



UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

PLATAFORMA DE ANÁLISIS DE CALIDAD DE SERVICIO EN REDES WIFI

MEMORIA PARA OPTAR AL TÍTULO DE  
INGENIERA CIVIL ELÉCTRICA

CATALINA MARES GONZÁLEZ INOSTROZA

PROFESOR GUÍA:  
JORGE SANDOVAL ARENAS

MIEMBROS DE LA COMISIÓN:  
CESAR AZURDIA MEZA  
MARCELO MARÍN SOTO

SANTIAGO DE CHILE  
2021

RESUMEN DE LA MEMORIA PARA OPTAR  
AL TÍTULO DE INGENIERA CIVIL ELÉCTRICA  
POR: CATALINA MARES GONZÁLEZ INOSTROZA  
FECHA: 2021  
PROF. GUÍA: JORGE SANDOVAL ARENAS

## PLATAFORMA DE ANÁLISIS DE CALIDAD DE SERVICIO EN REDES WIFI

Debido al desarrollo de las tecnologías de la información y la comunicación, y que las aplicaciones de uso diario han evolucionado a procesos de comunicación complejos, los reclamos hacia las empresas de telecomunicaciones han aumentado mucho, principalmente por problemas de conectividad de sus redes WiFi, lo cual se traduce en incumplimientos en el contrato. De acuerdo a esto, nace la necesidad de crear un método de fiscalización de la calidad de servicio ofrecida por los proveedores de servicios de internet, el cual pueda ser manejado por un usuario promedio que desconoce del tema.

La solución planteada busca brindar apoyo a los clientes, para que estos tengan a su disposición la información necesaria para reconocer la calidad de servicio recibida. A partir de esto, se crea una aplicación para android que recopila información de la red WiFi con la finalidad de analizar estos datos y mostrar la información generada de forma sencilla para el usuario. Para esto se crea un proyecto en “android studio”, donde las clases y funciones están en java y son la base del proceso de obtención y análisis de los datos. Luego de esto, se procede a almacenar los datos con el gestor de base de datos “Realm” y a mostrar el análisis generado de forma comprensible en la aplicación.

El trabajo desarrollado es probado en un edificio de santiago centro, donde se comprueba que el análisis es verosímil de acuerdo a la teoría, la cual indica que la señal y el throughput máximo empeora a medida que el usuario se aleja del router y si aumenta la interferencia. Con este análisis se busca informar al usuario si las personas conectadas a la red WiFi pueden utilizar las aplicaciones que desean, al mismo tiempo, en la posición que se encuentran.



*Dedicado a mi madre, gracias por tu apoyo incondicional.*



# Agradecimientos

Para comenzar, quiero agradecer a mi madre por todo su apoyo, preocupación, consejos, sacrificios y, sobre todo, por todo el amor que me ha dado todos estos años. Todo lo que he logrado hasta ahora ha sido gracias a ti, porque me has enseñado que soy capaz de hacer todo lo que me proponga. Gracias ha ti me convertí en la gran mujer que soy hoy. También quiero agradecer a mi abuela, por apoyarme y amarme incondicionalmente, y por haber formado parte de mi crecimiento con su linda compañía y enseñanzas.

Quiero agradecer a Jhon, mi pareja, por todo el amor y confianza que me ha dado. Gracias por ser mi amigo y compañero de vida, gracias por apoyarme y por darme ánimo siempre que lo necesito, sobre todo este año tan complicado. Espero que se vengán muchas aventuras juntos los próximos años.

Gracias al profesor Jorge Sandoval, por su dedicación y apoyo en este proceso tan importante para mí. Fue un gusto trabajar junto a usted y espero que todo le salga muy bien en este año lleno de nuevas metas.

Por último, quiero agradecer a todos los equipos en los que he estado, futbolito FCFM, futsal U. Chile, Zenteno, LCBF y a todas las personas maravillosas que he conocido ahí, porque ustedes han alegrado mis días y me han hecho crecer como persona. Gracias a este deporte conocí a mi mejor amiga Pauli, gracias por todo tu apoyo, cariño y por confiar en mí.



# TABLA DE CONTENIDO

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Descripción del Problema . . . . .	2
1.3. Objetivo General . . . . .	2
1.4. Objetivos Específicos . . . . .	3
1.5. Metodología . . . . .	3
<b>2. Antecedentes</b>	<b>5</b>
2.1. Redes Fijas . . . . .	5
2.1.1. Alcance de las Redes . . . . .	5
2.1.2. Componentes Básicos . . . . .	6
2.1.3. Tecnologías de Redes LAN . . . . .	7
2.1.4. Estado Actual de Redes Fijas en Chile . . . . .	7
2.2. Servicios de Internet . . . . .	8
2.2.1. Proveedores de Servicios de Internet en Chile . . . . .	9
2.2.2. Efectos del Covid-19 en Servicios de Internet . . . . .	10
2.2.3. Ley 21.046 . . . . .	11
2.3. Calidad de Servicio . . . . .	12
2.3.1. Servicios Integrados: IntServ . . . . .	13
2.3.2. Servicios Diferenciados: DiffServ . . . . .	15
2.4. Estándar IEEE 802.11 . . . . .	16
2.4.1. Arquitectura . . . . .	16
2.4.2. Funcionamiento en Capa Física . . . . .	17
2.4.3. IEEE 802.11 MAC - Nivel de Enlace . . . . .	19
2.4.4. Estándar 802.11e . . . . .	20
<b>3. Metodología</b>	<b>22</b>
3.1. Descripción Básica del Sistema . . . . .	22
3.2. Obtención de los Datos . . . . .	24
3.3. Estimación de Calidad . . . . .	24
<b>4. Descripción de la Plataforma</b>	<b>26</b>
4.1. Estructura del Proyecto . . . . .	26
4.2. Flujo de Navegación . . . . .	28
4.3. Obtención de Datos . . . . .	32
4.4. Conexión con la Base de Datos . . . . .	38

4.5. Análisis de Calidad de Servicio . . . . .	41
<b>5. Funcionamiento y Análisis</b>	<b>51</b>
5.1. Resultados y Análisis . . . . .	51
<b>6. Conclusiones</b>	<b>56</b>
6.1. Continuación del Proyecto . . . . .	56
<b>Bibliografía</b>	<b>58</b>
<b>A. Índice de Siglas</b>	<b>61</b>

# Índice de Tablas

2.1. Familia de protocolos 802.11 . . . . .	17
4.1. Valores de throughput máximo escogidos a partir de la figura 4.16. . . . .	45
4.2. Valores referenciales de throughput mínimo requeridos por las aplicaciones. .	49
5.1. Resultados de la aplicación en la mañana. . . . .	52
5.2. Detalle del escáner de las redes WiFi alrededor en la mañana en la posición A.	53
5.3. Resultados de la aplicación en la tarde. . . . .	53
5.4. Detalles de los resultados de un usuario en la posición C. . . . .	54
5.5. Detalle del escáner de las redes WiFi alrededor en la mañana en la posición C.	55

# Índice de Ilustraciones

2.1. Conexiones fijas totales . . . . .	8
2.2. Líneas de banda ancha fija por tecnología . . . . .	8
2.3. Número de servicios por habitantes. . . . .	9
2.4. Total de servicios entregados. . . . .	9
2.5. Conexiones de internet fijo . . . . .	10
2.6. Tráfico total de aplicaciones globales . . . . .	11
2.7. Esquema de un nodo IntServ. . . . .	14
2.8. Dominio DiffServ . . . . .	15
2.9. Arquitectura del estándar 802.11 . . . . .	18
2.10. Diferencia entre OFDM y OFDMA . . . . .	18
2.11. Separación de los canales en la banda de 2.4 GHz . . . . .	19
2.12. Modelo de funcionamiento de DCF . . . . .	20
2.13. Modelo de funcionamiento de PCF . . . . .	20
2.14. Categorías de acceso según prioridad del usuario . . . . .	21
3.1. Medición pasiva en la red . . . . .	23
3.2. Diagrama de las condiciones de funcionamiento de la aplicación . . . . .	23
3.3. Distribución de las versiones de android . . . . .	25
4.1. Estructura general de las carpetas del proyecto . . . . .	26
4.2. Flujo de la primera actividad o pantalla. . . . .	29
4.3. Flujo de la segunda actividad o pantalla. . . . .	31
4.4. Flujo de la tercera actividad o pantalla. . . . .	32
4.5. Flujo de la cuarta actividad o pantalla. . . . .	32
4.6. Mensaje al usuario sobre su estado de conectividad . . . . .	33
4.7. Mensaje al usuario sobre su conexión WiFi . . . . .	34
4.8. Diagrama general del proceso de validación . . . . .	34
4.9. Solicitudes de permiso para acceder a información del dispositivo. . . . .	35
4.10. Proceso de redirigir la aplicación a configuraciones. . . . .	36
4.11. Diagrama general del proceso de medición . . . . .	37
4.12. Diagrama general del proceso de obtención de datos . . . . .	38
4.13. Tabla de datos de WiFi . . . . .	40
4.14. Descripción gráfica del análisis . . . . .	42
4.15. Relación entre throughput y RSSI . . . . .	43
4.16. Variación del throughput para 802.11ac . . . . .	43
4.17. Uso de circuitos integrados de estándares 802.11 . . . . .	44

4.18. Máximo valor de throughput alcanzado según ancho de banda en 802.11ac. . . . .	45
4.19. Lógica de la función getMaxThroughput. . . . .	46
4.20. Señales sin interferencia . . . . .	47
4.21. Interferencia completa entre señales . . . . .	47
4.22. Interferencia parcial entre señales . . . . .	48
5.1. Estructura del hogar donde se realizan las pruebas . . . . .	52



# Capítulo 1

## Introducción

### 1.1. Motivación

El desarrollo de las tecnologías de la información y la comunicación (TICs) <sup>1</sup> ha provocado profundos cambios, no sólo en el estilo de vida de las personas, sino que también en los proveedores de servicios de internet (ISPs, por sus siglas en inglés), pues estos han tenido que ajustarse a las necesidades de comunicación en cuando a su variedad y complejidad. Actualmente, la penetración de tecnologías existentes en la vida cotidiana es muy alta, por ejemplo, en Chile se estima que se cuentan con 126,16 abonados móviles cada 100 habitantes, cuyo dato es obtenido para septiembre del 2020 a partir de las estadísticas de la subsecretaría de telecomunicaciones (SUBTEL). Además, el acceso a internet es considerado prácticamente una necesidad, por lo que cada vez más personas tienen acceso a él. En el caso de Chile, la SUBTEL señala que se tiene una penetración de internet fijo de 18,94 por cada 100 habitantes, lo cual se encuentra en continuo crecimiento. No obstante, se debe tener en cuenta que este último valor es determinado a partir de la cantidad de servicios contratados en relación al número total de habitantes, por lo que si se considera que una familia promedio se compone de cuatro personas, esta cifra se puede representar como una penetración de 75,76 cada 100 familias.

El acceso a tecnologías y a internet no son los únicos progresos que se han presentado este último tiempo, igualmente las plataformas y aplicaciones de uso diario han evolucionado rápidamente a procesos de comunicación complejos, como lo son las videoconferencias, servicios de televisión por protocolo de internet (IPTV, por sus siglas en inglés) o servicios de libre transmisión (OTT, por sus siglas en inglés). Estas aplicaciones necesitan grandes requisitos en cuando a ancho de banda, retardo, throughput y cantidad de datos transmitidos, provocando una necesidad de mejoras tecnológicas en las redes de acceso y de protocolos en la calidad del servicio ofrecido al cliente.

---

<sup>1</sup>Las siglas pueden ser observadas en el Apéndice A. Índice de Siglas.

## 1.2. Descripción del Problema

La calidad de servicio es un factor importante para las empresas, pues ésta marca una gran diferencia para el usuario al momento de elegir a un proveedor de servicios de internet y al exigir sus derechos según lo contratado. Por esta razón, las ISPs deberían asegurar al cliente un servicio otorgado igual al contratado y, además, transparentar los problemas que puedan tener algunos componentes. Para esto, actualmente se está implementando la ley 21,046, la cual establece la obligación de una velocidad mínima garantizada de acceso a internet mediante un sistema de medición de calidad, donde se analice que lo contratado por los usuarios sea efectivamente lo que reciben en sus dispositivos. A pesar de esto, el usuario no está siendo informado directamente sobre el cumplimiento o incumplimiento de su contrato y, aunque esta información se le haga llegar al usuario, éste no necesariamente es capaz de reconocer o comprender las fallas técnicas o las posibles soluciones al problema. Sin embargo, la mayoría de las personas sí son capaces de percibir ciertos indicadores de calidad, como la velocidad o la conectividad, por lo que es importante encontrar la manera de mantener al cliente informado.

Actualmente, a pesar de que existan normas que se deban cumplir, es normal escuchar reclamos hacia las empresas de telecomunicaciones, ya sea por un servicio defectuoso, errores de facturación o incumplimiento de las condiciones del contrato. Esto ocurre debido a que no existen métodos de continua fiscalización de las redes de las distintas compañías, o bien, son tantos los clientes de éstas que un análisis global no detecta problemas de un usuario en específico. Por lo tanto, sin que alguien fiscalice continuamente la calidad de servicio ofrecida, las empresas tienen cierta libertad en su operación, provocando que algunos clientes presenten inconvenientes debido a errores en la red. Un ejemplo de lo anterior es lo acontecido con las ISPs en abril del 2020, en plena crisis sanitaria, donde la SUBTEL indica un exceso en las tasas de sobreventa, lo cual provoca alta congestión en las horas punta y velocidades reales más bajas que las contratadas.

Los usuarios no necesariamente son expertos en redes, en consecuencia, se puede suponer que no comprenden lo que sucede en un router, por lo que no es un trabajo sencillo mantenerlos informados de lo que ocurre. Debido a esto, es entendible que muchos clientes no sepan si su proveedor cumple con el servicio contratado, o bien, han normalizado algunos problemas, por ejemplo, de conectividad.

## 1.3. Objetivo General

En consecuencia a los problemas mencionados, el siguiente trabajo se enfoca en una solución que permita brindar apoyo a los usuarios de redes WiFi, permitiendo que estos tengan a su disposición el conocimiento de los servicios que pueden usar en su hogar sin mayores inconvenientes. De acuerdo a esto, como objetivo general, se pretende elaborar y validar una herramienta que utiliza métricas claves de redes WiFi para determinar la calidad de servicio del usuario en tiempo real.

## 1.4. Objetivos Específicos

Para poder llevar a cabo el objetivo general, se plantean cuatro objetivos específicos, los cuales corresponden a los pasos previos que se deben realizar para cumplir con el objetivo propuesto, y son los que se enumeran a continuación.

1. Determinar los parámetros que influyen en la percepción de calidad de servicio del cliente en la red WiFi.
2. Recopilar la información del nivel de señal de la red WiFi propia del usuario y de las redes desplegadas en el área, las cuales que pueden influir en la calidad del servicio
3. Analizar la información obtenida para generar reportes comprensibles por el usuario.
4. Crear una aplicación para android que permita visualizar de forma clara la información obtenida.

## 1.5. Metodología

Para lograr los objetivos propuestos se da uso al software de administración de proyectos llamado “trello”, con la finalidad de utilizar el sistema kanban para registrar actividades. A propósito de la obtención de datos de la red inalámbrica WiFi, se propone utilizar “android studio”, pues así se puede desarrollar una aplicación en java que los consiga y muestre esta información de forma clara en el dispositivo móvil. Por consiguiente, es necesario un sistema de administración de base de datos, por lo cual se utiliza Realm. Además, como repositorio del proyecto<sup>2</sup>, se da uso a “github” para gestionar el proyecto y utilizar un sistema de control de versiones.

El análisis de calidad se basa en que el usuario utiliza la aplicación en un ambiente residencial, es decir, con pocas personas, donde todas se encuentran conectadas a la red WiFi de manera inalámbrica, es decir, no se consideran aquellos usuarios que están conectados vía cableada. El objetivo de la aplicación, por el momento, es darle a conocer al usuario si las personas conectadas a la red pueden utilizar las aplicaciones que desean en la ubicación del hogar en que se encuentran, para lo cual se recopilan datos de la red y se realiza un cálculo para obtener el throughput máximo de cada uno de los usuario.

Las pruebas de la aplicación se realizan en el séptimo piso de un departamento de la comuna de Santiago en la Región Metropolitana de Chile <sup>3</sup>, el cual tiene diez departamentos en cada piso y se encuentra rodeado de otros edificios. Las pruebas consisten en mostrar de forma detallada los datos obtenidos y los cálculos realizados para un usuario en distintas partes del hogar, los cuales están a diferentes distancias del único router del departamento. El dispositivo con el que realizan las pruebas corresponde a un motorola g6 play que tiene una versión de android 8.0.

En el siguiente informe se presenta un capítulo de “Antecedentes”, donde se especifican los conceptos más importantes para comprender el proyecto en su totalidad, además, se describen algunos hechos actuales que han sido influyentes en el origen de esta memoria

---

<sup>2</sup>El repositorio del proyecto se encuentra en <https://github.com/Catagzzi/App>

<sup>3</sup>Por motivos de privacidad la ubicación exacta no es descrita.

. Luego, se tiene un capítulo de “Metodología”, donde se procede a explicar los métodos y procedimientos utilizados para la obtención de los datos y la lógica utilizada para la confección de la aplicación para android. A continuación, se presenta un capítulo de “Descripción de la plataforma”, donde se muestra y explica el proyecto en su totalidad, desde la plataforma que observa el usuario hasta los cálculos realizados para realizar el análisis de calidad de servicio. En el capítulo siguiente a éste, llamado “Funcionamiento y Análisis”, se muestran algunos resultados del análisis a realizar, los cuales son explicados y analizados, además, se describen los alcances que puede llegar a tener esta aplicación a futuro. Finalmente, se tiene un capítulo de “Conclusiones”, donde se realiza una reflexión final del trabajo realizado.

# Capítulo 2

## Antecedentes

### 2.1. Redes Fijas

En [8] se refiere a una red fija como aquella en la que los usuarios tienen una movilidad reducida o nula, pues el acceso se realiza mediante un cable.

#### 2.1.1. Alcance de las Redes

El alcance de una red se refiere al radio que abarca al brindar acceso, es decir, el área de cobertura. Éste puede variar de pequeñas a grandes distancias dependiendo de la disposición de equipos conectados entre sí, por lo tanto, se debe tener en consideración el diseño de la red y los componentes físicos utilizados para su construcción. A partir de esto existen distintos tipos de alcances de una red y son los que se describen a continuación según lo mencionado en [8].

- PAN (Personal Area Network): Corresponde a las redes de área personal, abarcando una zona de pocos metros de alcance de un usuario. Además, permite un acceso cableado o inalámbrico para conectar diversos dispositivos, tales como: celulares, impresoras o notebooks. Un ejemplo de este tipo de red es una computadora conectada con sus periféricos, como lo son el teclado, la impresora, el ratón, etc.
- LAN (Local Area Network): Corresponde a las redes de área local, abarcando un área de hasta 200 [m] o 1[km] de distancia con el uso de repetidores, lo cual permite interconectar un conjunto de computadores o dispositivos periféricos. Las WLAN son redes de área local que permiten una comunicación inalámbrica, donde los dispositivos se conectan a punto de acceso a la red (AP, por sus siglas en inglés).
- CAN (Campus Area Network): Corresponde a las redes de área de campus, lo cual se refiere a que abarca zonas de un campus universitario o de una base militar. Es muy similar a una MAN, con la diferencia de que es una red privada, pues sus componentes son propiedad de una organización.
- MAN (Metropolitan Area Network): Corresponde a las redes de área metropolitana, abarcando zonas de cobertura extensas dentro de una ciudad, además, pueden ser públicas o privadas.

- WAN (Wide Area Network): Corresponde a las redes de área amplia, lo cual se refiere a que puede cubrir un país o un continente.

### 2.1.2. Componentes Básicos

De acuerdo a lo mencionado en [8], se pueden utilizar varios medios físicos para la transmisión de información desde una máquina a otra. Dentro de los medios guiados se encuentran los cables, los cuales son aquellos elementos de la red que, a nivel de capa física, permiten transportar bits. Estos pueden ser de distintos tipos, pues varían en términos de ancho de banda, retardo, costo o facilidad de instalación, y son los que se mencionan a continuación.

- Cable de par trenzado: Consiste en dos cables de cobre aislados trenzados en forma helicoidal y, en general, de 1 mm de grosor. El ancho de banda depende del grosor del cable y de la distancia que recorre, pero en muchos casos se pueden lograr varios megabits/seg durante pocos kilómetros. Debido a su adecuado desempeño y bajo costo, los pares trenzados se utilizan mucho.
- Cable coaxial: Consiste en un alambre de cobre rígido como núcleo, rodeado por un material aislante, el cual se encuentra forrado de un conductor cilíndrico que, por lo general, es una malla de tejido fuertemente trenzado y, además, el conductor externo está cubierto con una funda protectora de plástico. Este cable permite un alto ancho de banda, dependiendo de su calidad y longitud, y una excelente inmunidad al ruido.
- Líneas eléctricas: En el cableado de líneas eléctricas la señal de datos está sobrepuesta en la señal eléctrica de baja frecuencia, pues ambas señales usan el cable al mismo tiempo. Las propiedades eléctricas varían de una casa a la otra y cambian a medida que los electrodomésticos se encienden y apagan, lo cual hace que las señales de datos reboten alrededor del cableado.
- Fibra óptica: Un sistema de transmisión óptico tiene tres componentes clave: la fuente de luz, el medio de transmisión y el detector. En la fibra óptica un pulso de luz viaja por medio de una fibra de vidrio ultradelgada hasta ser percibida por el detector, el cual genera el pulso eléctrico de la señal. Éstas son más costosas, sin embargo, logran una transmisión de datos de 100 [Gbps] por 100 [km] sin necesidad de amplificación.

Otro elemento importante en la red es el hardware del dispositivo en sí, pues ahí se ejecutan procesos de capa física y de enlace. El componente del hardware más importantes para el funcionamiento de la red es la tarjeta de red (NIC, por sus siglas en inglés), la cual corresponde a la interfaz entre el host y el cable que lo conecta a la red, por lo tanto, su función es asegurar que los datos lleguen a su destino. Para esto, la información es convertida en señales eléctricas, las cuales son almacenadas en paquetes de datos para que el sistema operativo logre comprender dicha información. Por lo tanto, las funciones que debe cumplir un adaptador de red son: recibir y enviar datos de equipo y transformarlos en señales, recibir señales eléctricas del cable y traducirlas en datos, determinar si los datos recibidos son para el equipo y controlar el flujo de datos [8].

Finalmente, se tienen los dispositivos inalámbricos, los cuales permiten que los equipos se conecten a la red. Por ejemplo, a nivel de WLAN se tienen los puntos de acceso inalámbricos que permite interconectar dispositivos, delegando la tarea de enrutamiento y direccionamiento a servidores, enrutadores y switches [8].

### 2.1.3. Tecnologías de Redes LAN

Las tecnologías LAN son aquellos métodos utilizados para garantizar la conexión entre dispositivos de la red mediante estándares diseñados para que las redes LAN funcionen de manera correcta. A continuación se procede a describir las tecnologías más comunes según lo descrito en [8].

