



UNIVERSIDAD NACIONAL DE CHIMBORAZO

FACULTAD DE INGENIERÍA

CARRERA DE INGENIERÍA ELECTRÓNICA Y TELECOMUNICACIONES

**“Trabajo de grado previo a la obtención del Título de Ingeniero en
Electrónica y Telecomunicaciones”**

TRABAJO DE GRADUACIÓN

Título del proyecto:

**“DISEÑO E IMPLEMENTACIÓN UN SISTEMA DE
TELECOMUNICACIÓN PARA LOS DISPOSITIVOS ELECTRÓNICOS
DE PERSONAS CON DISCAPACIDADES DE LA UNIVERSIDAD
NACIONAL DE CHIMBORAZO, QUE PERMITA AYUDAR EN LA
ORIENTACIÓN, UBICACIÓN E INTERACCIÓN ENTRE LOS
USUARIOS Y LA SOCIEDAD”**

Autores:

Perkins Santiago Haro Parra
Luis Fernando Inca Guamán

Director: Ing. Aníbal Llanga

**Riobamba – Ecuador
2016**

Los miembros del Tribunal de Graduación del proyecto de investigación de título: **DISEÑO E IMPLEMENTACIÓN UN SISTEMA DE TELECOMUNICACIÓN PARA LOS DISPOSITIVOS ELECTRÓNICOS DE PERSONAS CON DISCAPACIDADES DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO, QUE PERMITA AYUDAR EN LA ORIENTACIÓN, UBICACIÓN E INTERACCIÓN ENTRE LOS USUARIOS Y LA SOCIEDAD**, presentado por: Perkins Santiago Haro Parra, Luis Fernando Inca Guamán, y dirigida por: Ing. Aníbal Llanga

Una vez escuchada la defensa oral y revisado el informe final del proyecto de investigación con fines de graduación escrito en la cual se ha constatado el cumplimiento de las observaciones realizadas, remite la presente para uso y custodia en la biblioteca de la Facultad de Ingeniería de la UNACH.

Para constancia de lo expuesto firman:

Ing. Paulina Vélez

Presidente del Tribunal



Firma

Ing. Aníbal Llanga

Director del Proyecto



Firma

Ing. Juan Carlos Cepeda

Miembro de Tribunal



Firma

CERTIFICACIÓN DEL TUTOR

Certifico que el presente trabajo de investigación previo a la obtención del grado de Ingeniero en ELECTRÓNICA Y TELECOMUNICACIONES. Con el tema:

“DISEÑO E IMPLEMENTACIÓN UN SISTEMA DE TELECOMUNICACIÓN PARA LOS DISPOSITIVOS ELECTRÓNICOS DE PERSONAS CON DISCAPACIDADES DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO, QUE PERMITA AYUDAR EN LA ORIENTACIÓN, UBICACIÓN E INTERACCIÓN ENTRE LOS USUARIOS Y LA SOCIEDAD.” ha sido elaborado por los estudiantes **Perkins Haro y Luis Inca**, el mismo que ha sido revisado y analizado en un cien por ciento con el asesoramiento permanente de mi persona en calidad de Tutor por lo que se encuentran aptos para su presentación y defensa respectiva.

Es todo cuanto puedo informar en honor de la verdad



Ing. Aníbal Llanga

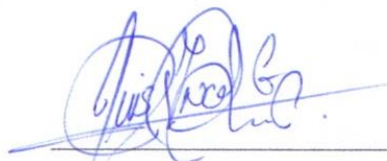
C.I. 060293332-7

AUTORÍA DE LA INVESTIGACIÓN

“La responsabilidad del contenido de este Proyecto de Graduación, nos corresponde exclusivamente a: Perkins Santiago Haro Parra, Luis Fernando Inca Guamán y al Ing. Aníbal Llanga y el patrimonio intelectual de la misma a la Universidad Nacional de Chimborazo.”



Perkins Santiago Haro Parra
C.I. 060341197-6



Luis Fernando Inca Guamán
C.I. 160064253-0

AGRADECIMIENTO

Agradecemos a nuestros padres cuyo apoyo incondicional nos han llevado hasta donde ahora estamos, son ejemplos de trabajo esfuerzo y constancia, durante toda nuestra formación académica, ellos han sido un pilar fundamental que jamás dudaron de nuestras capacidades, a nuestro tutor Ing. Aníbal Llanga, que gracias a la amistad y confianza que posee hacía con nosotros nos permitió el desarrollo del presente proyecto, al Ing Iván Usigña por la generosidad y don de gente al momento de prestarnos su ayuda en cualquier circunstancias y al Ing. Javier Haro por su amistad, su experiencia y conocimientos dentro del ámbito de redes institucional, gracias a lo cual se dio la culminación de este trabajo, y como final agradecemos a todas las amistades sinceras que se han formado en toda nuestra vida académica.

DEDICATORIA

Este presente trabajo está dedicado a nuestras queridas familias que siempre confiaron en nosotros y que nos apoyaron durante toda nuestra vida académica, al enseñarnos a no rendirnos y que un tropezón no es caída a luchar por los sueños anhelados y a lograr conseguir cada uno de los títulos que nos proponemos, además cabe recalcar el esfuerzo que nuestras familias han puesto en nosotros física, económica y emocionalmente durante cada una de las etapas de nuestras vidas, es muy grato para nosotros dar un agradecimiento muy profundo a todas y cada una de las personas que han formado parte de nuestra formación académica durante todo el transcurso de la universidad.

ÍNDICE GENERAL

UNIVERSIDAD NACIONAL DE CHIMBORAZO.....	I
Autores:.....	I
CERTIFICACIÓN DEL TUTOR	III
AUTORÍA DE LA INVESTIGACIÓN.....	IV
AGRADECIMIENTO	V
DEDICATORIA	VI
ÍNDICE DE FIGURAS.....	XI
ÍNDICE DE TABLAS.....	XIII
RESUMEN.....	XIV
ABSTRACT	XV
INTRODUCCIÓN	16
1. FUNDAMENTACIÓN TEÓRICA	19
1.1 Discapacidades.....	19
1.1.1 Introducción:	19
1.1.2 Tipos de discapacidades.....	20
1.2 Sistema de Base de datos (SBD):.....	21
1.2.1 Introducción:	21
1.2.2 Características del Sistema de Base de Datos (SBD)	22
1.2.2.1 Componentes de un Sistema de Base de Datos.....	22
1.2.2.2 Distintos niveles de un Sistema de Base de Datos (SBD)....	23
1.2.3 Plataforma XAMPP	24
1.2.3.1 MySQL:	25
1.2.3.2 Iniciar y detener XAMPP.....	25
1.2.3.3 Creación de la base de datos.....	26
1.3 Lenguaje de programación NETBEANS	27
1.3.1 Introducción	27
1.3.2 Desarrollador Java Netbeans.....	28
1.3.2.1 Estructura de la programación Java.....	28
1.3.2.2 Programación detallada en java	28
1.3.2.3 Identificadores	29
1.4 Controlador del sistema PHP.....	30

1.4.1	Sistema operativo Brakets.....	30
1.4.1.1	Definición	30
1.4.1.2	Características	30
1.4.1.3	Creación de un nuevo proyecto.....	31
1.5	Android Studio	32
1.5.1	Introducción	32
1.5.1.1	Creación de un proyecto en Android Studio	32
1.5.1.2	Llamar las clases de java por Android Studio (Activity)	37
1.5.1.3	Código en Android Studio para llamar una de clases en Java .	38
1.6	Red inalámbrica (WiFi) interna para el proyecto STAPD	39
1.6.1	Introducción	39
1.6.1.1	Redes inalámbricas WiFi.....	39
1.6.1.2	Rango y Flujo de Datos.....	39
1.6.2	Recursos para la Implementación de la Red Inalámbrica	40
1.6.2.1	Router.....	40
1.6.2.2	Switch.....	40
1.6.2.3	Cable de Red.....	41
1.6.3	Equipos para la Implementación de la Red Inalámbrica	41
1.6.3.1	Router Cisco Linksys E4200	41
1.6.3.2	Switch cisco Catalyst 2960.....	42
1.6.3.3	Cable UTP con conectores RJ45.....	43
1.6.4	Subneteo y Configuración de la Red Inalámbrica	45
1.6.4.1	Subnetear	45
1.6.4.2	PuTTY	46
1.	METODOLOGÍA	47
2.1	Tipo de estudio.....	47
2.1.1	Descriptivo	47
2.2	Métodos, Técnicas e Instrumentos.....	47
2.2.1	Métodos	47
2.2.1.1	Investigación Predictiva Experimental	47
2.3	Técnicas	47
2.4	Población y muestra	47
2.5	Hipótesis	48
2.6	Operacionalización de las variables.....	48
2.7	Procedimientos.....	50

2.8	Procedimiento y análisis	51
2.8.1	Discapacidades que utiliza el sistema STAPD.....	51
2.8.2	Desarrollo de la aplicación Android.....	52
2.8.2.1	Programación en Java Netbeans	52
2.8.2.2	Creación y conexión de la base de datos XAMPP	52
2.8.2.3	Controlador del sistema PHP en Android Studio.....	54
2.8.2.4	Programación y compilación de la aplicación en Android Studio....	55
2.8.2.4.1	Programación para recuperación de datos en Brackets.	55
2.8.2.4.2	Registros en base de datos programado en Brackets.....	56
2.8.2.4.3	Eliminar la dirección IP del sistema de base de datos.....	56
2.8.2.4.4	Extracción de la dirección IP en Android Studio.	56
2.8.2.4.5	Ayuda y registro en el sistema de base de datos.	56
2.8.2.4.6	Declaración de variables y la programación general.	57
2.8.2.4.7	Interfaz de la aplicación móvil Android.	57
2.8.2.4.8	Diagrama de flujo de la aplicación STAPD.....	59
2.8.3	Red Inalámbrica STAPD.....	60
2.8.3.1	Diseño de la Red Inalámbrica STAPD.....	60
2.8.3.2	Configuración de los Reuters con sus direcciones IP	60
2.8.3.3	Ubicación en los puertos de los Routers inalámbricos.	63
2.8.3.4	Topología de la red inalámbrica STAPD.....	64
2.8.3.5	Ubicación de los puntos de acceso de la red STAPD.	65
2.8.4	Ubicación de los routers de la Red Inalámbrica STAPD	68
2.8.4.1	Ubicación de los routers en el edificio de la facultada Ingeniera ..	68
2.8.5	Configuración para los Routers de la Red Inalámbrica STAPD.....	71
2.8.6	Pruebas de interactividad de la aplicación Android (STAPD)	72
2.8.7	Comprobación de hipótesis	74
2.8.8	Planteamiento de la hipótesis estadística	74
2.8.9	Establecimiento del nivel de significancia.....	75
2.8.10	Determinación del valor estadístico de prueba.....	75
	CAPITULO III.....	78
	2. Resultados	78
	CAPITULO IV	80
	3. Discusión.....	80
	CAPITULO V.....	82

4. Conclusiones y Recomendaciones	82
5.1 Conclusiones.....	82
5.2 Recomendaciones.....	84
CAPITULO VI	85
5. PROPUESTA	85
6.1 Título de la propuesta	85
6.2 Introducción.....	85
6.3 Objetivos.....	87
6.3.1 Objetivo General	87
6.3.2 Objetivos Específicos	88
6.4 Fundamentación Científico-Técnico.....	88
6.5 Descripción de la propuesta.....	88
6.6 Diseño organizacional	89
6.7 Monitoreo y Evaluación de la propuesta.....	89
6. BIBLIOGRAFÍA	90
ANEXO 1	93
ANEXO 2	114

ÍNDICE DE FIGURAS.

