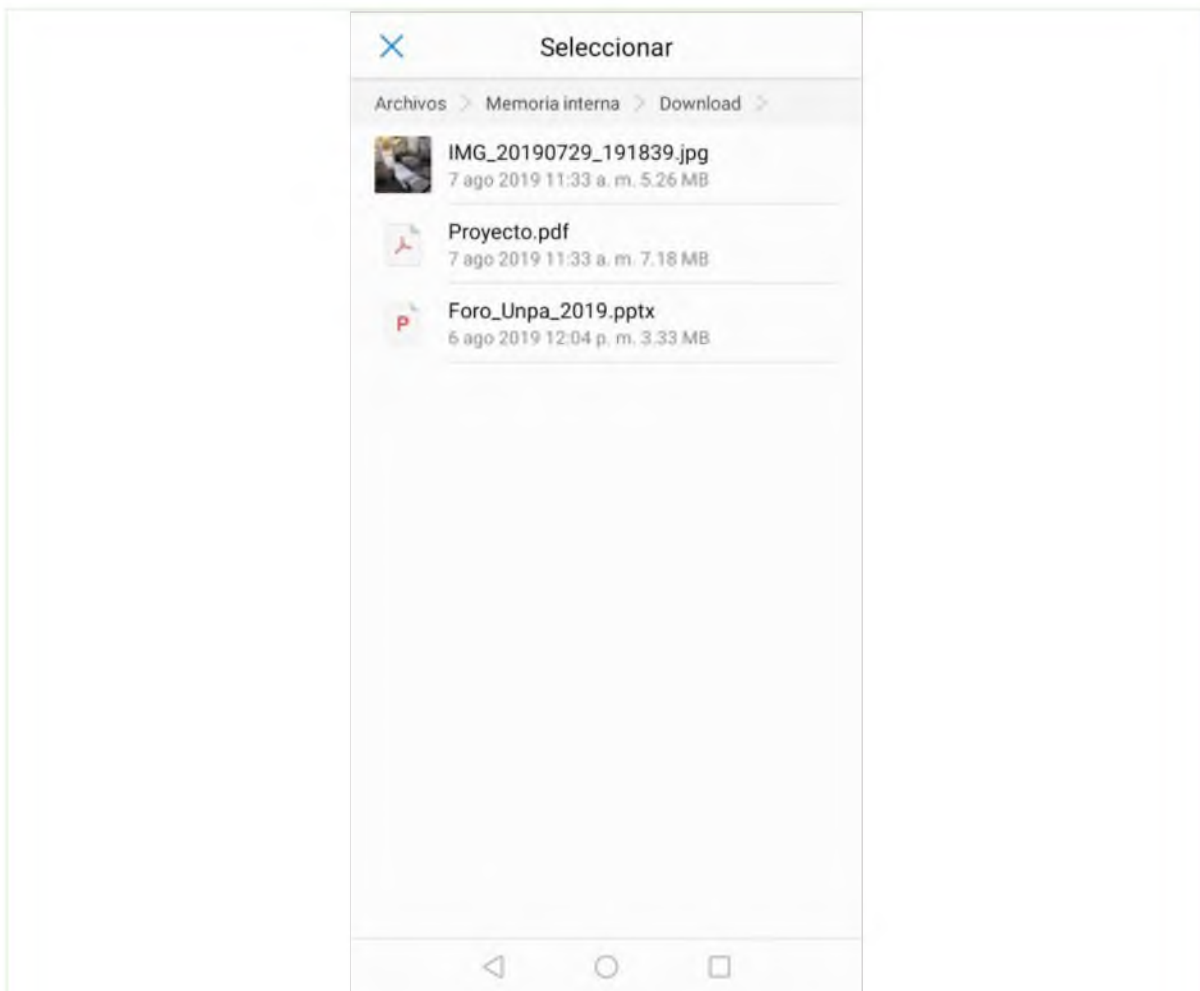


Figura 6.9: Explorador de archivos.

Para el caso donde el explorador de archivos no existe, se debe a la versión del teléfono inteligente, mostrando como salida el cierre de la aplicación.

#### 6.2.2.8.4 Análisis de Resultados

La primera entrada se cumple si el dispositivo que se utiliza tiene una versión mayor a la versión 4.4.2 y un nivel de API 19 como mínima, al cumplirse la entrada, la aplicación cliente muestra el explorador de archivos el cual es propio de teléfono inteligente.

Para el caso donde el explorador de archivos no existe, es porque el dispositivo cuenta con una versión menor a la versión 4.4.2 y un nivel de API menor que la API 19 y esto impide su visualización.

### 6.2.2.9 Especificación APC-01-03.9

“Caso de prueba para seleccionar un archivo de presentación multimedia”.

#### 6.2.2.9.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.9*

#### 6.2.2.9.2 Especificaciones de entrada

En esta prueba se recibe como entrada:

- Se visualiza la información básica del archivo.
- No se visualiza la información básica del archivo.

#### 6.2.2.9.3 Especificaciones de salida

La aplicación cliente muestra al usuario la información básica del archivo, tal como se muestra en la figura 6.10.

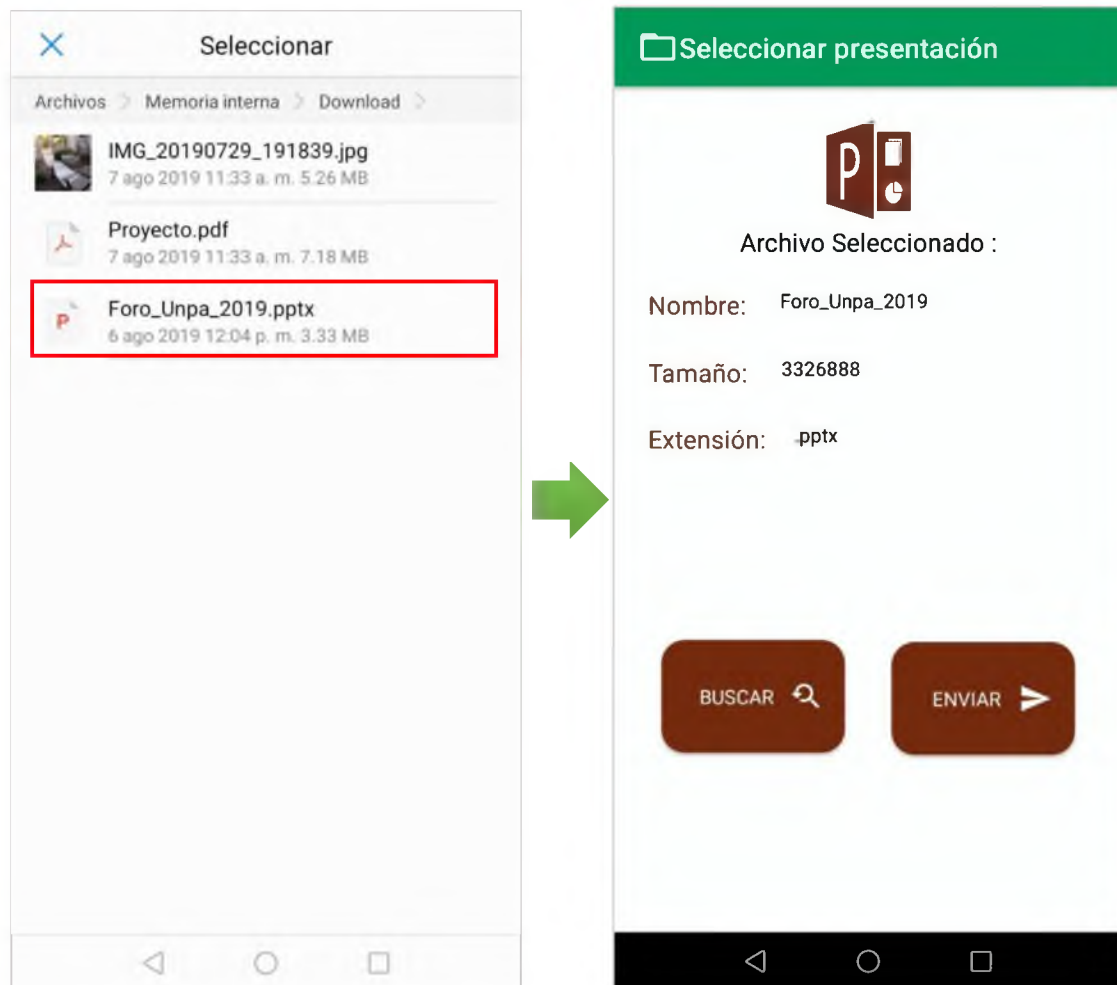


Figura 6.10: Información básica del archivo.

Para el caso donde no se visualiza la información, se debe a que no se genera el explorador de archivos.

#### 6.2.2.9.4 Análisis de Resultados

Si se visualiza la ventana de la información básica del archivo (nombre, tamaño y extensión), el usuario tiene la opción de enviar a la aplicación servidor el archivo de presentación multimedia.

Para el caso donde no se visualiza la información básica del archivo, ocasionado por no generarse el explorador de archivos debido a que el dispositivo utilizado cuenta con una versión de Android no compatible [ver sección 6.2.1.1 “Nomenclatura”], razón por la cual el archivo no podrá ser seleccionado.

#### 6.2.2.10 Especificación APC-01-03.10

“Caso de prueba para buscar otra presentación multimedia”.

##### 6.2.2.10.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.10*

##### 6.2.2.10.2 Especificaciones de entrada

En esta prueba se recibe como entrada:

- La ventana para seleccionar otra presentación.
- Ausencia de la ventana para seleccionar otra presentación.

##### 6.2.2.10.3 Especificaciones de salida

La aplicación permitirá al usuario elegir otra presentación multimedia al seleccionar el botón “**BUSCAR**”, como se muestra a continuación en la figura 6.11.

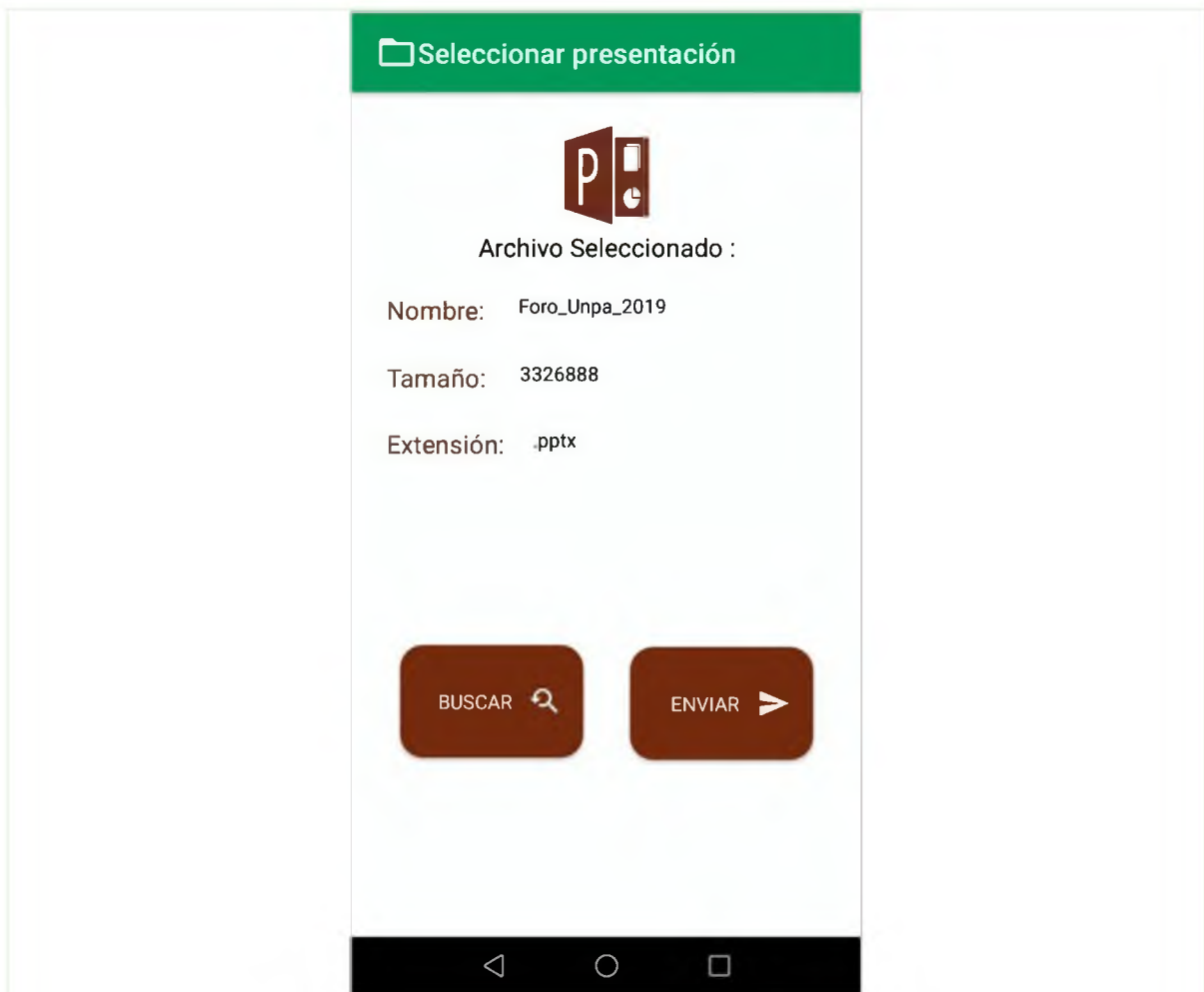


Figura 6.11: Buscar otro archivo de presentación multimedia.