- Ethernet

Dentro de las tecnologías LAN, ethernet es la más usada y, a su vez, ésta se divide en dos tipos: Ethernet clásica y Ethernet conmutada. La primera resuelve el problema de acceso múltiple y opera a tasas de transmisión de 3 a 10 Mbps, mientras que la segunda utiliza dispositivos llamados switches para conectar distintas computadoras. Ésta última es en lo que se conoce como la Ethernet y opera a 100, 1000 y 10000 Mbps, en formas conocidas como Fast Ethernet, Gigabit Ethernet y 10 Gigabit Ethernet.

El método de acceso al medio que utiliza ethernet es CSMA/CD o acceso múltiple con portadora y detección de errores. Este método consiste en que cada estación debe escuchar el medio antes de transmitir datos y, en caso de que haya ocurrido una colisión en la red, la estación que haya detectado este problema procede a enviar un mensaje de notificación al resto. Luego de esto, las transmisiones se detienen y se ejecuta un algoritmo de backoff, el cual consiste en esperar un tiempo aleatorio antes de volver a intentar la transmisión.

- Token Ring

Es una arquitectura de red que consiste en una topología de anillo, sin embargo, los ordenadores se encuentran conectados a una unidad de acceso multi-estación (MSAU, por sus siglas en inglés). Esta tecnología utiliza un método de acceso a partir de un testigo, el cual es una serie especial de bits que viaja sobre la red concediendo permisos, es decir, un equipo puede transmitir sólo si posee el testigo, mientras que los otros permanecen en espera. El problema de esta tecnología radica en su complejidad, por lo que se requerían chips especiales y precios altos.

- FDDI

La interfaz de radio distribuida por fibra (FDDI, por sus siglas en inglés) corresponde a una red compuesta por dos topologías de anillos, similares al caso de token ring, donde la información fluye en sentidos opuestos. El presentarse un fallo en el anillo primario, la red se reconfigura para que la información se continúe transmitiendo por el secundario. Esta tecnología utiliza el mismo método de acceso que token ring, es decir, a partir de testigos.

### 2.1.4. Estado Actual de Redes Fijas en Chile

Debido al desarrollo tecnológico ha aumentado el alcance económico para optar por la instalación y contrato de servicios de internet fijo, por lo que comienza una tendencia de crecimiento del número de conexiones fijas. Esto se puede observar en la figura 2.1, donde se tiene, para marzo del 2020, una penetración de conexiones fijas de internet del 18.01 %.

La novena encuesta de accesos y usos de internet realizada por la SUBTEL indica las siguientes razones por las que la población no tiene acceso a internet.

- 34.6% señala que es debido al costo elevado del servicio.
- 28.2% señala que no les interesa.
- 25.4% no saben utilizar el computador.
- 23.4% explica que no saben o no les han informado que beneficios ofrece internet al hogar.
- 15.1% indica que no saben o no les han informado como utilizar internet.

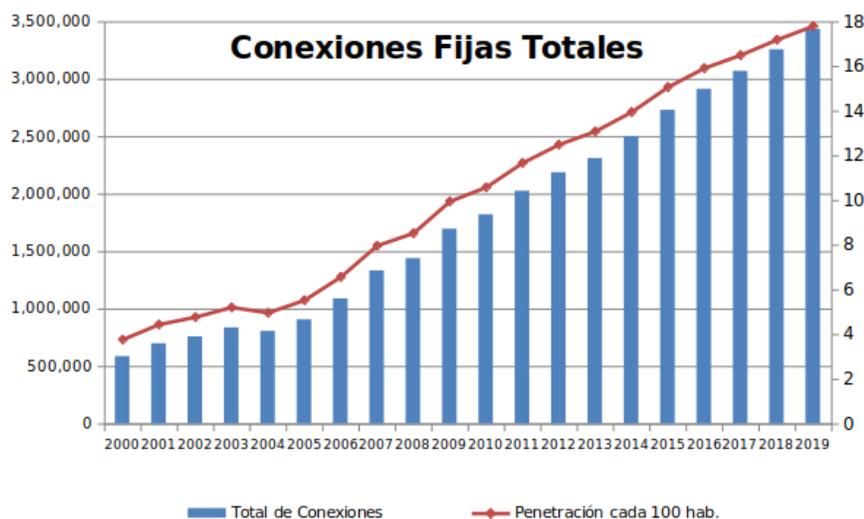


Figura 2.1: Número de conexiones fijas por año. Figura obtenida de la SUBTEL.

Considerando las conexiones fijas de internet se puede observar en la figura 2.2 que HFC y fibra óptica son las tecnologías más usadas. Cabe destacar que el uso de fibra ha aumentado notablemente respecto al 2012, a pesar de su alto costo de instalación.

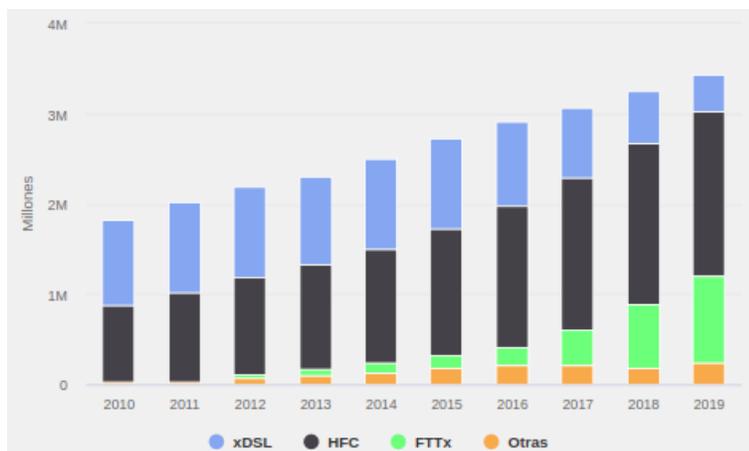


Figura 2.2: Cantidad de líneas de banda ancha fija por tecnología.

## 2.2. Servicios de Internet

En Chile, el mayor número de conexiones a internet se realizan a través de dispositivos móviles, lo cual demuestra que la gente se está adaptando a los cambios digitales, pues cada

vez se utilizan más servicios de entretenimiento, información y comunicación. De acuerdo a esto, en los últimos años, el uso de internet móvil ha crecido notablemente, mientras que la telefonía fija ha sido el servicio que ha tenido una mayor disminución en su uso. En la figura 2.3 se puede visualizar el número de servicios por habitantes en Chile, llegando al 2018 a un número de 2.8 servicios contratados por persona. En la figura 2.4 se observa dicho aumento de manera detallada, donde se aprecia el dominio de los servicios de internet y voz móvil, mientras que la TV pagada, el internet y la voz fija se mantienen bajos hasta el 2018 [10]. Sin embargo, el uso de internet fijo no está siendo bien representado, pues se debe considerar que cada servicio contratado corresponde a una familia entera con conexión a internet, es decir, si se contratan 1 millón de servicios de internet fijo se tienen 4 millones de personas con conexión a internet, considerando que una familia está compuesta por cuatro personas en promedio. Por lo tanto, el 100 % de contrataciones de este tipo de servicio corresponde a un cuarto de la población total aproximadamente y, por cada unidad de servicio contratada, el número de personas beneficiadas es el cuádruple. De acuerdo a las estadísticas realizadas por la SUBTEL, se tiene que para diciembre del 2018 se cuenta con una penetración de servicios de internet fijo de 49 cada 100 hogares.

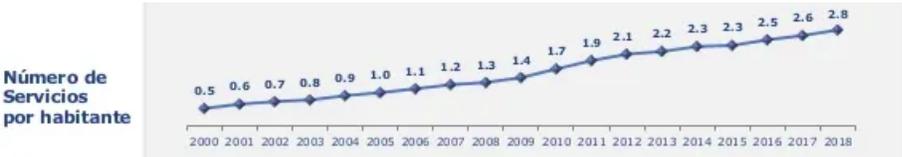


Figura 2.3: Número de servicios por habitantes. Figuras obtenidas de [10].

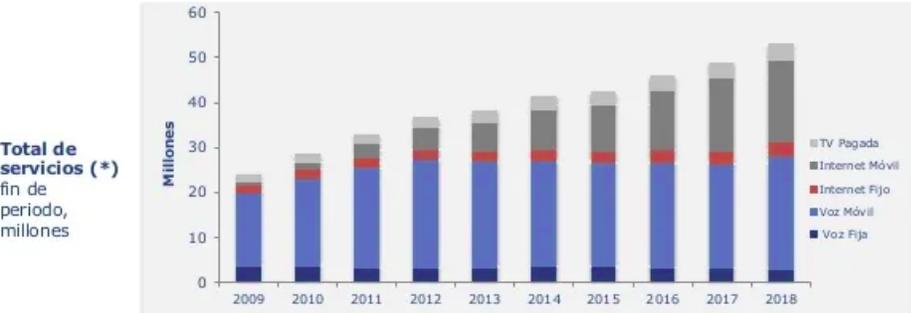


Figura 2.4: Total de servicios entregados. Figuras obtenidas de [10].

### 2.2.1. Proveedores de Servicios de Internet en Chile

Según [5], los proveedores de servicios de internet (ISP) son aquellas empresas u organizaciones que proporcionan a los usuarios el acceso a internet y a servicios relacionados, lo cual logran a partir del despliegue de toda la infraestructura necesaria para cumplir dicha función. A menudo, son empresas que proporcionan servicios de telecomunicaciones, incluyendo el acceso a las comunicaciones de datos y la conexión telefónica. A pesar de que existan muchos de estos proveedores, los usuarios no tiene problemas de interoperabilidad, pues estos se encuentran conectados entre sí por medio de redes. A su vez, los operadores son empresas que contratan la infraestructura de estos proveedores y venden sus diversos servicios a los usuarios.

En Chile existen varios proveedores de servicios de internet, los cuales se presentan en la figura 2.5, donde se observa el porcentaje de participación de conexiones de internet fijo según la empresa y la tecnología utilizada, siendo VTR el ganador con un 38.7% del total del mercado, seguido por Movistar con un 31.5% y Claro con 14.1%. Con respecto a la infraestructura, las conexiones HFC (Híbrido de Fibra Coaxial) es el que tiene mayor parte del mercado con un 54.8%, luego le sigue la fibra óptica con un 21.6%, ADSL con un 18.1% y otras tecnologías con un 5.5% [10].

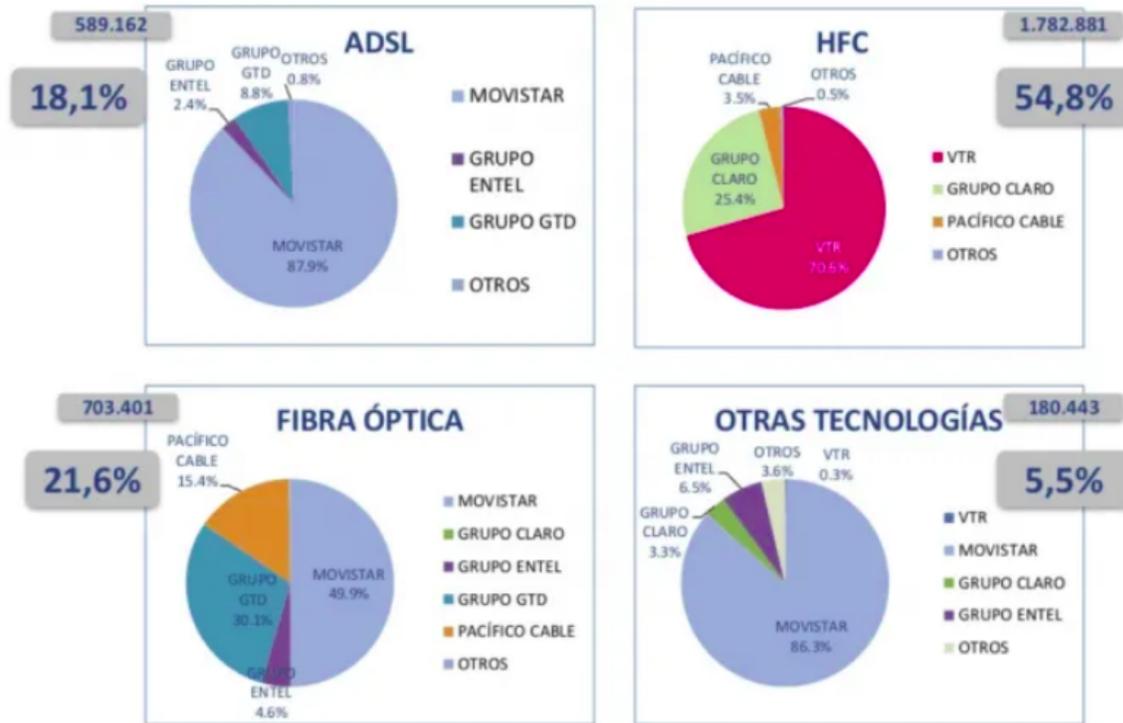


Figura 2.5: Porcentaje de participación de conexiones por empresa y tecnología. Figura obtenida de [10].

### 2.2.2. Efectos del Covid-19 en Servicios de Internet

El efecto de la pandemia del presente año trae consigo un aumento en el uso de ciertas aplicaciones tras las órdenes de quedarse en casa, lo cual se traduce a que las redes de banda ancha están pasando por un cambio muy importante. En [6] se señala que, normalmente, el tráfico de datos se distribuye entre las redes empresariales, educativas, públicas de WiFi, móvil y satelital, sin embargo, ahora debe realizarse mediante las redes de banda ancha fija para el consumidor. A nivel mundial, el tráfico ha mostrado un aumento de casi un 40% entre el 1 de febrero y el 19 de abril del 2020, lo cual ha generado un gran impacto en el rendimiento de la red y la calidad de servicio (QoS). Las aplicaciones que tienen un mayor impacto en este crecimiento son las que se describen a continuación según lo mencionado en [6].

- Conferencias remotas: Las escuelas y las empresas comienzan a utilizar herramientas de conferencia para mantenerse en contacto y tener sesiones interactivas, por ejemplo, Zoom.

- VPN: Conexiones de tiempo completo a redes corporativas de forma remota mediante VPN (Red privada virtual). Una VPN corresponde a una tecnología de red que permite conectar hosts y redes mediante el uso de otras redes como Internet y que tienen muchas propiedades de redes privadas [8].
- Juegos: Descarga masiva de juegos interactivos para entretener a los usuarios.
- Redes sociales: Su uso aumenta considerablemente, incluyendo las aplicaciones de mensajería instantánea y de almacenamiento en la nube.

Como resultado a este aumento de tráfico de internet fijo, se puede observar en la figura 2.6 el uso que le dan los consumidores a su acceso a internet, sin una interferencia externa como el acceso de internet de la empresa o de la educación. En el primer puesto se tiene a youtube que es una aplicaciones de entretenimiento con un aumento de 7.25 % [6].

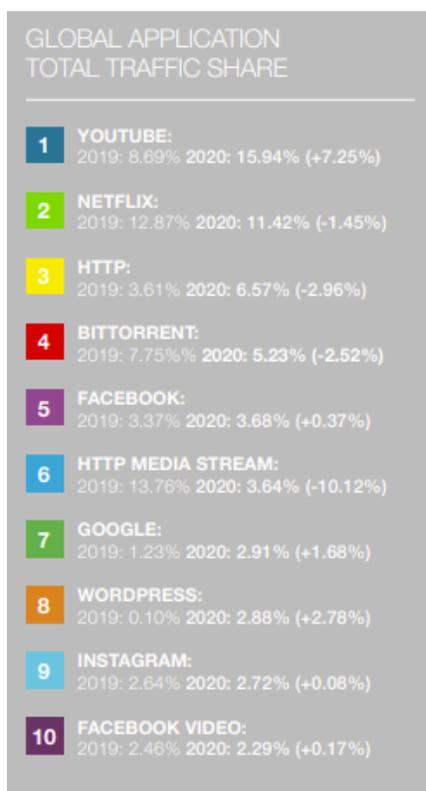


Figura 2.6: Tráfico total de aplicaciones globales. Figura obtenida de [6].

### 2.2.3. Ley 21.046

Según el Servicio Nacional del Consumidor (SERNAC), el mercado de las telecomunicaciones es el tercero con mayor cantidad de reclamos en el país, dentro del cual el mercado de internet fijo corresponde a un 12,7 % del total de las reclamaciones [7]. Debido a esto, de acuerdo a lo registrado en la Biblioteca del Congreso Nacional de Chile (BCN), el 11 de Septiembre del 2012 se ingresa un proyecto que busca establecer la obligación de una velocidad mínima garantizada de acceso a internet y, luego de cinco años de tramitación, esta ley es promulgada por la presidenta Michelle Bachelet el 3 de Noviembre del 2017, finalmente, dos días después la ley es publicada y aplicada. Sin embargo, recién el día 20 de marzo del 2020

la SUBTEL inicia una consulta pública para conocer la opinión de los ciudadanos y de las empresas, la cual finaliza el 8 de Abril del presente año y, hasta ahora, no se tienen claras las normas exigidas para dar inicio a la aplicación de la ley [2].

Según la SUBTEL, el objetivo principal de esta ley es que los proveedores de servicios de internet garanticen a los usuarios un porcentaje de velocidad de acceso promedio según el rango horario de la conexión, lo cual debe quedar estipulado en el contrato e informado en la publicidad de la empresa. Además, busca crear una herramienta que permita al cliente medir la velocidad y desempeño del servicio y, en caso de incumplimiento del contrato, la empresa debe solucionar el problema y compensar dicha falla. Cabe destacar que en el caso de conexión fija a internet no se considera el acceso mediante WiFi, por lo que se puede generar un error de interpretación por parte del cliente.

## 2.3. Calidad de Servicio

La UIT (Union internacional de telecomunicaciones) define la calidad de servicio o QoS como “el efecto global de la calidad de funcionamiento de un servicio, que determina el grado de satisfacción de los usuarios”. Por lo tanto, QoS corresponde a la capacidad de las redes, tanto móviles como fijas, de establecer las condiciones en las que se desarrolla la transmisión de los datos [17].

En una red existen distintos tipos de tráfico que requieren ciertas características para asegurar una buena calidad de servicio. Esta diferenciación es realizada por el estándar de la tecnología de acceso al medio utilizado, el cual garantiza los parámetros requeridos. Según lo indicado en [17],[18], los principales parámetros involucrados en la calidad de servicio son los que se describen a continuación.

- Latencia: Ésta corresponde al tiempo de ida del RTT, el cual corresponde al tiempo que transcurre entre el envío y la recepción de un paquete dentro de la red, y otras demoras de la conexión. Otra forma de verlo es que corresponde a la suma de todos los retrasos temporales, los cuales pueden ser: de propagación, de cola o buffer, de procesado y de transmisión.
- Jitter: Corresponde a la varianza entre dos mediciones consecutivas de RTT. En una comunicación óptima los paquetes deben llegar en forma periódica, pero esto puede verse afectado por problemas en la red como: congestión, pérdida de sincronización o transmisión de datos por diferentes caminos.
- Porcentaje de pérdida de paquetes: Corresponde al porcentaje de mensajes ICMP (protocolo de control de mensajes de internet) que no reciben respuesta con respecto al total de mensajes enviados. Esto proviene de que el protocolo ICMP permite controlar el estado de un paquete, entregando un mensaje en caso de que éste no llegue a su destino.
- Throughput: Corresponde a la tasa en que la información es transferida, por lo tanto, puede ser calculada a partir de la cantidad de bits recibidos por unidad de tiempo. Su valor máximo es determinado por el ancho de banda disponible y la relación señal-interferencia más el ruido (SINR).
- RSSI: Es un indicador de intensidad de la señal recibida o detectada por cualquier dispositivo conectado a un punto de acceso o router específico. Esta escala se mide en

decibelios (dB) desde 0 hasta -120, donde los valores cercanos a 0 reflejan un mayor nivel de señal. El RSSI no indica la calidad de la señal, pues se debe considerar la relación señal/ruido.

- GBR

Por un lado, la calidad de servicio puede analizarse desde un punto de vista del usuario, correspondiente a la percepción del servicio por parte de los usuarios. Por otro lado, se tiene un punto de vista de la red, el cual se enfoca en tres aspectos que son mencionados a continuación según lo descrito en [17].

1. Accesibilidad de la red: Se refiere a la disponibilidad de los recursos de red que permiten a los usuarios utilizar los servicios. Se puede medir a través de parámetros de intensidad, los cuales varían dependiendo de la tecnología de red.
2. Accesibilidad de servicio: Se refiere a la disponibilidad de los servicios que ofrece la red. Se pueden medir ciertos parámetros que permiten su evaluación, como por ejemplo, la latencia.
3. Integridad de servicio: Corresponde a la calidad entregada durante el servicio, lo cual puede verse afectado por la calidad de señal de voz, la calidad de transmisión, el throughput o caída del servicio.

En el actual modelo que ofrecen las redes fijas se requieren ciertas garantías debido a las necesidades específicas de algunos servicios, lo cual se traducen en una distinción y diferenciación de paquetes de datos. De acuerdo a esto, la implementación de QoS en los routers de una red IP costa de dos fases, una distinción de paquetes y un tratamiento diferenciado.

Los routers en sí también utiliza algunas técnicas que permiten mejorar la eficiencia, con lo cual se reduce el tráfico circulante, como por ejemplo el caché local, el cual es un bloque de memoria que permite mantener a mano los datos requeridos frecuentemente por varios procesos. Cuando un proceso requiere información, primero consulta el cache, si la información se encuentra allí se produce una mejora en el desempeño, reduciendo el retardo de procesamiento [12].

Este tipo de redes tiene tres modelos para responder a la calidad de servicio y son Best-Effort o mejor esfuerzo, IntServ o servicios integrados y DiffServ o servicios diferenciados. El primero es más bien considerado un comportamiento de la red donde se define un servicio único, por lo que las aplicaciones transmiten datos siempre que deban [13].

### **2.3.1. Servicios Integrados: IntServ**

IntServ es un modelo que propone una solución para el soporte de QoS de extremo a extremo, lo cual logra a partir de la reserva de recursos en los diferentes equipos de conmutación (routers) que componen el trayecto de la información. Este modelo permite que las mismas aplicaciones especifiquen los requisitos de calidad de servicio que necesitan, los cuales deben difundirse entre los diferentes routers por los que se encaminan los paquetes. Estos equipos deben proporcionar mecanismos para el control de la calidad de servicio ofrecida a estos flujos de información, lo que se consigue mediante la reserva de recursos [13].

IntServ se basa en la definición de dos elementos, una arquitectura donde se permite la

reserva de recursos de conmutación, y un protocolo que permita a las aplicaciones transmitir sus requisitos, el cual corresponde al protocolo RSVP. Éste se utiliza para hacer las reservaciones, además, brinda la posibilidad de que varios emisores transmiten a múltiples grupos de receptores, permite que receptores individuales cambien de canal libremente, optimiza el uso de ancho de banda y elimina la congestión. Con respecto a la arquitectura, ésta consiste en nodos que realizan todas las funciones descritas a partir de módulos, lo cual se puede observar en la figura 2.7, donde se muestra el esquema interno de un nodo [13],[8].