Figura 1. Los 4 Tipos de Discapacidades generalizadas en la actualidad.....	21
Figura 2. Arquitectura de los niveles del sistema de base de datos.	24
Figura 3. Iniciación del apache en XAMPP.....	26
Figura 4. Inicialización del MySQL en XAMPP	26
Figura 5. Interfaz del phpMyAdmin para configuración del a base de datos	27
Figura 6. Entorno desarrollador abierto para algunos sistemas operativos.....	27
Figura 7. Estructura básica detallada en java	28
Figura 8. Ventana de edición de programa en Breakets.....	31
Figura 9. Logo de software Android Studio.....	32
Figura 10. Presentación de entrada para una selección de nuevo proyecto	32
Figura 11. Campos de información para un nuevo proyecto	33
Figura 12. Configuración delas plataformas para nuestra aplicación	33
Figura 13. Configuración de la selección de sistema Android	34
Figura 14. Elección del asistente para la actividad principal de la aplicación.	34
Figura 15. Se coloca el nombre de la actividad del proyecto.....	35
Figura 16. Estructura del proyecto para ser programado	35
Figura 17. Ejecución del Activity_main.xml configuración de la interfaz	36
Figura 18. Ejecución del MainActivity.java va toda la programación.	36
Figura 19. Ejecución de AndroidManifest da permisos de programación.....	37
Figura 20. Llamado de las clases creadas en Java Neatbeans.....	37
Figura 21. Seleccionador de la aplicación para ser utilizado en Android Studio .	38
Figura 22. Router Cisco Linksys E4200, proporciona red inalámbrica STAPD. .	41
Figura 23. Switch Cisco Catalyst 2960, configuración puertos.	43
Figura 24. Cable UTP con conector RJ45, conexión entre APs.	44
Figura 25. software para la configuración de los routers	46
Figura 26. Cuadro de inicio con rango de botones en Java Neatbeans	53
Figura 27. Base de Datos creada junto con la tabla de registro	53
Figura 28. Activación de los demás bonotes de la conexión a la base de datos ...	53
Figura 29. Mensaje de creación de la base de datos	54
Figura 30. Cara principal de la aplicación STAPD.....	57
Figura 31. Cara de ayuda de la aplicación STAPD.....	57
Figura 32. Cara de mensaje de la aplicación STAPD	58
Figura 33. Cara donde se muestra el mensaje de la aplicación STAPD	58
Figura 34. Diagrama de flujo de sistema STAPD.....	59
Figura 35. Configuración del router.....	61
Figura 36. Ingreso del rango de las IPs en el router.....	62
Figura 37. Configuración del nombre de la Red inalámbrica.	62
Figura 38. Asignación de la contraseña de acceso a la red	62

Figura 39. Topología de árbol de la red inalámbrica STAPD.	65
Figura 40. Topología de árbol del primer piso junto al rango de direcciones IP..	66
Figura 41. Topología árbol del segundo piso junto al rango de direcciones IP. ...	67
Figura 42. Topología árbol del tercer piso junto al rango de direcciones IP.	68
Figura 43. Ubicación de los routers en el plano arquitectónico del primero piso.	69
Figura 44. Ubicación de los routers en el plano arquitectónico del segundo piso	70
Figura 45. Ubicación de los routers en el plano arquitectónico del tercer piso. ...	71

ÍNDICE DE TABLAS

Tabla 1. Condiciones para programación en cada clase en Android	30
Tabla 2. Estándar de protocolos 802.11 a, 802.11b, 802.11g.	40
Tabla 3. IPs de acceso a los routers junto con las contraseñas	61
Tabla 4. Identificación de Routers con los racks en la UNACH	64
Tabla 5. Identificación de los routers en el primer piso.	65
Tabla 6. Identificación de los routers en el segundo piso.	66
Tabla 7. Identificación de los routers en el tercer piso.	67
Tabla 8. Tiempos de la conexión a la base de datos	72
Tabla 9. Tiempo de repuesta entre el servidor al usuario	73
Tabla 10. Valores Obtenidos.....	75
Tabla 11. Valores de frecuencias esperadas.....	76
Tabla 12. Valores Críticos Método Chi-Cuadrado	76
Tabla 13. Resultados del método estadístico del CHI-CUADRADO	77

RESUMEN

El proyecto de investigación “DISEÑO E IMPLEMENTACIÓN UN SISTEMA DE TELECOMUNICACIÓN PARA LOS DISPOSITIVOS ELECTRÓNICOS DE PERSONAS CON DISCAPACIDADES DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO, QUE PERMITA AYUDAR EN LA ORIENTACIÓN, UBICACIÓN E INTERACCIÓN ENTRE LOS USUARIOS Y LA SOCIEDAD” es desarrollado para brindar ayuda a las personas con discapacidad que deseen moverse dentro del edificio principal de la Facultad de Ingeniería. El Sistema de Telecomunicaciones para Ayuda a Personas Discapacitadas (STAPD), está diseñado bajo los parámetros de un sistema de base de datos siendo su motor principal Windows, Apache, MySQL, PHP (“XAMPP”), programado con un software libre llamado Netbeans, utiliza las librerías necesarias para el desarrollo de un algoritmo en java, además utiliza el software llamado Brackets que es un motor de desarrollo para PHP, el cual permite realizar una conexión estable con la base de datos creada en MySQL, el entorno de desarrollo Android Studio da la posibilidad de crear la aplicación móvil con todos los parámetros necesarios para que puedan interactuar los dispositivos móviles, esto posibilita obtener las direcciones estáticas IPs de una red estructurada interna de internet inalámbrico en el edificio principal de la Facultad de Ingeniería. El desarrollo de este sistema de telecomunicaciones tendrá como fin interactuar entre personas discapacitadas y personas sin discapacidad creando una plataforma de ubicación, ayuda y movilización mutua, con el propósito de cumplir con la inclusión social a personas con discapacidad como lo ordena el plan nacional del buen vivir. El sistema STAPD permite la interacción entre personas discapacitadas que solicitan ayuda para poder moverse dentro del edificio principal de la facultad y las personas que deseen brindar la ayuda, para poder facilitar de esta forma la orientación y la ubicación de a qué lugar se desean mover las personas que poseen alguna discapacidad.



UNIVERSIDAD NACIONAL DE CHIMBORAZO

CENTRO DE IDIOMAS INSTITUCIONA

Lic. Byron Soria

20 de Julio de 2016

ABSTRACT

The study project "Design and Implementation TELECOMMUNICATIONS SYSTEM FOR ELECTRONIC DEVICES OF PEOPLE WITH DISABILITIES NATIONAL UNIVERSITY OF CHIMBORAZO, ALLOWING help guide, LOCATION AND INTERACTION BETWEEN USERS AND SOCIETY" It is made to give help to people with special capacities who wish to move without problem through main building of the Faculty of Engineering.

Telecommunications System to help special capacities. It is designed under the parameters of a database system being its main processor Windows, Apache, MySQL, PHP ("XAMPP"), programmed with a free software called NetBeans, using the libraries necessary for the development of an algorithm in java, also uses software called brackets this is a development processor for PHP, which allows a steady connection to the database created in MySQL, Android development environment studied the possibility to make the mobile application with all the necessary parameters to enable mobile devices to interact, this allows to get the static IP address of an internal structured network of wireless internet in the main building of the Faculty of Engineering.

The development of this telecommunication system will purpose to interact with people with special abilities and regular people building a platform location, and bilateral help in the mobilization, in order to meet social inclusion for people with special abilities as a rule by "Plan Nacional del buen Vivir".

The STAPD system lets an interaction between people with special abilities that are asking for help to move inside the main building of the Faculty, and people who wish to provide help, in others words facilitate the orientation and location of what place you want to mobilize people who have special abilities.

CENTRO DE IDIOMAS



COORDINACION

INTRODUCCIÓN

El Gobierno Nacional con la Secretaria Nacional de Planificación y Desarrollo crea en el año 2013 el Plan Nacional del Buen Vivir el cual indica en sus objetivos la igualdad, la inclusión y la equidad social y territorial en la diversidad, por lo cual la Universidad Nacional de Chimborazo ha realizado diferentes tipos de proyectos de investigación el cual ayuda a las personas con discapacidad a la inclusión social siendo como base de prueba en el edificio principal de la facultad de ingeniería, teniendo la idea de darle un plus extra a estos proyectos de investigación se tuvo la iniciativa de realizar un prototipo de un sistema de telecomunicaciones para ayuda a personas con discapacidad (STAPD), con el fin de dar la ubicación actual de alguna persona con discapacidad o sin discapacidad que tengan la necesidad de una ayuda o a su vez de ayudar, para esto le va a llegar un aviso a su teléfono celular con la ayuda de una aplicación móvil y con el sistema (STAPD).

Este proyecto de inclusión a la sociedad podría cumplir los objetivos del plan nacional del buen vivir obtenido resultados positivos y solidificando las ideas presentados en este estudio.

- Estudio y análisis de un sistema de base de datos el cual permita el almacenamiento de registros realizados por los usuarios del sistema de telecomunicaciones para ayuda a personas discapacitadas (STAPD). Con la plataforma XAMPP y MySQL se realizó un servidor independiente, que consiste en el registro y comparación de datos registrados por medio de la aplicación móvil android. (Xampp, s.f.)
- El desarrollador NetBeans es el software el cual se ha desarrollado la aplicación móvil con las funciones de dar la ubicación mediante a la red inalámbrica wiffi, obteniendo la dirección IP (Internet Protocolo), dando

rangos prioritarios para la sectorización del edificio de la Facultad de Ingeniería de la Universidad Nacional de Chimborazo.

- Diseño de la interface y estética de la aplicación utilizando Android Studio, software que permitió la compilación del programa realizado en NeatBeans, dando como formato APK, formato necesario para la instalación del sistema de telecomunicaciones para ayuda a personas con discapacidad (STAPD) en los dispositivos móvil con sistema operativo Android.

- Implementación de una red inalámbrica de internet exclusiva para el sistema de telecomunicaciones para ayuda a personas con discapacidad (STAPD) con subneteos de direcciones IPs estáticas ubicadas estratégicamente junto a los routers con rangos diferentes de IPs. (Dordoigne, 2014)

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. Desde la versión "5.6.15", XAMPP cambió la base de datos de MySQL A MariaDB. El cual es un fork de MySQL con licencia GPL.

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos.

NETBEANS permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. un módulo es un archivo java que contiene clases de java escritas para interactuar con las apis de netbeans y un archivo especial (manifest file) que lo identifica como módulo. Las

aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma netbeans pueden ser extendidas fácilmente por otros desarrolladores de software. (NetBeans, s.f.)

ANDROID STUDIO es un entorno de desarrollo integrado (IDE), basado en IntelliJ IDEA de la compañía JetBrains, que proporciona varias **mejoras con respecto al plugin ADT** (Android Developer Tools) para Eclipse. Android Studio utiliza una licencia de software libre Apache 2.0, está programado en Java y es multiplataforma. (Android, s.f.).

CAPITULO I

1. FUNDAMENTACIÓN TEÓRICA

1.1 Discapacidades

1.1.1 Introducción:

Una discapacidad es una condición que hace que una persona sea considerada como discapacitada. Esto quiere decir que el sujeto en cuestión tendrá dificultades para desarrollar tareas cotidianas y corrientes que, al resto de los individuos, no les resultan complicadas. El origen de una discapacidad suele ser algún trastorno en las facultades físicas o mentales.

A lo largo de los años, la discapacidad ha sido percibida de distintas maneras por la sociedad. En el siglo XX se encontraba vinculada a una cierta función que se calificaba como dañada en comparación al estado general de un sujeto. Podía tratarse de una discapacidad física, intelectual o de otro tipo, originada por un trastorno mental o por una enfermedad de características crónicas.

Partiendo de lo expuesto se subrayó que básicamente cuando se habla de discapacidad de tipo físico se refiere a la ausencia o a la disminución de capacidades motoras que tiene una persona lo que supone que le sea muy complicado llevar a cabo distintas actividades de la vida rutinaria.

Problemas durante la gestación o al nacer, así como diversos accidentes que le hayan podido causar graves daños en su cuerpo, y en concreto en zonas como la médula, son algunas de las principales causas que traen consigo que alguien tenga

una discapacidad física. Un ejemplo sería: “Manuel sufre una discapacidad como consecuencia de un accidente de tráfico que ahora le hace ir en silla de ruedas”.

También existe lo que se conoce como discapacidad psíquica. Está en concreto es la que surge en un ser humano como consecuencia de que el mismo se encuentra sufriendo cuadros de esquizofrenia, trastorno bipolar o síndrome de Asperger, entre otros. Todo ello trae consigo que aquel tenga dificultades a la hora de integrarse y de comportarse en el ámbito social.

Y finalmente posee la discapacidad mental. Con esta denominación se hace referencia a que alguien cuenta con un desarrollo y funcionamiento mental que es inferior a la media. Por todo ello, se establece que entre las personas que poseen este tipo de discapacidad se encontrarían las que tienen síndrome de Down así como todas aquellas que por problemas en el parto, por accidentes de tráfico o por lesiones prenatales hayan visto disminuida dicha capacidad intelectual.

En los últimos años, en cambio, la discapacidad comenzó a ser considerada a partir de una perspectiva de derechos humanos. El objetivo pasó a ser la integración de los discapacitados en la comunidad, facilitando esto a partir de la idea de accesibilidad.

Este último concepto (accesibilidad) contempla la adecuación de entornos y dispositivos para que las personas discapacitadas puedan usarlos de la misma manera que el resto. Para esto se necesitan considerar ciertos aspectos técnicos que reduzcan las barreras para quienes sufren de alguna discapacidad (como rampas para las personas que se movilizan en sillas de ruedas o semáforos con sonido para los no videntes). (Definiciones, s.f.)

1.1.2 Tipos de discapacidades

En la actualidad se crearon 4 tipos generales de discapacidades el cual definen cada una a un grupo de discapacidades según la similitud o el concepto del mismo, Las 4 discapacidades se muestra en el figura 1.