#### 6.2.2.10.4 Análisis de Resultados

Si se elige el botón “**BUSCAR**” el usuario podrá navegar otra vez en el explorador de archivos para elegir otra presentación multimedia.

#### 6.2.2.11 Especificación APC-01-03.11

“Caso de prueba para validar la extensión del archivo”.

##### 6.2.2.11.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.11*

##### 6.2.2.11.2 Especificaciones de entrada

Al elegir el botón “SELECCIONAR PRESENTACIÓN” las entradas esperadas se presentan a continuación, para este caso de prueba:

- La extensión del archivo seleccionado es pptx.
- La extensión del archivo seleccionado es diferente a pptx.

### 6.2.2.11.3 Especificaciones de salida

La aplicación cliente puede enviar la presentación multimedia a la aplicación servidor.

Para el caso donde la extensión del archivo seleccionado es diferente a pptx, el resultado por parte de la aplicación cliente es un mensaje de advertencia para elegir un archivo con la extensión correcta, tal como se muestra en la figura 6.12.



Figura 6.12: Validación de la extensión del archivo.

#### 6.2.2.11.4 Análisis de Resultados

Si el archivo seleccionado tiene la extensión pptx, éste podrá ser enviado a la aplicación servidor. Para el caso donde la extensión del archivo seleccionado es diferente a pptx, la aplicación cliente no podrá enviar el archivo a la aplicación servidor.

#### 6.2.2.12 Especificación APC-01-03.12

“Caso de prueba para enviar la presentación multimedia”.

##### 6.2.2.12.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.12*

##### 6.2.2.12.2 Especificaciones de entrada

En esta prueba se recibe como entrada:

- La presentación multimedia es enviada a la aplicación servidor.
- La presentación multimedia no pudo ser enviada a la aplicación servidor.

##### 6.2.2.12.3 Especificaciones de salida

La presentación multimedia será guardada en el directorio definido en el servidor Raspberry Pi y posteriormente proyectada.

Para el caso donde la presentación multimedia no pudo ser enviada a la aplicación servidor, la aplicación cliente muestra la ventana principal para volver a seleccionar una presentación.

##### 6.2.2.12.4 Análisis de Resultados

Si la presentación multimedia es enviada y recibida exitosamente por la aplicación servidor, la aplicación cliente muestra la ventana principal con el botón **“CONTROL REMOTO”** habilitado para posteriormente controlar la presentación.

Por el contrario, si la presentación multimedia no pudo ser enviada a la aplicación servidor, puede ser porque no se ha ejecutado o iniciado la aplicación servidor, para este caso, se recomienda ejecutar o iniciar primero la aplicación servidor.

#### 6.2.2.13 Especificación APC-01-03.13

“Caso de prueba para el listado del servidor de LibreOffice”.

##### 6.2.2.13.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.13*

#### 6.2.2.13.2 Especificaciones de entrada

Al elegir el botón “CONTROL REMOTO” las entradas esperadas se presentan a continuación, para este caso de prueba:

- El nombre del servidor de LibreOffice se muestra en la lista.
- El nombre del servidor de LibreOffice no se muestra en la lista.

#### 6.2.2.13.3 Especificaciones de salida

La ventana principal del módulo control remoto muestra el nombre del servidor de LibreOffice, tal como se visualiza en la figura 6.13.

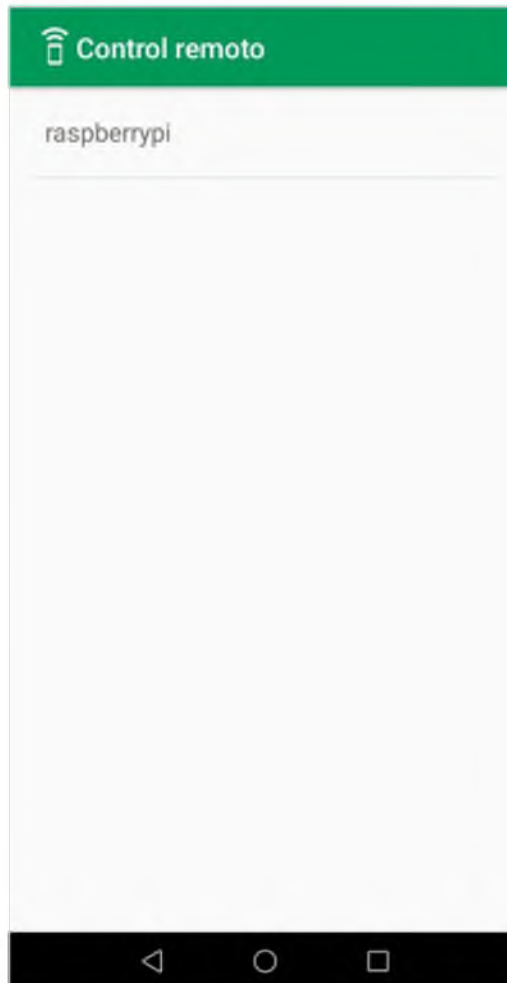


Figura 6.13: Servidor de LibreOffice.

La segunda entrada indica que la lista se encuentra vacía, por lo tanto, la aplicación muestra una ProgressBar (barra de progreso) y un mensaje de advertencia en pantalla para asegurar que LibreOffice se ejecuta en el servidor, tal como se muestra en la figura 6.14.

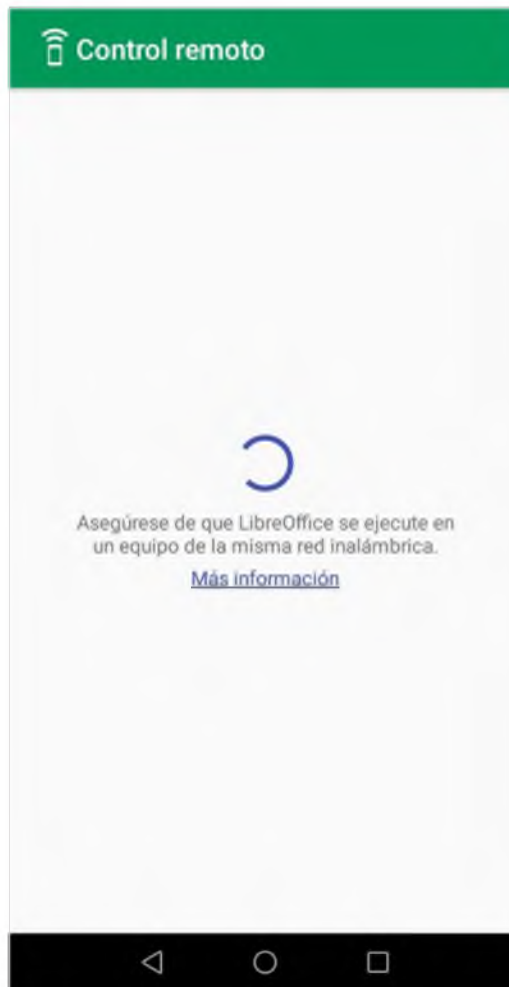


Figura 6.14: Servidor de LibreOffice sin ejecutarse.

#### 6.2.2.13.4 Análisis de Resultados

Si el nombre del servidor de LibreOffice se muestra en la lista de la ventana principal del módulo control remoto, la presentación multimedia estará lista para ser controlada por medio del teléfono inteligente.

Para el caso de la segunda entrada donde no se visualiza el nombre del servidor de LibreOffice, la presentación multimedia no podrá ser controlada.

**6.2.2.14 Especificación APC-01-03.14**

“Caso de prueba para validar el emparejamiento y ejecución de la presentación”.

**6.2.2.14.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.14*

**6.2.2.14.2 Especificaciones de entrada**

Al elegir el nombre del servidor de LibreOffice, las entradas esperadas se presentan a continuación, para este caso de prueba:

- LibreOffice es ejecutado en el servidor.
- LibreOffice no es ejecutado en el servidor.

**6.2.2.14.3 Especificaciones de salida**

Si LibreOffice es ejecutado en el servidor y hay un correcto emparejamiento se ejecuta la presentación multimedia, tal como se visualiza en la figura 6.15.

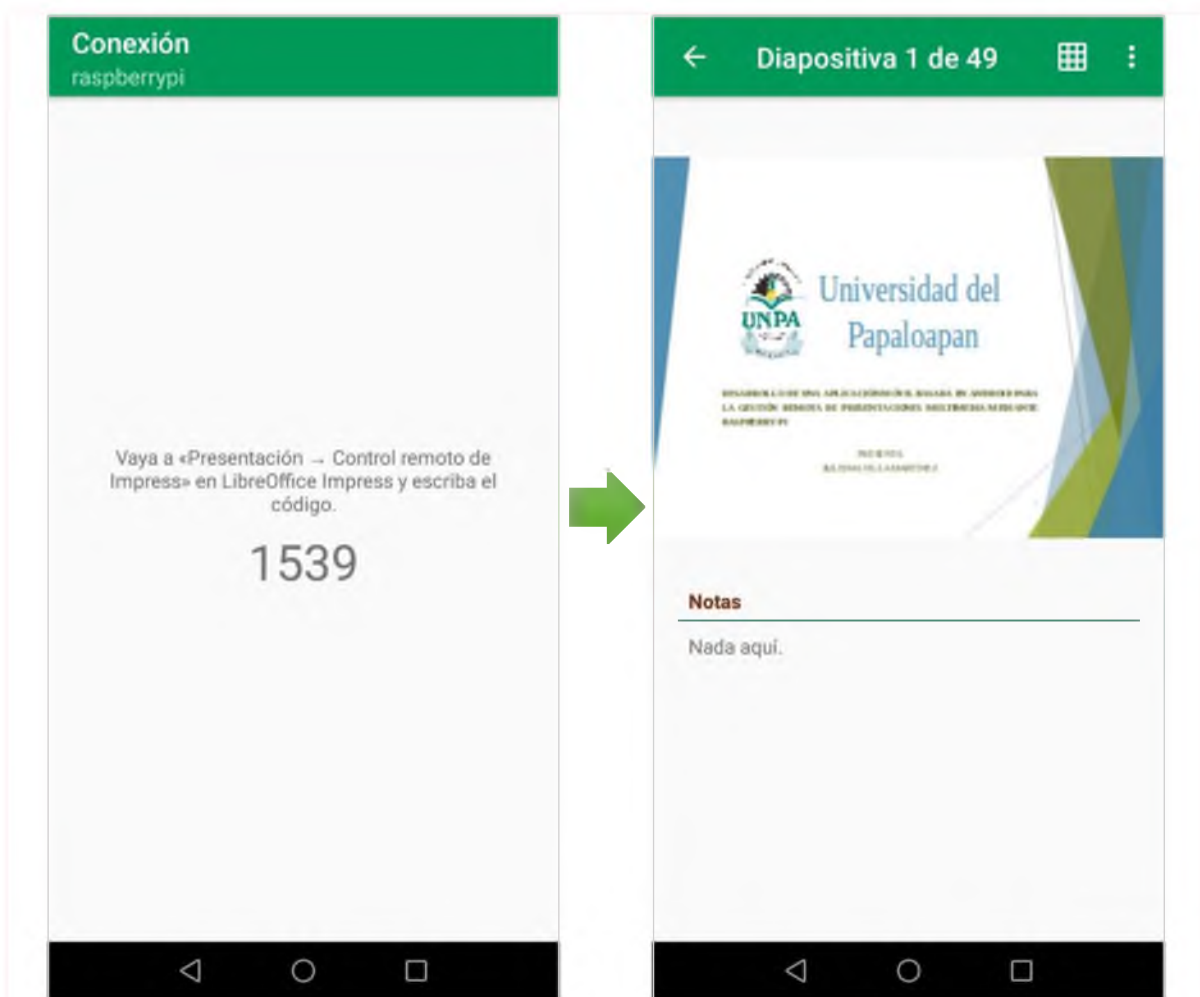


Figura 6.15: Emparejamiento y ejecución de una presentación multimedia.

#### 6.2.2.14.4 Análisis de Resultados

Si LibreOffice Impress es ejecutado en el servidor, se puede lograr el emparejamiento entre el teléfono inteligente y el servidor de LibreOffice. Por el contrario, si LibreOffice Impress no es ejecutado en el servidor no se logrará el emparejamiento y por lo tanto no se podrá ejecutar la presentación multimedia.

#### 6.2.2.15 Especificación APC-01-03.15

“Caso de prueba para visualizar las diapositivas en forma de maya”.

##### 6.2.2.15.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: **APC-01-02.15**

#### 6.2.2.15.2 Especificaciones de entrada

Las entradas esperadas se presentan a continuación, para este caso de prueba:

- La presentación multimedia es ejecutada en el servidor.
- La presentación multimedia no es ejecutada en el servidor.

#### 6.2.2.15.3 Especificaciones de salida

Si la presentación multimedia es ejecutada en el servidor, se podrá elegir la opción para visualizar las diapositivas en forma de maya, así como el total de diapositivas y el número actual de la diapositiva seleccionada, tal como se visualiza en la figura 6.16.



Figura 6.16: Visualización de diapositivas en forma de maya.

#### 6.2.2.15.4 Análisis de Resultados

La visualización de las diapositivas en forma de maya depende de la ejecución de la presentación multimedia, además del correcto emparejamiento entre el teléfono inteligente y el servidor de LibreOffice.

### **6.2.2.16 Especificación APC-01-03.16**

“Caso de prueba para el botón “Parar presentación””.

#### **6.2.2.16.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APC-01-02.16*

#### **6.2.2.16.2 Especificaciones de entrada**

La entrada esperada se presenta a continuación, para este caso de prueba:

- La presentación multimedia es ejecutada en el servidor o se visualizan las diapositivas en forma de maya.