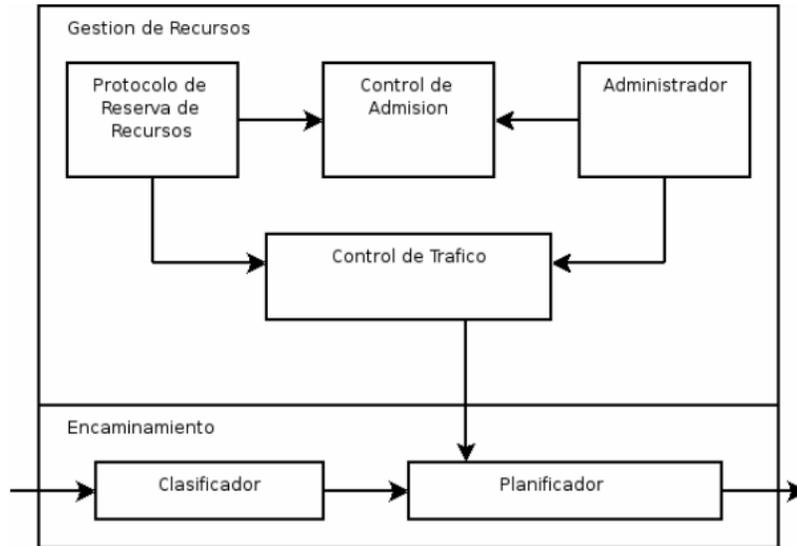


Figura 2.7: Esquema de funciones internas de un nodo IntServ. Figura obtenida de [13].

En cada nodo, el encaminamiento se realiza mediante los módulos clasificador y planificador, mientras que la gestión de recursos se lleva a cabo en el Control de Admisión, Protocolo de reservas y Administrador, los cuales son descritos a continuación según [13].

- Control de admisión: Corresponde a un algoritmo encargado de decidir si una petición puede ser aceptada de acuerdo a los recursos disponibles en el nodo y los solicitados por la aplicación.
- Administrador: Establece diferentes políticas en relación a la configuración adecuada del control de admisión, clasificador y planificador.
- Protocolo de reserva de recursos: Es el encargado de recibir y retransmitir los mensajes de reserva de recursos que generan las aplicaciones.
- Clasificador: Se encarga de asignar la clase apropiada a cada paquete recibida en función de las tablas de clasificación que almacena.
- Planificador: Es el encargado de conformar el tráfico de salida en función de las reservas realizadas.

La gran cantidad de usuarios y el elevado número de flujos que pueden generar provoca que existan graves problemas de escalabilidad en el núcleo de la red. Además, cada router debe almacenar un listado de todos los flujos activos y los correspondientes recursos asignados. Por lo tanto, estos factores provocan que el modelo Intserv sea difícilmente implementable en una red de dimensiones considerables [13].

### 2.3.2. Servicios Diferenciados: DiffServ

DiffServ es un modelo que propone una solución para el soporte de QoS basado en la priorización de clases de tráfico. Al igual que IntServ, se realiza una reserva de recursos en los nodos intermedios, pero en este caso se realizan por conjuntos de clase de tráfico en lugar de por flujos. Otra diferencia es que las aplicaciones no realizan ninguna petición de recursos, si no que deben marcar el tráfico que generen adecuadamente para que éste reciba un tratamiento específico en función de la clase a la que pertenece [13].

La arquitectura del modelo DiffServ consiste en un dominio, cuyo esquema se puede observar en la figura 2.8, donde aparecen dos tipos de router y son los que se describen a continuación a partir de [13].

- Nodos interiores o CR: Son aquellos routers que forman el núcleo de la red, cuya función es proporcionar un sistema de colas que permita ofrecer diferentes tratamientos a los conjuntos de clase de tráfico.
- Nodos frontera o ER: Son aquellos routers que se encuentran en los límites del dominio y presentan algún interfaz con un nodo fuera del dominio DiffServ o con una red de acceso. Además, son los encargados de implementar las funciones descritas para los nodos interiores y deben encargarse de las funciones de clasificación y acondicionamiento de tráfico.

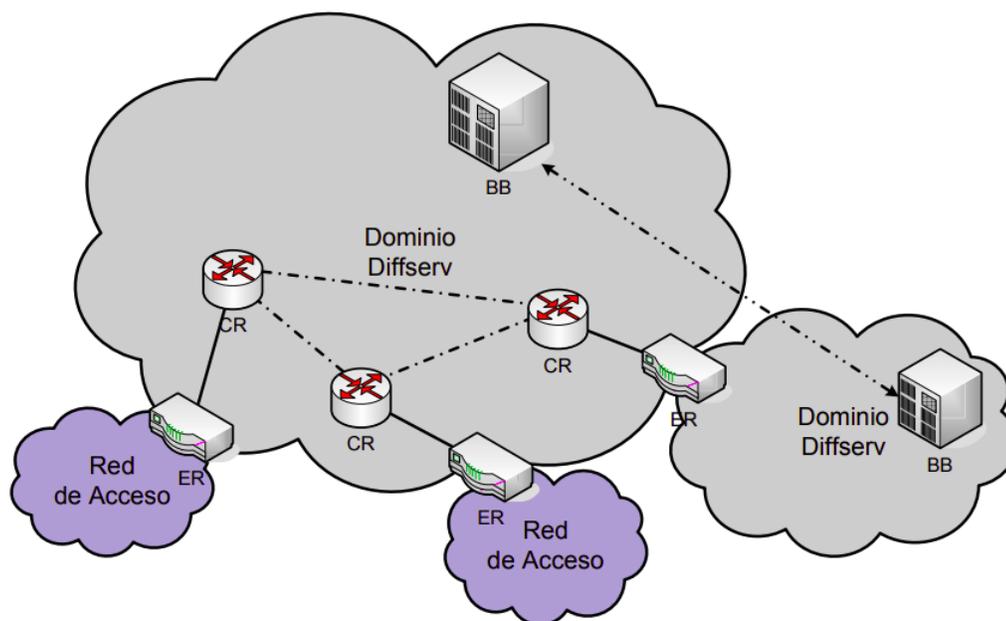


Figura 2.8: Elementos que conforman un dominio DiffServ. Figura obtenida de [13].

Para la identificación de las diferentes clases de tráfico se define un código llamado DSCP, el cual se mapea en el campo de tipo de servicio (ToS) en el caso de IPv4 y en el campo de clase de tráfico en IPv6. Los primeros 6 bits se utilizan para identificar la clase a la que pertenece cada paquete, de forma que los routers puedan aplicar el tratamiento correspondiente. Este procedimiento es lo que se conoce como comportamiento por salto (PHB, por sus siglas en inglés), el cual corresponde a un conjunto de condiciones para el tratamiento de tráfico.

Existen tres tipos de perfiles de comportamiento y son los que se describen a continuación según [13],[8].

- Reenvío expedito o EF: Está indicado para aquellos flujos de tráfico que requieren un caudal mínimo asegurado, así como un retardo limitado y una variación de retardo máxima determinada, Este perfil se ajusta a aplicaciones para tráfico en tiempo real, pues los paquetes asociados tienen un tratamiento preferencial garantizado por todos los routers diffserv en la ruta.
- Reenvío asegurado o AF: Este perfil está indicado para flujos de tráfico con menores requisitos que los indicados en EF, pues no se garantizan requisitos temporales en este caso. Dentro de cada clase definida en AF se especifica que deben haber cuatro clases de prioridades, cada una con sus propios recursos. Además, se definen tres probabilidades de descarte para los paquetes que están en congestión: baja, media y alta. Por lo tanto, estos dos factores en conjunto definen 12 clases de servicios.
- Mejor esfuerzo o BE: Aunque no pertenece exclusivamente al modelo Diffserv, este perfil se utiliza para el tráfico que no tiene requisitos de calidad de servicio, por lo que es adecuado para aplicaciones que trabajan en background o que no requieren trabajo en tiempo real.

Tanto los nodos interiores como los frontera deben cumplir con dos funciones específicas dentro de un dominio diffserv, clasificación y acondicionamiento. Por un lado, la clasificación consiste en identificar el perfil PHB al que corresponde un flujo de tráfico y, por otro lado, el acondicionamiento pretende conseguir que el tráfico ingresado en un dominio se ajuste a las condiciones. Para cumplir con lo último se debe comprobar si el tráfico de entrada se ajusta a un patrón determinado, asignar un código DSCP a los paquetes de entrada y asegurar que el tráfico se adapte a las condiciones.

## 2.4. Estándar IEEE 802.11

Un estándar define un conjunto de reglas de acceso y de transmisiones al interior de la red. En específico, el estándar IEEE 802.11 o WiFi está dedicado a la normalización de redes LAN, enfocándose en la capa física y de enlace del modelo OSI. Esta tecnología surge por la necesidad de establecer un mecanismo universal de conexión inalámbrica, siendo así creado el primer estándar IEEE 802.11 en 1997, el cual presenta una extrema lentitud de conexión para la mayoría de las aplicaciones [9]. Por esta razón se crean versiones mejoradas a los largo de los años, generando así una familia de estándares, los cuales se pueden observar en la tabla 2.1.

### 2.4.1. Arquitectura

La arquitectura del estándar 802.11 se basa en un sistema dividido en celdas o células denominadas conjunto básico de servicio (BSS), lo cual se observa en la figura 2.9. Cada BSS está conformada por nodos, fijos o móviles, llamados ‘‘Estaciones’’ y está gobernada por un punto de acceso (AP). Los AP son elementos muy importantes para las redes WLAN, pues actúan como transmisores y receptores de las señales transmitidas, es decir, proporcionan cobertura a las estaciones que forman parte de la red [9]. Las funciones básicas que pueden

Estándar	Característica
IEEE 802.11	Estándar original con tasas de 1 y 2 Mbps trabajando a 2.4 GHz
IEEE 802.11a	Tasas de hasta 54 Mbps en 5 GHz, utiliza 52 subportadoras de OFDM
IEEE 802.11b	Tasas de hasta 11 Mbps en 2.4 GHz
IEEE 802.11c	Permite la comunicación de dos redes distintas
IEEE 802.11d	Permite que distintos dispositivos intercambien información
IEEE 802.11e	Mejoras para el soporte de calidad de servicio
IEEE 802.11f	Protocolo para la comunicación entre puntos de acceso
IEEE 802.11g	Tasas de hasta 54 Mbps en 2.4 GHz (compatible con 802.11b)
IEEE 802.11h	Trabaja en 5 GHz y propone extensiones para la compatibilidad con Europa
IEEE 802.11i	Mejoras en seguridad
IEEE 802.11j	Extensiones para Japón
IEEE 802.11k	Permite calcular y valorar los recursos de radiofrecuencia de los clientes
IEEE 802.11n	Mejoras mayores en la tasa de transmisión, trabaja en 2.4GHz y 5 GHz
IEEE 802.11p	Uso de 802.11 en vehículos, opera en 5.9 GHz y 6.2 GHz
IEEE 802.11r	Permite una rápida transición entre nodos
IEEE 802.11s	Redes en malla inalámbricas
IEEE 802.11t	Predicción de rendimiento inalámbrico para probar estándares y métricas
IEEE 802.11u	Uso conjunto con otras redes no-802
IEEE 802.11v	Gestión de la configuración de dispositivos cliente
IEEE 802.11w	Permite proteger la red ante ataques en las tramas de gestión inalámbrica
IEEE 802.11ac	Mejora las tasas de transferencia hasta 433 Mbit/s, opera en 5GHz
IEEE 802.11ax	Mayor rendimiento y eficiencia espectral global, opera en 2.4 GHz y 5 GHz

Tabla 2.1: Familia de protocolos 802.11, información obtenida de [9], [13].

realizar los AP son las que se mencionan a continuación de acuerdo a descrito en [9].

- Portal para interconectar la red WLAN y otra red LAN 802.x de otro tipo.
- Puente hacia otros puntos de acceso, con la finalidad de extender los servicios de acceso.
- Router para encaminar los datos dentro de la zona de cobertura de acuerdo a los requerimientos.

Las estaciones obtienen acceso al sistema de distribución (DS) por medio del AP, donde pueden conectarse a otros nodos fuera de su área de cobertura gracias a que este componente se encarga de conducir las tramas hacia su destino. Las estaciones o clientes inalámbricos pueden ser cualquier dispositivo capaz de administrar una red inalámbrica y que soporte el protocolo utilizado. Por último, hay que mencionar que la unión de varias BSS representa una unidad lógica llamada conjunto de servicio extendido (ESS) [9].

### 2.4.2. Funcionamiento en Capa Física

De acuerdo a lo descrito en [27], para mejorar la eficiencia en el protocolo 802.11 en ambientes con alta interferencia de canales y propagación por trayectos múltiples, se utilizan distintas técnicas enfocadas en mejorar la comunicación. Una de éstas es la modulación OFDM, en la cual los bits enviados se encuentran espaciados en canales de frecuencia y



Figura 2.9: Arquitectura del estándar 802.11. Figura obtenida de [9].

son ortogonales uno del otro, lo cual permite utilizar portadoras solapadas entre sí sin interferencia. Esto evoluciona en el protocolo 802.11ax a OFDMA, donde estos canales son subdivididos para ser utilizados por distintos usuarios y dispositivos, lo cual se puede observar con mayor claridad en la figura 2.10. Tanto OFDM como OFDMA dividen un canal en subportadoras mediante una función matemática conocida como transformada rápida inversa de Fourier (IFFT). En un canal 802.11 n/ac con un ancho de banda de 20 [MHz] se tienen 52 subportadoras útiles, donde 48 son de datos y 4 son piloto.

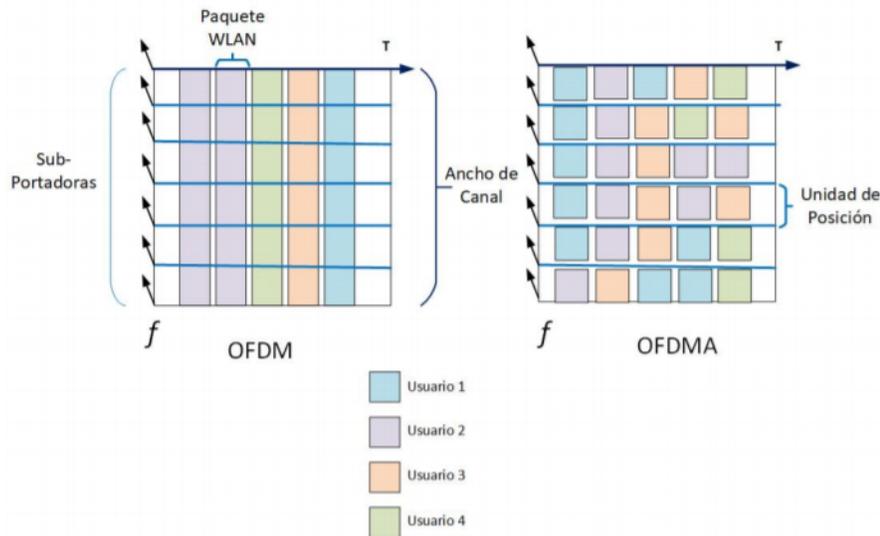


Figura 2.10: Diferencia entre OFDM y OFDMA. Figura obtenida de [27].

De acuerdo a esto, el estándar 802.11 funciona en rangos de frecuencias, los cuales se encuentran separados en canales, tal y como se observa en la figura 2.11 para la banda de 2.4 [GHz]. En el caso de Chile, se tienen 13 separaciones, las cuales ocupan parte del espectro de manera solapada, es decir, el canal 1 coincide con el 2 o el 3, por lo que existe interferencia entre ellos [11].

Otra tecnología utilizada en este estándar es MIMO, lo cual permite incrementar la capacidad de los enlaces al utilizar múltiples antenas que pueden recibir y transmitir en simultáneo.

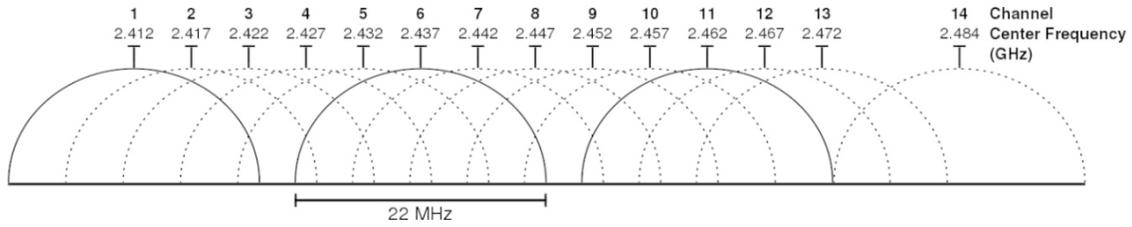


Figura 2.11: Separación de los canales en la banda de 2.4 [GHz]. Figura obtenida de [21].

Esto es muy beneficioso cuando se tienen muchos dispositivos conectados a la red. Además, al utilizar varias antenas se tiene la ventaja de que la señal puede recorrer diferentes caminos hasta su objetivo, por lo que se puede aprovechar los rebotes para potenciar el rendimiento. Sin embargo, se tiene un procesamiento más complejo, por lo que el costo del sistema es mayor.

### 2.4.3. IEEE 802.11 MAC - Nivel de Enlace

En el estándar IEEE 802.11, la capa de acceso al medio se encarga de proporcionar un servicio fiable al usuario y, al mismo tiempo, permite un acceso equitativo al medio inalámbrico compartido. Para esto, el estándar utiliza un protocolo para el intercambio de información llamado CSMA/CA, el cual consiste en enviar la trama de origen a destino con un mensaje de confirmación llamado ACK por parte de este último. Además, todas las tramas tienen un campo de “control de errores de tramas” (FCS) que es comprobado cada vez que se recibe la información. En el caso de que exista un error en el envío de la trama, ya sea porque no se recibe el mensaje ACK o por un fallo en el campo de control, ésta es reenviada hasta que la información llegue al destino sin problemas. Además del mecanismo anterior, el estándar incorpora un sistema que otorga robustez al protocolo, el cual es llamado RTS/CTS. Cuando una estación utiliza este mecanismo, éste debe mandar una trama RTS al destino antes de mandar los datos (MSDU o unidad de datos de servicio MAC). La finalidad de esto es que sea recibido por el destino para que éste responda con otra trama llamada CTS, la cual sirve como señal de que el origen puede mandar las tramas MSDU [13].

El estándar 802.11 define dos funciones para el acceso al canal y son las que se mencionan a continuación de acuerdo a lo descrito en [13].

1. Función de Coordinación Distribuida (DCF): Esta función permite que dispositivos con la misma capa física tengan un acceso compartido al medio gracias al protocolo CSMA/CA. Las estaciones deben realizar una lectura del canal antes de intentar una transmisión, con el objetivo de comprobar que el medio se encuentra vacío por un periodo de tiempo (IFS), para lo cual utilizan el método RTS/CTS. Cabe destacar que el periodo de tiempo utilizado justo antes de una transmisión en condiciones normales es llamado DIFS. Una vez que la estación consigue acceso al medio, ésta puede transmitir la trama MSDU hacia el destino, el cual espera un tiempo SIFS hasta poder transmitir el mensaje ACK de confirmación. En el esquema de la figura 2.12 se puede observar, de forma más clara, los tiempos de espera entre mensajes de comprobación del medio para transmitir la data.
2. Función de Coordinación Centralizada (PCF): Esta función es opcional para los pro-

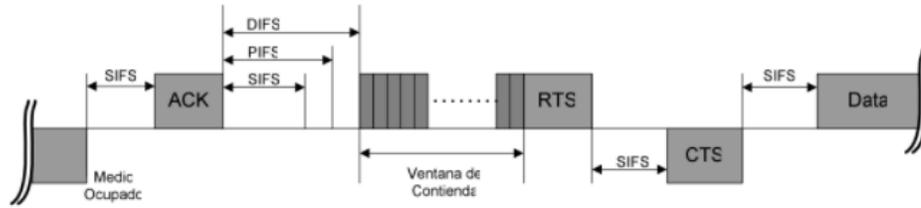


Figura 2.12: Modelo de funcionamiento de DCF. Figura obtenida de [13]

ductos 802.11, pues está diseñado para ofrecer soporte de calidad de servicio. Este mecanismo utiliza un nuevo elemento llamado punto de coordinación (PC), el cual se encuentra en el PA y se encarga de priorizar el acceso al medio de estaciones determinadas. Durante el periodo de contienda (CP) las estaciones compiten por el medio usando el mecanismo DCF y, durante el periodo de libre contienda (CFP), el PA clasifica las transmisiones. Para realizar esta clasificación se utiliza un mecanismo de sondeo, el cual comienza cuando el PC envía una trama CF-poll a una de las posibles estaciones, luego, ésta puede responder con datos junto a una trama CF-ACK o simplemente con un mensaje de confirmación si no desea enviar más información. Una vez que el intercambio de información con esta estación está lista, el PC envía el CF-Poll a otra estación que esté en la lista y así sucesivamente hasta finalizar, momento en el que transmite por difusión una trama CF-end anunciando el final del ciclo CFP. En el esquema de la figura 2.12 se puede observar, de forma más clara, los periodos entre mensajes de comprobación del medio y clasificación de transmisiones.

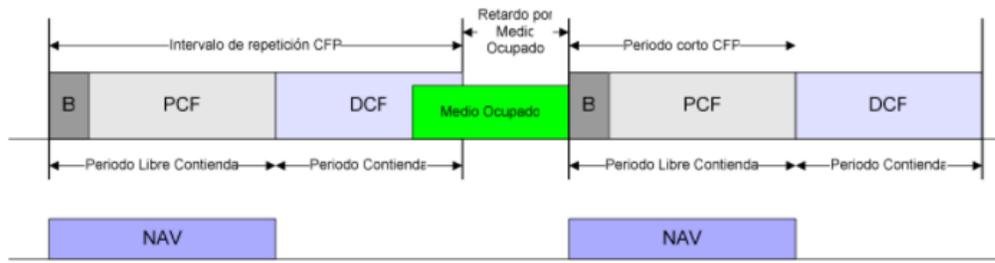


Figura 2.13: Modelo de funcionamiento de PCF. Figura obtenida de [13]

#### 2.4.4. Estándar 802.11e

Según [13], el estándar 802.11e define un conjunto de mejoras para el 802.11, para lo cual se hace una diferenciación, en primero lugar, entre estaciones con soporte de calidad de servicio (QSTA) y aquellas que no lo soporta (STA) y, en segundo lugar, entre punto de acceso con soporte de calidad de servicio (QAP) y sin soporte (AP). Por último, dentro de este estándar se puede distinguir entre las funciones de acceso al canal y la gestión de especificación de tráfico (TSPEC).

A nivel de enlace, este protocolo define una función de coordinación híbrida (HCF), la cual se emplea para el conjunto de servicios básicos con soporte QoS (QBSS) y define dos modos de operación, el acceso a canal distribuido mejorado (EDCA) y el acceso al canal controlado

(HCCA). Ambas funciones mejoran la funcionalidad de los métodos originales (DCF y PCF), pues EDCA permite soportar la priorización de tráfico, tal y como lo hace DiffServ, mientras que HCCA soporta tráfico parametrizado, de la misma forma que IntServ. Además, en este estándar, no se utilizan mensajes de confirmación para cada trama recibida correctamente, por lo que se mejora el rendimiento general de la capa MAC para el tráfico sensible al retardo [13].

Como se menciona anteriormente, EDCA realiza una priorización del tráfico, la cual se consigue introduciendo cuatro categorías de acceso (AC) que se pueden apreciar en la cuarta columna de la figura 2.14. A partir de esto, se logra un envío de tráfico asociado a las prioridades del usuario, pues cada categoría dispone de su propia cola de transmisión. Cuando los datos llegan al PA, la capa MAC de 802.11e los clasifica adecuadamente para enviar la MSDU a la cola correspondiente. Luego, el algoritmo de contienda interno calcula la espera (backoff) independientemente para cada cola, donde aquella con menos backoff es la que gana la competición. Una vez determinada la ganadora, ésta pasa a competir externamente por el acceso al medio inalámbrico, donde el algoritmo de contienda es similar a DCF, pero con tiempos de espera variable según la cola [13].