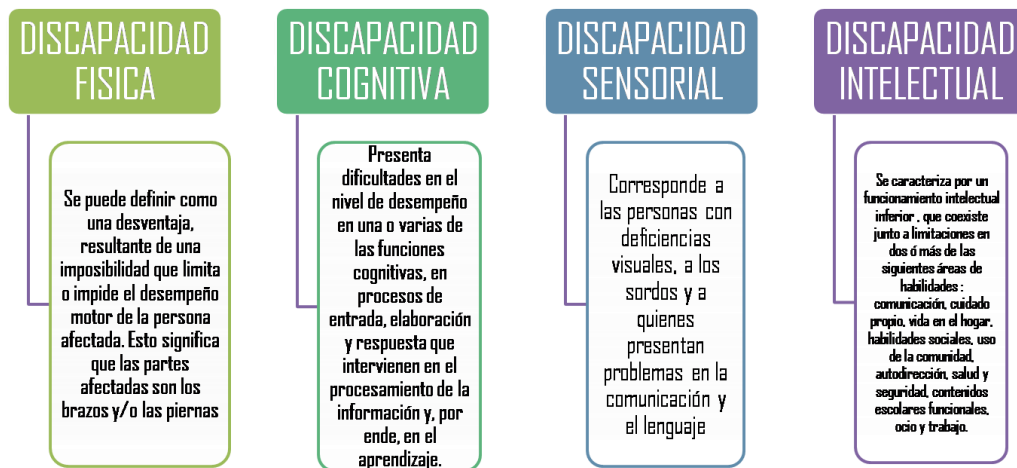


Figura 1. Los 4 Tipos de Discapacidades generalizadas en la actualidad
Fuente: CONADIS

1.2 Sistema de Base de datos (SBD):

1.2.1 Introducción:

El sistema de base de datos se considera como un fichero el cual se puede almacenar información de diferentes tipos, en este el fichero mencionado la información se puede guardar en campos o delimitadores, en él se podría almacenar el nombre y el apellido de varias personas en modos separados con esta forma se obtuvo el fichero únicamente de los nombres o apellidos de forma separada o en conjunto.

Un sistema de base de datos (SBD) tiene varios recursos para manejar grandes volúmenes de información, pero se debe tomar en cuenta que no todos los sistemas de base de datos no manejan información.

1.2.2 Características del Sistema de Base de Datos (SBD)

Un sistema de base de datos responde a las siguientes características:

- Independencia de los Datos. - Se refiere que los datos no sean dependientes del programa y de esta forma cualquier aplicación pueda hacer uso de los datos.
- Reducción de la Redundancia. - Se llama redundancia a la duplicación de datos, al optimizar asta al máximo se consiguió un aprovechamiento del espacio y también se optimizo la existencia de inconsistencias entre los datos. La inconsistencia existe cuando se encontró datos contradictorios.
- Seguridad. - El sistema de base de datos permite un control de la seguridad de los datos.

1.2.2.1 Componentes de un Sistema de Base de Datos

Un sistema de base de datos está compuesto por cuatro parámetros necesarios para su funcionamiento como los son:

- Personas
- Máquinas
- Programas
- Datos

Los Datos

Es el fundamento esencial del sistema de base de datos, es la parte fundamental el cual se puede manipular los datos en diferentes programas y aplicaciones la cual se haya almacenado.

Programas

Son los encargados de manipular los datos ingresados, tiene diferentes nombres como DBMS (Data Base Management System) o también llamados SGBD

(Sistema Gestor de Base de Datos). Los programas tienen dos funciones importantes que son las siguientes:

- La declaración o definición de las estructuras de almacenamiento de datos
- La dirección y manipulación de datos.

1.2.2.2 Distintos niveles de un Sistema de Base de Datos (SBD)

El sistema de base de datos puede ser estudiado en distintos tres niveles como lo son:

1. Nivel Físico

Es el nivel donde se almacenan los datos, en otra forma de decir cómo se almacena los datos, ya sea en registros o de la forma que se haya programado. Este nivel es muy poco común en el uso de las personas ya que deben estar calificado para su manipulación, este nivel está asociada a una representación de datos más conocida como Esquema Físico.

2. Nivel Conceptual

Este corresponde a la visión de la base de datos desde un punto de vista real, en otras palabras, se trata con la entidad u objeto presentado sin importar como esta presentado o almacenado. Este nivel también está asociado como Esquema Conceptual.

3. Nivel Visión

Es parte del esquema conceptual, este nivel presenta en conjunto toda la base de datos, El nivel de visión se encarga de dividir los segmentos con el fin de los usuarios no tengan acceso a los datos de otros usuarios. Este nivel es asociado como Esquema de Visión, como se muestra en la figura 2.

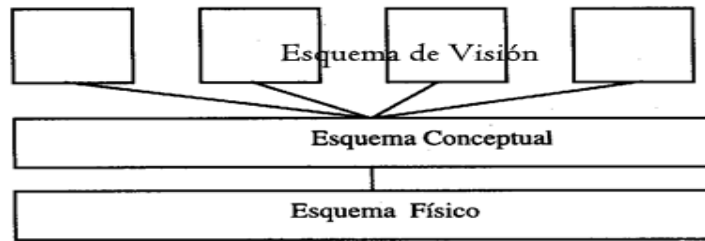


Figura 2. Arquitectura de los niveles del sistema de base de datos.
Fuente: *Introducción a la base de datos relacionales* (Domingo abarca Ramírez)

1.2.3 Plataforma XAMPP

XAMPP es un servidor independiente de plataforma de código libre. Te permite instalar de forma sencilla Apache en tu propio ordenador, sin importar tu sistema operativo (Linux, Windows, MAC o Solaris). Y lo mejor de todo es que su uso es gratuito.

XAMPP incluye además servidores de bases de datos como MySQL y SQLite con sus respectivos gestores phpMyAdmin y phpSQLiteAdmin. Incorpora también el intérprete de PHP, el intérprete de Perl, servidores de FTP como ProFTPD ó FileZilla FTP Serve, etc. entre muchas cosas más, una de las ventajas de usar XAMPP es que su instalación es de lo más sencilla, basta descargarlo, extraerlo y comenzar a usarlo.

Paquetes básicos de XAMPP:

- Apache: El servidor Web más común
- MySQL: Base de datos de código libre
- PHP y Perl: Lenguajes de programación
- ProFTPD: El servidor FTP
- Open SSL: Soporte de la capa de sockets segura

(Zapata Cristian, 2013)

1.2.3.1 MySQL:

Todo el sistema de permisos de acceso al servidor, a las bases de datos y sus tablas, MySQL lo almacena en una tabla llamada `mysql`, que como todas estará en el directorio `/data`. La principal herramienta de MySQL es `mysqladmin`, la cual como parece indicar su nombre es la encargada de la administración.

MySQL crea por defecto al usuario `root` con todos los permisos posibles habilitados, se pudo utilizar este usuario como administrador o crear otro, por ejemplo, `mysqladmi`. Como el usuario `root` lo crea sin clave de acceso, lo primero que se debe hacer es asignarle una:

```
mysqladmin -u root password "miclave"
```

A partir de ahora cualquier operación que se haga como `root` se debe especificar la clave. Hay que destacar que entre el modificador `-p` y la clave no debe haber espacios.

```
mysqladmin -u root -pmiclave
```

(Rouse, 2010)

1.2.3.2 Iniciar y detener XAMPP

En momentos es necesario iniciar, detener y reiniciar los servidores, los archivos de la configuración de Apache se cargan al dar inicio Apache, al momento de modificar algún archivo de configuración Apache, es necesario detener y reiniciar el servidor para poder cargar los archivos modificados.

Para tener un buen arranque y poder abrir el MyAdmin es necesario tener en cuenta los puertos de enlace como en Apache el puerto es 8080 y para phpMySQL 3306, con estos puertos se inició de forma efectiva la base de datos.

Para poder utilizar el servidor Apache debe ser muy necesario arrancar en panel de control XAMPP Apache, dando click en el botón Start. En la figura 3 se muestra XAMPP.

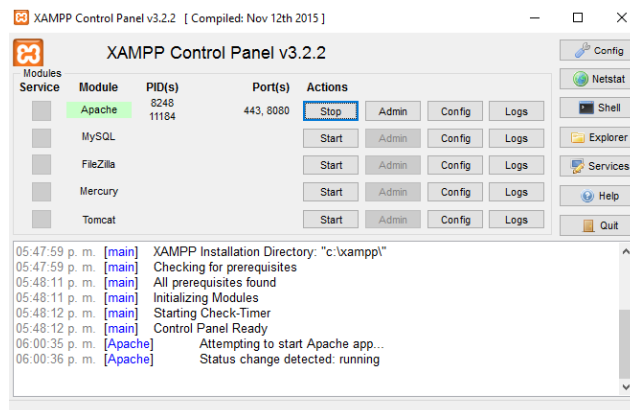


Figura 3. Iniciación del apache en XAMPP

Fuente: Autores

A su vez si el Apache no arranca o no inicia es necesario cambiar los puertos que estén por defecto a los puertos con numeración 8080, esto se logra entrando a Confing y cambiar los puertos.

De igual forma es muy necesario arrancar MySQL para poder utilizar la base de datos, dando click en el botón Start, como se muestra en la figura 4.

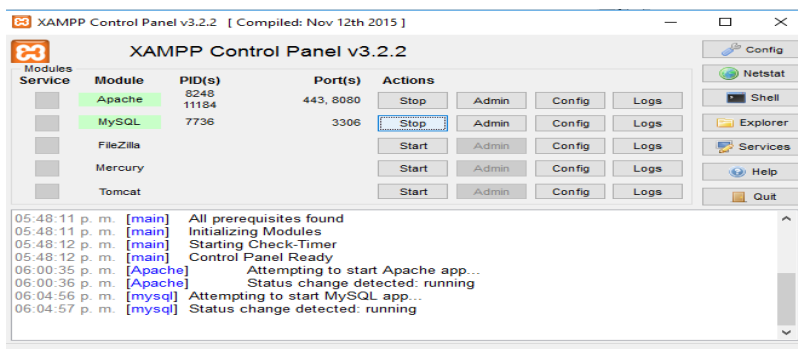


Figura 4. Inicialización del MySQL en XAMPP

Fuente: Autores

De igual forma si el MySQL no arranca o no inicia es necesario cambiar los puertos que estén por defecto a los puertos con numeración 3306, esto se logra entrando a Confing y cambiar los puertos.

1.2.3.3 Creación de la base de datos.

Después de cargar y arrancar el apache y MySQL, está listo para entrar al phpMyAdmin poniendo en el navegador localhost:8080 con esto se tiene acceso a

la base de datos y revisar todas las configuraciones echas en la base de datos, como se muestra en la figura 5.

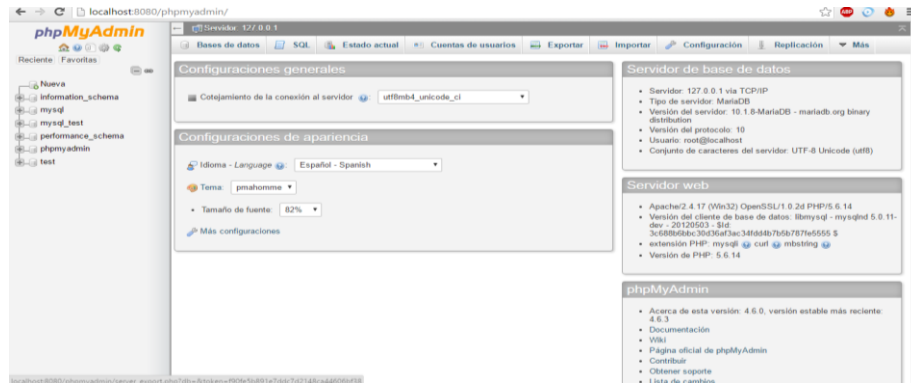


Figura 5. Interfaz del phpMyAdmin para configuración de a base de datos
Fuente: Autores

1.3 Lenguaje de programación NETBEANS

1.3.1 Introducción

Es un entorno de desarrollador gratuito con un código abierto que permite el uso de un amplio rango de tecnologías como desarrollador para escritorio, como aplicaciones Web o también dispositivos móviles. Entre sus tecnologías de desarrollo existen: Java, PHP, Groovy, C/C++, HTML etc. Su instalación es compatible para diferentes sistemas operativos: Windows, Linux, Mac OS, etc. como se muestra en la figura 6.