#### **6.2.2.16.3 Especificaciones de salida**

Si la presentación multimedia está siendo ejecutada o se visualizan las diapositivas en forma de maya, se podrá seleccionar el botón “Parar presentación”, tal como se muestra en la figura 6.17.



Figura 6.17: Parar presentación.

#### 6.2.2.16.4 Análisis de Resultados

Si la presentación multimedia es ejecutada o se visualizan las diapositivas en forma de maya, se puede detener la presentación de tres formas posibles: 1) seleccionando el menú "Parar presentación" (como se muestra en la figura anterior), 2) seleccionando la flecha atrás de la barra superior (Toolbar) y 3) seleccionando el botón atrás del teléfono inteligente.

### 6.3 Plan de pruebas de la aplicación servidor (APS-01-01)

En este plan de pruebas se verifica que la funcionalidad de la aplicación servidor sea exitosa al esperar conexiones entrantes de clientes y al recibir exitosamente archivos de presentación multimedia en un directorio definido; y así proyectarlos por medio de un proyector digital.

El nombre de la aplicación servidor se abrevia Aplicación Servidor APS para referirlo de una forma más simple en este documento.

Las características de los requerimientos para realizar las pruebas a la aplicación servidor se muestran en la tabla 6.6:

Tabla 6.6: Requerimientos de prueba de la App servidor.

Servidor	
Equipo	Sistema Operativo
Tarjeta Raspberry Pi	Raspbian Stretch (Linux)
Otros requerimientos	
Proyector digital	Indefinido

El plan de pruebas de la aplicación (APS-01-01) está formado por una especificación del plan de pruebas y una especificación de los casos de prueba. Tales especificaciones se describen a continuación:

#### 6.3.1 Especificación del Plan de Pruebas de la aplicación servidor

##### 6.3.1.1 Nomenclatura

Se presenta a continuación la nomenclatura que será utilizada a lo largo del documento referente al plan de pruebas para la aplicación servidor, la cual forma parte del sistema “Raspdroid-Projection” presentado en este documento de tesis. El formato establecido para referirse a cada apartado del plan de pruebas de la aplicación servidor es **APS-XX-YY**. Donde:

- **APS** es el prefijo utilizado para referirse a Aplicación Servidor, como referencia a la aplicación servidor.

- **XX** es el prefijo utilizado para representar el tipo de concepto a ser evaluado.
- **YY** es el prefijo utilizado para representar un valor consecutivo del concepto.

A continuación, se definen cada uno de los conceptos y valores consecutivos para cada uno de ellos utilizados en el plan de prueba:

Tabla 6.7: Concepto del plan de pruebas de la aplicación servidor.

Valor	Tipo
01	Documentación de Pruebas
02	Módulos de la Aplicación Servidor
03	Funciones de la Aplicación Servidor

Tabla 6.8: Documentación de pruebas de la aplicación servidor.

Formato	Descripción
APS-01-01	Plan de Pruebas de la aplicación
APS-01-02	Especificación del Diseño de Pruebas
APS-01-03	Especificación de los Casos de Prueba

Tabla 6.9: Módulos de la aplicación servidor.

Formato	Descripción
APS-02-01	Módulo de la red hotspot Wi-Fi.
APS-02-02	Módulo de la aplicación servidor.
APS-02-03	Módulo del servidor de LibreOffice.

Tabla 6.10: Funciones de la aplicación servidor.

Formato	Descripción
APS-03-01	Permitir a los teléfonos inteligentes conectarse a un punto de acceso Wi-Fi.
APS-03-02	Gestionar la interfaz de usuario de la aplicación servidor.
APS-03-03	Utilizar LibreOffice Impress como servidor para abrir y controlar archivos de presentación multimedia.

Las pruebas en la aplicación servidor no contemplan probar la siguiente característica:

- Pruebas de la aplicación con un sistema operativo diferente al sistema operativo GNU/Linux basado en Debian Raspbian Stretch.

### 6.3.2 Especificación de los Casos de Prueba de la aplicación servidor (APS-01-01)

#### 6.3.2.1 Especificación APS-01-03.1

“Caso de prueba para comprobar que la red hotspot Wi-Fi emite señal”.

##### 6.3.2.1.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APS-01-02.1*

##### 6.3.2.1.2 Especificaciones de entrada

Al iniciar el servidor Raspberry Pi las entradas esperadas se presentan a continuación, para este caso de prueba:

- La red hotspot Wi-Fi emite señal.
- La red hotspot Wi-Fi no emite señal.

##### 6.3.2.1.3 Especificaciones de salida

Si la red hotspot creada emite señal, es visible el nombre del punto de acceso Wi-Fi creado [como se visualiza en la figura 6.18], de esta forma los teléfonos inteligentes podrán conectarse al hotspot.

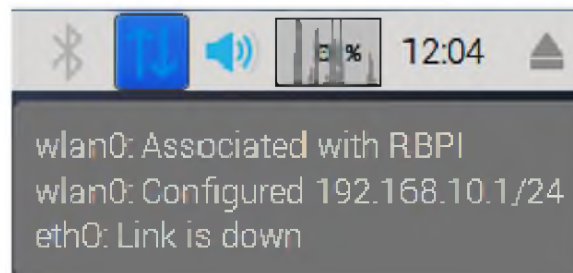


Figura 6.18: Red hotspot Wi-Fi emitiendo señal.

Para el caso de la entrada donde la red hotspot Wi-Fi no emite señal, no se visualiza el nombre del punto de acceso Wi-Fi creado.

##### 6.3.2.1.4 Análisis de Resultados

Para el caso de la entrada donde la red hotspot Wi-Fi emite señal, se obtiene como resultado la visualización del nombre del punto de acceso Wi-Fi creado, para posteriormente conectarse a ese punto de acceso. En el caso de la entrada donde la red hotspot Wi-Fi no emite señal, no se visualiza el nombre del punto de acceso Wi-Fi creado, de tal forma que el usuario no podrá conectarse al punto de acceso.

### 6.3.2.2 Especificación APS-01-03.2

“Caso de prueba para la ejecución de la aplicación servidor”.

#### 6.3.2.2.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APS-01-02.2*

#### 6.3.2.2.2 Especificaciones de entrada

Al ejecutar la aplicación, las entradas esperadas se presentan a continuación, para este caso de prueba:

- La aplicación servidor se ejecuta de forma correcta.
- La aplicación servidor no se ejecuta de forma correcta.

#### 6.3.2.2.3 Especificaciones de salida

Si la aplicación servidor se ejecuta de forma correcta, se muestra como resultado la interfaz de la aplicación como se visualiza en la figura 6.19.

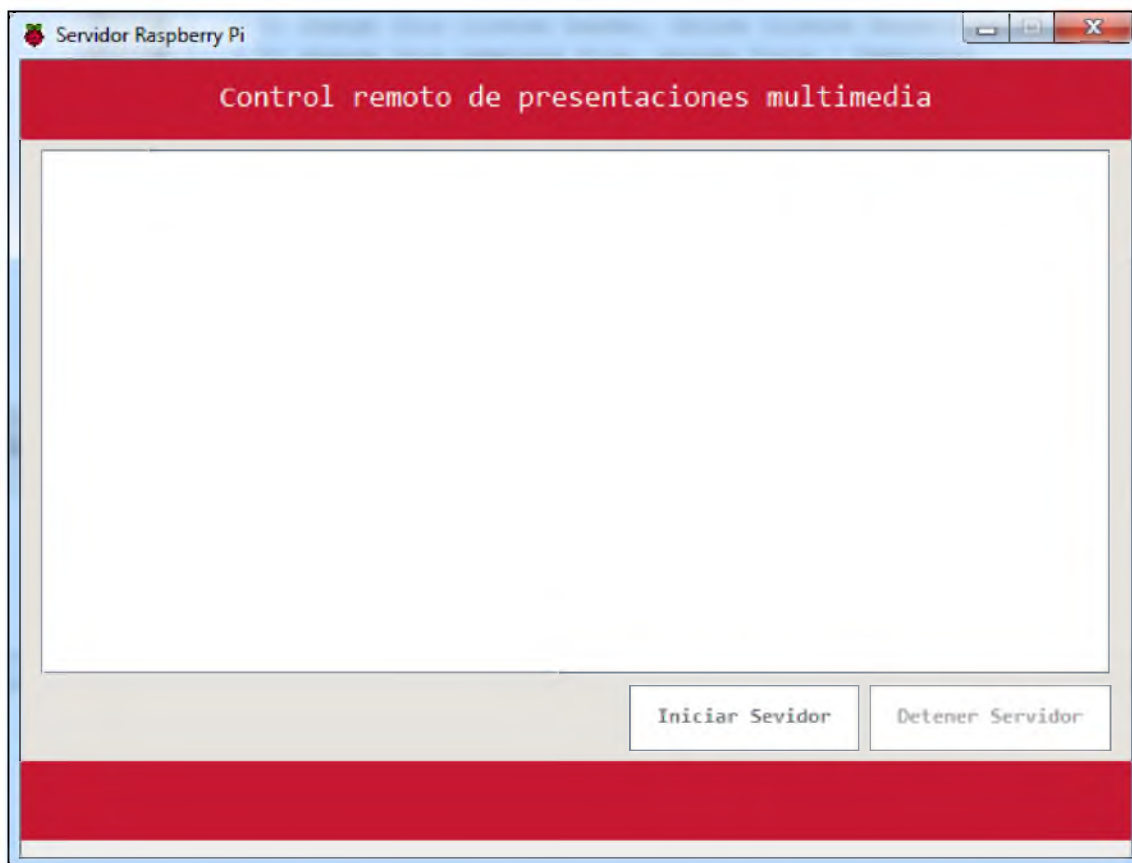


Figura 6.19: Ejecución de la aplicación servidor.

Para el caso de la entrada donde la aplicación servidor no se ejecuta de forma correcta, no se muestra la interfaz de la aplicación.

### **6.3.2.2.4 Análisis de Resultados**

Para el caso de la entrada donde la aplicación servidor se ejecuta de forma correcta, se obtiene como resultado la visualización de la interfaz principal de la aplicación servidor. En el caso de la entrada donde la aplicación servidor no se ejecuta de forma correcta, simplemente la aplicación no muestra su interfaz principal.

### **6.3.2.3 Especificación APS-01-03.3**

“Caso de prueba para la selección del botón “Iniciar Servidor” de la aplicación servidor”.

#### **6.3.2.3.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APS-01-02.3*

#### **6.3.2.3.2 Especificaciones de entrada**

Al seleccionar el botón “Iniciar Servidor”, las entradas esperadas se presentan a continuación, para este caso de prueba:

- Se ejecuta de forma correcta la multitarea iniciada en la aplicación servidor.
- No se ejecuta de forma correcta la multitarea iniciada en la aplicación servidor.

#### **6.3.2.3.3 Especificaciones de salida**

Si se ejecuta de forma correcta la multitarea iniciada en la aplicación servidor, se muestra como resultado la interfaz principal como se visualiza en la figura 6.20.

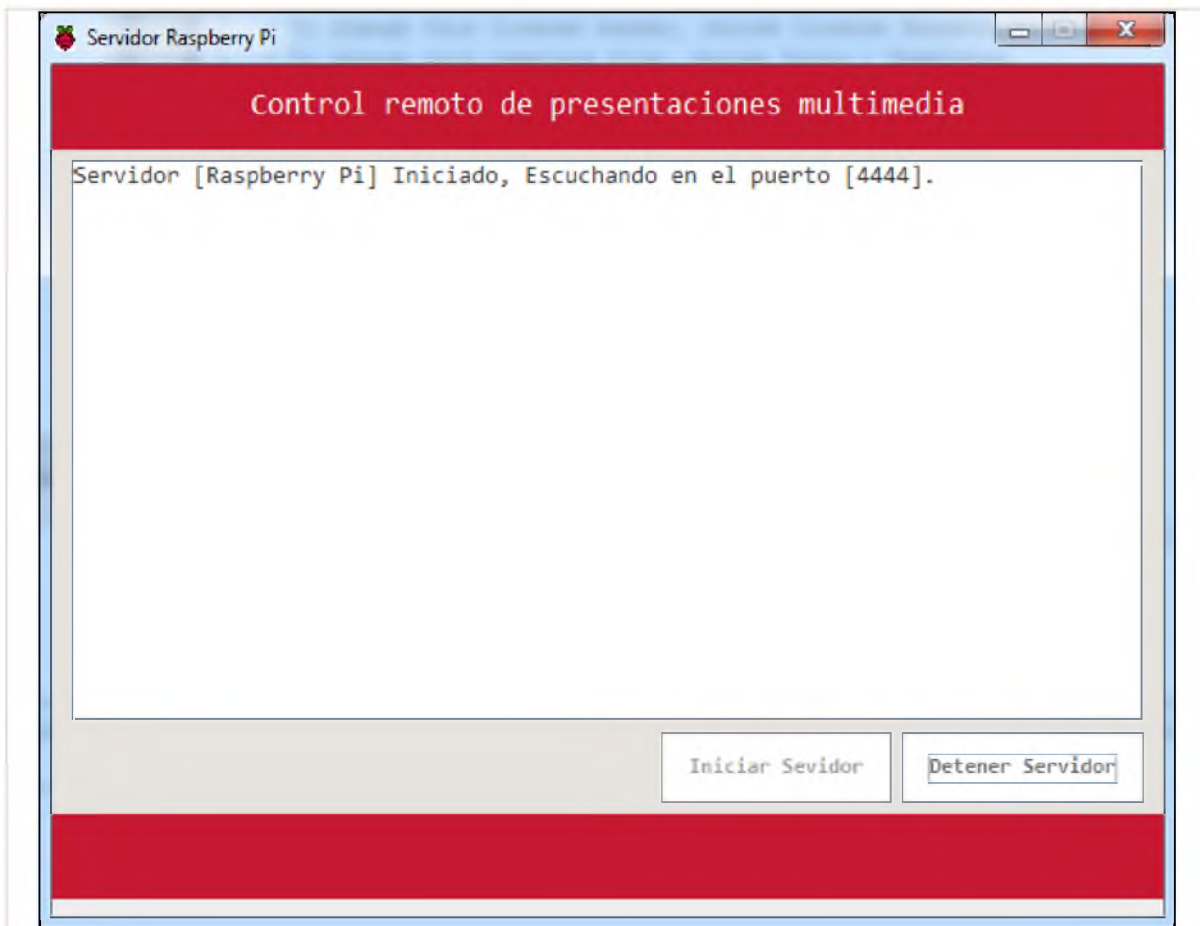


Figura 6.20: Servidor iniciado.