Prioridad	Prioridad 802.1D	Descripción 802.1D	Categoría de Acceso 802.11e	Descripción 802.11e
Menor	1	Background	AC_BK	Best Effort
...	2	-	AC_BK	Best Effort
...	0	Best Effort	AC_BE	Best Effort
...	3	Excellent Effort	AC_BE	Prueba Video
...	4	Carga Controlada	AC_VI	Video
...	5	Video	AC_VI	Video
...	6	Voz, Video	AC_VO	Voz
Mayor	7	Señalización Red	AC_VO	Voz

Figura 2.14: Categorías de acceso según prioridad del usuario. Figura obtenida de [13].

En cuanto al método HCCA, éste es muy similar a PCF, pero proporciona soporte de calidad de servicio basado en parametrización, pues proporciona acceso basándose en un sondeo QoS al medio inalámbrico, además, la planificación de paquetes se basa en los perfiles TSPEC (especificaciones del tráfico) admitidos. El principal concepto de HCCA es la fase de acceso controlada (CAP), la cual consiste en un intervalo de tiempo limitado formado por la concatenación de tiempos en los que una QSTA puede transmitir una serie de tramas. El CAP puede ser inicializado por el AP cuando éste necesite mandar o recibir información de una estación, con el objetivo de controlar el acceso al medio. Además, en este estándar se incorporan nuevos sub-tipos de tramas de datos para el soporte de calidad de servicio, cuyas funciones van desde garantizar el inicio de la transferencia de datos hasta para darle fin en caso de que la estación no tenga más información que enviar [13].

# Capítulo 3

## Metodología

### 3.1. Descripción Básica del Sistema

En el capítulo anterior se observa que en las redes WiFi, el correcto flujo de datos, el nivel de señal y la calidad de ésta pueden ser medidos por indicadores cuantitativos, los cuales pueden ser obtenidos por mediciones activas o pasivas a partir de estaciones de la red inalámbrica.

Por un lado, las mediciones activas corresponden a aquellas en las que se introducen paquetes a la red, estableciendo una comunicación con otro nodo, por lo que se recopila información mediante el tráfico de información [19]. Una herramienta de software que permite realizar este tipo de mediciones es el ping, el cual envía paquetes ICMP de tipo echo request y recibe como respuesta paquetes ICMP de tipo echo reply, lo cual permite obtener los indicadores de latencia, jitter y el porcentaje de paquetes perdidos. Otra medición activa que se puede realizar es la obtención de la velocidad de transmisión de datos a partir del intercambio de datos con un servidor en la red [18].

Por otro lado, las mediciones pasivas son aquellas en las que se observa el estado de la red en un punto particular, por lo que no se crea ni modifica el tráfico de red, lo cual es beneficioso al momento de obtener indicadores de calidad debido a que no se genera un tráfico extra que pueda afectar el desempeño de las conexiones [19]. A partir de este tipo de mediciones se puede obtener indicadores de volumen de tráfico de datos, latencia, throughput, intensidad de señal recibida y disponibilidad de conexión.

El sistema a realizar está compuesto por una aplicación, generada en el entorno de desarrollo “android studio”, que realiza mediciones pasivas desde el teléfono en que se encuentre. Para esto se envían peticiones hacia distintos nodos servidores de la red, los cuales envían una respuesta al dispositivo o nodo agente, tal y como se observa en la figura 3.1 [18]. Además de realizar estas mediciones, la aplicación contiene un módulo que tiene como objetivo obtener información adicional de la red, como por ejemplo si el dispositivo tiene conexión a internet o si se encuentra conectado a WiFi.

La aplicación es desarrollada tomando en consideración un hogar con un router y una

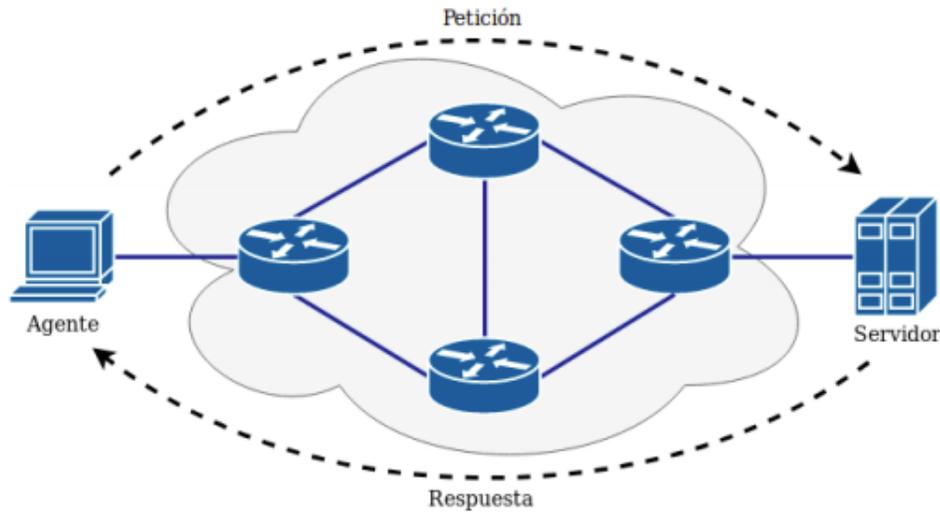


Figura 3.1: Medición pasiva en la red. Figura obtenida de [18].

pequeña cantidad de usuarios, tal y como se observa en la figura 3.2. En este caso las dos personas representadas tienen un nivel de señal distinto debido a interferencias, obstáculos y lejanía al router. De acuerdo al nivel de señal recibido, los parámetros de calidad van a ser distintos para cada usuario, por lo que las mediciones van a ser distintas para cada uno de acuerdo a muchos factores. A partir de estos parámetros se procede a realizar un análisis que involucra a ambos usuarios del router, con la finalidad de estimar la calidad de servicio que recibe cada uno.

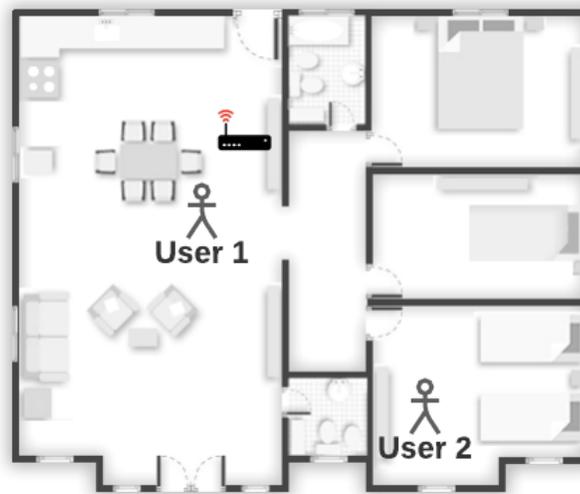


Figura 3.2: Diagrama de las condiciones de funcionamiento de la aplicación

El objetivo principal de la aplicación consiste en decirle al usuario de forma clara si las personas conectadas a la red WiFi, de forma inalámbrica, pueden ocupar las aplicaciones que desean en la ubicación en la que están, lo cual se determina a partir de los requerimientos de throughput. De acuerdo a esto, el usuario debe ingresar la cantidad de personas conectadas a la red, luego, debe ponerse en la posición de cada uno de ellos para realizar un proceso de

obtención de datos de la red, con los cuales se procede a calcular el throughput máximo que pueden alcanzar cada una de estas personas. Este dato debe ser comparado con el throughput necesario de cada una de las aplicaciones a utilizar, por lo que se procede a pedirle al usuario que seleccione el servicio que desea utilizar cada persona. Con esto datos de throughput máximo y requerido se procede a hacer un análisis de porcentaje de uso de la red, para determinar si los usuarios pueden o no utilizar el servicio que quieren. Finalmente, esta información es mostrada al usuario de una manera simple y amigable.

## 3.2. Obtención de los Datos

Como se menciona anteriormente, los datos son obtenidos por medio de mediciones pasivas realizadas por un dispositivo móvil. Para las primeras pruebas de la aplicación se utiliza un modelo motorola g6 play con sistema operativo android versión 8.0.

En la aplicación se utiliza una versión mínima de sdk 21, lo cual quiere decir que la mínima versión de sistema operativo de android en la que funciona la aplicación es android 5.0, es decir, el sistema de android impide que el usuario realice esta instalación si el dispositivo cuenta con un nivel menor de API del sistema. De acuerdo a la figura 3.3, al utilizar esta versión se abarca un 94.1 % de la población, pues los dispositivos están cada vez más avanzados, estrenando actualizaciones cada vez que sea posible.

Antes de comenzar con la obtención de datos en sí, se debe obtener información básica de la conectividad, como por ejemplo, si el usuario tiene conexión a internet y si se encuentra conectado a una red WiFi. Además de esto, se deben solicitar los permisos pertinentes al usuario, lo cual depende estrechamente de los datos que se desean conseguir.

Los primeros datos a obtener corresponden a elementos de identificación del dispositivo móvil y de la red WiFi a la que se encuentra conectado el usuario, como lo serían el IMEI, el nombre de la red y la dirección MAC del AP. Luego, se procede a obtener datos con los que se pueden realizar análisis de calidad de servicio, como la latencia, RSSI, porcentaje de paquetes perdidos, canal, ancho de banda y un escaneo de las redes WiFi alrededor.

## 3.3. Estimación de Calidad

El análisis de calidad que se realiza en esta aplicación consiste en determinar el throughput máximo que tiene cada usuario en una posición específica, el cual no puede ser obtenido mediante mediciones, pues al obtenerlo de esta forma no se tendría el máximo necesariamente. Este dato debe ser comparado con el throughput requerido por la aplicación que desea usar cada usuario, con la finalidad de obtener un porcentaje de uso del canal para cada usuario. Finalmente, se busca que estos porcentajes por sí solos y su suma reflejen información importante para el usuario con respecto al funcionamiento de la red.

Para este análisis se considera una situación estática en que la red no mejora las tecnologías utilizada para transmitir información, es decir, se considera que el usuario tiene un throughput máximo gracias a que utiliza el estándar 802.11ac, MIMO 2x2 y un intervalo entre transmisiones de 800 ns, lo cual permanece constante incluso si la red está saturada.

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99,8%
4.2 Jelly Bean	17	99,2%
4.3 Jelly Bean	18	98,4%
4.4 KitKat	19	98,1%
5.0 Lollipop	21	94,1%
5.1 Lollipop	22	92,3%
6.0 Marshmallow	23	84,9%
7.0 Nougat	24	73,7%
7.1 Nougat	25	66,2%
8.0 Oreo	26	60,8%
8.1 Oreo	27	53,5%
9.0 Pie	28	39,5%
10. Android 10	29	8,2%

Figura 3.3: Distribución de las versiones de android. Figura obtenida del IDE de Android Studio.

El análisis de calidad de servicio es realizado considerando sólo a aquellos usuarios conectados a la red WiFi de manera inalámbrica, sin embargo, debe tenerse en cuenta que pueden existir personas conectadas por ethernet, lo cual le otorga un grado de error a los resultados del análisis. Además de esto, se considera la condición ideal en que la aplicación es usada en un ambiente con pocos usuarios utilizando la red y que cuenta con un router para todos.

Con respecto a la ubicación geográfica, no se tienen restricciones de zonas urbanas o rurales, sin embargo, las pruebas de la aplicación son realizadas en un departamento en la comuna de Santiago de la región Metropolitana de Chile. Específicamente, el hogar a analizar está formado por dos personas y un router, además, se encuentra en el séptimo piso del edificio, donde cada piso consta de 10 departamentos. Es importante destacar la ubicación del edificio debido a que corresponde a una zona con mucha presencia de señales WiFi externas, con lo cual se debe mencionar que en el análisis se está asumiendo que éstas generan interferencia, sin embargo, en la práctica, éstas no tienen impacto en la calidad de la señal a menos que se encuentren transmitiendo.

# Capítulo 4

## Descripción de la Plataforma

### 4.1. Estructura del Proyecto

Este trabajo consiste en desarrollar una aplicación para android, por lo que se crea un proyecto en el entorno de desarrollo integrado “Android Studio”, donde el frontend está diseñado en el lenguaje de etiquetas XML, mientras que el backend está escrito en el lenguaje de programación Java. El proyecto generado se encuentra formado por un conjunto de carpetas y archivos que contienen el código fuente, los recursos, las configuraciones y la construcción de la aplicación [22]. La estructura general es la que se puede observar en la figura 4.1, donde las carpetas principales son manifests, java y res, las cuales son descritas a continuación junto con una explicación de los scripts “Gradle”.

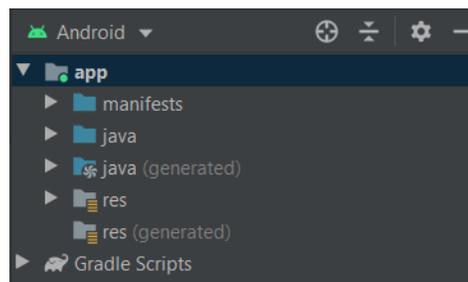


Figura 4.1: Estructura general de las carpetas del proyecto.

La carpeta “manifests” contiene el archivo “AndroidManifest.xml”, el cual incluye la descripción de las características de la aplicación creada, las cuales se nombran a continuación según lo descrito en [1].

- Permisos requeridos: Corresponden a todos los permisos que debe solicitar la aplicación android para acceder a datos confidenciales del usuario o a funciones del sistema.
- Nombre de la aplicación: Se describe con el atributo “label” y corresponde a una etiqueta de texto que se muestra a los usuarios, en este caso, el texto utilizado es “Memoria”.
- ID de la aplicación: Corresponde al nombre del paquete de la aplicación y se utiliza como identificador único de ésta en el sistema, en este caso, el ID utilizado es de “com.example.memoria”.

- Componentes o actividades: Corresponde a la declaración de las componentes o actividades creadas en la aplicación, las cuales son descritas más adelante.
- Subclase: Corresponde al camino de la subclase creada en la aplicación, en la cual se inicializa la base de datos.

En la carpeta “java” se encuentra el código fuente de la aplicación, pues contiene a los archivos java que dan origen al funcionamiento de cada una de las actividades. Una actividad corresponde a un componente principal de la aplicación que se asocia con una ventana de la interfaz gráfica donde el usuario puede interactuar. Estos archivos java se han organizado en cuatro carpetas de acuerdo a su funcionalidad y son las que se describen a continuación.

- Activities: Contiene los archivos de cada una de las actividades, donde se tiene el código fuente para que éstas interactúen de manera correcta con el usuario. Las actividades creadas son las que se muestran a continuación y son descritas con mayor profundidad más adelante.
  - ThirdActivity.java
  - UsersMeasureActivity.java
  - FirstAnalysisActivity.java
  - FirstAnalysisActivityII.java
- Adapters: Contiene los archivos de los adaptadores creados en la aplicación. Un adaptador es una clase que actúa como puente entre un AdapterView y los datos requeridos para esta vista. En otras palabras, el AdapterView es un tipo de vista de Android que permite mostrar un conjunto de datos cargados y procesados por un adaptador. Los archivos creados son los que se muestran a continuación y son descritos con mayor profundidad más adelante.
  - SecondPageAdapter.java
  - ThirdPageAdapter.java
  - SpinnerAdapter.java
- App: En esta carpeta se encuentra el archivo “MyApplication”, el cual contiene una clase que actúa como contexto en android para inicializar Realm como administrador de base de datos del proyecto.
- Models: En esta carpeta se encuentran los archivos que contienen las tablas que conforman el modelo de la base de datos. Éste consiste en dos tablas definidas a partir de clases y se encuentran en los archivos que se muestran a continuación.
  - WiFi.java

La carpeta “res” es aquella que contiene todos los recursos del proyecto y, de acuerdo al tipo de elemento, se dividen en cuatro carpetas principales que son las que se describen a continuación.

- Drawable: Esta carpeta contiene todas las imágenes utilizadas en el proyecto, además, contiene archivos que describen el diseño gráfico de ciertos elementos de la aplicación, como los bordes, el fondo y las listas desplegadas.
- Layout: Esta carpeta contiene todos los archivos XML que constituyen a cada una de las actividades formadas por distintos tipos de vistas. Los archivos XML creados son

los que se muestran a continuación, los cuales son descritos con mayor profundidad más adelante.

- Activity\_analysis1.xml
- Activity\_show\_analysis.xml
- Activity\_third.xml
- Activity\_users\_measure.xml
- Item\_page2\_analysis1.xml
- Item\_page3\_analysis1.xml
- Item\_spinner.xml
- Mipmap: Esta carpeta contiene el icono que aparece en el menú principal para distintas densidades de la pantalla, por lo que contiene esta misma imagen en distintas con distintas dimensiones.
- Values: Esta carpeta contiene tres archivos, uno para definir los colores utilizados, otro para los estilos, y el último para definir algunas cadenas de caracteres a usar en la aplicación y, respectivamente, son nombrados a continuación.
  - Colors.xml
  - Styles.xml
  - Strings.xml

La última carpeta a describir corresponde a “Gradle Scripts”, la cual contiene una serie de archivos que permiten compilar y construir la aplicación, sin embargo, el más importante es “build.gradle(Module:app)”, pues en éste se configuran todas las opciones de compilación [22]. Una de estas opciones es la versión sdk con que se compila la aplicación, para esto se define el parámetro “compileSdkVersion” y, en este caso, se utiliza la versión 29, lo cual permite que se tengan muchas funcionalidades. A pesar de que se defina una versión de compilación tan alta, también se define el parámetro “minSdkVersion”, el cual especifica la mínima versión de API que requiere la aplicación, es decir, ésta no puede ser instalada en versiones anteriores a la definida. Además de esto, también se configuran las opciones como la versión sdk más alta probada, la versión de la aplicación y las dependencias utilizadas, donde se indican todas las librerías que deben ser incluidas en el proyecto.

## 4.2. Flujo de Navegación

Cuando el usuario ingresa a la aplicación, se abre la pantalla de la figura 4.2(a), donde se le da la opción de ingresar la cantidad de personas que planean utilizar la red WiFi para conectarse a internet. Si el usuario oprime el botón “Listo” sin haber agregado un valor, aparece el mensaje que se observa en la figura 4.2(b) “No ha ingresado ningún valor”, el cual aparece siempre que se cumpla esta condición. Cuando el usuario hace click en la casilla, se despliega un teclado numérico, con la finalidad de orientar al lector para que ingrese un número, como se observa en la figura 4.2(c). Una vez que haya ingresado una cantidad, el usuario puede oprimir el botón para proseguir con la siguiente actividad.

En esta segunda actividad, se implementa una lista de vistas tipo **LinearLayout** a partir de un adaptador, la cual es tan extensa como la cantidad de usuarios ingresados en la actividad

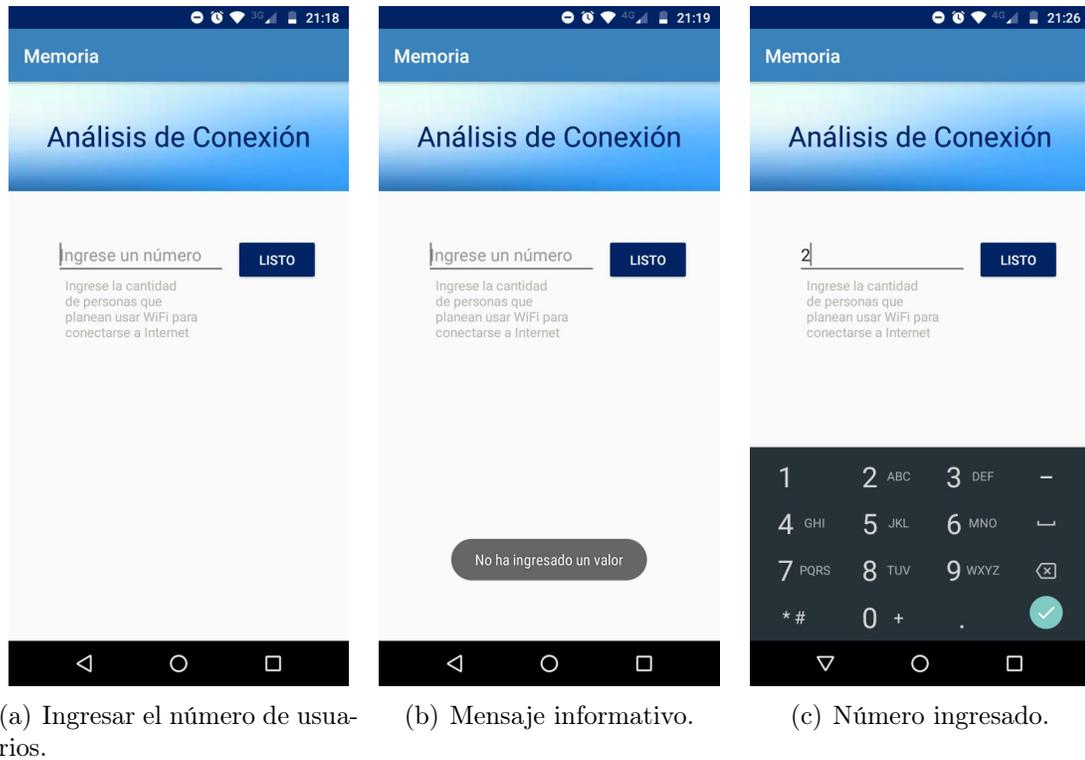


Figura 4.2: Flujo de la primera actividad o pantalla.

anterior, tal y como se observae en la figura 4.3(a). Una vista **LinearLayout** es un tipo de vista que alinea todos los elementos en una sola dirección, en este caso vertical. En cada una de estas vistas se pueden observar los siguientes elementos.

- Un icono que representa un usuario.
- Un nombre que hace referencia al número del usuario.
- Una barra para ingresar el nombre real del usuario.
- Un icono de advertencia para que el usuario lea las instrucciones.
- Unas breves instrucciones de lo que debe hacer antes de presionar el botón “medir”.
- Un botón “medir” con el cual se obtienen los parámetros de la señal a analizar.
- Un icono de vacío que representa que aún no se realiza la medición para ese usuario.

Un adaptador corresponde a un objeto que implementa una interfaz entre un conjunto de datos y una vista, de acuerdo a esto, su función es recuperar datos de un conjunto de datos y generar vistas a partir de estos [15]. El adaptador utilizado en esta actividad se llama “SecondPageAdapter.java” y sus funciones son las que se describen a continuación.

- Generar una lista de vistas **LinearLayout** utilizando la cantidad de usuarios ingresadas anteriormente.
- Recopilar los nombres ingresados en la casilla de texto. No importa si estos han sido actualizados, pues se guarda el último nombre ingresado.
- Al oprimir el botón “medir”, realiza todo el proceso de obtención y almacenamiento de datos descrito anteriormente.

Si el usuario oprime el botón medir sin haber ingresado el nombre respectivo, se muestra el mensaje de la figura 4.3(b) indicando que por favor se ingrese un nombre. Si el usuario oprime en la casilla para agregar un texto, se abre un teclado alfabético para que escriba el nombre del usuario que corresponda, lo cual se puede observar en la figura 4.3(c). Una vez que el nombre esté listo, se puede oprimir el botón “medir” para obtener los datos de ese usuario y, al finalizar este proceso, el botón cambia a “Actualizar” y el icono de vacío se transforma en un ticket, con la finalidad de indicarle a la persona que los datos ya han sido obtenidos, pero que puede volver a realizar esta medición en caso de error. Si no se ha realizado la medición para todos los usuarios y se oprime el botón analizar, aparece el texto que se observa en la figura 4.3(e), advirtiendo que faltan usuarios por medir.