Figura 6. Entorno desarrollador abierto para algunos sistemas operativos
Fuente: Netbeans Java

1.3.2 Desarrollador Java Neatbeans

Java se ha caracterizado por ser un lenguaje de programación con propósito general, concurrente, orientado a objetos proponiendo su independencia de implementos como fuera posible. Su propósito es que los desarrolladores escriban una programación una sola vez y pueda ser ejecutado en cualquier dispositivo, es decir que el código de programación que fue ejecutado en una plataforma no debería ser copilado correr otra vez.

Es necesario para el uso de Java en el desarrollo de Netbeans las clases JDK, junto con sus complementos de librerías y bibliotecas estándar.

1.3.2.1 Estructura de la programación Java

```
public class ClasePrincipal {  
    public static void main(String[ ] args) {  
        sentencia_1;  
        sentencia_2;  
        // ...  
        sentencia_N;  
    }  
}
```

1.3.2.2 Programación detallada en java

Como se muestra en la figura 7.

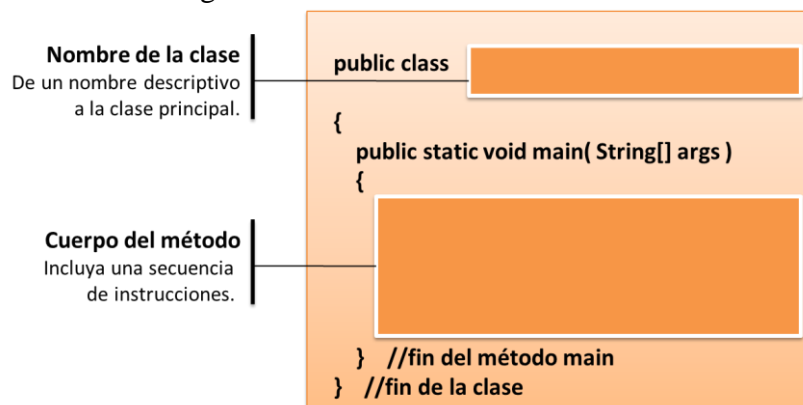


Figura 7. Estructura básica detallada en java
Fuente: Neatbeans Java

Comentario. El programa comienza con un comentario. El delimitador de inicio de un comentario es `/*` y el delimitador de fin de comentario es `*/`. Para que el programa escriba “Hola Mundo” en la consola se utilizó el método `System.out.println()`. Los comentarios son ignorados por el compilador y son útiles para el programador. Los comentarios ayudan a explicar aspectos relevantes de un programa y lo hacen más legible. En un comentario se puede escribir todo lo que desee, el texto puede ser una o más líneas. (Guevara, 2011)

Definición de clase. La primera línea del programa, después del primer comentario. Define una clase, la definición de la clase comienza por el carácter `{` y termina con el carácter `}`. El nombre de la clase lo define el programador.

Definición del método. Después de la definición de clase se escribe la definición del método `main ()`. Todos los programas Java deben incluir un método `main ()`. Este método indica las sentencias a realizar cuando se ejecuta un programa. Un método es una secuencia de sentencias ejecutables. Las sentencias de un método quedan delimitadas por los caracteres `{` y `}` que indican el inicio y el fin del método, respectivamente.

Sentencia. Dentro del método `main ()` se incluye una sentencia para mostrar un texto por la consola. Los textos siempre del lenguaje. Todas las sentencias de un programa Java deben terminar con el símbolo punto y coma. Este símbolo indica el comparador que ya ha finalizado una sentencia.

1.3.2.3 Identificadores

Los identificadores son nombres asignados a variables, métodos. Clases en el código fuente de la programación. Todo identificador nuevo que se vaya a implementar en una programación Java de ser identificadas previamente a su utilización, como se muestra en la Tabla 1.

Tabla 1. Condiciones para programación en cada clase en Android

abstract	do	implements	protected	throw
boolean	double	import	public	throws
break	else	instanceof	rest	transient
byte	extends	int	return	true
case	false	interface	short	try
catch	final	long	static	void
char	finally	native	strictfp	volatile
class	float	new	super	while
const	for	null	switch	
continue	goto*	package	synchronized	
default	if	private	this	

Fuente: Programación en Java, García Beltrán y J.M Arranz (2009)

1.4 Controlador del sistema PHP

1.4.1 Sistema operativo Brackets

1.4.1.1 Definición

Brackets es un editor de código abierto para el diseño y desarrollo web construido sobre tecnologías como HTML, CSS y JavaScript. El proyecto fue creado y es mantenido por Adobe , y se distribuye bajo una licencia MIT .

Brackets se diferencia de los demás editores gracias a la facilidad de mostrar el código específico de acuerdo al contexto usado, esto mediante su interfaz tipo Quick Edit UI.

Brackets te permite trabajar directamente en el navegador editando el código al instante, estableciendo breakpoints y moviéndose con fluidez entre las diferentes vistas de código y del mismo navegador. El editor de texto dentro de Brackets se encuentra basado en CodeMirror.

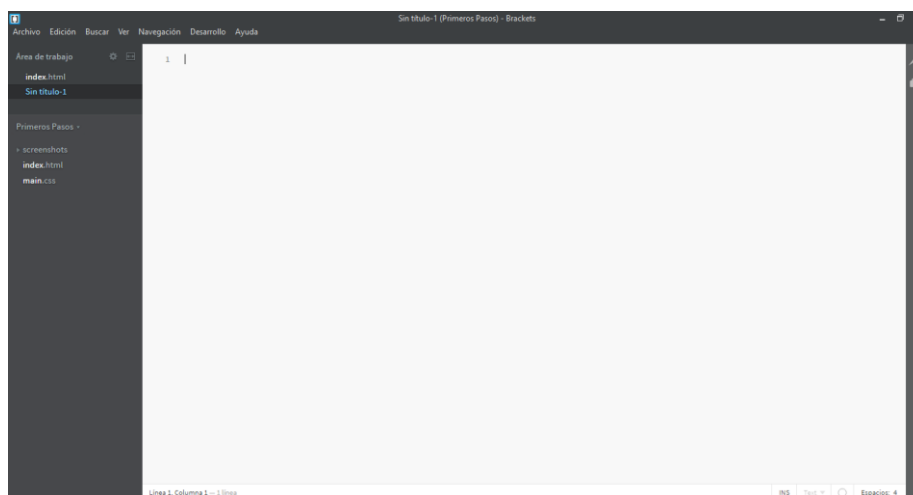
1.4.1.2 Características

- Útil sin hacerse cargo. Con características como la vista previa y edición rápida, los soportes agilizan el desarrollo sin ponerse en tu camino
- Construido con la web para la web. Hecho con mucho amor y JavaScript, Brackets es un proyecto de código abierto, con una activa comunidad de desarrolladores y en crecimiento.

- Un editor de código para la web. Centrarse en una cosa, y hacerlo muy bien; Brackets está construido desde cero con un enfoque en el desarrollo web.
 - Tienes el poder. Brackets está construido con HTML, CSS y JavaScript, lo que significa que es fácil de contribuir el proyecto y extender su funcionalidad.
 - Las herramientas no deben ser obstáculo. En lugar de un espacio de trabajo desordenado con paneles flotantes, barras de herramientas e íco, Brackets se centra en la prestación de “edición rápida “, vistas en línea que proporcionan acceso contextual a su contenido.
 - Funciona con el navegador. El navegador es su punto de vista del diseño. Brackets conecta directamente en el navegador, lo que le permite diseñar y desarrollar en el mismo entorno en el que se implemente.
- (Martínez, 2013)

1.4.1.3 Creación de un nuevo proyecto

Los pasos son sencillos solo se da click en nuevo y enseguida tiene una página en blanco para editar, esto queda como se muestra en la figura 8.



*Figura 8. Ventana de edición de programa en Breakets
Fuente: Autores*

Este lenguaje de programación servirá para poder conectar la aplicación echa en Android Studio con el sistema de base de datos XAMPP.

1.5 Android Studio

1.5.1 Introducción

Se puede decir que es un entorno de desarrollo integrado (IDE), que está basado en IntelliJ IDEA de una compañía JetBrains, el cual proporciona varias mejoras con respecto a un plugin ADT (Android Developer Tools) para Eclipse. Android Studio utiliza una licencia de software libre llamado Apache 2.0, está programado en Java y es un multiplataforma, como se muestra en la figura 9.



Figura 9. Logo de software Android Studio
Fuente: Autores

1.5.1.1 Creación de un proyecto en Android Studio

Inicialmente click en:

- ✓ “Start a new Android Studio project”, como se muestra en la figura 10.

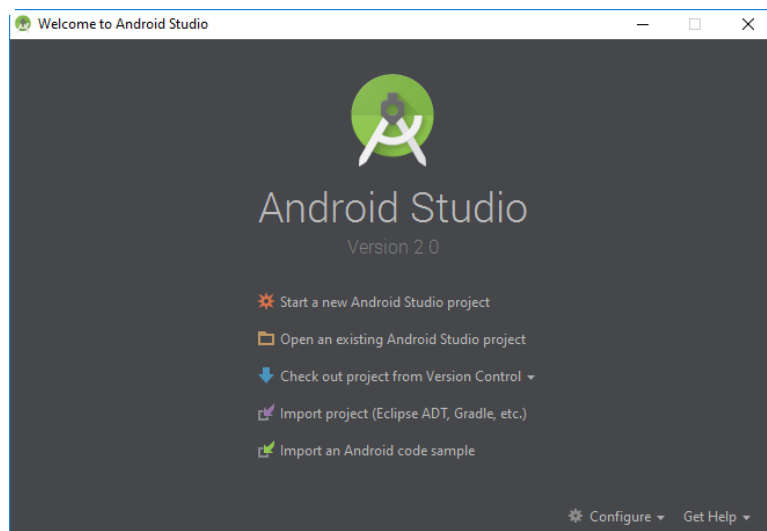


Figura 10. Presentación de entrada para una selección de nuevo proyecto
Fuente: Autores

El asistente de creación del proyecto guiará por las distintas opciones de creación y configuración de un nuevo proyecto Android.

- ✓ Enseguida se llenó los campos como nombre del proyecto, el dominio, y la ruta donde crear el proyecto, como se muestra en la figura 11.

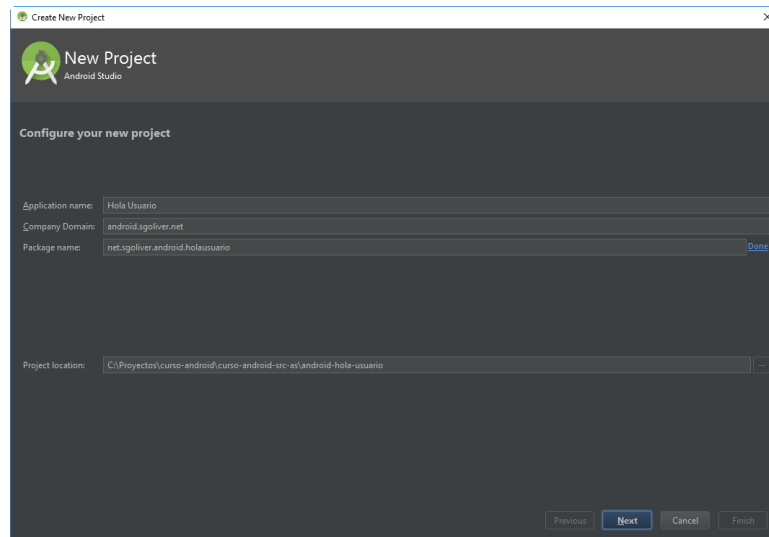


Figura 11. Campos de información para un nuevo proyecto
Fuente: Autores

- ✓ Se configuro las plataformas y APIs que va a utilizar nuestra aplicación, se encontró aplicaciones para teléfonos y tablets, como se muestra en la figura 12.

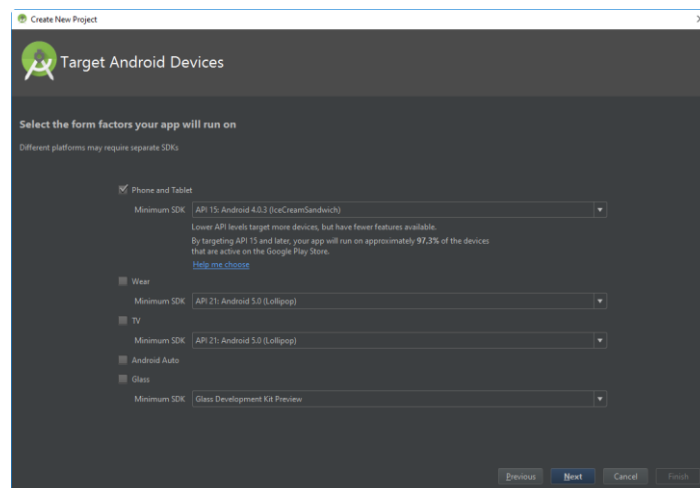


Figura 12. Configuración de las plataformas para nuestra aplicación
Fuente: Autores

- ✓ La versión mínima que se seleccionó en esta pantalla implicará que nuestra aplicación se pueda ejecutar en más o menos dispositivos. De esta forma, cuanto menor sea esta, a más dispositivos podrá llegar nuestra aplicación, como se muestra en la figura 13.