Para el caso de la entrada donde la multitarea iniciada en la aplicación servidor no se ejecuta de forma correcta, se genera un error.

#### 6.3.2.3.4 Análisis de Resultados

Para el caso de la entrada donde se ejecuta de forma correcta la multitarea iniciada en la aplicación servidor, se obtiene como resultado la interfaz principal con un mensaje que se ha iniciado la aplicación servidor y que está escuchando en el puerto indicado. En el caso de la entrada donde no se ejecuta de forma correcta la multitarea iniciada en la aplicación servidor, la aplicación genera un error al seleccionar el botón “Iniciar Servidor”.

#### 6.3.2.4 Especificación APS-01-03.4

“Caso de prueba para recibir un archivo de presentación multimedia”.

##### 6.3.2.4.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: **APS-01-02.4**

#### 6.3.2.4.2 Especificaciones de entrada

Las entradas esperadas se presentan a continuación, para este caso de prueba:

- El servidor Raspberry Pi recibe el archivo de presentación multimedia.
- El servidor Raspberry Pi no recibe el archivo de presentación multimedia.

#### 6.3.2.4.3 Especificaciones de salida

Si el servidor Raspberry Pi recibe el archivo de presentación multimedia de forma exitosa, la interfaz de la aplicación servidor muestra el mensaje apropiado como se visualiza en la figura 6.21.

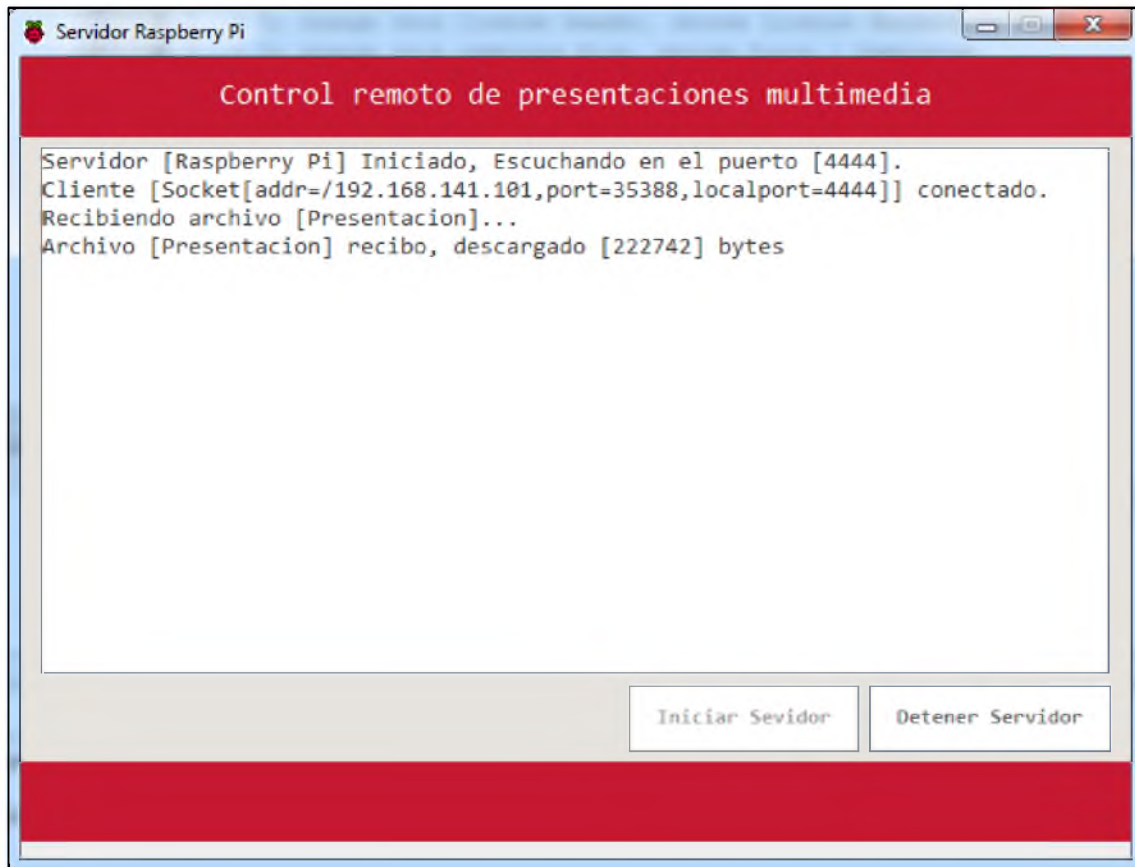


Figura 6.21: Recepción de un archivo de presentación multimedia.

Para el caso de la entrada donde el servidor Raspberry Pi no recibe el archivo de presentación multimedia, muestra un mensaje de error.

#### 6.3.2.4.4 Análisis de Resultados

Para el caso de la entrada donde el servidor Raspberry Pi recibe el archivo de presentación multimedia, éste se guarda en un directorio definido, mostrando así el progreso de descarga del archivo. En el caso de la entrada donde el servidor Raspberry Pi no recibe el archivo de presentación multimedia, no se guarda nada en el directorio definido.

### **6.3.2.5 Especificación APS-01-03.5**

“Caso de prueba para la selección del botón “Detener Servidor” de la aplicación servidor”.

#### **6.3.2.5.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: **APS-01-02.5**

#### **6.3.2.5.2 Especificaciones de entrada**

Al detener la aplicación servidor las entradas esperadas se presentan a continuación, para este caso de prueba:

- La aplicación servidor se detiene de forma correcta.
- La aplicación servidor no se detiene de forma correcta.

#### **6.3.2.5.3 Especificaciones de salida**

Si la aplicación servidor se detiene de forma correcta, se muestra como resultado un mensaje como se visualiza en la figura 6.22.

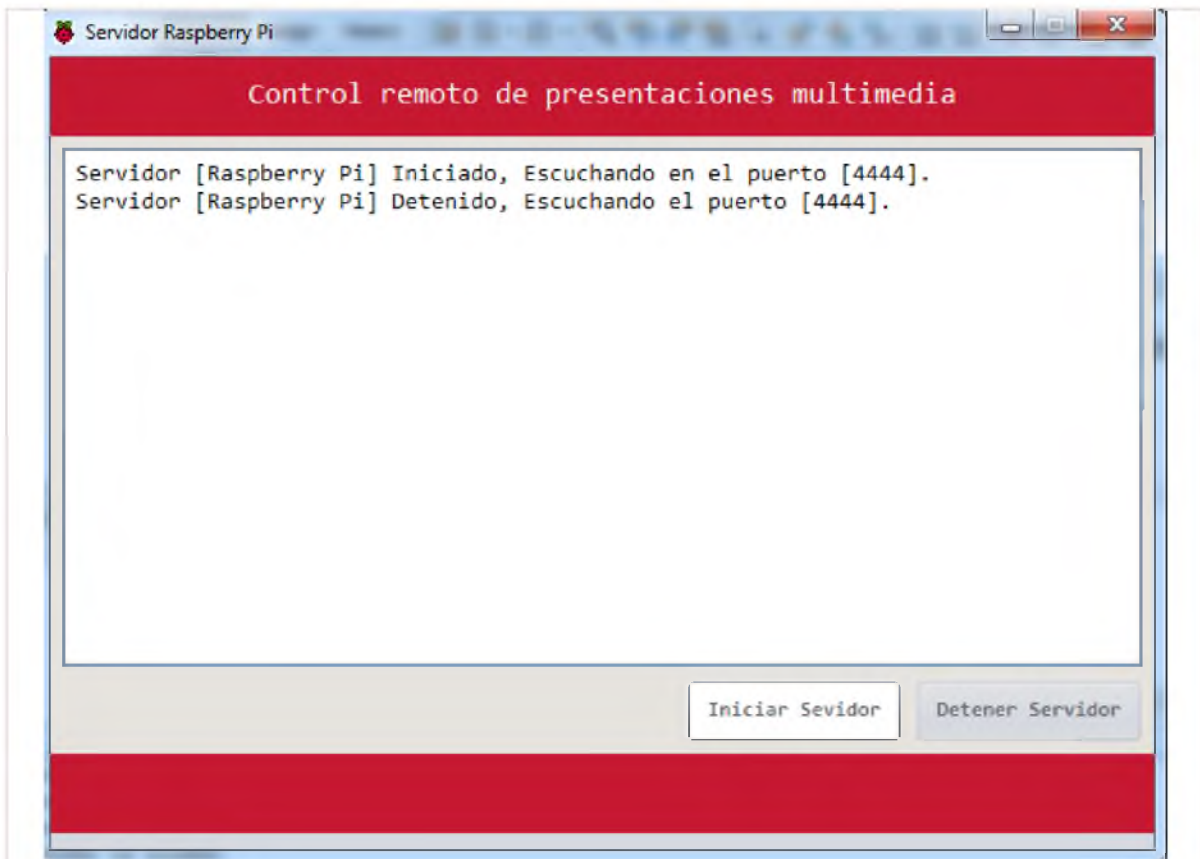


Figura 6.22: Servidor detenido.

Para el caso de la entrada donde la aplicación servidor no se detiene de forma correcta, se genera como salida un bloqueo de la aplicación.

#### 6.3.2.5.4 Análisis de Resultados

Para el caso de la entrada donde la aplicación servidor se detiene de forma correcta, los clientes no podrán establecer una conexión segura con la aplicación servidor, hasta que ésta vuelva a iniciarse; además de que no podrán enviar una presentación multimedia al directorio definido. En el caso de la entrada donde la aplicación servidor no se detiene de forma correcta, la aplicación servidor se bloquea.

#### 6.3.2.6 Especificación APS-01-03.6

“Caso de prueba para comprobar que el servidor de LibreOffice es iniciado”.

##### 6.3.2.6.1 Elementos de Prueba

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: **APS-01-02.6**

#### 6.3.2.6.2 Especificaciones de entrada

Las entradas esperadas se presentan a continuación, para este caso de prueba:

- LibreOffice Impress se ha iniciado en el servidor Raspberry Pi.
- LibreOffice Impress no se ha iniciado en el servidor Raspberry Pi.

#### 6.3.2.6.3 Especificaciones de salida

Si LibreOffice Impress se ha iniciado en el servidor Raspberry Pi, se podrá abrir la presentación multimedia, obteniendo como resultado la ventana que se muestra en la figura 6.23.



Figura 6.23: LibreOffice Impress iniciado.

Para el caso de la entrada donde LibreOffice Impress no se ha iniciado en el servidor, no se podrá abrir la presentación multimedia.

#### 6.3.2.6.4 Análisis de Resultados

Para el caso de la entrada donde LibreOffice Impress se ha iniciado en el servidor Raspberry Pi, se puede abrir la presentación multimedia guardada en el directorio del servidor. En el caso de la entrada donde LibreOffice Impress no se ha iniciado en el servidor Raspberry Pi, no se podrá abrir ni controlar la presentación multimedia.

**6.3.2.7 Especificación APS-01-03.7**

“Caso de prueba para el emparejamiento”.

**6.3.2.7.1 Elementos de Prueba**

Las características a ser probadas son aquellas definidas en la especificación del diseño de prueba: *APS-01-02.7*

**6.3.2.7.2 Especificaciones de entrada**

Las entradas esperadas se presentan a continuación, para este caso de prueba:

- Se realiza el emparejamiento.
- No se realiza el emparejamiento.

**6.3.2.7.3 Especificaciones de salida**

Si se realiza el emparejamiento, las ventanas de conexión se visualizan en la figura 6.24 y figura 6.25, respectivamente.

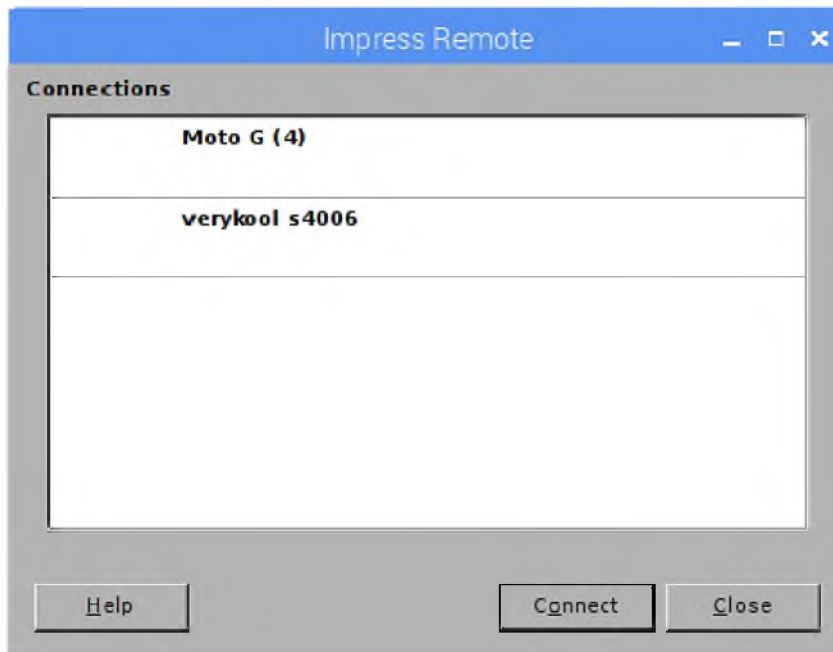


Figura 6.24: Emparejamiento (1).