En cuanto se hayan ingresado todos los nombres y se hayan realizado las mediciones respectivas, el usuario puede oprimir el botón “Analizar” para avanzar a la siguiente actividad, la cual se observa en la figura 4.4(a). El adaptador utilizado en esta pantalla se llama “SpinnerAdapter” y sus funciones son las que se describen a continuación.

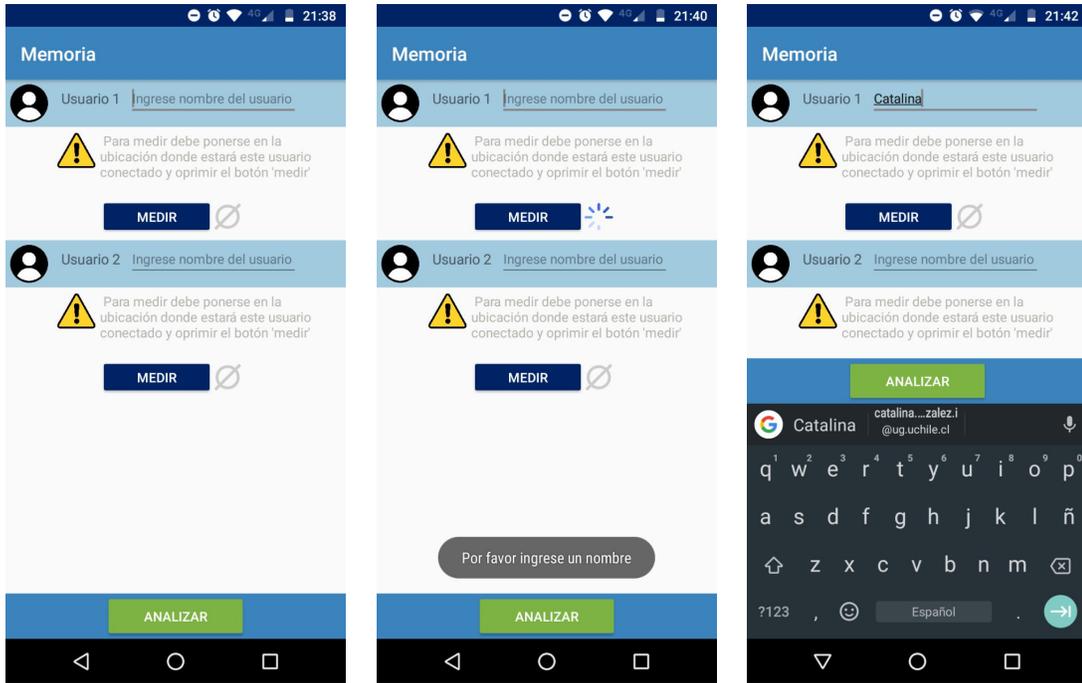
- Generar una lista desplegable para cada usuario.
- Recopilar la última opción seleccionada para cada uno de los usuarios.

Al hacer click en una lista desplegable, el usuario puede observar las opciones que se muestran en la figura 4.4(b), las cuales son aplicaciones conocidas por el usuario que representan servicios de entretenimiento y trabajo. Estas opciones pueden ser cambiadas a futuro por distintas opciones que abarquen una mayor cantidad de servicios, lo cual es sencillo de realizar en el código, pues los servicios son construidos como una clase, a la cual se le entregan los atributos del nombre de la aplicación, throughput necesario e icono. Cuando se hayan seleccionado las aplicaciones respectivas para cada usuario, se puede hacer click en el botón “Listo”, con lo cual se avanza a la cuarta y última actividad. Cuando se oprime este botón comienza un proceso de análisis entre los throughput máximos de los usuarios y los necesarios por las aplicaciones, lo cual se explica con mayor detalle en la siguiente sección.

La última actividad es la que se muestra en la figura 4.5, en la cual se utiliza el adaptador “ThirdPageAdapter” para generar vistas en forma de tarjetas de información y sus funciones son las que se describen a continuación.

- Compara el throughput máximo de los usuarios con respecto a los requeridos por la aplicación para determinar un porcentaje de uso de la red. Si es mayor al 100% se muestra una cruz y, si es menor, un ticket. Este análisis se describe con mayor detalle en la sección 4.5 “Análisis de Calidad de Servicio”.
- Generar una lista de cartas con la información de cada usuario, como el nombre, la aplicación que desea usar y si ésta puede ser utilizada según los requerimientos de throughput.

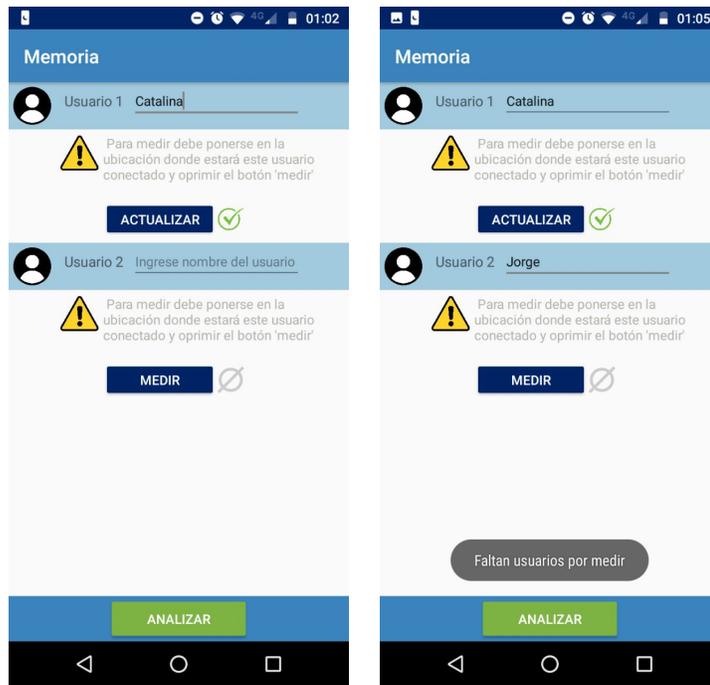
La ventaja de entregar la información en tarjetas es que éstas pueden ser presionadas para entregar información más detallada al usuario, ya sea entregando un pequeño mensaje o abriendo una nueva vista.



(a) Lista de vistas para cada usuario.

(b) Primer mensaje informativo.

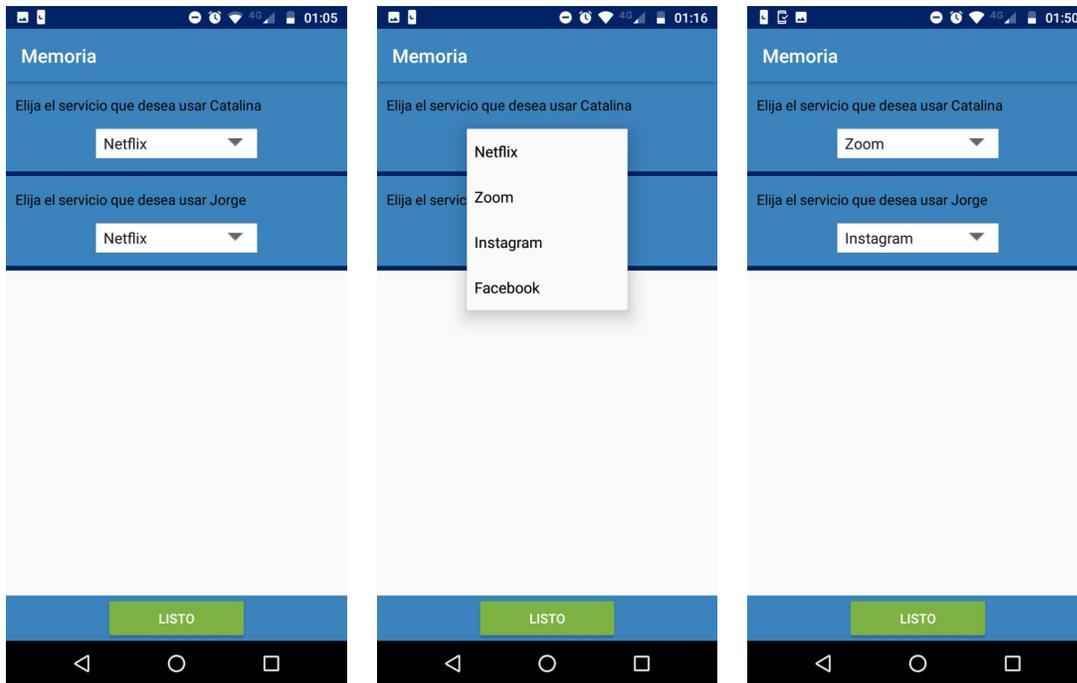
(c) Nombre ingresado.



(d) Datos del primero usuario medidos.

(e) Segundo mensaje informativo.

Figura 4.3: Flujo de la segunda actividad o pantalla.



(a) Lista de vistas para cada usuario. (b) Opciones de la lista desplegable. (c) Aplicaciones seleccionadas.

Figura 4.4: Flujo de la tercera actividad o pantalla.

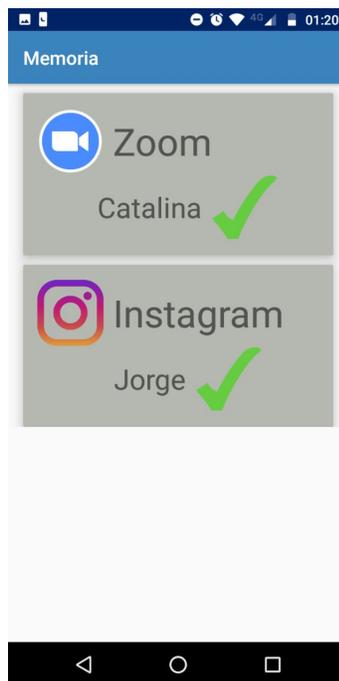


Figura 4.5: Flujo de la cuarta actividad o pantalla.

### 4.3. Obtención de Datos

Para comenzar el análisis de calidad de servicio en una red WiFi, se necesitan datos que puedan caracterizar el comportamiento de ésta bajo ciertos estándares de funcionamiento.

De acuerdo a esto, se procede a realizar un código en java que permita obtener datos de la conexión WiFi que tenga el dispositivo en un momento y lugar determinado. Sin embargo, para esto se necesita que el usuario tenga conexión a internet, esté conectado a una red WiFi y que éste conceda a la aplicación los permisos necesarios.

Para determinar si se tiene una buena conexión a internet, es necesario considerar los parámetros de latencia y throughput, sin embargo, al obtener el valor de este último por medio del dispositivo móvil, puede que no corresponda al valor máximo que pueda alcanzar en ese momento y lugar determinado. Por esta razón, se plantea utilizar el parámetro RSSI o nivel de señal para calcular el valor del throughput máximo, sin embargo, este dato por sí solo puede variar mucho entre dispositivos, además de que no considera la interferencia de otras señales ni la concurrencia con otros usuarios de la misma red, por lo que se decide complementar con el SINR. Cabe destacar que el throughput máximo estimado viene limitado por el router dado el protocolo que esté utilizando, es decir, el throughput máximo puede ser más bajo del estimado según la tecnología utilizada. Además de estos dos parámetros, se decide obtener una lista con las redes WiFi que detecta el dispositivo y la información de cada una, con la finalidad de cuantificar la interferencia presente. Sin embargo, de esta lista sólo sirven aquellas redes que estén utilizando el mismo canal de la red del usuario o aquellas que presentan un traslape, por lo que es importante obtener el canal de frecuencia para poder comparar con las otras redes y determinar si corresponden o no a interferencia. Finalmente, para poder identificar los datos conseguidos, se decide obtener la dirección MAC del AP o router, el nombre de la red y el IMEI del dispositivo, el cual es importante para identificar el dispositivo del usuario.

Al iniciar este análisis, el usuario debe ingresar la cantidad de personas que van a utilizar la red, como se muestra en la pantalla de la figura 4.2(a). Luego, al oprimir el botón “Listo”, se procede a la actividad de mediciones, con lo cual comienza el proceso de validación que se observa en la figura 4.8. En este proceso se verifica, en primer lugar, si el dispositivo tiene conexión a internet, para lo cual se utiliza la función `context.getSystemService()` que retorna un objeto “ConnectivityManager” que contiene información sobre el estado de la conectividad de la red. Luego, a éste se le extrae el objeto “NetworkInfo” a partir de la función `getActiveNetworkInfo()`, el cual describe el estado de la interfaz de red. En el caso de que el dispositivo no tenga conexión, el objeto “NetworkInfo” retorna nulo, por lo que se comprueba esta igualdad para validar que el usuario tenga conectividad. Si el objeto es nulo, se procede a mostrar el mensaje de la figura 4.6 para informar que no hay conexión a internet, en caso contrario, se procede a validar la conexión WiFi.

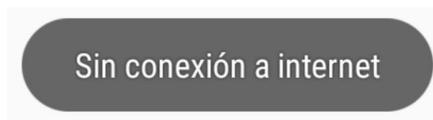


Figura 4.6: Mensaje al usuario sobre su estado de conectividad.

Para verificar la conexión WiFi del dispositivo se utiliza el objeto “NetworkInfo” obtenido de la validación anterior, al cual se le extrae la información del tipo de conexión a partir de la función `getType()`. Si este dato es igual a una conexión tipo `ConnectivityManager.TYPE_WIFI`, entonces se da inicio al proceso de medición, en el caso contrario se

muestra el mensaje de la figura 4.7 para informar que el usuario no está conectado a una red WiFi.

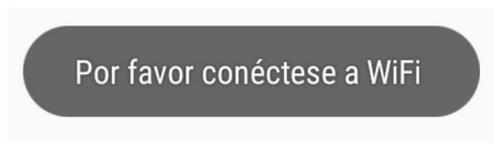


Figura 4.7: Mensaje al usuario sobre su conexión WiFi.

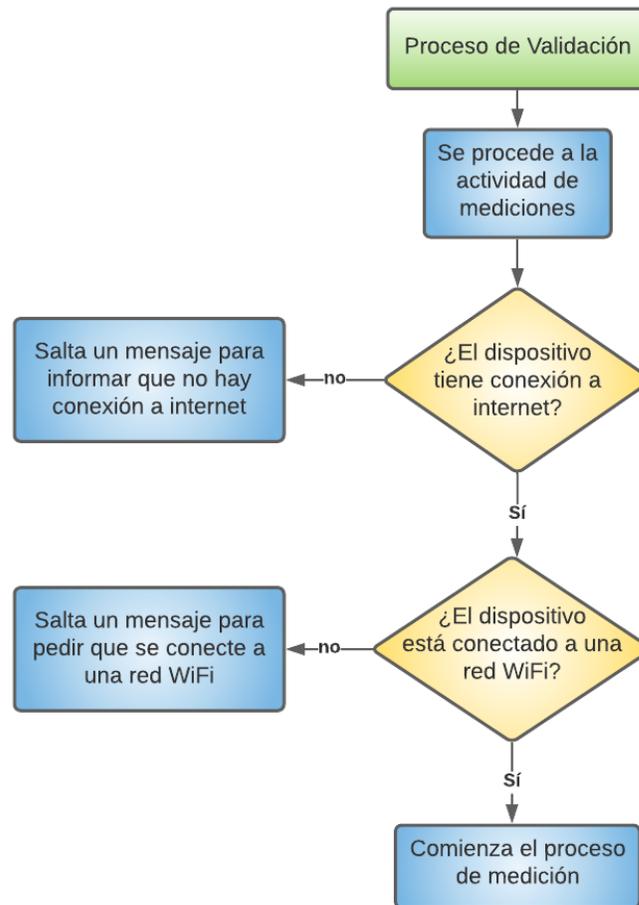
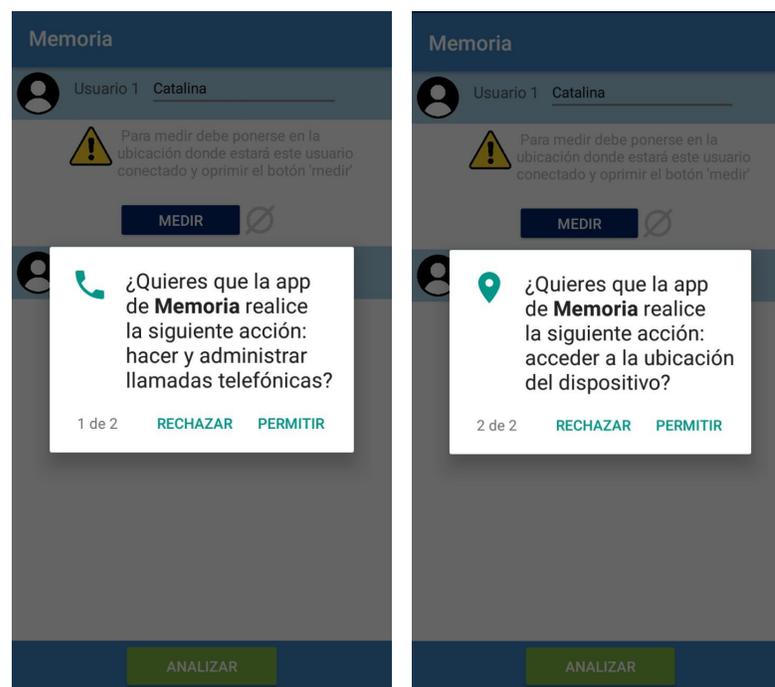


Figura 4.8: Diagrama general del proceso de validación.

Una vez que haya terminado el proceso de validación, la aplicación puede avanzar a la siguiente actividad, la cual se observa en la figura 4.3(a), en donde se tiene un ítem para cada usuario con una casilla para ingresar un nombre y un botón “medir”. Cuando el usuario oprime este botón, comienza el proceso de petición de permisos que se observa en la figura 4.11, en el cual se pide al usuario que otorgue los permisos necesarios para obtener información confidencial o utilizar funciones del dispositivo. Sin embargo, estos permisos se piden sólo la primera vez que la aplicación es utilizada o si el usuario ha modificado la configuración de los permisos manualmente.

Para determinar si el usuario ha concedido o no los permisos se utiliza la función `ActivityCompat.checkSelfPermission(contexto, permiso)`, la cual tiene como parámetros de entrada el contexto y el permiso a evaluar. En este caso, se requiere un permiso para obtener información del teléfono y otro para acceder a la ubicación de éste y, para referirse a estos, se utilizan los códigos `Manifest.permission.READ_PHONE_STATE` y `Manifest.permission.ACCESS_COARSE_LOCATION` respectivamente. El estado inicial de estas variables es de `PERMISSION_DENIED` o petición denegada, pues el usuario no ha aceptado aún la solicitud. Entonces, al hacer esta comparación y ver que los permisos no tienen un estado `PERMISSION_GRANTED` o petición concedida, se realizan las preguntas de las figuras 4.9(a) y 4.9(b) a partir de la función `requestPermissions()`. Ésta tiene como parámetros de entrada una lista con los permisos a solicitar y un código numérico que permite confirmar el estado de la petición.

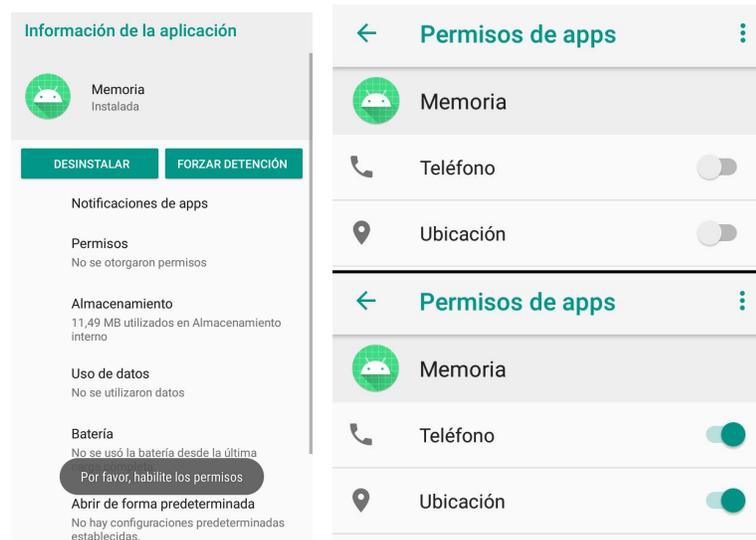


(a) Permiso para obtener informa- (b) Permiso para acceder a la ubi-  
ción del teléfono. cación.

Figura 4.9: Solicitudes de permiso para acceder a información del dispositivo.

Como se puede observar en la figura 4.9, las solicitudes hacen referencia a hacer y administrar llamadas telefónicas y a acceder a la ubicación del dispositivo, sin embargo, de acuerdo a lo mencionado anteriormente, los datos a obtener no tienen relación directa con estos temas. Esto se debe a que, para obtener el IMEI y un escaneo de las redes WiFi de alrededor, se requieren permisos relacionados con información del dispositivo y del ambiente en la posición en que éste se encuentra. Por ello, se requieren los permisos `READ_PHONE_STATE` y `ACCESS_COARSE_LOCATION`, los cuales forman parte de los grupos de permisos relacionados al teléfono y a la ubicación respectivamente y, de acuerdo a esto, android muestra en el cuadro de diálogo la pregunta relacionada a cada grupo en vez de hacer una tan específica.

Cuando aparecen estos cuadros, el usuario tiene las opciones de conceder o denegar los permisos, pero si se rechaza la petición, la aplicación no puede funcionar desde este punto en adelante, pues el proceso de obtención de datos no se puede llevar a cabo. Por esto, si el usuario insiste en oprimir el botón “medir”, aparece un mensaje pidiendo que se habiliten los permisos y se redirige la aplicación a las configuraciones para que el usuario los otorgue manualmente, tal y como se observa en la figura 4.10.



(a) Información de la aplicación. (b) Habilitar los permisos manualmente.

Figura 4.10: Proceso de redirigir la aplicación a configuraciones.

Cuando los permisos hayan sido concedidos, el usuario puede volver a la aplicación para continuar con el proceso y, finalmente, oprimir el botón “medir” sin problemas. Una vez que se hayan validado todos los permisos, se ejecuta el código para obtener los datos, con lo cual cambia el texto del botón a “actualizar” y el icono que lo acompaña, lo que se puede observar en la figura 4.3(d). En el caso de que el usuario se haya equivocado de posición, éste puede oprimir el botón “actualizar” cuantas veces quiera para que se ejecute nuevamente el código anterior, con la finalidad de que los datos obtenidos sean los correspondientes. Si está todo correcto, se puede proseguir con el resto de los usuarios hasta finalizar el proceso de obtención de datos.

A estas alturas, el proceso de obtención de datos se ejecuta sin problemas, sin embargo, está compuesto por varios pasos a seguir, los cuales se observan en la figura 4.12, por lo que es importante hacer una descripción más detallada de este proceso.