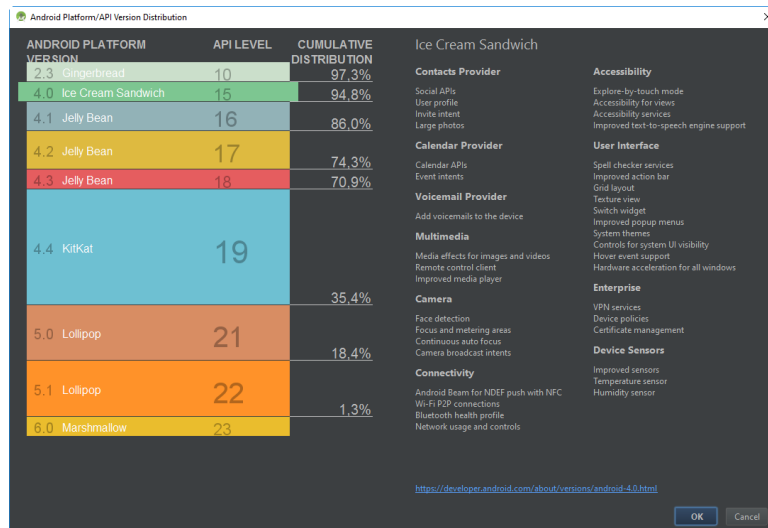


Figura 13. Configuración de la selección de sistema Android
Fuente: Autores

- ✓ En la siguiente figura en el asistente se eligió el tipo de actividad principal de la aplicación. Se puede decir que por actividad es una ventana o pantalla de la aplicación, como se muestra en la figura 14.

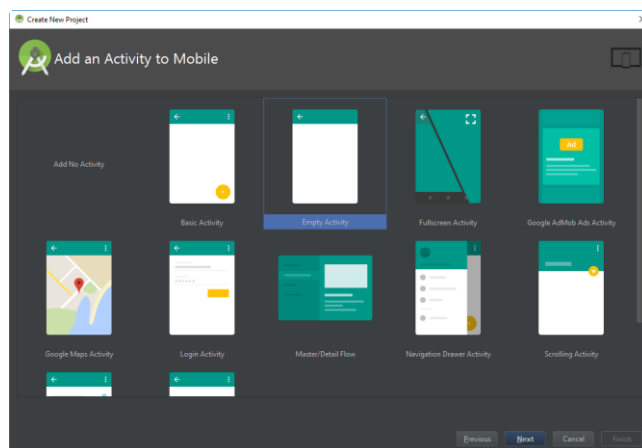


Figura 14. Elección del asistente para la actividad principal de la aplicación.
Fuente: Autores

- ✓ Para el siguiente paso se indica los datos asociados a esta actividad que hace un momento se eligió, indicando el nombre de su clase con java asociada esto así (Activity Name) y el nombre layout xml (Algo referido a una interfaz gráfica de una actividad seleccionada), como se muestra en la figura 15.

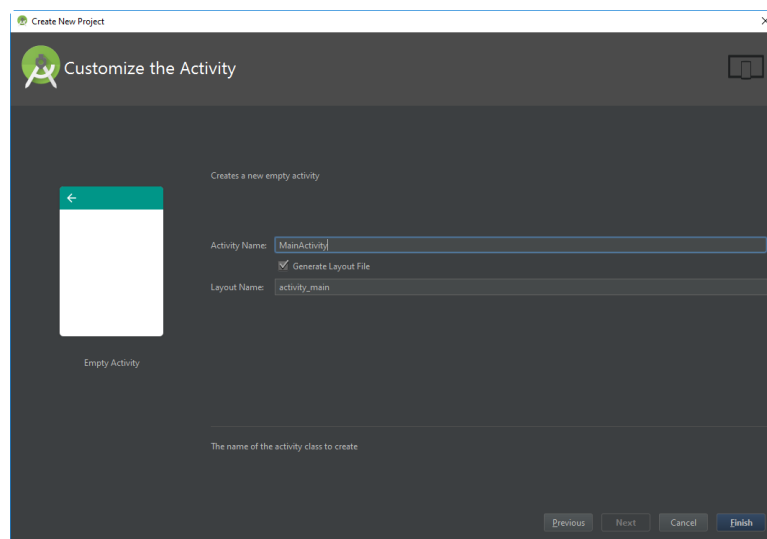


Figura 15. Se coloca el nombre de la actividad del proyecto
Fuente: Autores

- ✓ Una vez de haber configurado todo se da un click en el botón Finish esto hará que Android Studio cree para que la estructura del proyecto configurado junto con los elementos dispensables necesarios, como se muestra en la figura 16.

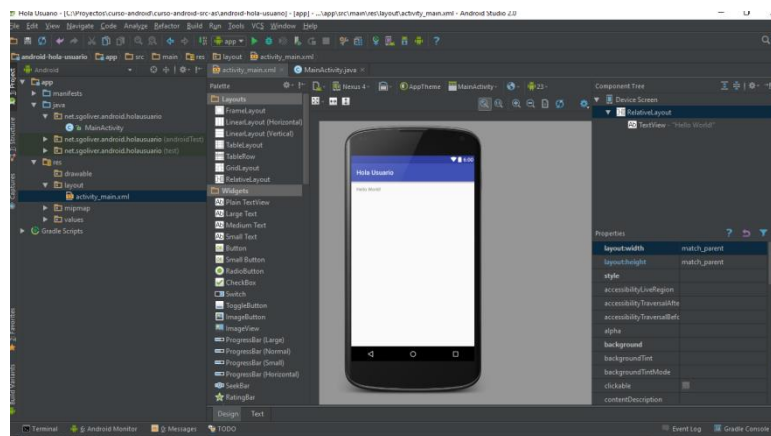


Figura 16. Estructura del proyecto para ser programado
Fuente: Autores

Al momento de terminar toda la iniciación del nuevo proyecto se activan tres indispensables herramientas de Android Studio como es lo es: Activity_main.xml, MainActivity.java y AndroidManifest, los cuales son necesarios para el proyecto.

- ✓ Activity_main.xml es donde se diseñe la interfaz de la aplicación de forma gráfica o de forma textual, como se muestra en la figura 17.

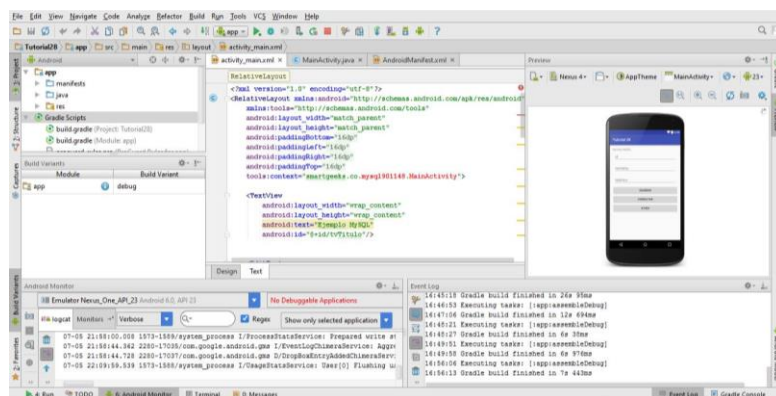


Figura 17. Ejecución del Activity_main.xml configuración de la interfaz
Fuente: Autores

- ✓ MainActivity.java es donde a realizar toda la programación de la aplicación Android, donde se modifica a la conveniencia del proyecto, como se muestra en la figura 18.

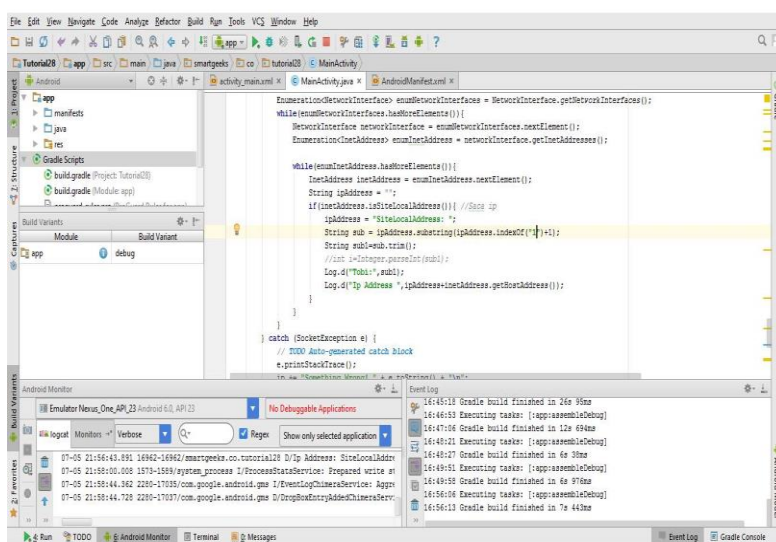


Figura 18. Ejecución del MainActivity.java va toda la programación.
Fuente: Autores

- ✓ AndroidManifest es donde permite todos los permisos de programación de forma general de la aplicación, como se muestra en la figura 19.

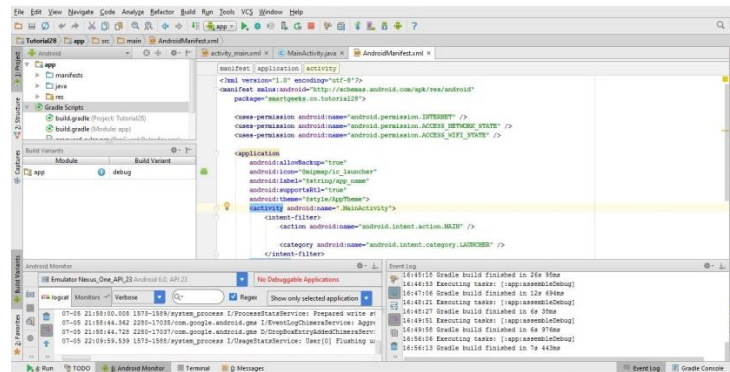


Figura 19. Ejecución de AndroidManifest da permisos de programación
Fuente: Autores

1.5.1.2 Llamar las clases de java por Android Studio (Activity)

Para la creación de la aplicación por medio de Android Studio, se debe llamar las clases creadas en Java Neatbeans con sus librerías y su programación, aquí se realiza la interfaz gráfica de la aplicación del proyecto de investigación.

- ✓ Retire incluir 'app' forma settings.gradle
- ✓ Eliminar el directorio de aplicaciones
- ✓ Reemplazar build.gradle con el ejemplo del final de este post (IntelliJ crea similar)
- ✓ Crear jerarquía de carpetas para su código Java (src / main / java)

Como se muestra en la figura 20.

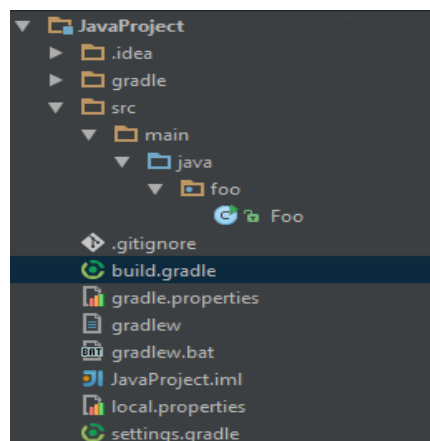


Figura 20. Llamado de las clases creadas en Java Neatbeans
Fuente: Autores

- ✓ Seleccione Editar configuración del menú desplegable donde normalmente se inicia proyecto
- ✓ Haga clic en Agregar nueva configuración y seleccione Aplicación como se muestra en la figura 21

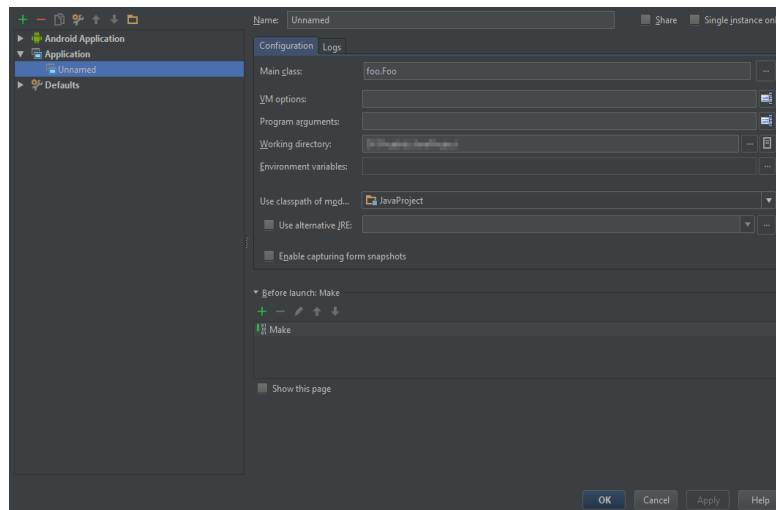


Figura 21. Seleccionador de la aplicación para ser utilizado en Android Studio
Fuente: Autores

- ✓ En la clase principal punto de su clase principal.