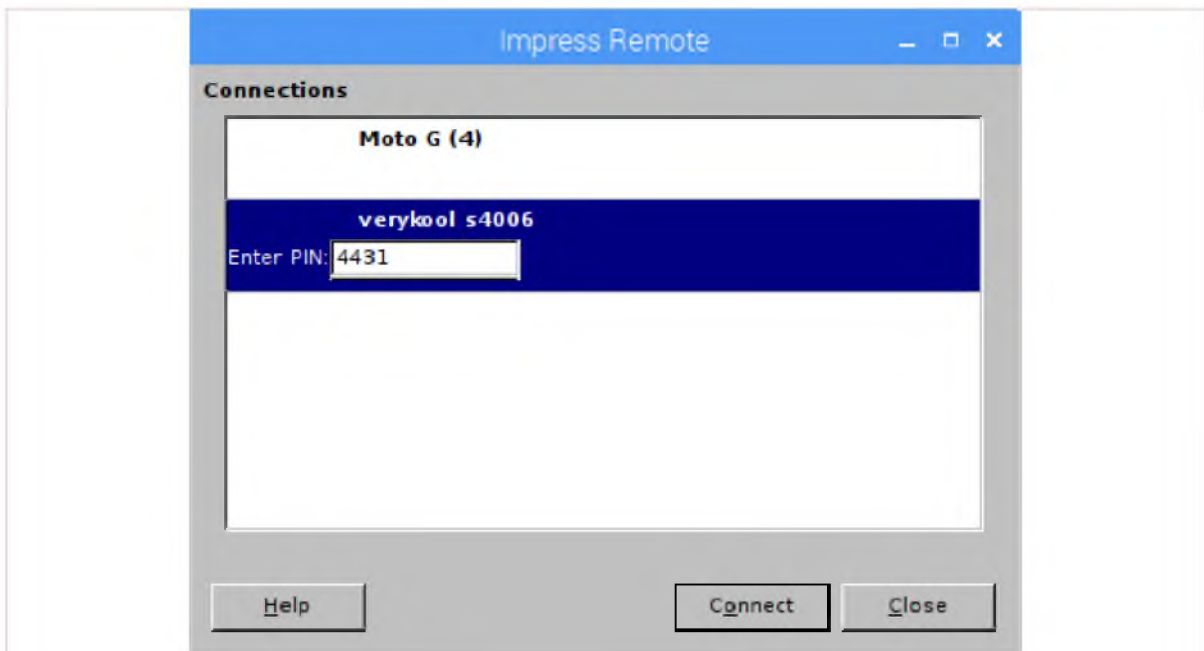


Figura 6.25: Emparejamiento (2).

Para el caso de la entrada donde no se realiza el emparejamiento, no se ejecuta la presentación multimedia.

#### 6.3.2.7.4 Análisis de Resultados

Para el caso de la entrada donde se realiza el emparejamiento entre el teléfono inteligente y el servidor de LibreOffice, se obtiene como resultado la correcta ejecución de la presentación multimedia. En el caso de la entrada donde no se realiza el emparejamiento entre el teléfono inteligente y el servidor de LibreOffice, no se podrá ejecutar tal presentación.

## 6.4 Resultados del sistema “Raspdroid-Projection”

A continuación, se describe cada uno de los resultados del sistema “Raspdroid-Projection” obtenidos con respecto a cada una de las pruebas realizadas a la aplicación cliente y a la aplicación servidor.

### 6.4.1 Resultados de los casos de prueba de la aplicación cliente

Los resultados obtenidos y registrados en esta sección son en base a las pruebas realizadas a la aplicación cliente utilizando tres teléfonos inteligentes con modelos y versiones completamente diferentes; esto permite analizar de una forma detallada el comportamiento de cada caso de prueba.

Tabla 6.11: Análisis de resultados del plan de pruebas con el dispositivo Verykool.

Pruebas con el modelo Verykool Leo 3G s4006			
Caso de prueba	Pruebas realizadas	Pruebas funcionales	Valor de referencia
1. Comprobar el estado del botón “ <b>BUSCAR RED WI-FI</b> ”	10	10	10/10=1
2. Comprobar el estado del botón “ <b>SELECCIONAR PRESENTACIÓN</b> ”	10	10	10/10=1
3. Comprobar el estado del botón “ <b>CONTROL REMOTO</b> ”	10	10	10/10=1
4. Comprobar el estado del dispositivo Wi-Fi del teléfono inteligente	10	5	5/10=0.5
5. Escaneo y listado de puntos de acceso Wi-Fi	10	10	10/10=1
6. Actualizar la lista de puntos de acceso Wi-Fi	10	10	10/10=1
7. Conexión a un punto de acceso Wi-Fi	10	10	10/10=1
8. Explorador de archivos del teléfono inteligente	10	10	10/10=1
9. Seleccionar un archivo de presentación multimedia	10	10	10/10=1
10. Buscar otra presentación multimedia	10	10	10/10=1

11. Validar la extensión del archivo	10	10	10/10=1
12. Enviar la presentación multimedia	10	10	10/10=1
13. Listado del servidor de LibreOffice	10	10	10/10=1
14. Validar el emparejamiento y ejecución de la presentación	10	10	10/10=1
15. Visualizar las diapositivas en forma de maya	10	10	10/10=1
16. Botón "Parar presentación"	10	10	10/10=1

En la tabla 6.11 se muestra un análisis de resultados de los casos de prueba que se realizaron con el teléfono inteligente modelo Verykool. Como se observa en esta tabla, en uno de los casos de prueba existe un valor de referencia de 0.5 al obtener únicamente 5 pruebas funcionales de las 10 pruebas realizadas; lo que significa que este caso de prueba fue 50% funcional, en comparación con los demás casos de prueba los cuales fueron 100% funcionales.

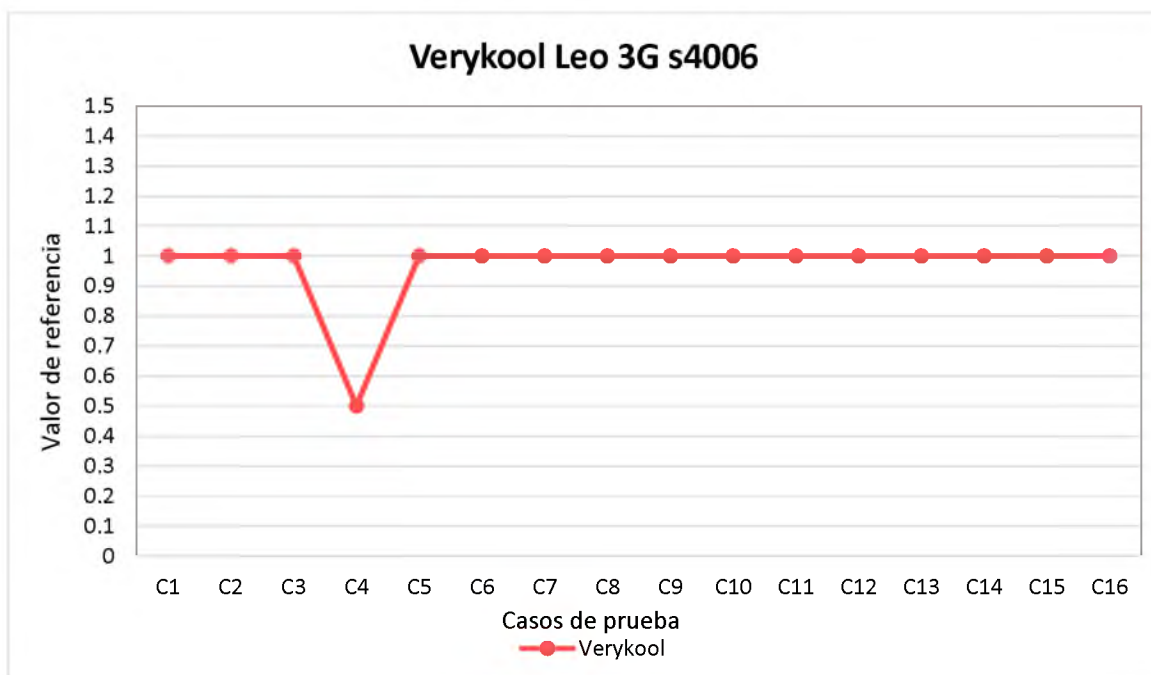


Figura 6.26: Gráfica de resultados del dispositivo Verykool.

En la figura 6.26 se muestra la gráfica de resultados del dispositivo Verykool con respecto a los casos de prueba realizados.

Tabla 6.12: Análisis de resultados del plan de pruebas con el dispositivo Motorola.

Pruebas con el modelo Motorola Moto G4			
Caso de prueba	Pruebas realizadas	Pruebas funcionales	Valor de referencia
1. Comprobar el estado del botón “ <b>BUSCAR RED WI-FI</b> ”	10	10	10/10=1
2. Comprobar el estado del botón “ <b>SELECCIONAR PRESENTACIÓN</b> ”	10	10	10/10=1
3. Comprobar el estado del botón “ <b>CONTROL REMOTO</b> ”	10	10	10/10=1
4. Comprobar el estado del dispositivo Wi-Fi del teléfono inteligente	10	5	5/10=0.5
5. Escaneo y listado de puntos de acceso Wi-Fi	10	10	10/10=1
6. Actualizar la lista de puntos de acceso Wi-Fi	10	10	10/10=1
7. Conexión a un punto de acceso Wi-Fi	10	10	10/10=1
8. Explorador de archivos del teléfono inteligente	10	10	10/10=1
9. Seleccionar un archivo de presentación multimedia	10	10	10/10=1
10. Buscar otra presentación multimedia	10	10	10/10=1
11. Validar la extensión del archivo	10	10	10/10=1
12. Enviar la presentación multimedia	10	10	10/10=1
13. Listado del servidor de LibreOffice	10	9	9/10=0.9
14. Validar el emparejamiento y ejecución de la presentación	10	10	10/10=1
15. Visualizar las diapositivas en forma de maya	10	10	10/10=1
16. Botón “Parar presentación”	10	10	10/10=1

En la tabla 6.12 se muestra un análisis de resultados de los casos de prueba que se realizaron con el teléfono inteligente modelo Motorola. En esta tabla se observa que de las 10 pruebas realizadas, en dos casos de prueba no se obtuvieron los valores esperados, para este caso, un valor de referencia igual a 1 el cual representa un caso de prueba 100% funcional.

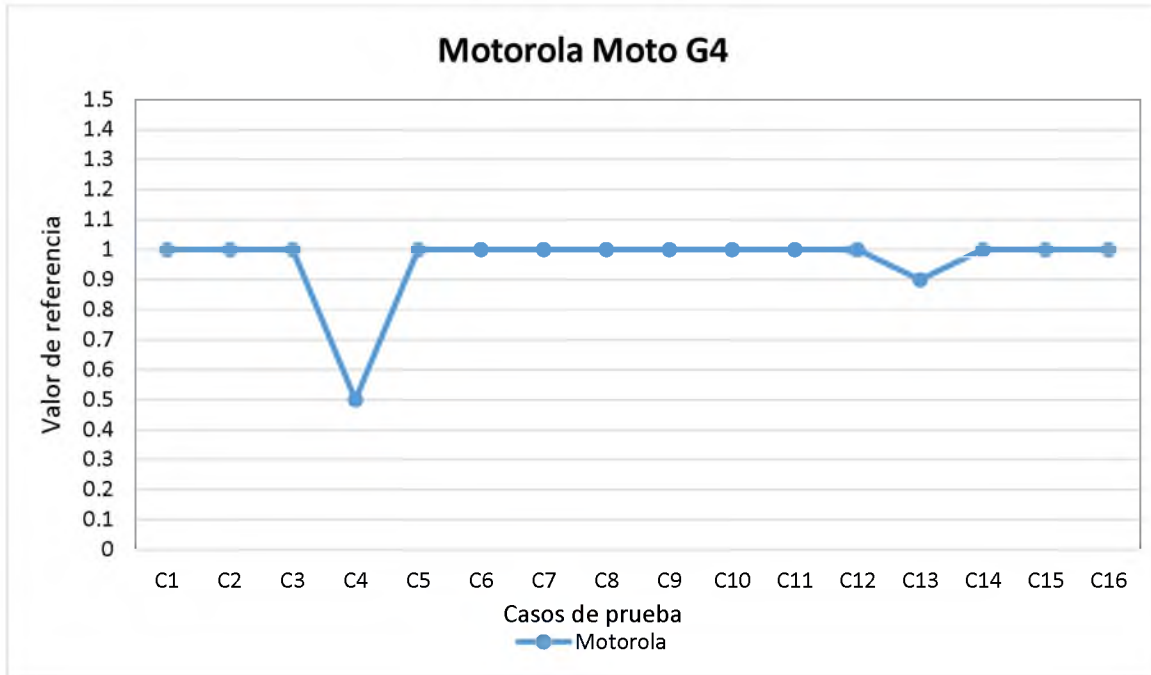


Figura 6.27: Gráfica de resultados del dispositivo Motorola.