Una vez que se haya completado el proceso de validación, hay que comprobar que el usuario a medir tiene un nombre ingresado al cual corresponder los datos. En caso de que no tenga un nombre, se muestra un mensaje indicando que se ingrese uno, de lo contrario, se sigue con el proceso y se procede a extraer un objeto “WifiManager” a partir de la función `getSystemService(Context.WIFI_SERVICE)`, el cual contiene la información de la conectividad WiFi. Luego, se utiliza la función `startScan()` para solicitar un escaneo de las redes WiFi percibidas por el dispositivo. Para visualizar los resultados se utiliza la función

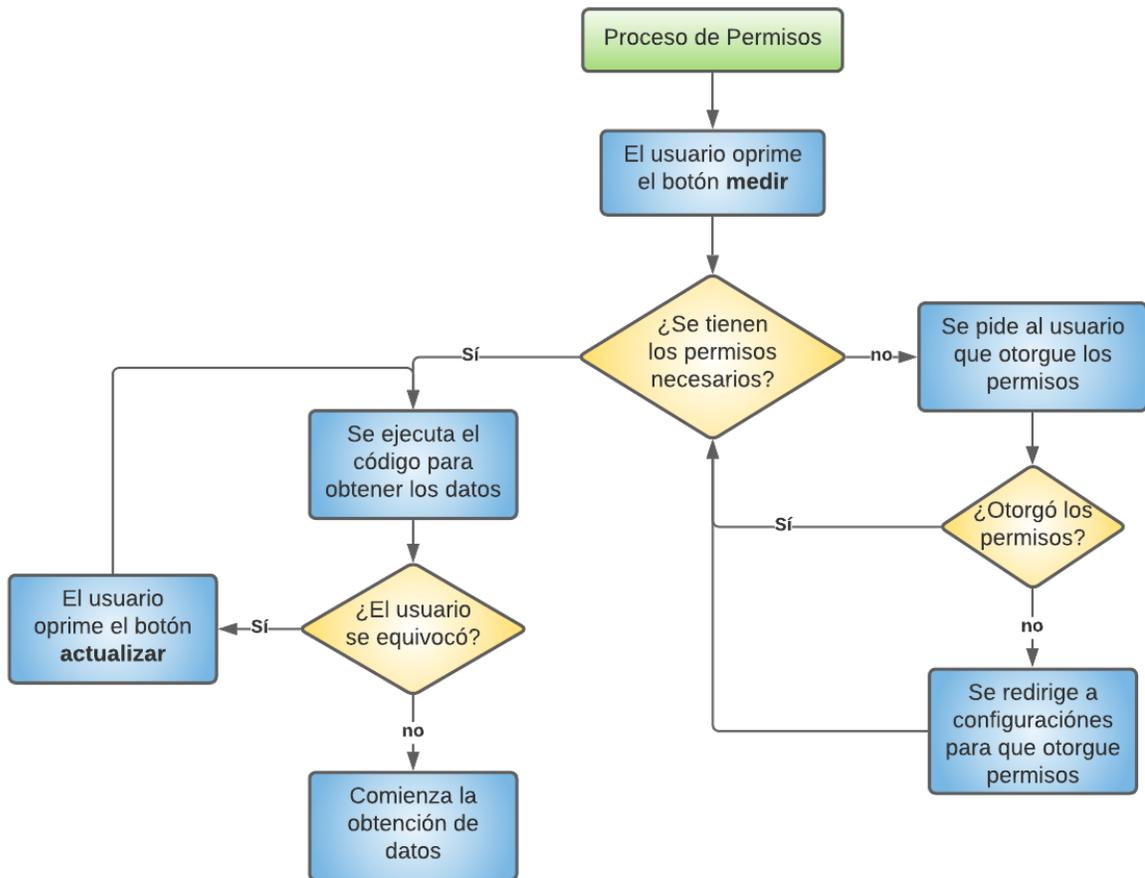


Figura 4.11: Diagrama general del proceso de medición.

`getScanResults()`, con lo cual se obtiene una lista de las redes WiFi encontradas con la información detallada de cada una, así como el nombre de la red, la dirección MAC del router, el RSSI, el canal y el ancho de banda.

Luego de realizar este escaneo, se procede a extraer información de la conexión WiFi del usuario utilizando el método `getConnectionInfo()`, el cual retorna un objeto “WifiInfo” que contiene información de la red a la que se encuentra conectado el dispositivo. Con esto se puede obtener el canal que está ocupando el usuario, lo cual se muestra como el valor de la frecuencia central en Hz, la dirección MAC, el RSSI y la latencia.

Además de estos datos, se procede a extraer datos de identificación de la red del usuario y del dispositivo utilizado. El nombre de la red se obtiene a partir del objeto “NetworkInfo”, mientras que el IMEI se consigue del objeto “WifiInfo”

Una vez que se hayan obtenido todos los datos, se procede a almacenarlos en la base de datos y a realizar el análisis de calidad de servicio calculando el throughput máximo del usuario. Estos pasos se explican con mayor detalle en las secciones siguientes.

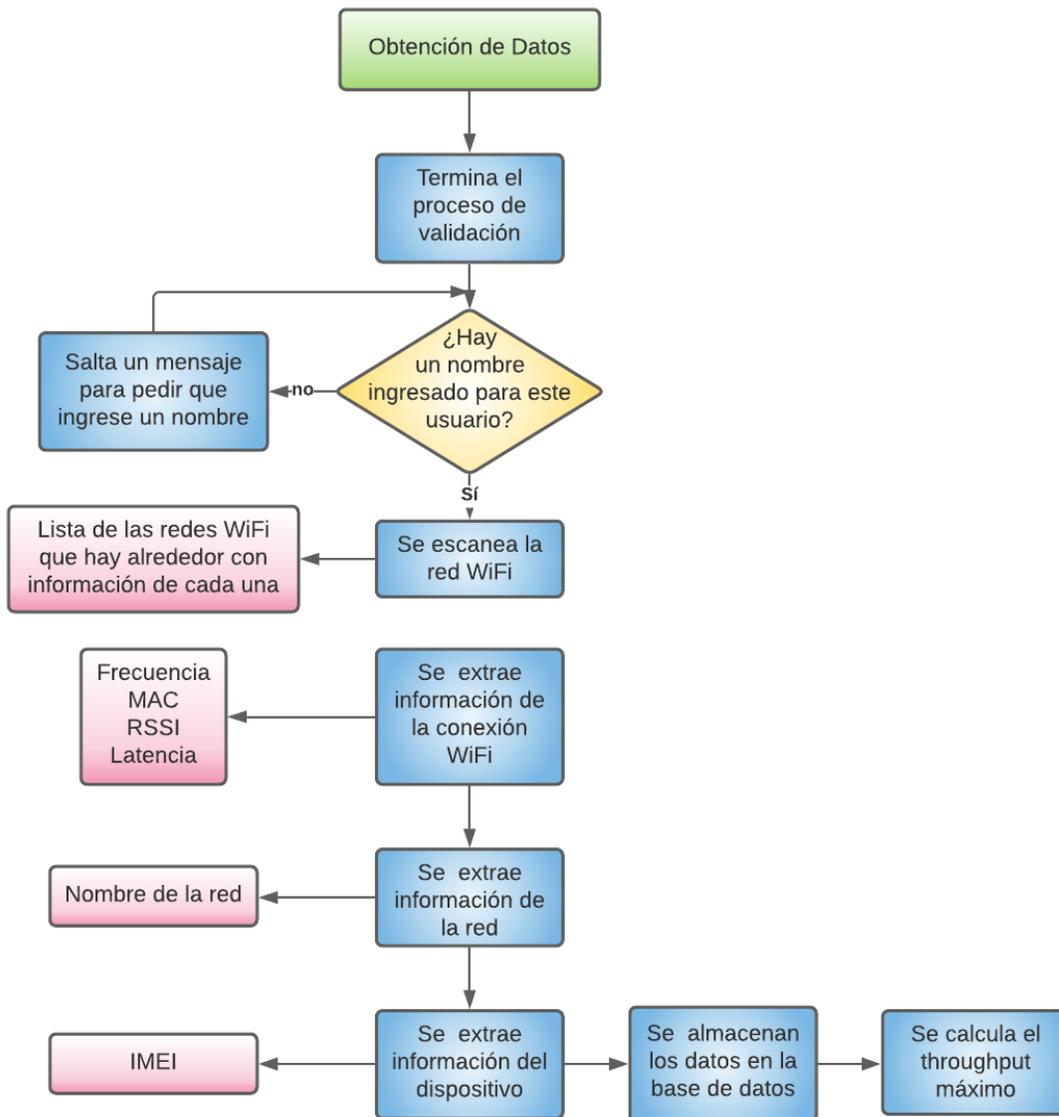


Figura 4.12: Diagrama general del proceso de obtención de datos.

## 4.4. Conexión con la Base de Datos

En la sección anterior se describe el proceso en que la aplicación obtiene una cantidad de datos que caracterizan la señal WiFi recibida por un usuario en específico. Estos datos van a variar de acuerdo a la posición del usuario con respecto al AP y a las interferencias presentes alrededor, por lo que las muestras van a ser distintas entre sí. De acuerdo a esto, es importante almacenar los datos para poder compararlos entre sí o para realizar estadísticas a largo plazo, con la finalidad de conocer el comportamiento de la red de manera general.

Para almacenar los datos se tiene la opción de utilizar un almacenamiento en una base de datos externa, con lo cual se puede obtener y escribir datos en ella a partir de una API REST como interfaz entre sistemas. Sin embargo, la cantidad de datos que requiere la aplicación para hacer los análisis no es tan grande, pues los datos importante que deben analizarse

son los obtenidos en un mes aproximadamente. De acuerdo a esto, se decide utilizar un almacenamiento interno del teléfono, para lo cual se escoge Realm como gestor de base de datos, el cual se caracteriza por su simplicidad y velocidad de procesamiento, pues utiliza C++ como núcleo.

Antes de utilizar Realm en la aplicación, hay que realizar un proceso para inicializar este generador de base de datos, para lo cual se crea una subclase de “Application”. En esta subclase se inicializa Realm bajo el contexto de la aplicación creada, luego, se procede a crear una instancia de Realm a partir del método **RealmConfiguration** con la configuración que viene por defecto. Para que Realm sea correctamente inicializado, se debe declarar esta subclase en el archivo Manifest, con el objetivo de que la aplicación sepa dónde y cómo sincronizar el almacenamiento de datos.

```
0 public class MyApplication extends Application {
1     @Override
2     public void onCreate() {
3         super.onCreate();
4         Realm.init(getApplicationContext());
5         setUpRealmConfig();
6
7     private void setUpRealmConfig() {
8         RealmConfiguration config = new RealmConfiguration.Builder().
9             deleteRealmIfMigrationNeeded().build();
10        Realm.setDefaultConfiguration(config);}}
```

Los modelos de datos en Realm se definen utilizando clases java que son una subclase de RealmObject, tal y como se puede observar en el siguiente segmento de código, donde se define el modelo de datos WiFi como objeto de Realm. Lo primero que se debe definir en la clase son los campos que componen el modelo, donde se debe indicar el tipo de dato al que corresponde e indicar las propiedades de llave primaria, llaves foráneas, campos obligatorios, etc. Como se puede observar en el código, en la línea 1 se está indicando que el siguiente campo, el id, corresponde a una llave primaria, mientras que en las líneas 3 y 6 se define que los siguientes campos, el IMEI y el MAC, son obligatorios, pues corresponden a datos de identificación. Luego de esto, se procede a inicializar las variables según corresponda, por ejemplo, el campo “createdAt” corresponde a la fecha en formato dd-MM-yyyy HH:mm:ss, por lo que se obtiene este valor cada vez que la clase es llamada, además, el “id” está formado por la concatenación de la fecha y el IMEI del dispositivo.

```
0 public class WiFi extends RealmObject {
1     @PrimaryKey
2     private String id;
3     @Required
4     private String IMEI;
5     private Date createdAt;
6     @Required
7     public String MAC;
8     public String red;
9     public int wifiNets;
10    public int frequency;
```

```

11 public int wifiRSSI;
12 public double latency;
13
14 public WiFi(String IMEI){
15     DateFormat df = new SimpleDateFormat("dd-MM-yyyy HH:mm:ss");
16     this.IMEI = IMEI;
17     this.createdAt = Calendar.getInstance().getTime();
18     this.id = IMEI + df.format(createdAt);
19     this.MAC = "";
20     this.red = "";
21     this.wifiNets = 0;
22     this.frequency = 0;
23     this.wifiRSSI = 0;
24     this.latency = 0.0;}}

```

De acuerdo al código anterior, la estructura de la tabla de datos de la red WiFi creada es la que se muestra en la figura 4.13, donde la llave primaria corresponde al campo “id” correspondiente al “IMEI + Fecha y hora”. Luego de esto, se agregan los datos correspondiente a identificación del usuario y de la red, los cuales sería el IMEI, la fecha y la hora y a dirección MAC. Finalmente, se agregan los datos encargados de caracterizar la calidad de la señal, como el canal, el número de redes WiFi detectadas, el RSSI, la latencia y el porcentaje de paquetes perdidos. Cabe destacar que se pueden agregar todos los datos que se deseen a esta tabla, sólo se requiere modificar la subclase descrita anteriormente.

WiFi
IMEI + Fecha y hora
IMEI
Fecha y hora
MAC
Canal
N° Redes disponibles
RSSI
Latencia
Porcentaje de paquetes perdidos

Figura 4.13: Tabla de datos de WiFi.

Para realizar operaciones de lectura y escritura en la base de datos en Realm, se debe comenzar y finalizar el proceso con una transacción. En este caso, una transacción corresponde

a una interacción compuesta de procesos, los cuales deben aplicarse uno después de otro y sin errores para que la transacción sea finalizada. De acuerdo a esto, para agregar datos a la base de datos se crea una función llamada `createNewWifi()`, a la cual se le ingresan los datos obtenidos y es la que se observa en el siguiente segmento de código.

```
0 public void createNewWifi(String IMEI, String MAC, String red, int nets
  , int freq, int rssi, double lat) {
1     realm.beginTransaction();
2     Wifi wifi = new Wifi(IMEI);
3     wifi.MAC = MAC;
4     wifi.red = red;
5     wifi.wifiNets = nets;
6     wifi.frequency = freq;
7     wifi.wifiRSSI = rssi;
8     wifi.latency = lat;
9     realm.copyToRealm(wifi);
10    realm.commitTransaction(); }
```

Como se puede observar, la función comienza en la línea 1 con un llamado a comenzar una transacción, luego, los datos son agregados como atributos de la subclase y, cuando todos los procesos hayan sido completados, se procede a copiar los datos en la base de datos en la línea 9 y se finaliza la transacción. Finalmente, esta función puede ser llamada en cualquier parte del código cada vez que se quiera guardar datos en la base de datos.

## 4.5. Análisis de Calidad de Servicio

En los capítulos anteriores se describe de forma detallada el funcionamiento de la aplicación, sin embargo, falta explicar todo el motor logístico que le da sentido y veracidad a ésta. Por consiguiente, en el presente capítulo se procede a describir la teoría detrás del análisis de calidad de servicio que aprecia el usuario.

Antes de comenzar con una descripción más detallada, se debe explicar de forma general el análisis que realiza esta aplicación por el momento, el cual se realiza considerando sólo a aquellos usuario con un tipo de conexión por medio de WiFi, es decir, no se toman en cuenta aquellos conectados por ethernet. En la figura 4.14 se observa el plano de una casa, en donde hay un router y dos personas conectadas a él, las cuales van a tener distinto nivel de señal dependiendo de su ubicación en el hogar, por ejemplo, “user 1” está más cerca que “user 2”, por lo que debería tener un nivel RSSI más alto. De acuerdo a esta diferencia en la señal recibida, ambos usuarios van a tener un throughput máximo distinto acorde a su RSSI, por lo que nace la pregunta si ambas personas pueden usar las aplicaciones que desean en la ubicación en la que están de acuerdo a las limitaciones de la red. A partir de esta pregunta, la aplicación busca obtener el máximo throughput que puede alcanzar cada uno de los usuarios, con tal de comparar con los requerimientos de la aplicación que desean utilizar y, finalmente, verificar si la red permite utilizar estos servicios de acuerdo al uso del canal. El objetivo de que el usuario tenga esta información, es que puedan utilizar las aplicaciones que desean sin tener problemas, por ejemplo, si ambos usuarios necesitan usar zoom para una reunión importante, pueden determinar previamente si tendrán problemas de conexión, si sólo uno de

ellos puede conectarse o si simplemente deben cambiar de ubicación. Cabe destacar que esta aplicación es de uso domiciliario, por lo que se espera una cantidad de usuarios razonable.

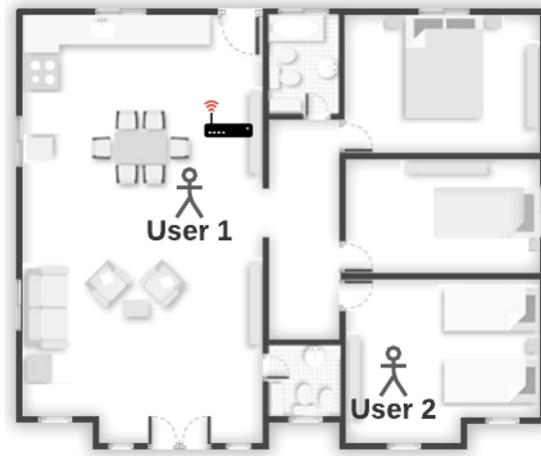


Figura 4.14: Descripción gráfica del análisis.

De acuerdo al capítulo 2, el throughput corresponde a la tasa en que la información es transferida, siendo el máximo valor el que utiliza la capacidad del canal más cercana a aquella dada por el teorema de Shannon-Hartley. Este teorema establece la máxima cantidad de datos que pueden ser transmitidos sobre un enlace con un ancho de banda específico y que está sometido a ruido, por ende, la capacidad viene dada por una relación matemática entre el ancho de banda, la potencia de la señal y la potencia del ruido [25]. De acuerdo a esto, al medir el throughput del usuario en cierta posición, no necesariamente se está midiendo su máximo valor, pues la capacidad del canal puede o no ser la máxima. Por esta razón, se decide utilizar otros parámetros de la señal para obtener cuantitativamente el valor del throughput máximo que puede alcanzar el usuario, y este procedimiento es que se describe a continuación.

Como se menciona anteriormente, cuando el usuario está en la segunda actividad debe presionar el botón “medir” cada vez que ingresa un nombre, con lo cual se procede a realizar las mediciones de los parámetros necesarios para obtener una estimación, lo más exacta posible, del throughput máximo del dispositivo conectado a WiFi en esa ubicación en particular. El primer parámetro a considerar es el nivel de señal o RSSI el cual tiene una relación directa con el throughput máximo, es decir, a mayor RSSI mayor throughput, lo cual se puede observar en el gráfico de la figura 4.15 para una tarjeta de red específica funcionando en WiFi 802.11g.

El problema con esta relación es que, para un valor de RSSI, la estimación del throughput puede variar en un rango demasiado amplio, incluso para el mismo AP, ya sea por las interferencias presentes o por las tecnologías de los dispositivos utilizados en la red WiFi, por lo que no se puede entregar una relación cuantitativa concluyente. Otro problema que se logra apreciar de esta relación, es que el RSSI no considera las otras señales recibidas, las cuales pueden provocar cierta interferencia, es decir, pueden ser consideradas como ruido. Éste es un factor importante para determinar el throughput final, por lo que se decide utilizar el SINR calculado a partir de la diferencia entre la señal recibida y el ruido. De acuerdo a esto, el análisis se basa en la tabla de la figura 4.16, utilizada por Cisco, que relaciona MCS, SINR y RSSI para indicar el throughput máximo esperado, donde también se considera la

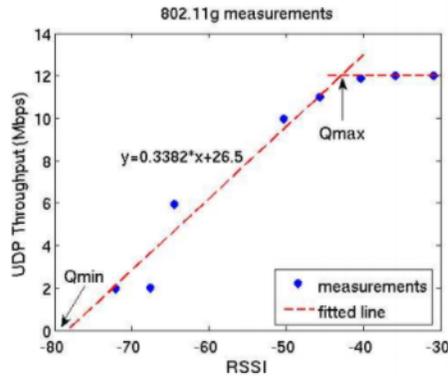


Figura 4.15: Relación entre throughput y RSSI para una tarjeta intel 2915ABG en 802.11g. Figura obtenida de [26].

modulación, el ancho del canal y el tiempo de espera entre transmisiones.

HT MCS	VHT MCS	Modulation	Coding	20MHz				40MHz				80MHz				160MHz			
				Data Rate		Min. SNR	RSSI	Data Rate		Min. SNR	RSSI	Data Rate		Min. SNR	RSSI	Data Rate		Min. SNR	RSSI
				800ns	400ns			800ns	400ns			800ns	400ns			800ns	400ns		
2 Spatial Streams																			
8	0	BPSK	1/2	13	14.4	2	-82	27	30	5	-79	58.5	65	8	-76	117	130	11	-73
9	1	QPSK	1/2	26	28.9	5	-79	54	60	8	-76	117	130	11	-73	234	260	14	-70
10	2	QPSK	3/4	39	43.3	9	-77	81	90	12	-74	175.5	195	15	-71	351	390	18	-68
11	3	16-QAM	1/2	52	57.8	11	-74	108	120	14	-71	234	260	17	-68	468	520	20	-65
12	4	16-QAM	3/4	78	86.7	15	-70	162	180	18	-67	351	390	21	-64	702	780	24	-61
13	5	64-QAM	2/3	104	115.6	18	-66	216	240	21	-63	468	520	24	-60	936	1040	27	-57
14	6	64-QAM	3/4	117	130.3	20	-65	243	270	23	-62	526.5	585	26	-59	1053	1170	29	-56
15	7	64-QAM	5/6	130	144.4	25	-64	270	300	28	-61	585	650	31	-58	1170	1300	34	-55
	8	256-QAM	3/4	156	173.3	29	-59	324	360	32	-56	702	780	35	-53	1404	1560	38	-50
	9	256-QAM	5/6			31	-57	360	400	34	-54	780	866.7	37	-51	1560	1733.3	40	-48

Figura 4.16: Variación del throughput para 802.11ac en relación a varios parámetros. Figura obtenida de [3].

Antes de comenzar a analizar la tabla hay que explicar el concepto de MCS para comprender mejor la relación establecida entre SINR y throughput. Según [24], el MCS o esquema de modulación y codificación corresponde a una lista de valores que permite mapear la señal a un SNRI del canal, para lo cual se proporciona una combinación de los parámetros: número de “spatial stream” (SM), tipos de modulación y tasa de codificación, además, MCS tiene una asignación para 802.11ac (VHT) y 802.11n (HT). De manera aclaratoria, se debe mencionar que SM corresponde a la mínima distancia entre distintas antenas de la señales a transmitir para cumplir con el desacople entre ellas , con lo cual se aumenta el rendimiento.

La tabla de la figura 4.16, muestra en las primeras cuatro columnas los valores de MCS con 2 “Spatial Stream” para VHT y HT con sus respectivas combinaciones de tipo de modulación y tasa de codificación. Luego, se muestra una división según el ancho del canal, dándose las opciones de 20 [MHZ], 40 [MHZ], 80 [MHZ] y 160 [MHZ] y, para cada una de estas opciones se muestran los valores de RSSI, mínimo SINR y throughput, éste último para dos intervalos de espera entre transmisiones, 800 [ns] y 400 [ns].