1.5.1.3 Código en Android Studio para llamar una de clases en Java

aplicar el plugin : 'java'

```
sourceCompatibility = 1.8
versión = '1.0'
```

```
repositorios {
    mavenCentral ()
}
```

```
dependencias {
    grupo testCompile : 'junit' , nombre : 'junit' , versión : '4.11'
}
```

(Dmitriev, 2015)

1.6 Red inalámbrica (WiFi) interna para el proyecto STAPD

1.6.1 Introducción

Como el nombre lo indica, es una red con dos o más terminales se puedan conectar y comunicar sin la necesidad de cables. La red inalámbrica puede estar conectado mientras se desplazar dentro de la una determinada zona geográfica esta es la ventaja de la red inalámbrica “La movilidad”.

Las redes inalámbricas utilizan las ondas electromagnéticas en lugar de un cable estándar. Hay muchas tecnologías que utilizan la frecuencia como medio de enlace y transmisión.

1.6.1.1 Redes inalámbricas WiFi

La tecnología WiFi ayuda crear redes informáticas inalámbricas (Wreless). En un protocolo IEEE llamado 802.11. Es una forma de conexión con dispositivos electrónicos de una forma inalámbrica, estos dispositivos electrónicos pueden ser una computadora, un SmartTV, un Smartphone, Smartmusic etc. Estos pueden conectarse al internet por un punto de acceso de red inalámbrica (AP). La distancia de la frecuencia de conexión esta entre los 20 metros en interiores y al aire libre es mayor.

1.6.1.2 Rango y Flujo de Datos

Los estándares 802.11^a, 802.11b y 802.11g, llamados “estándares físicos”, son modificaciones del estándar 802.11 y operan de modos diferentes, lo que les permite alcanzar distintas velocidades en la transferencia de datos según sus rangos. (CCM, 2016), como se muestra en la Tabla 2.

Tabla 2. Estándar de protocolos 802.11 a, 802.11b, 802.11g.

Estándar	Frecuencia	Velocidad	Rango
WiFi a (802.11a)	5 GHz	54 Mbit/s	10 m
WiFi B (802.11b)	2,4 GHz	11 Mbit/s	100 m
WiFi G (802.11g)	2,4 GHz	54 Mbit/s	100 m

Fuente: *Introducción a WiFi (802.11 o WiFi)*, Enciclopedia CCM (2016)

1.6.2 Recursos para la Implementación de la Red Inalámbrica

1.6.2.1 Router

Es un dispositivo de red el cual no permite enrutamiento de los paquetes de redes independientes. El enrutamiento es realizado de acuerdo a las reglas y conjuntos que forma la tabla de enrutamiento. Su función operable es realizada en la capa 3 del modelo OSI.

La función de enrutamiento es direccionar las IP en función de sus direcciones de red ya definidas por la máscara de subred y las asigna junto al algoritmo de enrutamiento y la tabla asociada. Estos protocolos de enrutamiento son creados conjuntamente a la arquitectura de nuestra red y los enlaces de comunicación entre los sitios y las redes.

1.6.2.2 Switch

Es dispositivo con el propósito especial de resolver problemas de rendimiento en la red local, puede ser por el ancho de banda y embotellamiento de datos. El switch puede ayudar proporcionando un ancho de banda, acelerar la salida y entrada de los paquetes, bajar el tiempo de espera y reduciendo el costo por cada puerto.

Este dispositivo opera en la capa 2 del modelo OSI y renvía paquetes en base a la dirección MAC.

Switch realiza un segmenta miento económico de la red dentro de pequeños dominios de colisiones, así reduciendo el porcentaje de colisiones por su alto porcentaje de ancho de banda, Este dispositivo no es diseñado como un control sobre la red o como la fuente de seguridad, redundancia o manejo.

1.6.2.3 Cable de Red

Este cable tiene una similitud como los cables de teléfono solo que un poco más ancho, El cable común es de ocho cables internos de diferentes colores llamado UTP, existe dos tipos de cables de red el cruzado y el directo necesarios para un diseño de una red.

El conector para ambos extremos permite conectar a diferentes equipos entre si, este conector tiene dos diferentes nombres, se les conoce como RJ45, donde “RJ” significa (registerd Jack) “conector certificado”, pero su nombre correcto es 8P8C el cual significa 8 pines, 8 contactos. Cada uno de estos pines es conectado a uno de los ocho cables dentro UTP.

1.6.3 Equipos para la Implementación de la Red Inalámbrica

1.6.3.1 Router Cisco Linksys E4200

Este router Cisco Linksys E4200 es el tope de gama de la marca, se cita como características principales su LAN y WAN Gigabit, su doble banda simultánea con velocidades de hasta 300Mbps en la banda de los 2.4GHz y velocidades de hasta 450Mbps en la banda de los 5GHz. También incluye un puerto USB para conectar discos duros o impresoras. (Cisco, 2013), como se muestra en la figura 22.



*Figura 22. Router Cisco Linksys E4200, proporciona red inalámbrica STAPD.
Fuente: Autores*

Características:

- ✓ CPU: Broadcom BCM4718A1KFBG a 480MHz
- ✓ Memoria RAM: Hynix H5PS5162FFR de 64MB de capacidad.
- ✓ Memoria FLASH: Winbond 25Q128BVFG de 16MB de capacidad
- ✓ Ethernet: Broadcom BCM53115SKFBB conectividad Gigabit.
- ✓ Fuente de alimentación: Necesita un voltaje de 12V y una intensidad de corriente de 2Amp para alimentar al propio equipo y a los dispositivos USB que se conecta.
- ✓ Antenas: Posee 3 antenas para la banda de los 2.4GHz en configuración 2T3R MIMO y para la banda de los 5GHz posee 3 antenas en configuración 3T3R MIMO con Three Stream para conseguir los 450Mbps. En la banda de los 2.4GHz no está disponible el Three-Stream.
- ✓ Banda 5GHz: Broadcom BCM4331KMLG.
- ✓ Amplificadores WiFi: Para la banda de los 2.4GHz posee dos amplificadores SiGe 2528L045CA.
- ✓ Para la banda de los 5GHz lleva tres amplificadores. (Cisco, 2013)

1.6.3.2 Switch cisco Catalyst 2960

Los switches Cisco Catalyst 2960 son switches Gigabit Ethernet (10/100/1000) apilables de configuración fija que ofrecen conectividad de red para grandes y medianas empresas, y sucursales. Permiten realizar operaciones empresariales de manera confiable y segura con un menor costo total de propiedad a través de diversas características innovadoras, tales como Cisco FlexStack-Plus, visibilidad y control de aplicaciones, Power over Ethernet Plus (PoE+), revolucionarias funciones de administración de energía y Smart Operations. (Cisco, 2013), como se muestra en la figura 23, se puede observar el equipo de forma real que facilitara la comunicación entre los diferentes router q se posee dentro de la red STAPD, los mismo que ayudaran a la obtención de los diferentes rangos de ip para poder dar un mensaje de alerta a las personas que quieren trabajar con esotos equipos.



Figura 23. Switch Cisco Catalyst 2960, configuración puertos.
Fuente: Autores

Características:

- ✓ Los switches Cisco Catalyst 2960-S y 2960-SF vienen en modelos de 24 y 48 puertos. El 2960-S.
 - ✓ Ofrece conectividad Gigabit Ethernet (10/100/1000) con enlaces de interconexión Small Form-Factor Pluggable (SFP/SFP+) de 10 G/1 G.
 - ✓ El 2960-SF ofrece conectividad Fast Ethernet (10/100) con enlaces de interconexión de 1 G SFP.
 - ✓ Estos switches de categoría empresarial admiten los dos conjuntos de funciones de software siguientes:
 - ✓ El software LAN Base incluye funciones avanzadas de capa 2 (L2), apilamiento FlexStack, enrutamiento estático y PoE+, y suele destinarse a implementaciones realizadas en grandes empresas.
 - ✓ El software LAN Lite incluye funciones de capa 2 de nivel básico y PoE+, y suele destinarse a implementaciones realizadas en medianas empresas.
- (Cisco, 2013)

1.6.3.3 Cable UTP con conectores RJ45

Un cable es un cordón que está resguardado por alguna clase de recubrimiento y que permite conducir electricidad o distintos tipos de señales. Los cables suelen estar confeccionados con aluminio o cobre.

UTP, por otra parte, es una sigla que significa *Unshielded Twisted Pair* (lo que puede traducirse como “Par trenzado no blindado”). El cable UTP, por lo tanto, es una clase de cable que no se encuentra blindado y que suele emplearse en

las telecomunicaciones. (Copyright, 2008 - 2016), como se muestra en la figura 24, un trozo de cable de 5 metros que se utiliza para poder conectar el servidor apache con un puerto específico para administrar los datos que se almacenan o se recuperan de la base de datos.



*Figura 24. Cable UTP con conector RJ45, conexión entre APs.
Fuente: Autores*

Conector RJ45

RJ-45 es una interfaz física comúnmente utilizada para conectar redes de computadoras con cableado estructurado (categorías 4, 5, 5e, 6 y 6a). Posee ocho pines o conexiones eléctricas, que normalmente se usan como extremos de cables de par trenzado(UTP).

Es utilizada comúnmente con estándares como TIA/EIA-568-B, que define la disposición de los pines (patillaje) o wiring pinout.

Una aplicación común es su uso en cables de red Ethernet, donde suelen usarse cuatro pares (ocho pines). Otras aplicaciones incluyen terminaciones de teléfonos (dos pares), por ejemplo: en Francia y Alemania, y otros servicios de red como RDSI, T1 e incluso RS-232. (Información, 2010)

1.6.4 Subneteo y Configuración de la Red Inalámbrica

1.6.4.1 Subnetear

Subnetear es la acción de tomar un rango de direcciones IP donde todas las IPs sean locales unas con otras y dividir las en diferentes rangos, o subnets, donde las direcciones IP de un rango serán remotas de las otras direcciones.

Si tú quieres determinar cuántos hosts tú tienes en un rango IP, primero debes determinar cuántos hosts bits se obtiene. Se toma el ejemplo anterior de 131.107.2.4 y 255.255.255.0. Ya establece anteriormente que el network ID es 131.107.2 y el host ID es 4. En otras palabras, se tiene 3 octetos para el Network ID y uno – un octeto – para el Host ID.

La división en subredes permite crear múltiples redes lógicas de un único bloque de direcciones. Como se usa un router para conectar estas redes, cada interfaz en un router debe tener un ID único de red.

Cada nodo en ese enlace está en la misma red. Se creó las subredes utilizando uno o más de los bits del host como bits de la red. Esto se hace ampliando la máscara para tomar prestado algunos de los bits de la porción de host de la dirección, a fin de crear bits de red adicionales. Cuanto más bit de host se usen, mayor será la cantidad de subredes que puedan definirse. Para cada bit que se tomó prestado, se duplica la cantidad de subredes disponibles. Por ejemplo: si se toma prestado 1 bit, es posible definir 2 subredes. Si se toman prestados 2 bits, es posible tener 4 subredes. Sin embargo, con cada bit que se toma prestado, se dispone de menos direcciones host por subred.

El primer Router a ser configurado posee dos interfaces para interconectar dos redes. Dado un bloque de direcciones 192.168.1.0 /24, se crearán dos subredes. Se toma prestado un bit de la porción de host utilizando una máscara de subred 255.255.255.128, en lugar de la máscara original 255.255.255.0. El bit más

significativo del último octeto se usa para diferenciar dos subredes. Para una de las subredes, este bit es "0" y para la otra subred, este bit es "1". (Recoleta, 2011)

1.6.4.2 PuTTY

Como se muestra en la figura 25.



*Figura 25. software para la configuración de los routers
Fuente: Autores*

PuTTY es un cliente SSH y Telnet con él se puede conectar a los servidores remotos iniciando una sesión en ellos que permite ejecutar comandos. El ejemplo más claro es cuando se inicia PuTTY para ejecutar comandos en un servidor VPS y así poder instalar algún programa o configurar alguna parte del servidor.

La parte de cliente Telnet es más desconocida pero el ejemplo más claro de uso es conectarse a nuestro router domestico para configurarlo a través de Telnet y abrir puertos, etc.