En la figura 6.27 se muestra la gráfica de resultados del dispositivo Motorola con respecto a los casos de prueba realizados.

Tabla 6.13: Análisis de resultados del plan de pruebas con el dispositivo Huawei.

Pruebas con el modelo Huawei Y6 2018 (ATU-LX3)			
Caso de prueba	Pruebas realizadas	Pruebas funcionales	Valor de referencia
1. Comprobar el estado del botón “ <b>BUSCAR RED WI-FI</b> ”	10	10	10/10=1
2. Comprobar el estado del botón “ <b>SELECCIONAR PRESENTACIÓN</b> ”	10	10	10/10=1
3. Comprobar el estado del botón “ <b>CONTROL REMOTO</b> ”	10	10	10/10=1

4. Comprobar el estado del dispositivo Wi-Fi del teléfono inteligente	10	5	5/10=0.5
5. Escaneo y listado de puntos de acceso Wi-Fi	10	10	10/10=1
6. Actualizar la lista de puntos de acceso Wi-Fi	10	10	10/10=1
7. Conexión a un punto de acceso Wi-Fi	10	10	10/10=1
8. Explorador de archivos del teléfono inteligente	10	9	9/10=0.9
9. Seleccionar un archivo de presentación multimedia	10	5	5/10=0.5
10. Buscar otra presentación multimedia	10	10	10/10=1
11. Validar la extensión del archivo	10	10	10/10=1
12. Enviar la presentación multimedia	10	10	10/10=1
13. Listado del servidor de LibreOffice	10	10	10/10=1
14. Validar el emparejamiento y ejecución de la presentación	10	10	10/10=1
15. Visualizar las diapositivas en forma de maya	10	10	10/10=1
16. Botón "Parar presentación"	10	10	10/10=1

En la tabla 6.13 se muestra un análisis de resultados de los casos de prueba que se realizaron con el teléfono inteligente modelo Huawei. Como se observa, fueron menos las pruebas funcionales para este dispositivo, debido a la falta de activación de algunos permisos requeridos manualmente en este tipo de versiones de Android.

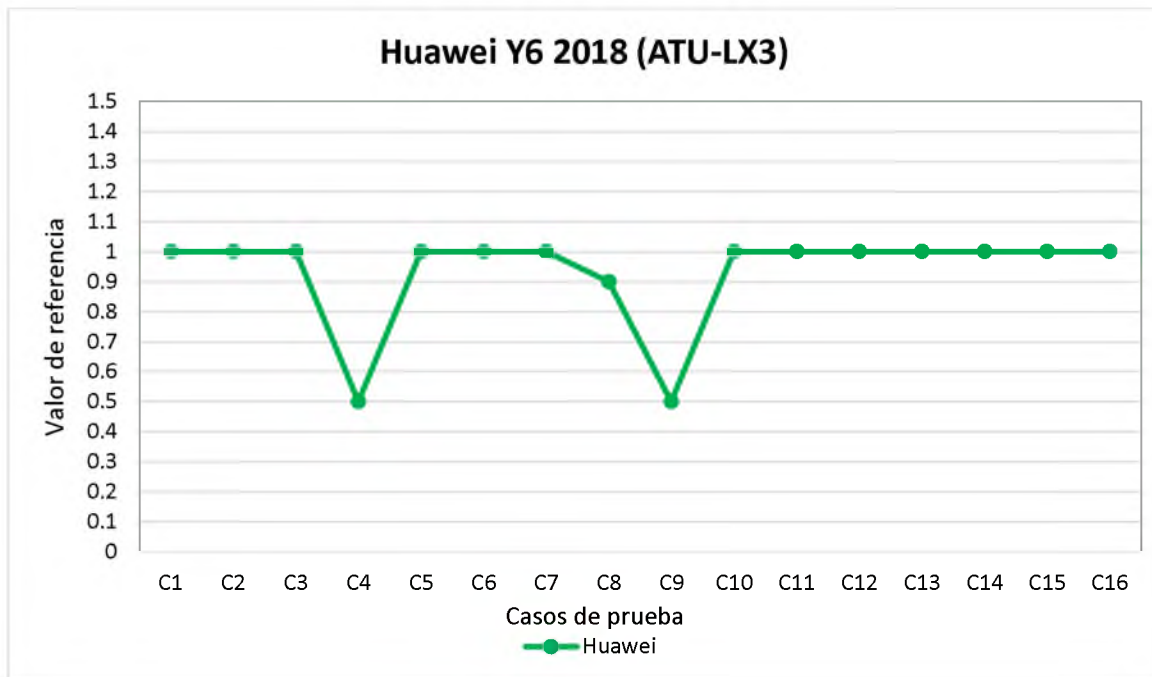


Figura 6.28: Gráfica de resultados del dispositivo Huawei.

En la figura 6.28 se muestra la gráfica de resultados del dispositivo Huawei con respecto a los casos de prueba realizados.

### 6.4.2 Resultados de los casos de prueba de la aplicación servidor

Los resultados obtenidos y registrados en esta sección son en base a las pruebas realizadas a la aplicación servidor utilizando una tarjeta Raspberry Pi.

Tabla 6.14: Análisis de resultados del plan de pruebas con la Raspberry Pi.

Pruebas con la Raspberry Pi			
Caso de prueba	Pruebas realizadas	Pruebas funcionales	Valor de referencia
1. Comprobar que la red hotspot Wi-Fi emite señal	30	30	30/30=1
2. Ejecución de la aplicación servidor	30	30	30/30=1

3. Selección del botón “Iniciar Servidor” de la aplicación servidor	30	30	30/30=1
4. Recibir un archivo de presentación multimedia	30	27	27/30=0.9
5. Selección del botón “Detener Servidor” de la aplicación servidor	30	30	30/30=1
6. Comprobar que el servidor de LibreOffice es iniciado	30	30	30/30=1
7. Emparejamiento	30	30	30/30=1

En la tabla 6.14 se muestra un análisis de resultados de los casos de prueba que se realizaron a la aplicación servidor, en esta tabla se incluyen las 30 pruebas realizadas; ya que se llevaron a cabo 10 pruebas por cada dispositivo móvil. Como se observa en esta tabla, un caso de prueba obtuvo únicamente 27 pruebas funcionales de las 30 pruebas realizadas, obteniendo un valor de referencia de 0.9, debido a que hubo un retraso en la recepción de un archivo por ciertos segundos.

En la figura 6.29 se muestra la gráfica de resultados de la tarjeta Raspberry Pi como servidor con respecto a los casos de prueba realizados.

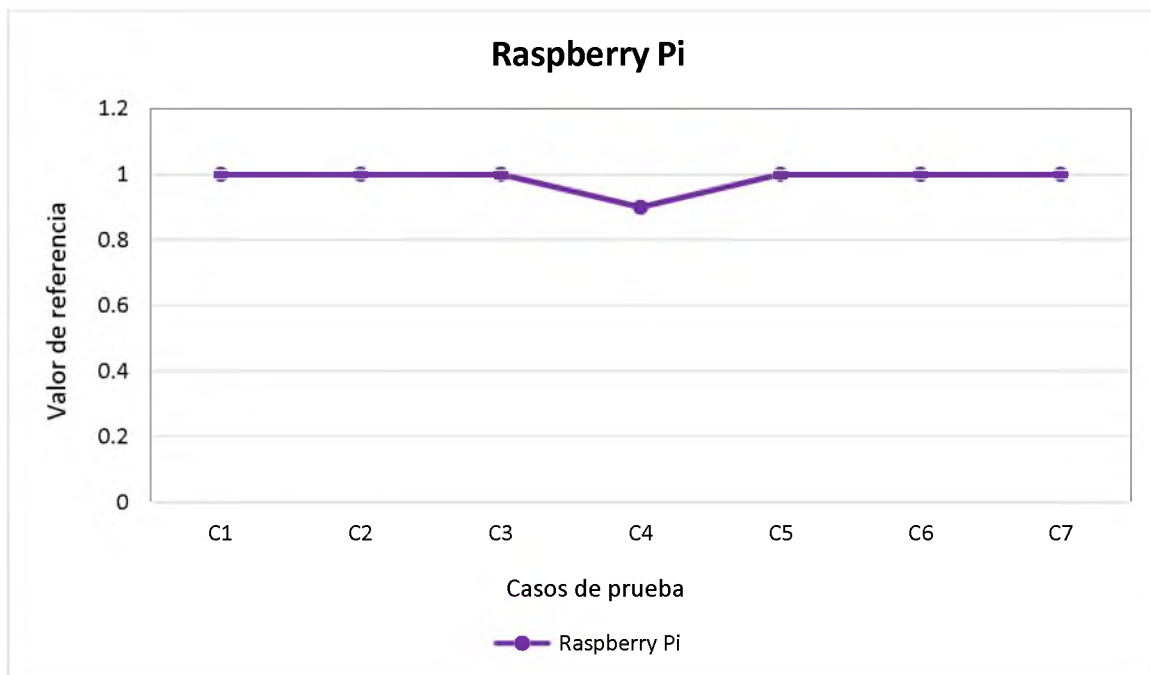


Figura 6.29: Gráfica de resultados de la Raspberry Pi.

## 6.5 Conclusiones

En este capítulo se realizaron los casos de prueba correspondientes a la aplicación cliente y a la aplicación servidor y la comparación de los resultados obtenidos.

Para la aplicación cliente se definieron 16 casos de prueba y para la aplicación servidor se definieron 7 casos de prueba. Las pruebas realizadas a la aplicación cliente se llevaron a cabo en tres teléfonos inteligentes de diferentes modelos y con diferentes versiones del sistema operativo Android, mientras que las pruebas realizadas a la aplicación servidor se llevaron a cabo en una Raspberry Pi.

En las gráficas mostradas anteriormente [ver figura 6.26, figura 6.27 y figura 6.28, respectivamente], se observa que las pruebas realizadas con cada dispositivo móvil tienen un resultado diferente. En la gráfica de la figura 6.26 correspondiente a los resultados del dispositivo Verykool, se observa que existe un mayor valor de referencia en sus casos de prueba; a diferencia de los resultados de los casos de prueba del dispositivo Motorola y Huawei. Esto se debe a que la App instalada en el dispositivo Verykool no requiere la activación de permisos para funcionar de forma correcta, debido a la versión del sistema operativo del dispositivo, en comparación con los dispositivos Motorola y Huawei con versiones más actuales de Android, ya que la App instalada en estos dispositivos móviles para funcionar adecuadamente requiere la activación de permisos de forma manual, para acceder a cierto tipo de información; por ejemplo, el acceso a la información de la conexión Wi-Fi para conocer si la señal Wi-Fi está activa y para ver las conexiones Wi-Fi disponibles; o el acceso a los directorios y archivos del teléfono al momento de seleccionar una presentación multimedia.

## Capítulo 7. Conclusiones y trabajos futuros

---

En este capítulo, se presentan las conclusiones obtenidas en base al proceso de investigación y a la metodología de solución llevados a cabo en este trabajo de tesis, y además, se listan los trabajos futuros.

### 7.1 Conclusiones

El motivo principal del desarrollo de este trabajo de tesis surge de la necesidad de solventar la movilidad del ponente al momento de exponer las presentaciones multimedia, además de aprovechar el gran potencial que tienen los teléfonos inteligentes, con los cuales, se busca omitir o eliminar la conexión del equipo de cómputo con los cables de energía y comunicación que son necesarios en el método tradicional para realizar una presentación multimedia con la ayuda de algún proyector.

El desarrollo de este sistema ofrece una solución práctica para el ponente al controlar remotamente las presentaciones multimedia a través de un teléfono inteligente, aportando una solución diferente al método tradicional utilizado actualmente, ya que, permite la movilidad del usuario al exponer, logrando solucionar la problemática mencionada y cumpliendo con el objetivo general de esta tesis.

Para llevar a cabo el desarrollo del trabajo de tesis, se utilizaron algunas tecnologías emergentes en dispositivos móviles como el desarrollo de una aplicación Android. Además, se utilizó una tarjeta Raspberry Pi con la configuración necesaria para utilizarla como un servidor de presentaciones multimedia, proporcionando con ello, características de un servidor de red y un entorno de mensajería basado en JSON con soporte de comunicación asíncrona mediante el uso de sockets.

### 7.2 Trabajos futuros

Para ampliar la investigación, se presenta a continuación una lista de trabajos futuros con la intención de mejorar la solución presentada en este trabajo de tesis:

- Utilizar la tecnología Bluetooth como alternativa de conexión.
- Proyectar tipos de archivos de acuerdo a la paquetería de LibreOffice.
- Implementar la aplicación en otra plataforma móvil como iOS.
- Gestionar la proyección de los dispositivos conectados, permitiendo proyectar y controlar contenido con más de dos dispositivos simultáneamente.

## Anexos

---

En esta sección se presentan los anexos relevantes para una mejor comprensión de algunos capítulos correspondientes al tema de tesis.

## Anexo A. Diseño de pruebas de la aplicación cliente

---

En este anexo se presenta la especificación del diseño de pruebas de la aplicación cliente del sistema "Raspdroid-Projection".

### A.1 Especificación del Diseño de Pruebas (APC-01-01)

#### A.1.1 Diseño de Prueba: APC-01-02.1

Aplicado a: **APC-02-01**

##### A.1.1.1 Características que se probarán

Se probará la función **APC-03-01** correspondiente a la ventana principal que forma parte de la aplicación cliente.