De acuerdo al gráfico de la figura 4.17, el estándar que tiene mayor cobertura de manera

global, y que se encuentra en continuo crecimiento, es el 802.11ac para el 2016, por lo que se utiliza la columna de VHT MCS como referencia, proyectando que este estándar es el más utilizado actualmente en la configuración inicial del AP. A pesar de que la aplicación es capaz de obtener el estándar utilizado, no muchos dispositivos móviles son capaces de realizar esta función, por lo que se decide simplificar el problema eligiendo el estándar con mayor cobertura. Además de esto, se consideran 2 “Spatial Stream”, es decir, dos flujos de datos, pues la mayoría de los dispositivos de hoy en día son capaces de dos flujos [16], es decir, se está considerando el mejor caso posible dentro de las opciones que abarcan la mayor cantidad de dispositivos. Otro factor que queda a elección es el ancho de banda; dado que éste es un factor variable y muy importante para el resultado final, se decide obtener el valor del ancho de banda utilizado por el AP para que sea un dato verosímil, por lo tanto, se ocupan las cuatro columnas de 20 [MHz], 40 [MHz], 80 [MHz] y 160 [MHz] para realizar el análisis final. Por último, el intervalo entre transmisiones escogido es de 800 [ns], pues éste es altamente compatible en el procesamiento de transmisores y receptores y es el valor que viene por defecto en la configuración de 802.11 [20]. Al considerar estos parámetros como fijos, el análisis a realizar es estático para estas tecnologías, por lo que los throughput calculados pueden llegar a ser mayores al considerar técnicas más avanzadas, sin embargo, el resultado final va a ser realista para este caso.

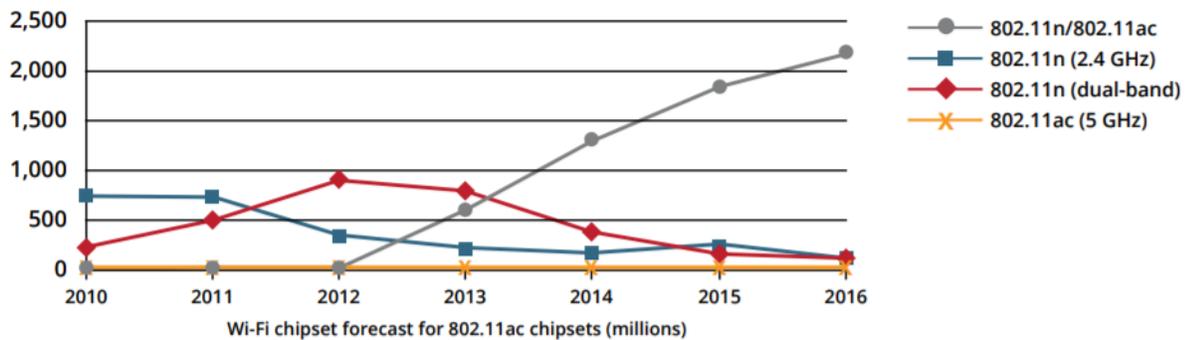


Figura 4.17: Uso de circuitos integrados de estándares 802.11. Figura obtenida de [4].

A partir de lo dicho anteriormente, en la tabla 4.1 se pueden observar las relaciones utilizadas entre SINR y throughput máximo para realizar el análisis de calidad de servicio. Cabe destacar que la casilla vacía corresponde a una combinación no válida para el proceso, pues los bits codificados no se dividen limpiamente en una serie de bloques.

Para ocupar los datos de la tabla 4.1 en la aplicación, se crea la función **getThroughput()**, la cual retorna una lista de valores tipo double, los que corresponden al throughput obtenido a partir del SINR determinado para cada señal percibida como ruido. Los parámetros de entrada para esta función son una lista de los valores SINR de las señales percibidas como posible ruido y el ancho de banda utilizado por la señal recibida. Si se observan las columnas de SINR de la tabla 4.1, se puede apreciar que son valores discretos, lo cual significa un problema si, por ejemplo, se quiere saber el throughput para un SINR de 6 en 20[MHz], por lo que se procede a proyectar estos datos como funciones continuas y lineales. Para esto, se utiliza la ecuación 4.1 llamada “Ecuación de la Recta”, donde la variable **X** corresponde a los valores de SINR e **Y** a los valores de throughput. En los casos extremos, como lo son aquellos valores de SINR cercanos a 0 o muy grandes, se utiliza esta ecuación considerando

MCS	20 [MHz]			40 [MHz]			80 [MHz]			160 [MHz]		
	D. rate [Mbps]	SINR	RSSI									
0	13	2	-82	27	5	-79	58.5	8	-76	117	11	-73
1	26	5	-79	54	8	-76	117	11	-73	234	14	-70
2	39	9	-77	81	12	-74	175.5	15	-71	351	18	-68
3	52	11	-74	108	14	-71	234	17	-68	468	20	-65
4	78	15	-70	162	18	-67	351	21	-64	702	24	-61
5	104	18	-66	216	21	-63	468	24	-60	936	27	-57
6	117	20	-65	243	23	-62	526.5	26	-59	1053	29	-56
7	130	25	-64	270	28	-61	585	31	-58	1170	34	-55
8	156	29	-59	324	32	-56	702	35	-53	1404	38	-50
9		31	-57	360	34	-54	780	37	-51	1560	40	-48

Tabla 4.1: Valores de throughput máximo escogidos a partir de la figura 4.16.

que el throughput es 0 para  $SINR = 0$  y es máximo para valores de SINR altos.

$$(Y - Y_1) = \frac{(Y_2 - Y_1)}{(X_2 - X_1)}(X - X_1) \quad (4.1)$$

Los valores de throughput máximo, para cada ancho de banda y utilizando el estándar 802.11ac, son los que se observan en la figura 4.18. Dado que en este análisis se consideran dos “Spatial Stream”, los valores de la figura que se deben considerar son los de la fila correspondiente a esta opción.

# Spatial Streams	Channel Width			
	20 MHz	40 MHz	80 MHz	160 MHz
1	86 Mbps	200 Mbps	433 Mbps	866 Mbps
2	173 Mbps	400 Mbps	866 Mbps	1.73 Gbps
3	288.9 Mbps	600 Mbps	1.3 Gbps	2.34 Gbps
4	346.7 Mbps	800 Mbps	1.73 Gbps	3.46 Gbps

Figura 4.18: Máximo valor de throughput alcanzado según ancho de banda en 802.11ac. Figura obtenida de [23].

De acuerdo a lo descrito anteriormente, en la función `getThroughput()` se itera sobre la lista de valores SINR, luego, se pregunta sobre el valor del ancho de banda, con la finalidad de encontrar el intervalo que le corresponda a dicho valor para aplicar la función que entrega el throughput. En el siguiente segmento de código, se muestra el caso de las funciones correspondientes a cada intervalo de SINR para un ancho de banda de 40 [MHz] (representado por el número 1), pues para los otros casos la lógica es la misma, pero con distintas funciones lineales. Según la tabla 4.1, el último intervalo corresponde a [32, 34] y su ecuación correspondiente es  $throughput = 18,0 * sinr - 252$ , sin embargo, en el código se utiliza [32, 36,2222] y se debe a que la función es proyectada hasta alcanzar el máximo throughput, en este caso 400 [Mbps], por lo que el punto 36,2222 corresponde al valor de SINR donde se alcanza.

```

11 else if (band ==1) {
12     if (sinr<5) {th.add(5.4*sinr);}
13     else if (5<=sinr && sinr<8) {th.add(9.0*sinr-18);}
14     else if (8<=sinr && sinr<12) {th.add(6.75*sinr);}
15     else if (12<=sinr && sinr<18) {th.add(13.5*sinr-81);}
16     else if (18<=sinr && sinr<21) {th.add(18.0*sinr-162);}
17     else if (21<=sinr && sinr<23) {th.add(13.5*sinr-67.5);}
18     else if (23<=sinr && sinr<28) {th.add(5.4*sinr+118.8);}
19     else if (28<=sinr && sinr<32) {th.add(13.5*sinr-108);}
20     else if (32<=sinr && sinr<36.2222) {th.add(18.0*sinr-252);}
21     else {th.add(400.0);}

```

La función `getThroughput()` es utilizada en una función más general llamada `getMaxThroughput()`, la cual retorna el throughput máximo final alcanzado por un usuario medido, y tiene como parámetros de entrada una lista de redes WiFi escaneadas por el dispositivo, el ancho de banda y el canal utilizado, la dirección MAC del AP, la versión del estándar 802.11 y el RSSI de la red WiFi utilizada. La lógica que sigue esta función es la que se muestra en el diagrama de la figura 4.19 y se describe con mayor detalle a continuación.

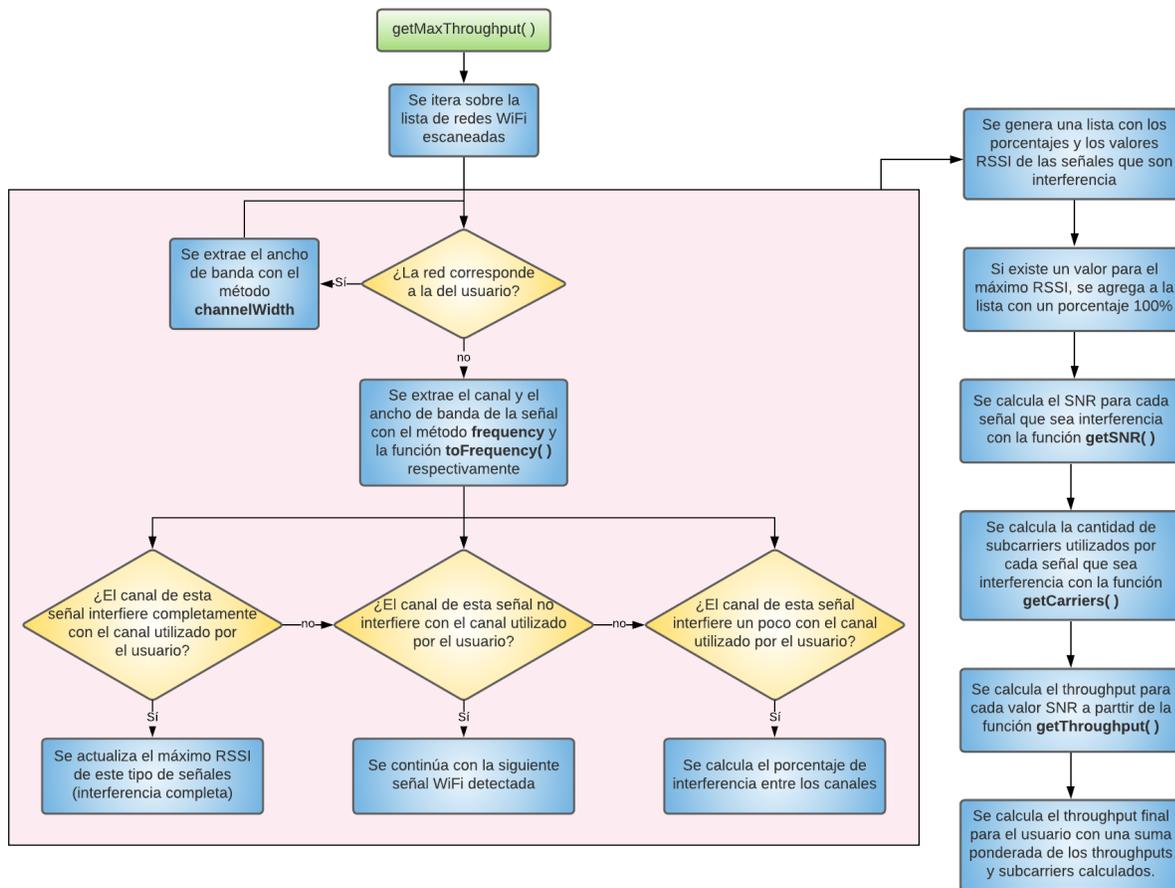


Figura 4.19: Lógica de la función `getMaxThroughput`.

Para conseguir el throughput final de un usuario, se procede a crear las listas vacías

“percentages” y “levels”, las cuales van a contener los porcentajes de interferencia y los valores RSSI de las señales WiFi escaneadas que correspondan a una interferencia de la señal utilizada por el usuario. Luego, se procede a inicializar el máximo RSSI como el mínimo posible  $maxRSSI = -121$ , un contador de señales que interfieren parcialmente con la del usuario  $quantity = 0$  y el ancho de banda utilizado  $band = 0$ , lo cual corresponde a 20 [MHz]. A continuación, se procede a iterar sobre todas las redes WiFi detectadas por el dispositivo, las cuales se encuentran definidas en una lista que es dada como parámetro de entrada. El objetivo de esto es determinar la interferencia producida por cada una de estas señales y, para esto, se debe determinar si éstas se encuentran en el mismo canal que el usuario, de manera parcial o total. Por ejemplo, observando la figura 2.11, si el usuario se encuentra en el canal 1 y otra red se encuentra en este mismo canal, se tiene una interferencia completa, por lo que el SINR puede ser calculado a partir de la resta de sus valores RSSI, sin embargo, si la otra red se encuentra en el canal 3, se tiene una interferencia parcial, por lo que el SINR es mayor a la resta entre dichos valores. De acuerdo a lo anterior, la función toma en cuenta los siguientes tres casos para cada una de las señales WiFi detectadas.

- El primer caso es que no exista interferencia entre los canales, como se puede observar en la figura 4.20. En este caso no se toma en cuenta de manera negativa en el cálculo de throughput máximo y se prosigue con la siguiente señal WiFi. En la función **getMaxThroughput()** se considera este caso utilizando el segmento de código que se muestra a continuación.



Figura 4.20: Señales sin interferencia.

```

0  else if ( (frec-(bandwidth/2)) >= (fOther + (bdOther/2)) || (
1  frec+(bandwidth/2)) < (fOther - (bdOther/2)) ){
    continue;}

```

- El segundo caso es que exista una interferencia completa entre los canales, como se puede observar en la figura 4.21. En este caso el SINR se calcula sencillamente como la resta entre los valores RSSI, sin embargo, si el RSSI del usuario es menor al de la otra señal, el SINR es 0, pues no existe una señal para transmitir que no esté interferida. En la función **getMaxThroughput()** se considera este caso utilizando el segmento de código que se muestra a continuación, donde se actualiza el valor “maxRSSI” por el máximo valor RSSI de aquellas señales que entren en esta categoría.

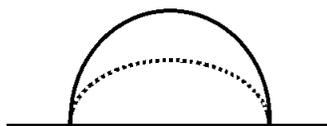


Figura 4.21: Interferencia completa entre señales.

```

0   if(fOther - (bdOther/2) <= frec-(bandwidth/2) && fOther + (
      bdOther/2) >= frec+(bandwidth/2) ){
1   if (maxRSSI < results.get(i).level){
2   maxRSSI = results.get(i).level; }}

```

Cabe destacar que “fOther” corresponde a la frecuencia central del canal de la señal WiFi externa y “bdOther” es el ancho de banda que utiliza. De manera análoga, “frec” es la frecuencia central del canal utilizado por el usuario y “bandwidth” su ancho de banda.

- El tercer caso es que exista una interferencia parcial entre los canales, como se puede observar en la figura 4.22. En este caso, el SINR se calcula restando los RSSI, con lo cual se obtiene un valor de throughput máximo  $i$  del usuario a partir de la interferencia  $i$ , luego, se pondera este valor por la cantidad de subcarriers que se están utilizando del canal del usuario. Para determinar esto, se procede a determinar el porcentaje de interferencia a partir del rango de frecuencias superpuestas entre los canales. En la función `getMaxThroughput()` se considera este caso utilizando el segmento de código que se muestra a continuación.

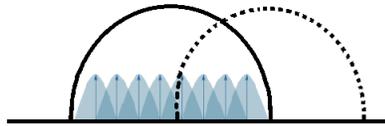


Figura 4.22: Interferencia parcial entre señales.

```

0   else {
1   if (frec+(bandwidth/2) <= fOther+(bdOther/2)){
2   double p = ( ((( float) (frec+(bandwidth/2)) - (fOther
      - (bdOther/2)) ))/bandwidth) *100.0);
3   percentage = p;
4   } else if ( frec-(bandwidth/2) >= fOther-(bdOther/2) ){
5   double p = ( ((( float) (fOther+(bdOther/2)) - (frec-(
      bandwidth/2)) ))/bandwidth) *100.0);
6   percentage = p;
7   } else {
8   double p = ( (( float) bdOther )/bandwidth) *100.0);
9   percentage = p;}}

```

Como se observa en el código, para calcular el porcentaje de interferencia hay que ponerse en tres casos, que ésta sea por la izquierda del canal, por la derecha o en el centro y, de acuerdo a esto, se determina el porcentaje de frecuencias superpuestas.

A partir de lo anterior, se obtiene una lista de los throughput máximos alcanzados por el usuario con respecto a cada una de las interferencias y una lista del porcentaje de interferencia de cada una de éstas con el canal del usuario. Con esto, se procede a hacer una suma ponderada de los throughput a partir de los subcarriers utilizados, como se observa en la ecuación 4.2, con la finalidad de obtener el throughput de un usuario en específico. Para

determinar los subcarriers utilizados, se debe tener en cuenta que un canal 802.11 n/ac de 20 MHz consta de 64 subportadoras, donde 52 de éstas se utilizan para transportar datos modulados [14], luego, este valor es multiplicado por el porcentaje de interferencia de la señal, por ejemplo, si se tiene un porcentaje del 32 %, significa que esta señal está ocupando 17 subcarriers.

$$Throughput_{usuario} = \frac{\sum_i \%Interferencia_i * \#Subcarriers * Throughput_i}{\#Subcarriers_{Total}} \quad (4.2)$$

Una vez que se obtiene este valor para todos los usuarios, lo cual se consigue al oprimir el botón medir para todas las personas ingresadas, se procede a pedirle al usuario que seleccione la aplicación que desea utilizar cada participante del análisis. Las opciones que observa el usuario son ejemplos de los servicios que más ha utilizado la gente en esta cuarentena, es decir, entretención y videollamadas. Los ejemplos usados son las aplicaciones Zoom, Netflix, Instagram y Facebook, las cuales pueden ser cambiadas sencillamente en el código, ya que éstas son creadas a partir de una clase llamada “services”, a la cual se le ingresa el nombre, el throughput que necesita para funcionar y su icono. Los valores de throughput mínimo requeridos por cada aplicación para tener un funcionamiento adecuado son los que se muestran en la tabla 4.2, donde se considera que Facebook e Instagram deben ser capaces de realizar videos en vivo, Netflix debe mostrar películas en buena calidad y Zoom debe permitir la comunicación entre los usuarios conectados<sup>1</sup>.

Aplicación	Throughput necesario
Zoom	5 Mbps
Netflix	10 Mbps
Instagram	5 Mbps
Facebook	5 Mbps

Tabla 4.2: Valores referenciales de throughput mínimo requeridos por las aplicaciones.

Para indicarle al usuario si las personas conectadas pueden utilizar la aplicación que desean en esa ubicación al mismo tiempo, se procede a hacer un análisis para determinar el rendimiento de la red. Éste consiste en calcular un porcentaje de uso de canal para cada usuario a partir de sus throughput máximos y del throughput que requieren. Por ejemplo, si el usuario 1 de la figura 4.14 tiene un throughput máximo  $X$  y desea usar una aplicación que requiere un throughput  $Z$ , su porcentaje de uso viene dado por  $\frac{Z}{X} \cdot 100$ , de igual forma, si el usuario 2 tiene un throughput máximo  $Y$  y desea usar una aplicación que requiere un throughput  $W$ , su porcentaje de uso viene dado por  $\frac{W}{Y} \cdot 100$ . Dado que ambos usuarios usan la red al mismo tiempo, si la suma de estos porcentajes es menor al 100 %, significa que el rendimiento de la red puede proporcionar el servicio de estas aplicaciones, sin embargo, si es mayor, los usuarios no pueden usarlas con las mejor calidad posible, por lo que pueden realizar las mediciones en otras posiciones del hogar o bien, utilizar una calidad más baja. Además, se debe realizar un análisis de estos porcentajes por separado, pues si uno de ellos es mayor al 100 % por sí sólo, se puede indicar que sólo éste no puede hacer uso de la aplicación

<sup>1</sup>Estos corresponden a valores de referencia para ejemplificar el uso de la aplicación y son obtenidos de <https://www.testvelocidad.es/2018/07/10/velocidad-internet-streaming/>

mientras que el otro sí, por lo que puede recomendarse que se acerque al router o que instalen otro AP.

# Capítulo 5

## Funcionamiento y Análisis

### 5.1. Resultados y Análisis

En el capítulo anterior se describe de forma detallada la construcción de la aplicación y el análisis de calidad de servicio realizada. De acuerdo a esto, en este capítulo se busca mostrar los resultados obtenidos más allá de lo que ve el usuario, con el objetivo de analizar los cálculos realizados y proponer posibles mejoras para una siguiente versión de la aplicación.

En la figura 5.1, se observa la estructura de la casa en que se realizan las pruebas que se muestran a continuación. Como se puede observar, el hogar consta de 5 ambientes distintos, sin embargo, sólo tres son lugares donde las personas suelen estar conectadas trabajando o usando servicios de entretenimiento. Estos lugares son descritos con las letras A,B y C, donde A se encuentra al lado del router, B también está muy cerca, pero se encuentra rodeado de murallas y C está lejos y rodeado de murallas. Estas posiciones han sido elegidas para observar el comportamiento de las predicciones de throughput máximo determinadas por la aplicación, pues estos tres lugares tienen distinta distancia al router y distintas interferencias debido a la detección de las señales de los vecinos.

A continuación, se procede a realizar pruebas de las mediciones y cálculos que ejecuta la aplicación en estos tres lugares para un sólo usuario. Para esto se consideran dos horarios, uno en la mañana y otro en la tarde, ambos en horario laboral, lo cual tiene un gran efecto en los resultados debido a que muchas personas trabajan de forma remota dada la pandemia.

En la tabla 5.1 se muestran los resultados obtenidos en estas tres posiciones un jueves entre las 9:15 am y 9:30 am. Los datos que se muestran corresponden a la cantidad de señales WiFi detectadas en el escaneo, el nivel RSSI de la señal del usuario y el throughput máximo calculado a partir de la lógica descrita en el capítulo anterior. Como se observa en la tabla, a medida que el usuario se aleja del router, el nivel de RSSI comienza a disminuir, lo cual se traduce en que el throughput máximo alcanzable por el usuario disminuye. Este resultado tiene coherencia con la realidad, pues a medida que el usuario se aleja del origen, la señal comienza a atenuarse y, además de la distancia, se tiene el factor de que las murallas obstaculizan el paso de la señal. La interferencia de las señales vecinas es uno de los factores más importantes a considerar, pero no se tiene un comportamiento predecible, pues la cantidad

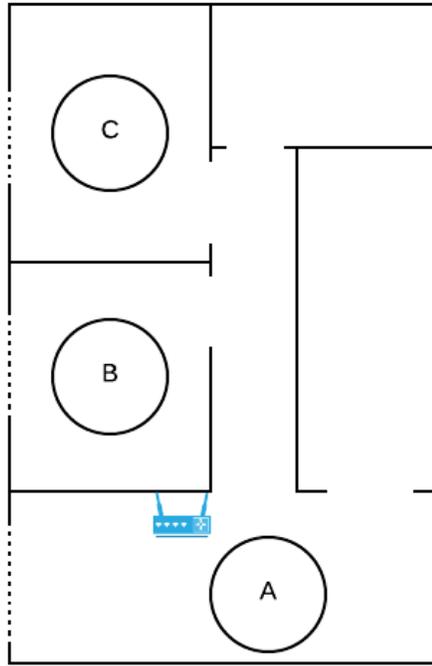


Figura 5.1: Estructura del hogar donde se realizan las pruebas de la aplicación.

de señales detectadas varía en cada medición y, además, éstas cambian el canal en que se encuentran, por lo que no siempre serán las mismas señales que produzcan interferencia.