Ventajas de PuTTY:

- ✓ Es gratuito y de código abierto.
- ✓ Disponible para varias plataformas (Windows y Linux).
- ✓ Es una aplicación portable.
- ✓ Interfaz sencilla y manejable.
- ✓ Muy completo y ofrece una gran flexibilidad con multitud de opciones.
- ✓ Está en constante desarrollo.

CAPITULO II

1. METODOLOGÍA

2.1 Tipo de estudio

2.1.1 Descriptivo

El tipo de investigación es descriptiva, puesto que se observa y describe el comportamiento de las personas al utilizar el sistema de telecomunicaciones de ayuda a personas con discapacidad (STAPD)

2.2 Métodos, Técnicas e Instrumentos

2.2.1 Métodos

2.2.1.1 Investigación Predictiva Experimental

El método utilizado en este proyecto es una investigación predictiva experimental, debido a que el proyecto será implementado en el edificio principal de la facultad de ingeniería de la Universidad Nacional de Chimborazo, para brindar ayuda a las personas discapacitadas a guiarse dentro del edificio.

2.3 Técnicas

2.4 Población y muestra

La población en este caso son las personas con diferentes discapacidades generales como Físicas, Cognitivas, Sensoriales e Intelectual que visiten la edificación principal de la facultad de Ingeniería de la Universidad Nacional de Chimborazo, en las que se realizarán las pruebas del sistema propuesto en el presente documento.

2.5 Hipótesis

El Sistema de Telecomunicaciones para los dispositivos electrónicos de personas con discapacidades de la Universidad Nacional de Chimborazo, ayudara la orientación, ubicación e interacción entre los usuarios.

2.6 Operacionalización de las variables

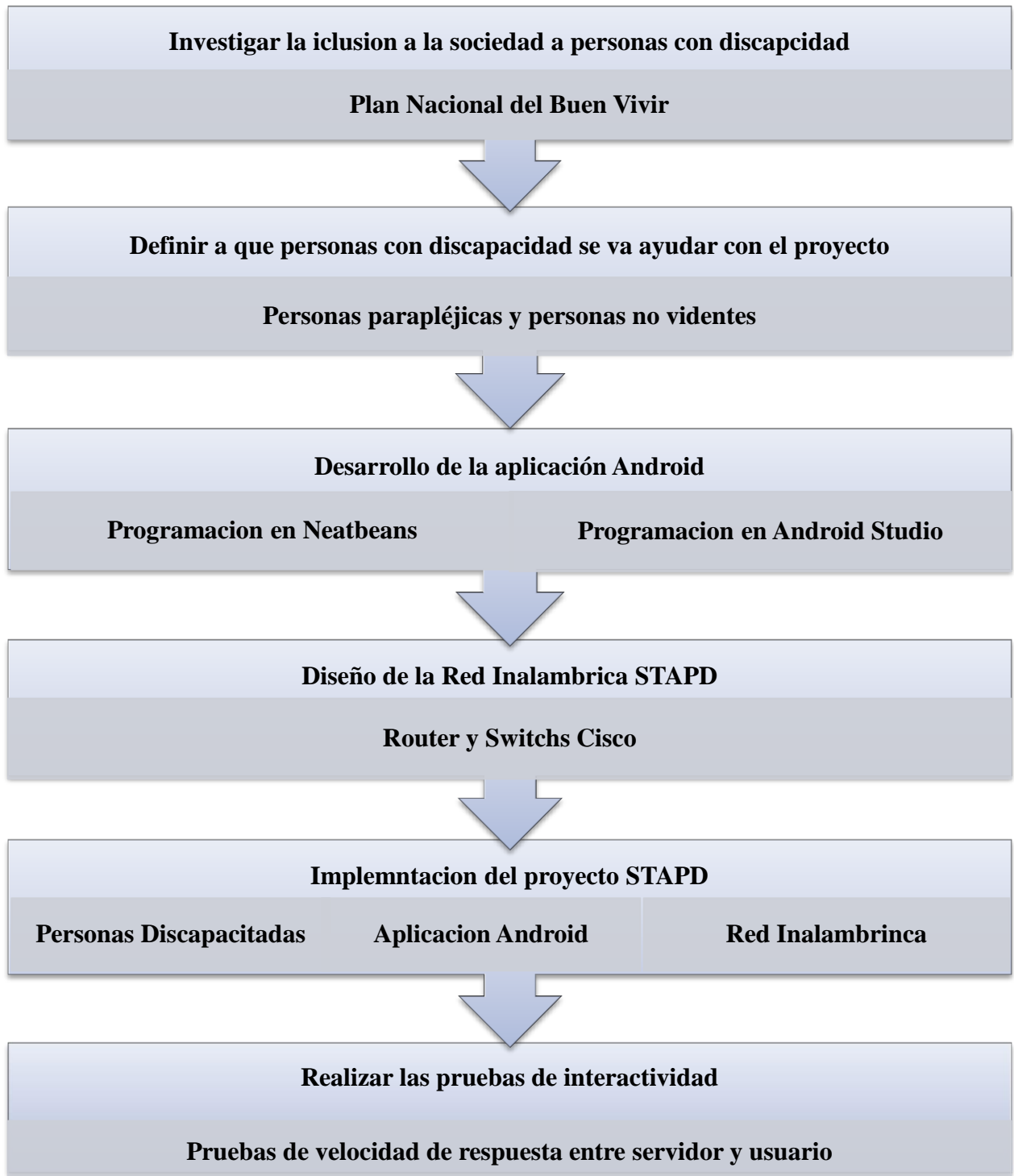
VARIABLE INDEPENDIENTE

VARIABLE	DIMENSIÓN	INDICADORES	INSTRUMENTOS
SISTEMA DE TELECOMUNICACIÓN PARA LOS DISPOSITIVOS ELECTRÓNICOS DE PERSONAS CON DISCAPACIDADES DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO	<ul style="list-style-type: none"> • Calidad de servicio en la comunicación • Confiabilidad de servicio en la comunicación 	<ul style="list-style-type: none"> • Tiempo de respuesta entre usuarios y base de datos • Tiempo de respuesta entre base de datos y usuarios • Disponibilidad permanente del servicio 	Aplicación Base de Datos Temporizadores

VARIABLE DEPENDIENTE

VARIABLE	DIMENSIÓN	INDICADORES	INSTRUMENTOS
AYUDA EN LA ORIENTACIÓN, UBICACIÓN E INTERACCIÓN ENTRE LOS USUARIOS.	<ul style="list-style-type: none">• Orientación• Ubicación• Interacción	<ul style="list-style-type: none">• Mensaje de alerta para conocer la orientación por medio de un rango de direcciones IPs.• Mensajes de alerta con información de la ubicación del usuario que necesita ayuda	Aplicación Base de Datos Temporizadores

2.7 Procedimientos



2.8 Procedimiento y análisis

2.8.1 Discapacidades que utiliza el sistema STAPD.

Tomando en cuenta la infinidad de discapacidades que existe en la actualidad, la aplicación móvil Android registra los cuatro tipos de discapacidades generales que son: Discapacidades Físicas, Discapacidad Cognitiva, Discapacidad Sensorial y Discapacidad Intelectual, con esto enfrasca las discapacidades más comunes los cuales puedan utilizar el sistema STAPD.

En el caso de discapacidades Físicas el sistema STAPD puede ayudar a las personas en silla de ruedas, personas con muletas, personas con monoplejías, etc. De igual forma las discapacidades cognitivas el sistema puede ayudar a las personas de lento aprendizaje y personas con síndrome de Down. En el caso de las discapacidades sensoriales el sistema puede ayudar a las personas de poca visión, personas sordas, personas mudas, personas con poca audición, personas con dificultad de lenguaje y de comunicación, y por último la discapacidad intelectual el sistema puede ayudar a personas con intelectual inferior que carecen de las siguientes habilidades como: comunicación, cuidado propio, sociales, salud y seguridad.

Con estas características de las discapacidades antes mencionadas, el sistema de telecomunicaciones STAPD obtiene la ubicación de una persona discapacitada la cual haya pedido ayuda y a su vez se encuentre en el edificio principal de la facultad de Ingeniería de la Universidad Nacional de Chimborazo, con este propósito las personas sin discapacidad tengan la facilidad de saber dónde se encuentre aquella persona y brindar una ayuda oportuna.

Al momento de que una persona con discapacidad tenga la necesidad de una ayuda de movilización o de diferente ayuda puede acudir a la aplicación móvil STAPD, la cual dando en el botón llamativo de color rojo con el título “Ayuda”, una pulsación este inmediatamente obtiene la dirección IP del dispositivo la cual este utilizando, enseguida esta se registra en la base de datos esperando si alguna

persona sin discapacidad pueda acudir a la ayuda, al momento que esto suceda la persona sin discapacidad dando una pulsación en el botón azul con el título de “Dar Ayuda”, de igual manera obtiene la dirección IP del dispositivo que esté utilizando y de esta forma compara la IP con la IP más parecida en la base de datos dando una alerta con el mensaje de la ubicación de la persona discapacitada.

2.8.2 Desarrollo de la aplicación Android

2.8.2.1 Programación en Java Netbeans

El lenguaje de programación Java Netbeans proporciona asistentes y plantillas que le permita crear Java EE, Java SE y aplicaciones Java ME. Las variedades de tecnologías y marcos son compatibles fuera de la caja. Por ejemplo, se puede utilizar el asistente y plantillas para crear aplicaciones que utilizan la infraestructura OSGi o el sistema de módulos de NetBeans como base de aplicaciones modulares.

El editor NetBeans en cuenta el idioma detecta errores mientras se escribe y le ayuda con las ventanas emergentes de documentación y realización -todos códigos inteligentes con la velocidad y la simplicidad de su editor de texto ligero favorito, a continuación, se detalla cada una de las instrucciones utilizadas en este proyecto.

2.8.2.2 Creación y conexión de la base de datos XAMPP

En el lenguaje de programación Java Netbeans, tiene la facilidad de crear de forma directa la base de datos referenciando con el nombre y el puerto de acceso ya configurado anteriormente en XAMPP, a continuación, se detalla las instrucciones utilizadas en el proyecto en el Anexo 1.

En la herramienta desing en netbeans se tiene la interfaz de botones, con las características de botones como panel, jButtonconnection, jButtonregister, jButtonobtenencion, jButtondelete, que permitio la iniciativa de poder tomar un set on click como se muestra en la Figura 26.

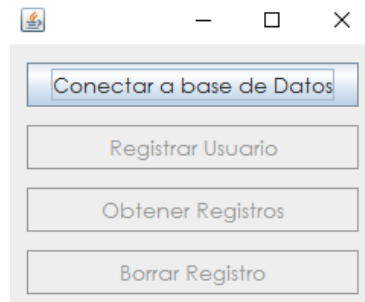


Figura 26. Cuadro de inicio con rango de botones en Java Neatbeans
Fuente: Autores

Este pequeño código permite la creación de la base de datos en XAMPP con el nombre de STAPD y en la tabla con el nombre de registro, como se muestra en la figura 27.



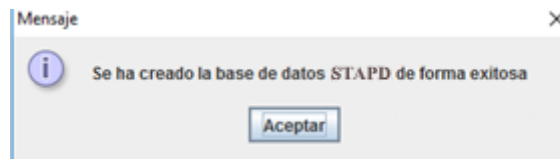
Figura 27. Base de Datos creada junto con la tabla de registro
Fuente: Autores

Luego de haberse creado la base de datos el botón de “Conectar a base de datos” se desactivará y activará los demás botones como lo muestra en la figura 28.



Figura 28. Activación de los demás botones de la conexión a la base de datos
Fuente: Autores

Mostrando un mensaje satisfactorio con el texto que se muestra en la figura 29.



*Figura 29. Mensaje de creación de la base de datos
Fuente: Autores*

Las líneas de instrucciones para activar los demás botones del cuadro de interfaz y de la desactivación del botón conectar a la base datos, junto con el mensaje de creación de la base de datos se muestran en Anexo 1.

2.8.2.3 Controlador del sistema PHP en Android Studio

2.8.2.3.1 Código de programación para la conexión a la base de datos en Brakets

Para la creación y conexión entre el proyecto y la base de datos, el lenguaje de programación Breakets permite mediante unas líneas de programación el nexco con XAMPP necesariamente para el funcionamiento de la aplicación Android, las instrucciones de la programación se detallan en Anexo 1.

Cuando el archivo de conexión y creación ala bases de datos no haya tenido ningún error se le guarda como archivos. PHP para luego ser llamados mediante la programación en Android Studio, con esto la base de datos esta creada y anexada con la aplicación.

2.8.2.3.2 Código de programación para el registro de informacion a la base de datos.

Para ingresar datos a la base de datos por medio del dispositivo móvil es necesario unas líneas de programación que se le presenta en Anexos 1.

Este código de programación será llamado como clases por medio de Android Studio, para que cumpla la función de registros a la base de datos.