##### A.1.1.2 Refinamiento del enfoque

Se habilita el botón **“BUSCAR RED WI-FI”** de la ventana principal de la aplicación cliente, el cual permite obtener el SSID y la IP del punto de acceso Wi-Fi que emite la Raspberry Pi.

##### A.1.1.3 Identificación de la prueba

El caso de prueba asociado con esta especificación es: **APC-01-03.1** “Caso de prueba para comprobar el estado del botón **“BUSCAR RED WI-FI”**”.

##### A.1.1.4 Criterio de aprobación o desaprobación de evaluación de las características.

La evaluación de este caso de prueba se realiza a partir de la correcta ejecución de la aplicación.

#### A.1.2 Diseño de Prueba: APC-01-02.2

Aplicado a: **APC-02-01**

##### A.1.2.1 Características que se probarán

Se probará la función **APC-03-01** correspondiente a la ventana principal que forma parte de la aplicación cliente.

##### A.1.2.2 Refinamiento del enfoque

Se habilita el botón **“SELECCIONAR PRESENTACIÓN”** de la ventana principal de la aplicación cliente, el cual permite seleccionar un archivo con extensión pptx.

##### A.1.2.3 Identificación de la prueba

El caso de prueba asociado con esta especificación es: **APC-01-03.2** “Caso de prueba para comprobar el estado del botón **“SELECCIONAR PRESENTACIÓN”**”.

##### A.1.2.4 Criterio de aprobación o desaprobación de evaluación de las características.

La evaluación de este caso de prueba se realiza a partir de la conexión del teléfono inteligente a un punto de acceso Wi-Fi.

### A.1.3 Diseño de Prueba: APC-01-02.3

Aplicado a: **APC-02-01**

#### A.1.3.1 Características que se probarán

Se probará la función **APC-03-01** correspondiente a la ventana principal que forma parte de la aplicación cliente.

#### A.1.3.2 Refinamiento del enfoque

Se habilita el botón **“CONTROL REMOTO”** de la ventana principal de la aplicación cliente, el cual permite gestionar de forma remota un archivo de presentación multimedia.

#### A.1.3.3 Identificación de la prueba

El caso de prueba asociado con esta especificación es: **APC-01-03.3** “Caso de prueba para comprobar el estado del botón **“CONTROL REMOTO”**”.

#### A.1.3.4 Criterio de aprobación o desaprobación de evaluación de las características.

La evaluación de este caso de prueba se realiza a partir del envío exitoso de un archivo con extensión pptx a la aplicación servidor.

### A.1.4 Diseño de Prueba: APC-01-02.4

Aplicado a: **APC-02-02**

#### A.1.4.1 Características que se probarán

Se probará la función **APC-03-02** correspondiente al botón **“BUSCAR RED WI-FI”** de la ventana principal que forma parte de la aplicación cliente, para el escaneo de puntos de acceso Wi-Fi que son visibles en el teléfono inteligente.

#### A.1.4.2 Refinamiento del enfoque

Al seleccionar el botón **“BUSCAR RED WI-FI”** de la ventana principal de la aplicación cliente se activa el dispositivo Wi-Fi del teléfono inteligente en caso de estar desactivado.

#### A.1.4.3 Identificación de la prueba

El caso de prueba asociado con esta especificación es: **APC-01-03.4** “Caso de prueba para comprobar el estado del dispositivo Wi-Fi del teléfono inteligente”.

#### A.1.4.4 Criterio de aprobación o desaprobación de evaluación de las características.

La evaluación de este caso de prueba se realiza a partir de la activación del dispositivo Wi-Fi del teléfono inteligente, esto para asegurar el correcto escaneo de puntos de acceso Wi-Fi visibles.

### **A.1.5 Diseño de Prueba: APC-01-02.5**

Aplicado a: **APC-02-02**

#### **A.1.5.1 Características que se probarán**

Se probará la función **APC-03-02** correspondiente al botón “**BUSCAR RED WI-FI**” de la ventana principal que forma parte de la aplicación cliente, para el escaneo de puntos de acceso Wi-Fi que son visibles en el teléfono inteligente.

#### **A.1.5.2 Refinamiento del enfoque**

Si el dispositivo Wi-Fi del teléfono inteligente está activado, la aplicación realiza un escaneo y obtiene una lista de puntos de acceso Wi-Fi visibles desde el teléfono inteligente.

#### **A.1.5.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.5** “Caso de prueba para el escaneo y listado de puntos de acceso Wi-Fi”.

#### **A.1.5.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la obtención de la lista en la cual se muestra el SSID de cada punto de acceso Wi-Fi.

### **A.1.6 Diseño de Prueba: APC-01-02.6**

Aplicado a: **APC-02-02**

#### **A.1.6.1 Características que se probarán**

Se probará la función **APC-03-02** correspondiente al botón “**BUSCAR RED WI-FI**” de la ventana principal que forma parte de la aplicación cliente, para el escaneo de puntos de acceso Wi-Fi que son visibles en el teléfono inteligente.

#### **A.1.6.2 Refinamiento del enfoque**

Al seleccionar el botón “Actualizar” se actualiza la lista de puntos de acceso Wi-Fi visibles en la aplicación.

#### **A.1.6.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.6** “Caso de prueba para actualizar la lista de puntos de acceso Wi-Fi”.

### **A.1.6.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la actualización de la lista de puntos de acceso Wi-Fi visibles en la aplicación.

### **A.1.7 Diseño de Prueba: APC-01-02.7**

Aplicado a: **APC-02-02**

#### **A.1.7.1 Características que se probarán**

Se probará la función **APC-03-02** correspondiente al botón “**BUSCAR RED WI-FI**” de la ventana principal que forma parte de la aplicación cliente, para el escaneo de puntos de acceso Wi-Fi que son visibles en el teléfono inteligente.

#### **A.1.7.2 Refinamiento del enfoque**

Al seleccionar un punto de acceso Wi-Fi se debe ingresar su contraseña correctamente para conectarse de forma exitosa.

#### **A.1.7.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.7** “Caso de prueba para la conexión a un punto de acceso Wi-Fi.

### **A.1.7.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la conexión exitosa del teléfono inteligente a un punto de acceso Wi-Fi.

### **A.1.8 Diseño de Prueba: APC-01-02.8**

Aplicado a: **APC-02-03**

#### **A.1.8.1 Características que se probarán**

Se probará la función **APC-03-03** correspondiente al botón “**SELECCIONAR PRESENTACIÓN**” de la ventana principal que forma parte de la aplicación cliente, para seleccionar y enviar un archivo de presentación multimedia.

#### **A.1.8.2 Refinamiento del enfoque**

Al elegir el botón “**SELECCIONAR PRESENTACIÓN**” de la ventana principal de la aplicación cliente, se muestra el explorador de archivos del teléfono inteligente.

#### **A.1.8.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.8** “Caso de prueba para el explorador de archivos del teléfono inteligente”.

### **A.1.8.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la ejecución de la aplicación cliente en un teléfono inteligente con una versión de Android 4.4 como mínima y con un nivel de API 19 o superior.

### **A.1.9 Diseño de Prueba: APC-01-02.9**

Aplicado a: **APC-02-03**

#### **A.1.9.1 Características que se probarán**

Se probará la función **APC-03-03** correspondiente al botón “**SELECCIONAR PRESENTACIÓN**” de la ventana principal que forma parte de la aplicación cliente, para seleccionar y enviar un archivo de presentación multimedia.

#### **A.1.9.2 Refinamiento del enfoque**

Al seleccionar un archivo de presentación multimedia, se obtienen datos básicos sobre éste, los cuales son: el nombre, el tamaño en bytes y la extensión del archivo.

#### **A.1.9.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.9** “Caso de prueba para seleccionar un archivo de presentación multimedia”.

### **A.1.9.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la visualización de la información básica correspondiente a la elección de un archivo de presentación multimedia.

### **A.1.10 Diseño de Prueba: APC-01-02.10**

Aplicado a: **APC-02-03**

#### **A.1.10.1 Características que se probarán**

Se probará la función **APC-03-03** correspondiente al botón “**SELECCIONAR PRESENTACIÓN**” de la ventana principal que forma parte de la aplicación cliente, para seleccionar y enviar un archivo de presentación multimedia.

#### **A.1.10.2 Refinamiento del enfoque**

Al seleccionar el botón “**BUSCAR**” se mostrará nuevamente el explorador de archivos propio del teléfono inteligente.

#### **A.1.10.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-10.10** “Caso de prueba para buscar otra presentación multimedia”.

### **A.1.10.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la selección del botón “**BUSCAR**” que hace referencia a buscar otro archivo de presentación multimedia.

### **A.1.11 Diseño de Prueba: APC-01-02.11**

Aplicado a: **APC-02-03**

#### **A.1.11.1 Características que se probarán**

Se probará la función **APC-03-03** correspondiente al botón “**SELECCIONAR PRESENTACIÓN**” de la ventana principal que forma parte de la aplicación cliente, para seleccionar y enviar un archivo de presentación multimedia.

#### **A.1.11.2 Refinamiento del enfoque**

Al seleccionar un archivo del explorador de archivos del teléfono inteligente, éste debe tener la extensión pptx, lo cual permitirá su correcto envío a la aplicación servidor.

#### **A.1.11.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.11** “Caso de prueba para validar la extensión del archivo”.

### **A.1.11.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la selección de un archivo válido con la extensión pptx.

### **A.1.12 Diseño de Prueba: APC-01-02.12**

Aplicado a: **APC-02-03**

#### **A.1.12.1 Características que se probarán**

Se probará la función **APC-03-03** correspondiente al botón “**SELECCIONAR PRESENTACIÓN**” de la ventana principal que forma parte de la aplicación cliente, para seleccionar y enviar un archivo de presentación multimedia.

#### **A.1.12.2 Refinamiento del enfoque**

Al seleccionar el botón “**ENVIAR**” la presentación multimedia es enviada a la aplicación servidor, para posteriormente ser proyectada y contralada.

#### **A.1.12.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.12** “Caso de prueba para enviar la presentación multimedia”.

### **A.1.12.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir del correcto envío de la presentación multimedia a la aplicación servidor, habilitando así el botón “**CONTROL REMOTO**” de la ventana principal de la aplicación cliente asegurando el control de forma remota de la presentación enviada.

### **A.1.13 Diseño de Prueba: APC-01-02.13**

Aplicado a: **APC-02-04**

#### **A.1.13.1 Características que se probarán**

Se probará la función **APC-03-04** correspondiente al botón “**CONTROL REMOTO**” de la ventana principal que forma parte de la aplicación cliente, para controlar remotamente una presentación multimedia.

#### **A.1.13.2 Refinamiento del enfoque**

Al seleccionar el botón “**CONTROL REMOTO**”, se muestra el nombre del servidor de LibreOffice en la ventana principal del módulo control remoto.

#### **A.1.13.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.13** “Caso de prueba para el listado del servidor de LibreOffice”.

### **A.1.13.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la visualización del nombre del servidor de LibreOffice “raspberrypi” en la lista generada.

### **A.1.14 Diseño de Prueba: APC-01-02.14**

Aplicado a: **APC-02-04**

#### **A.1.14.1 Características que se probarán**

Se probará la función **APC-03-04** correspondiente al botón “**CONTROL REMOTO**” de la ventana principal que forma parte de la aplicación cliente, para controlar remotamente una presentación multimedia.

#### **A.1.14.2 Refinamiento del enfoque**

Al seleccionar el nombre del servidor de LibreOffice “raspberrypi”, se muestra la ventana de conexión que muestra un código de cuatro dígitos generado de forma aleatoria por la propia aplicación, el cual se ingresa en la ventana de conexión de la aplicación servidor para llevar a cabo el emparejamiento del el teléfono inteligente con el servidor de LibreOffice, logrando así, la ejecución de la presentación multimedia.

### **A.1.14.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.14** “Caso de prueba para validar el emparejamiento y ejecución de la presentación”.

### **A.1.14.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la ejecución de LibreOffice Impress en la Raspberry Pi.

### **A.1.15 Diseño de Prueba: APC-01-02.15**

Aplicado a: **APC-02-04**

#### **A.1.15.1 Características que se probarán**

Se probará la función **APC-03-04** correspondiente al botón “**CONTROL REMOTO**” de la ventana principal que forma parte de la aplicación cliente, para controlar remotamente una presentación multimedia.

#### **A.1.15.2 Refinamiento del enfoque**

Si la presentación multimedia es ejecutada, el usuario tiene la opción de ver las diapositivas de la presentación multimedia en forma de maya (GridView).

#### **A.1.15.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.15** “Caso de prueba para visualizar las diapositivas en forma de maya”.

#### **A.1.15.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir del emparejamiento del teléfono inteligente con el servidor de LibreOffice y de la ejecución de la presentación multimedia.

### **A.1.16 Diseño de Prueba: APC-01-02.16**

Aplicado a: **APC-02-04**

#### **A.1.16.1 Características que se probarán**

Se probará la función **APC-03-04** correspondiente al botón **“CONTROL REMOTO”** de la ventana principal que forma parte de la aplicación cliente, para controlar remotamente una presentación multimedia.

### **A.1.16.2 Refinamiento del enfoque**

Al seleccionar el botón “Parar presentación”, se detiene la ejecución de la presentación multimedia.

### **A.1.16.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APC-01-03.16** “Caso de prueba para el botón “Parar presentación”.

### **A.1.16.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la ejecución de la presentación multimedia o de la visualización de las diapositivas en forma de maya.

## Anexo B. Diseño de pruebas de la aplicación servidor

---

En este anexo se presenta la especificación del diseño de pruebas de la aplicación servidor del sistema "Raspdroid-Projection".