A pesar de que se detecte una cierta cantidad de redes WiFi, no todas corresponden a interferencia, pues esas señales no necesariamente están utilizando un canal que tenga un solapamiento parcial o total con la del usuario. Para observar esto, se muestra en la tabla 5.2 el resultado del escaneo para el primer caso de la tabla 5.1, donde el usuario se encuentra en la posición A. La red con la que se realizan las pruebas corresponde a la número 0 de la tabla 5.2, la cual tiene un RSSI de -25, además, se encuentra en el canal de frecuencia central de 2412 [MHz], correspondiente al canal 1, y utiliza un ancho de banda tipo 0, es decir, de 20 [MHz]. De acuerdo a esto, las redes número 3, 10 y 14 interfieren con la del usuario, las cuales tienen interferencia total y no parcial; el resto de las redes simplemente no se encuentran solapadas con la red 0, por lo que no corresponden a ruido. Entonces, de las 15 señales detectadas, sólo tres corresponden a interferencia, por lo que se trabaja con éstas para determinar el SINR entre cada una con la señal del usuario, el número de subcarriers utilizados por cada una y, finalmente, el throughput máximo.

Hora	Posición	# Señales Detectadas	RSSI [dBm]	Throughput [Mbps]
9:16	A	15	-25	173
9:24	B	15	-47	104
9:28	C	18	-56	83

Tabla 5.1: Resultados de la aplicación en la mañana.

El siguiente horario a analizar es en la tarde de un día laboral, para esto se toman las mediciones de un usuario en las tres posiciones mencionadas anteriormente, lo cual se muestra

# Red	Nombre	BSSID	RSSI	Frecuencia	Ancho de Banda
0	Depto 705	b0:ea:bc:41:85:4a	-25	2412	0
1	San Feña	84:aa:9c:32:b7:fe	-50	2437	0
2	Familiarojasaburto	c8:b4:22:27:98:70	-52	2437	0
3	VTR-6867369	18:35:d1:1c:ac:a1	-57	2412	0
4	VTR-3295307	40:0d:10:d0:1b:d9	-66	2462	0
5	AMELIA	84:aa:9c:32:d5:b8	-68	2437	0
6	IGNACIA	94:91:7f:ce:37:6a	-72	2437	0
7	Amelia D-LINK	90:94:e4:cb:e4:5a	-72	2442	0
8	VTR-9600781	48:d3:43:8c:e9:99	-74	2412	0
9	Wifi susi	84:aa:9c:e1:a7:da	-74	2462	0
10	WIFICTM	50:0f:f5:bd:23:08	-74	2412	1
11	VTR-3872939	48:d3:43:33:c0:89	-78	2437	0
12	Minina	c8:b4:22:26:91:a5	-78	2462	0
13	VTR-2131454	48:d3:43:b1:6c:29	-78	2462	0
14	VTR-6715742	c0:05:c2:a0:42:31	-80	2412	0

Tabla 5.2: Detalle del escáner de las redes WiFi alrededor en la mañana en la posición A.

en la tabla 5.3. Se puede observar que ocurre lo mismo que en el caso anterior, a medida que el usuario se aleja del router, el RSSI disminuye y, por ende, el throughput máximo calculado también. En este caso se toma una cuarta medición en la posición C a las 17:18 p.m, con la finalidad de demostrar que, en una misma posición, la cantidad de señales detectadas puede variar en distintos instantes de tiempo, sin embargo, esto no necesariamente es influyente en la interferencia generada, pues, para ambas mediciones en la zona C del hogar, se tienen estimaciones del throughput máximo similares.

Hora	Posición	# Señales Detectadas	RSSI [dBm]	Throughput [Mbps]
16:00	A	16	-19	173
16:46	B	14	-45	125
16:52	C	7	-53	120
17:18	C	12	-54	123

Tabla 5.3: Resultados de la aplicación en la tarde.

Con la finalidad de comprender mejor el procedimiento que se realiza para el cálculo de throughput máximo, se muestra en la tabla 5.4 las listas generadas para el último escaneo de la tabla 5.3. Como se puede observar, las listas están compuestas por dos elementos, es decir, de las 12 señales detectadas, sólo dos corresponde a interferencias, donde la primera es parcial y la segunda completa. Esto se puede observar en la lista de porcentaje de uso del canal, donde una tiene un porcentaje del 77 % y la otra del 100 %, por consiguiente, de acuerdo a estos valores se obtiene la cantidad de subcarriers que utilizan cada una de estas señales, lo cual se consigue ponderando los 52 subcarriers totales por su porcentaje de uso, tal y como se muestra a continuación.

$$Subcarriers_0 = \frac{77,72}{77,27 + 100} \cdot 52 \quad (5.1)$$

Luego, para cada una de estas señales detectadas se procede a restar el nivel RSSI de éstas con el del usuario (-54), con la finalidad de obtener el SINR. Con estos valores se procede a utilizar las funciones lineales creadas a partir de la tabla 4.16, para obtener una lista con los valores de throughput máximo con respecto a cada una de las interferencias. Finalmente, estos valores de throughput son ponderados según la ecuación 4.2, lo cual corresponde a los cálculos que se muestran a continuación.

$$Throughput_{max} = \frac{23 \cdot 124,8 + 29 \cdot 122,2}{52} = 123,35 [Mbps] \quad (5.2)$$

De esta forma se logra obtener el throughput máximo que puede alcanzar este usuario en ese momento y en esa ubicación en particular. A continuación, se procede a utilizar este valor para determinar el porcentaje de uso del canal de este usuario, para lo cual se debe considerar la tabla 4.2, donde se tienen valores referenciales del throughput mínimo requerido por las aplicaciones utilizadas de ejemplo. Si este usuario señala que quiere utilizar Netflix, entonces se cumple que el porcentaje de uso del canal corresponde a lo siguiente:

$$Uso_{canal} = \frac{10Mbps}{123,35Mbps} = 0,081 = 8,1\% \quad (5.3)$$

Este valor quiere decir que este usuario, dado su throughput máximo alcanzable y el requerimiento mínimo de throughput de la aplicación elegida, va a ocupar un 8,1 % del 100 % de disponibilidad del canal, es decir, este usuario por sí solo puede hacer uso de esta aplicación sin problema. Finalmente, si la suma de los porcentajes de uso de cada una de las personas que va a usar la red WiFi es mayor al 100 % de disponibilidad del canal, se concluye que en conjunto van a tener problemas al momento de utilizar las aplicaciones que desean, lo cual se informa en la aplicación.

Porcentajes de uso del canal de la señal i	0 = 77.27 %
	1 = 100.00 %
Subcarriers utilizados	0 = 23
	1 = 29
SINR	0 = 23
	1 = 22
Throughput del usuario en relación a la interferencia i	0 = 124.80
	1 = 122.20

Tabla 5.4: Detalles de los resultados de un usuario en la posición C.

Como se puede apreciar hasta ahora, los resultados son consistentes con la realidad, sin embargo, hay un caso particular que disminuye los valores del throughput estimado, e incluso puede retornar valores de cero. Este caso ocurre algunas veces cuando el usuario se encuentra en el lugar C de la imagen 5.1, pues aquí sucede, de manera poco frecuente, que alguna de

las redes detectadas tiene un valor de RSSI mayor al de la red del usuario. Este caso se muestra en la tabla 5.5, donde el usuario se encuentra conectado a la red número 1, la cual tiene interferencia completa con la red 0. Siguiendo los pasos para obtener el throughput máximo, al obtener los valores de SINR se tiene el problema de que el RSSI de la señal vecina es mayor a la del usuario, es decir que la diferencia de RSSI es menor que 0, y de acuerdo al código, en este caso el SINR queda fijo en 0. Luego, este valor es reemplazado en la ecuación lineal correspondiente al caso, la cual es  $Throughput = 6,5 \cdot SINR$ , por lo que el valor de throughput del usuario con respecto a esta interferencia es 0. Este caso ocurre rara vez, pero de todas formas debe ser solucionado en versiones posteriores de la aplicación, ya sea considerando de otra manera aquellas señales que tienen un RSSI mayor o estableciendo un throughput mínimo dada cierta interferencia.

Sin considerar estos casos puntuales, los resultados obtenidos son adecuados para comparar entre el throughput máximo y el requerido por la aplicación que desea usar el usuario. En la realidad, los valores reales son mayores a los estimados, pero esto se debe a que se está tomando un caso estático en que la red no cambia a tecnologías mejores, si no que se queda en un estado base descrito en el capítulo anterior. A pesar de esto, se está considerando un caso estándar y generalizable para todos los usuarios, por lo que el valor estimado es una aproximación real tomando en cuenta estas tecnologías por defecto.

# Red	Nombre	BSSID	RSSI	Frecuencia	Ancho de Banda
0	WIFICTM	50:0f:f5:bd:23:08	-52	2412	1
1	Depto 705	b0:ea:bc:41:85:4a	-54	2412	0
2	San Feña	84:aa:9c:32:b7:fe	-56	2437	0
3	Familiarojasaburto	c8:b4:22:27:98:70	-58	2437	0
4	VTR-3295307	40:0d:10:d0:1b:d9	-63	2462	0
5	SAM	7c:db:98:de:56:4f	-71	2412	0
6	VTR-9302903	18:35:d1:48:bf:91	-72	2412	0
7	DEPTO 604	1c:b0:44:be:9e:26	-73	2462	0
8	Amelia D-LINK	90:94:e4:cb:e4:5a	-76	2442	0
9	Betty	7c:db:98:57:13:ac	-78	2462	0
10	Minina	c8:b4:22:26:91:a5	-79	2462	0
11	VTR-0477277	40:0d:10:30:e0:b9	-80	2462	0
12	AMELIA	84:aa:9c:32:d5:b8	-81	2437	0
13	-	ae:ae:19:d0:9a:a0	-81	2452	0
14	HUAWEI-2B76	88:10:8f:cb:2b:76	-82	2452	0

Tabla 5.5: Detalle del escáner de las redes WiFi alrededor en la mañana en la posición C.

# Capítulo 6

## Conclusiones

Durante el desarrollo de este trabajo, los alcances fueron modificados de acuerdo a las nuevas ideas que surgían y a los imprevistos relacionados con el tipo de información que es posible extraer de la red WiFi a partir de dispositivos móviles. A pesar de esto, este trabajo se considera el inicio de un proceso de desarrollo en la calidad de servicio, pues se busca garantizar la transparencia de información al mantener al usuario informado del servicio que recibe respecto a lo que contrata.

Dado el trabajo realizado, se logra reconocer los parámetros que influyen en la percepción de calidad de servicio del cliente y, a partir de esto, se recopilan datos de la red WiFi desde el dispositivo móvil. A partir de esta recopilación se logra realizar un análisis de los datos obtenidos, en el cual se modela el throughput máximo esperado para distintos usuarios dentro de un ambiente indoor domiciliario, el cual se encuentra sujeto a interferencia producto de la presencia de otras redes operando en la misma frecuencia. Además de esto, se logra mostrar los resultados obtenidos en una aplicación para android, lo cual se realiza de una forma sencilla mostrando si es posible que los usuarios utilicen los servicios que desean. De acuerdo a esto, se puede decir que los objetivos específicos son cumplidos correctamente y, por lo tanto, el objetivo general también, es decir, se logra crear una herramienta que utiliza parámetros claves de la red WiFi para determinar la calidad de servicio del usuario en tiempo real.

Tras el análisis de calidad de servicio realizado y al probar el modelo en un ambiente indoor domiciliario con mucha interferencia, se puede apreciar que la aplicación es consistente con la realidad, pues el nivel de señal y el throughput máximo de los usuarios disminuye a medida de la distancia al router aumenta, al igual que cuando se presentan más señales en el ambiente que provocan interferencia en el canal del usuario.

### 6.1. Continuación del Proyecto

El proyecto en general tiene una base firme, pues la estructura se encuentra ordenada para seguir agregando actividades y funcionalidades, en consecuencia, se encuentra abierto a agregar nuevos análisis para el usuario y mejoras a esta primera versión.

Por un lado, la simplicidad de la base de datos utilizada permite agregar nuevos datos al

modelo WiFi creado y, además, se pueden crear nuevos modelos rápidamente y sin problemas, pues estos son creados mediante clases de java. Debido a la estructura del modelo de datos WiFi creado, se pueden obtener datos de la red por rangos de tiempo, pues uno de los campos agregados corresponde a la fecha y la hora. Con esta información se puede agregar una sección donde se muestre en gráficos el comportamiento de los datos más relevantes y, para que el usuario los entienda fácilmente, deben tener una referencia, como por ejemplo, si se decide graficar la velocidad a través del tiempo, se debe mostrar con una línea la velocidad mínima garantizada.

Por otro lado, al diseño de la plataforma se le pueden hacer muchos cambios con tal de que sea más personalizada al usuario y más privada. En primer lugar, se debe hacer un log in para que el usuario pueda personalizar su cuenta con sus datos personales, con lo cual se puede agregar una pantalla en que el usuario pueda agregar los datos de su plan contratado, con la finalidad de monitorear el cumplimiento de éste. En segundo lugar, debe agregarse una pantalla de inicio, donde el usuario pueda elegir el análisis que desea observar y donde pueda observar su perfil de forma sencilla.

Finalmente, se debe modificar el análisis que realiza actualmente la aplicación, con tal de considerar de otra forma el caso en que la interferencia tenga un RSSI mayor que la señal del usuario. Además, se debe agregar información más detallada en las cartas de la última pantalla que ve el usuario, la cual corresponde a la figura 4.5, pues se tiene mucha información generada a partir del cálculo del throughput máximo, sin embargo, ésta debe ser mostrada de una manera comprensible, por ejemplo, mediante dibujos, gráficos o pestañas con información explicativa. Este último análisis es realizado con los throughput necesarios por las aplicaciones para que se tenga una máxima calidad, por lo que también se puede agregar la información de cómo funcionarían las aplicaciones en una calidad media o baja.

# Bibliografía

- [1] Descripción general del manifiesto de una app. <https://developer.android.com/guide>. Accedido 09-12-2020.
- [2] Historia de la ley nº 21.046. <https://www.bcn.cl/historiadelaley/historia-de-la-ley/vista-expandida/7117/>. Accedido 26-11-2020.
- [3] Prueba inalámbrica de la producción 802.11ac y guía de la validación. [https://www.cisco.com/c/es\\_mx/support/docs/wireless-mobility/wireless-lan-wlan/212892-802-11ac-wireless-throughput-testing-and.html](https://www.cisco.com/c/es_mx/support/docs/wireless-mobility/wireless-lan-wlan/212892-802-11ac-wireless-throughput-testing-and.html). Accedido 03-12-2020.
- [4] *802.11ac In-Depth*. Aruba Networks, 2015.
- [5] ¿qué son los proveedores de servicios de internet? <https://es.xfinity.com/hub/internet/internet-service-providers>, 2017. Accedido 02-07-2020.
- [6] Phenomena report covid-19 spotlight. <https://www.sandvine.com/covid-internet-spotlight-report?hsCtaTracking=69c3275d-0a47-4def-b46d-506266477a50%7Cac52173f-34c1-42df-8469-a091e7219e7a>, 2020. Accedido 01-07-2020.
- [7] Ranking de reclamos en mercado telecomunicaciones 2018-2019. <https://www.sernac.cl/portal/619/w3-article-57955.html>, 2020. Accedido 26-11-2020.
- [8] D. Wetherall A. Tanenbaum. *Redes de Computadoras*. Pearson, México, 5ta edition, 2012.
- [9] J. Anguís. *Redes de Área Local Inalámbricas: Diseño de la WLAN de Wheelers Lane Technology College*. Universidad de Sevilla, Escuela Superior de Ingenieros, Sevilla, España, 2008.
- [10] J. Carvajal. Las conexiones 4g en chile crecen un 35% y llegan a 15 millones. <https://digitalpolicylaw.com/las-conexiones-4g-en-chile-crecen-un-35-y-llegan-a-15-millones/>, 2019. Accedido 01-07-2020.
- [11] R. Vidal J. Paradells E. Garcia, E. López-Aguilera. Effect of adjacent-channel interference in ieee 802.11 wlans. *Technical University of Catalonia (UPC)*, 2007.
- [12] F. Gacharná. *Calidad de servicio en redes IP*. Universidad Autónoma de Colombia,

Colombia, 2007.

- [13] C. García. *Propuesta de arquitectura de QoS en entorno inalámbrico 802.11e basado en DiffServ con ajuste dinámico de parámetros*. Universidad Carlos III de Madrid, Leganés, España, 2006.
- [14] P. Nasiopoulos H. Alnuweiri, Y. Pourmohammadi Fallah and S. Khan. Ofdma-based medium access control for next-generation wlans. <https://doi.org/10.1155/2009/512865>, 2009. Accedido 03-01-2021.
- [15] A. Hathibelagal. Android desde cero: Entender adaptadores y adaptador vista. <https://code.tutsplus.com/es/tutorials/android-from-scratch-understanding-adapters-and-adapter-views-cms-26646>, 2016. Accedido 11-01-2021.
- [16] Y. Jay Guo J. Zhang. *Hybrid antenna array for mmWave massive MIMO*. Science Direct, 2017.
- [17] F. Lema. *Estudio para la Recomendación de Parámetros Técnicos a utilizarse en el Control de Redes LTE 4G en Ecuador*. Escuela Politécnica Nacional, Ecuador, 2017.
- [18] D. Madariaga. *Predicción de la Calidad de Servicio de Internet Móvil Basada en Mediciones Pasivas*. Universidad de Chile, Santiago, Chile, 2018.
- [19] A. Mortin. *Active and Passive Metrics and Methods (with Hybrid Types In-Between)*. Internet Engineering Task Force, Leganés, España, 2016.
- [20] V. Patel. 802.11ac mcs rates. <https://community.cisco.com/t5/wireless-mobility-documents/802-11ac-mcs-rates/ta-p/3155920>. Accedido 07-12-2020.
- [21] E. Pérez. Por qué existen los distintos canales wifi y cómo podemos configurarlos para evitar interferencias. <https://www.xataka.com/servicios/que-existen-distintos-canales-wifi-como-podemos-configurarlos-para-evitar-interferencias>, 2019. Accedido 03-01-2021.
- [22] J. Revelo. Estructura de un proyecto en android studio. <https://www.develou.com/android-studio-proyecto-en/>, 2014. Accedido 09-12-2020.
- [23] J. Rink. 802.11ac wireless: Channel bonding, mimo, spatial streams, and beamforming. <https://www.sourceonetechnology.com/802-11ac-wireless-channel-bonding-mimo-spatial-streams-and-beamforming/>, 2016. Accedido 02-01-2021.
- [24] W. Wang S. Liu, X. Zhang. Analysis of modulation and coding scheme selection in mimo-ofdm systems. *Beijing University of Posts and Telecommunications*, 2006.
- [25] C. E. Shannon. *A Mathematical Theory of Communication*. The bell system Technical Journal, 1948.
- [26] K. Papagiannaki V. Mhatre. *Active and Passive Metrics and Methods (with Hybrid Types In-Between)*. Internet Engineering Task Force, Leganés, España, 2016.

- [27] H. Zegarra. *Propuesta de extensión del protocolo IEEE 802.11 en modo PCF para ampliar la cobertura de celdas en una WLAN mediante nodos relay*. Universidad Católica San Pablo, Arequipa, Perú, 2019.

# Apéndice A

## Índice de Siglas

<b>TICs</b>	Tecnologías de la información y la comunicación
<b>ISPs</b>	Internet Service Provider
<b>SUBTEL</b>	Subsecretaría de telecomunicaciones
<b>IPTV</b>	Internet Protocol Television
<b>OTT</b>	Over-The-Top
<b>QoS</b>	Quality of Service
<b>PAN</b>	Personal Area Network
<b>LAN</b>	Local Area Network
<b>CAN</b>	Campus Area Network
<b>MAN</b>	Metropolitan Area Network
<b>WAN</b>	Wide Area Network
<b>AP</b>	Access Point
<b>NIC</b>	Network Interface Card
<b>DSL</b>	Digital Subscriber Line
<b>ADSL</b>	Asymmetric Digital Subscriber Line
<b>DSLAM</b>	Digital Subscriber Line Access Multiplexer
<b>IP</b>	Internet Protocol
<b>CSMA/CD</b>	Carrier Sense Multiple Access with Collision Detection
<b>MSAU</b>	Multistation Access Unit
<b>FDDI</b>	Fiber Distributed Data Interface
<b>HFC</b>	Hybrid Fiber-Coaxial
<b>SERNAC</b>	Servicio Nacional del Consumidor
<b>BCN</b>	Biblioteca del Congreso Nacional
<b>UIT</b>	Unión Internacional de Telecomunicaciones
<b>RTT</b>	Round-Trip Time
<b>ICMP</b>	Internet Control Message Protocol
<b>DNS</b>	Domain Name System
<b>ARP</b>	Address Resolution Protocol
<b>IntServ</b>	Integrated services
<b>Diffserv</b>	Differentiated services
<b>RSVP</b>	Resource Reservation Protocol

<b>SNR</b>	Signal to Noise Ratio
<b>CR</b>	Core Routers
<b>ER</b>	Egress Routers
<b>DSCP</b>	DiffServ Code Point
<b>ToS</b>	Type of Service
<b>PHB</b>	Per Hop Behavior
<b>EF</b>	Expedited Forwarding
<b>AF</b>	Assured Forwarding
<b>BE</b>	Best Effort
<b>BSS</b>	Basic Service Set
<b>DS</b>	Distribution System
<b>ESS</b>	Extended Service Set
<b>CSMA/CA</b>	Carrier Sense Multiple Access with Collision Avoidance
<b>ACK</b>	Acknowledgment
<b>FCS</b>	Frame Check Sequence
<b>RTS</b>	Request To send
<b>CTS</b>	Clear To send
<b>MSDU</b>	MAC Service Data Unit
<b>DFC</b>	Distributed Coordination Function
<b>PFC</b>	Point Coordination Function
<b>IFS</b>	Inter Frame Space
<b>DIFS</b>	Distributed Inter Frame Space
<b>SIFS</b>	Short Inter Frame Space
<b>PC</b>	Point Coordinator
<b>PC</b>	Contention Period
<b>PC</b>	Contention Free Period
<b>PIFS</b>	IFS of Point Coordination Function
<b>QSTA</b>	QoS-aware Station
<b>STA</b>	Station
<b>QAP</b>	QoS-aware Access Point
<b>TSPEC</b>	Traffic Specification
<b>HCF</b>	Hybrid Coordination Function
<b>EDCA</b>	Enhanced Distributed Channel Access
<b>HCCA</b>	HCF Controlled Channel Access
<b>AC</b>	Access Categories
<b>CAP</b>	Controlled Access Phase
<b>IDE</b>	Integrated Development Environment
<b>MCS</b>	Modulation and Coding Scheme
<b>HT</b>	High Throughput
<b>VHT</b>	Very High Throughput
<b>SM</b>	Spatial Stream
<b>OFDM</b>	Orthogonal Frequency Division Multiplexing
<b>MIMO</b>	Multiple Input Multiple Output