## B.1 Especificación del Diseño de Pruebas (APS-01-01)

### B.1.1 Diseño de Prueba: APS-01-02.1

Aplicado a: *APS-02-01*

#### B.1.1.1 Características que se probarán

Se probará la función *APS-03-01* correspondiente a la configuración de la red hotspot para permitir a los teléfonos inteligentes conectarse al punto de acceso Wi-Fi.

#### B.1.1.2 Refinamiento del enfoque

La red hotspot Wi-Fi emite señal de manera automática al iniciar el servidor Raspberry Pi.

#### B.1.1.3 Identificación de la prueba

El caso de prueba asociado con esta especificación es: *APS-01-03.1* "Caso de prueba para comprobar que la red hotspot Wi-Fi emite señal".

#### B.1.1.4 Criterio de aprobación o desaprobación de evaluación de las características.

La evaluación de este caso de prueba se realiza a partir de que la red hotspot Wi-Fi RBPI creada se emita correctamente para ser visible.

### B.1.2 Diseño de Prueba: APS-01-02.2

Aplicado a: *APS-02-02*

#### B.1.2.1 Características que se probarán

Se probará la función *APS-03-02* correspondiente a la interfaz principal de la aplicación servidor.

#### B.1.2.2 Refinamiento del enfoque

Cuando la aplicación servidor es ejecutada, se muestra la interfaz principal de la aplicación con los botones funcionales correspondientes.

#### B.1.2.3 Identificación de la prueba

El caso de prueba asociado con esta especificación es: *APS-01-03.2* "Caso de prueba para la ejecución de la aplicación servidor".

#### B.1.2.4 Criterio de aprobación o desaprobación de evaluación de las características.

La evaluación de este caso de prueba se realiza a partir de la correcta ejecución de la aplicación servidor en el servidor Raspberry Pi.

**B.1.3 Diseño de Prueba: APS-01-02.3**

Aplicado a: **APS-02-02**

**B.1.3.1 Características que se probarán**

Se probará la función **APS-03-02** correspondiente a la interfaz principal de la aplicación servidor.

**B.1.3.2 Refinamiento del enfoque**

Al seleccionar el botón “Iniciar Servidor” de la interfaz principal, se ejecutan varios procesos (multitarea) en la aplicación servidor, esperando conexiones entrantes de clientes para recibir archivos de presentaciones multimedia en un directorio definido en la Raspberry Pi.

**B.1.3.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APS-01-03.3** “Caso de prueba para la selección del botón “Iniciar Servidor” de la aplicación servidor”.

**B.1.3.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la ejecución correcta de los procesos (multitarea) en la aplicación servidor en la Raspberry Pi.

**B.1.4 Diseño de Prueba: APS-01-02.4**

Aplicado a: **APS-02-02**

**B.1.4.1 Características que se probarán**

Se probará la función **APS-03-02** correspondiente a la interfaz principal de la aplicación servidor.

**B.1.4.2 Refinamiento del enfoque**

Al recibir el archivo de presentación multimedia, éste se guarda en el directorio “/home/pi/Desktop/presentacion” definido en el servidor Raspberry Pi.

**B.1.4.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APS-01-03.4** “Caso de prueba para recibir un archivo de presentación multimedia”.

**B.1.4.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza al recibir y guardar de forma exitosa el archivo de presentación multimedia.

### **B.1.5 Diseño de Prueba: APS-01-02.5**

Aplicado a: **APS-02-02**

#### **B.1.5.1 Características que se probarán**

Se probará la función **APS-03-02** correspondiente a la interfaz principal de la aplicación servidor.

#### **B.1.5.2 Refinamiento del enfoque**

Al seleccionar el botón “Detener” de la interfaz de la aplicación servidor, se cierra la comunicación con la aplicación cliente, por lo tanto, los clientes conectados al punto de acceso RBPI no podrán realizar intentos de conexión hasta que se vuelva a iniciar la aplicación servidor.

#### **B.1.5.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APS-01-03.5** “Caso de prueba para la selección del botón “Detener Servidor” de la aplicación servidor”.

#### **B.1.5.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de que la aplicación servidor cierre la comunicación con el cliente de forma correcta.

### **B.1.6 Diseño de Prueba: APS-01-02.6**

Aplicado a: **APS-02-03**

#### **B.1.6.1 Características que se probarán**

Se probará la función **APS-03-03** correspondiente al servidor de LibreOffice para abrir y controlar archivos de presentaciones multimedia.

#### **B.1.6.2 Refinamiento del enfoque**

Los archivos de presentación multimedia recibidos en el directorio definido en el servidor Raspberry Pi son abiertos con el servidor de LibreOffice Impress para ser controlados.

#### **B.1.6.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APS-01-03.6** “Caso de prueba para comprobar que el servidor de LibreOffice es iniciado”.

#### **B.1.6.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la ejecución de LibreOffice Impress en el servidor Raspberry Pi.

**B.1.7 Diseño de Prueba: APS-01-02.7**

Aplicado a: **APS-02-03**

**B.1.7.1 Características que se probarán**

Se probará la función **APS-03-03** correspondiente al servidor de LibreOffice para abrir y controlar archivos de presentaciones multimedia.

**B.1.7.2 Refinamiento del enfoque**

Se establece la comunicación entre el teléfono inteligente y el servidor de LibreOffice para ejecutar y controlar un archivo de presentación multimedia.

**B.1.7.3 Identificación de la prueba**

El caso de prueba asociado con esta especificación es: **APS-01-03.7** "Caso de prueba para el emparejamiento".

**B.1.7.4 Criterio de aprobación o desaprobación de evaluación de las características.**

La evaluación de este caso de prueba se realiza a partir de la conexión y el emparejamiento entre el teléfono inteligente y el servidor de LibreOffice, a través de un código correcto al momento del emparejamiento.

## 8. Bibliografía

- [1] H. Molina, "Nuevas Tecnologías inalámbricas en la Proyección de contenidos en las aulas mediante la utilización de dispositivos móviles," *Historia y Comunicación Social*, vol. 19, no. Esp., pp. 261-272, Feb. 2014.
- [2] K. Malave, y J. L. Beauperthuy, ""ANDROID" GOOGLE'S OPERATING SYSTEM FOR MOBILE DEVICES," *Negotium*, vol. 7, no. 19, pp. 79–96, Jul. 2011.
- [3] What is a Raspberry Pi?. Raspberry Pi, 2018. [En línea]. Disponible: <https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/>. Accedido: Mayo 1, 2018.
- [4] Uso de los componentes físicos de la computadora. Cca.org.mx, 2018. [En línea]. Disponible: <http://cca.org.mx/cca/cursos/cucfc/modulo4/tema5-03.html>. Accedido: Ag. 1, 2018.
- [5] J. Rodríguez, "Metodología para la Implementación de Interfaces sobre Dispositivos Multisensoriales Aplicada a Sistemas de Recomendación Sensibles al Contexto," tesis M.C., cenidet@, Cuernavaca, Morelos, 2014.
- [6] Paneles de control | Desarrolladores de Android | Android Developers. Android Developers, 2018. [En línea]. Disponible: <https://developer.android.com/about/dashboards?hl=es-419>. Accedido: Octubre 5, 2019.
- [7] FrontPage – Raspbian. Raspbian.org, 2018. [En línea]. Disponible: <http://www.raspbian.org/>. Accedido: Mayo 1, 2018.
- [8] Conoce Android Studio | Android Developers. Android Developers, 2018. [En línea]. Disponible: <https://developer.android.com/studio/intro/?hl=es-419>. Accedido: Julio 1, 2018.
- [9] Bienvenido a NetBeans y [www.netbeans.org](http://www.netbeans.org), Portal del IDE Java de Código Abierto. Netbeans.org, 2018. [En línea]. Disponible: [https://netbeans.org/index\\_es.html](https://netbeans.org/index_es.html). Accedido: Junio 1, 2018.
- [10] Android SDK | Android Developers. Stuff.mit.edu, 2018. [En línea]. Disponible: <https://stuff.mit.edu/afs/sipb/project/android/docs/sdk/index.html>. Accedido: Julio 1, 2018.
- [11] H. Suhas, y K. Mahima, "ANDROID BASED MOBILE APPLICATION DEVELOPMENT and its SECURITY," *International Journal of Computer Trends and Technology*, vol. 3, no. 19, pp. 486–490, 2012.

- [12] Shubhankar Mukherjee et al, "Android Application Development & Its Security," *International Journal of Computer Science and Mobile Computing*, vol. 4, no. 3, pp. 714–719, Mar. 2015.
- [13] IBM Knowledge Center – Kit de desarrolladores de software (SDK) de IBM. Ibm.com, 2018. [En línea]. Disponible: [https://www.ibm.com/support/knowledgecenter/es/SSYKE2\\_8.0.0/com.ibm.java.80.doc/user/java\\_sdk.html](https://www.ibm.com/support/knowledgecenter/es/SSYKE2_8.0.0/com.ibm.java.80.doc/user/java_sdk.html). Accedido: Julio 1, 2018.
- [14] LibreOffice 6.0 SDK – Examples. Api.libreoffice.org, 2018. [En línea]. Disponible: <https://api.libreoffice.org/examples/examples.html>. Accedido: Julio 1, 2018.
- [15] Java, t., Conceptos básicos del lenguaje Java. Ibm.com, 2018. [En línea]. Disponible: <https://www.ibm.com/developerworks/ssa/java/tutorials/j-introtojava1/index.html>. Accedido: Agosto 1, 2018.
- [16] Conozca más sobre la tecnología Java. Java.com, 2018. [En línea]. Disponible: <https://www.java.com/es/about/>. Accedido: Agosto 1, 2018.
- [17] M. Jiménez, "IMPLANTACION DEL USO DE LAS PRESENTACIONES MULTIMEDIA COMO ESTRATEGIA PARA LA ENSEÑANZA DEL TIEMPO PASADO EN EL IDIOMA INGLES EN EL BACHILLERATO PRESENCIAL," *VIDA CIENTÍFICA Boletín Científico de la Preparatoria no.4*, vol. 2, no. 4, pp. 79–96, Jul. 2014.
- [18] Formatos compatibles de presentaciones electrónicas – oficina online. Helpcenter.onlyoffice.com, 2018. [En línea]. Disponible: <https://helpcenter.onlyoffice.com/es/ONLYOFFICE-Editors/ONLYOFFICE-Presentation-Editor/HelpfulHints/SupportedFormats.aspx>. Accedido: Sep. 1, 2018.
- [19] ¿Qué es LibreOffice? | LibreOffice en español - el paquete de oficina por excelencia. Es.libreoffice.org, 2018. [En línea]. Disponible: <https://es.libreoffice.org/descubre/libreoffice/>. Accedido: Julio 1, 2018.
- [20] Impress | LibreOffice en español - el paquete de oficina por excelencia. Es.libreoffice.org, 2018. [En línea]. Disponible: <https://es.libreoffice.org/descubre/impress/>. Accedido: Julio 1, 2018.
- [21] What Is a Socket? (The Java™ Tutorials > Custom Networking > All About Sockets). Docs.oracle.com, 2018. [En línea]. Disponible: <https://docs.oracle.com/javase/tutorial/networking/sockets/definition.html>. Accedido: Mayo 1, 2018.
- [22] JSON. Json.org, 2018. [En línea]. Disponible: <https://www.json.org/json-es.html>. Accedido: Agosto 1, 2018.

- [23] What is a Hotspot?. Intel, 2018. [En línea]. Disponible: <https://www.intel.com/content/www/us/en/tech-tips-and-tricks/what-is-a-hotspot.html>. Accedido: Agosto 1, 2018.
- [24] Aplicación iProjection | Epson México. Epson.com.mx, 2018. [En línea]. Disponible: <https://epson.com.mx/wireless-projector-app>. Accedido: Mayo 1, 2018.
- [25] Proyecta desde tu iPad / iPhone con Epson iProjection. Epson.es, 2018. [En línea]. Disponible: <https://www.epson.es/viewcon/corporatesite/cms/index/10140>. Accedido: Mayo 1, 2018.
- [26] Proyección inalámbrica mediante iOS/Android - Visual System Solutions | Panasonic Business. Business.panasonic.es, 2018. [En línea]. Disponible: <https://business.panasonic.es/sistemas-visuales/productos-y-accesorios/tecnologia-visual/iOS-Android>. Accedido: Mayo 1, 2018.
- [27] R. CEC, PROYECCIÓN SIN CABLES. Revista CEC, 2018. [En línea]. Disponible: <http://www.revistacec.com/1421-proyeccion-sin-cables-1421.html>. Accedido: Mayo 1, 2018.
- [28] J. R. Gutiérrez, "Servicio de proyección de material docente basado en Raspberry Pi," trabajo fin de grado, Estudios de Grado en Ing. de Tecnologías de Telecomunicación, Univ. de la Granada, Granada, 2014.
- [29] B. Fernández, "Control remoto de presentaciones mediante Raspberry PI y Android," Proyecto Fin de Carrera, Esc. Tec. Superior de Ing. y Sistemas de Telecomunicación, Univ. Politécnica Madrid, Madrid, 2015.
- [30] What is a Wireless Presentation System?, Black Box, 2018. [En línea]. Disponible: <https://www.blackbox.be/en-be/page/40742/Resources/Technical-Resources/Black-Box-Explains/multimedia/What-is-a-Wireless-Presentation-System>. Accedido: Agosto 1, 2018.
- [31] IEEE Computer Society, "IEEE Standard for Software and System Test Documentation," IEEE Std 829™-2008, New York, NY 10016-5997, USA, Jul. 2008.