2.8.2.3.3 Código de programación para consultar información ingresada en la base de datos.

Una vez ingresado una cantidad de información en la base de datos, se puede consultar información ya ingresadas en la base de datos, así mismo por la Id de identificación, para ello enseguida le presenta las instrucciones de programación.

Como los anteriores archivos se debe guardar como una librería o clases, para luego ser llamado por medio de Android Studio.

2.8.2.4 Programación y compilación de la aplicación en Android Studio

Para la creación de la aplicación Android del Sistema de telecomunicaciones para ayuda a personas con discapacidad, necesita de otro programa antes mencionado llamado “Brakets”, este programa le permite además de la conexión a la base de datos le permite la consulta y la eliminación de un dato registrado a la base de datos.

2.8.2.4.1 Programación para recuperación de datos en Brakets.

Para poder recuperar información del a base de datos es necesaria una pequeña programación que permitió la comunicación entre el usuario y la base de datos, toda la programación que permite establecer la conexión con la base de datos y la forma con la que se registra los datos necesarios para poder obtener la ubicación de cada persona que desee utilizar el Sistema de Telecomunicaciones para Ayuda a personas con discapacidades se muestra en Anexo 1.

Es importante saber que este proyecto consta de varias programaciones las cuales permite varias consultas en este caso es obtener la dirección IP ya registrada, todas las líneas de programación necesarias se encuentran en el Anexo 1.

2.8.2.4.2 Registros en base de datos programado en Brakets.

Esta aplicación tiene dos registros más, aparte del registro mostrado en el Anexo 1 el cual fue mencionado en el punto anterior, a continuación se puede consultar todas las líneas de la programación para registrar el tipo de discapacidad en el Anexo 1.

2.8.2.4.3 Eliminar la dirección IP del sistema de base de datos.

Una vez que el dispositivo de la persona que fue ayudada la aplicación tiene la opción de eliminar la dirección IP para que su registro sea eliminado del sistema de base de datos y no pueda pedir ayuda en tiempo futuro, la programación se encuentra en Anexo 1.

2.8.2.4.4 Extracción de la dirección IP en Android Studio.

Para poder obtener la dirección ip que posee el dispositivo del usuario que va a utilizar el Sistema de Telecomunicaciones para Ayuda a Personas con Discapacidad es necesario hacer un bucle dentro de un while, el cual permite ir obteniendo cada carácter que registra la dirección ip y de esta forma almacenar dentro de un string para poder utilizarlo a lo posterior para dar los rangos que posee cada router y así determinar la ubicación de la persona que desee solicitar la ayuda. Para realizar este proceso se puede consultar en el Anexo 1 donde se encuentra la programación necesaria para obtener la dirección ip del dispositivo móvil.

2.8.2.4.5 Ayuda y registro en el sistema de base de datos.

En el momento de que una persona necesite ayuda la aplicación extrae la dirección IP del dispositivo y lo guarda en el sistema de base de datos, que luego será extraído para comparar y determinar qué persona puede acudir a la solicitud de ayuda, toda la programación se encuentra en el Anexos 1.

2.8.2.4.6 Declaración de variables y la programación general.

Es necesario tener un encabezado y declaración de variables pues para esto se muestra una pequeña parte de toda la programación, toda la programación se encuentra en Anexo 1.

2.8.2.4.7 Interfaz de la aplicación móvil Android.

Cara principal de la aplicación Android del sistema de telecomunicaciones para ayuda a personas con discapacidad, como se muestra en la figura 30.



Figura 30. Cara principal de la aplicación STAPD
Fuente: Autores

Cara del botón ayuda de la aplicación Android del sistema de telecomunicaciones para ayuda a personas con discapacidad, como se muestra en la figura 31.



Figura 31. Cara de ayuda de la aplicación STAPD
Fuente: Autores

Cara del mensaje de ayuda de la aplicación Android del sistema de telecomunicaciones para ayuda a personas con discapacidad, como se muestra en la figura 32.



Figura 32. Cara de mensaje de la aplicación STAPD
Fuente: Autores

Pantalla donde se muestra el mensaje de ayuda de la aplicación Android del sistema de telecomunicaciones para ayuda a personas con discapacidad, como se muestra en la figura 33.

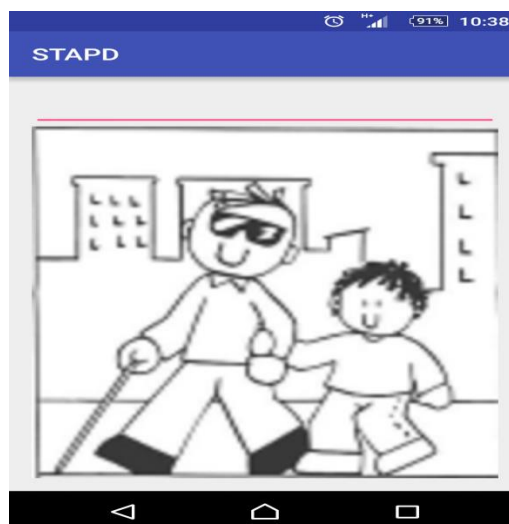


Figura 33. Cara donde se muestra el mensaje de la aplicación STAPD
Fuente: Autores

2.8.2.4.8 Diagrama de flujo de la aplicación STAPD.

El Sistema de Telecomunicaciones para Ayuda a Personas con Discapacidad se muestra a continuación en la figura 34.

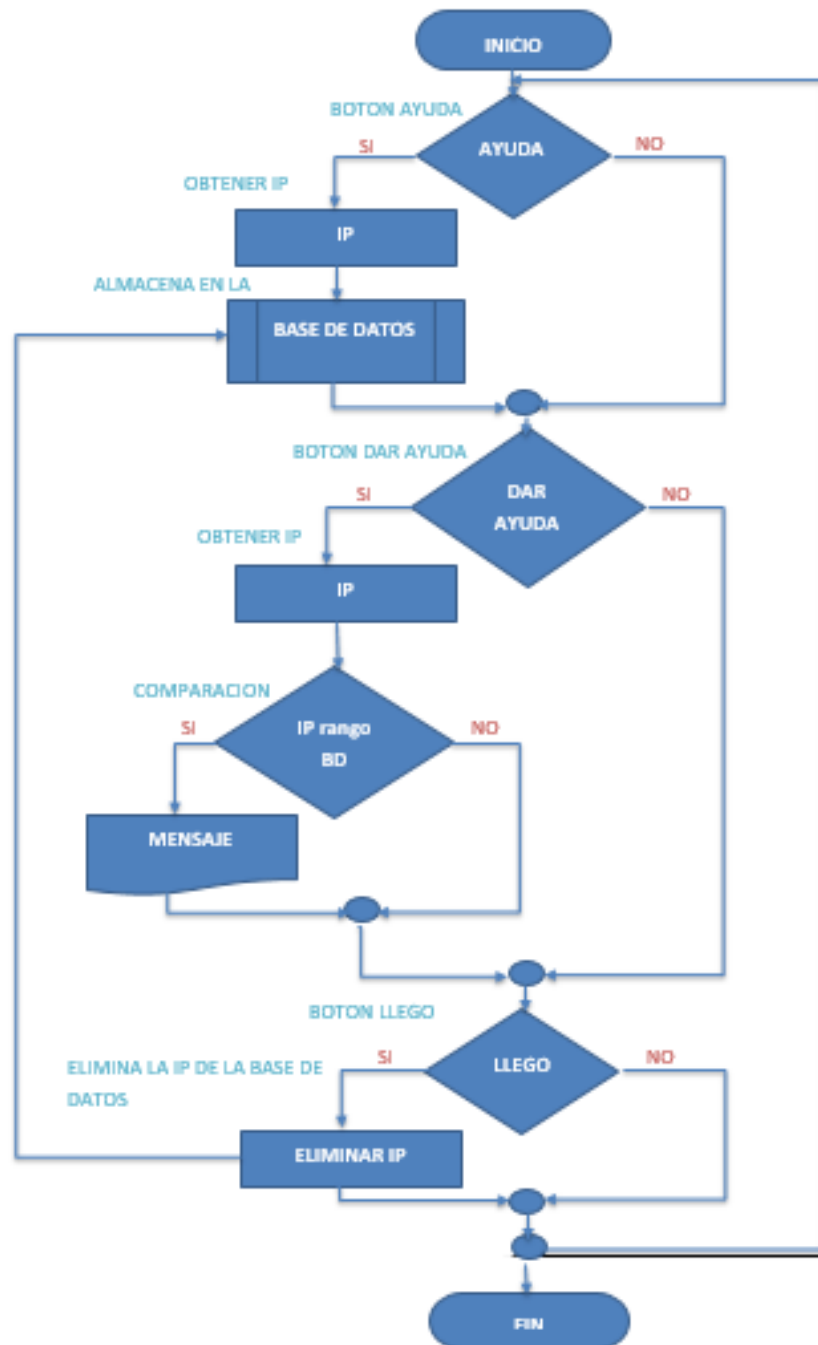


Figura 34. Diagrama de flujo de sistema STAPD
Fuente: Autores

2.8.3 Red Inalámbrica STAPD

2.8.3.1 Diseño de la Red Inalámbrica STAPD

En nuestro proyecto se ha visto la necesidad de crear una red independiente de direcciones estáticas, puesto que los puntos de acceso (AP) ya instalados en la Facultad de Ingeniería de la Universidad Nacional de Chimborazo en el edificio bloque A, dan direcciones dinámicas el cual no facilita la obtención de la dirección IP invariables en el punto de acceso (AP) donde se conecte nuestro dispositivo necesario para que nuestro proyecto funcione.

En el edificio de Ingeniería bloque A, se ha realizado una sectorización de tres diferentes puntos para cada piso el cual permita saber la ubicación de la persona que necesita ayudar o a su vez la persona que necesita ayuda, los diferentes puntos sectorizados tienen la nominación de Ala A, Centro, Ala B para cada piso que conforma el edificio.

Para el diseño de la nueva red inalámbrica STAPD, se ha instalado nueve puntos de acceso en lugares estratégicos junto a diferentes puntos de alimentación para cada router inalámbrico utilizado para el proyecto, utilizando los materiales previos mencionados en el documento.

2.8.3.2 Configuración de los Routers con sus direcciones IP

Para poder configurar los rangos de direcciones IP, el cual se desee cambiar dependiendo en la circunstancia en las que se esté utilizando el proyecto es necesario la IP de acceso al router junto a la contraseña que se muestra en la Tabla 3.

Tabla 3. IPs de acceso a los routers junto con las contraseñas

ROUTERS	IPS DE ACCESO AL ROUTER	CONTRASEÑAS
Router 1	192.168.1.10	Usuario: admin Contraseña: admin
Router 2	192.168.1.20	
Router 3	192.168.1.30	
Router 4	192.168.1.40	
Router 5	192.168.1.50	
Router 6	192.168.1.60	
Router 7	192.168.1.70	
Router 8	192.168.1.80	
Router 9	192.168.1.90	

Fuente: Autores

Configuración de cada router junto a sus direcciones IPs estáticas y contraseñas de acceso, en este caso se configuro el router 1 presentada en la ilustración.

- ✓ Con la IP de acceso de cada router en este caso del Router 1 es 192.168.1.10 se ingresó a la configuración del router, como se muestra en la figura 35.



Figura 35. Configuración del router

Fuente: Autores

- ✓ A continuación, con el ingreso del rango de las IPs dependiendo del router. Como se muestra en la figura 36.



Figura 36. Ingreso del rango de las IPs en el router
Fuente: Autores

- ✓ A continuación, se ingresó a la configuración inalámbrica para configurar algunos parámetros Como se muestra en la figura 37.

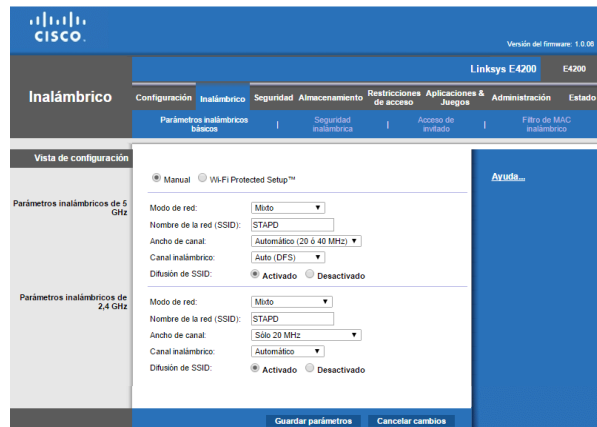


Figura 37. Configuración del nombre de la Red inalámbrica.
Fuente: Autores

- ✓ Y por último se asignó la seguridad para cada router en este caso la contraseña va a ser Tesis*2o16. La misma contraseña para todos los routers instalados en el proyecto, Como se muestra en la figura 38.

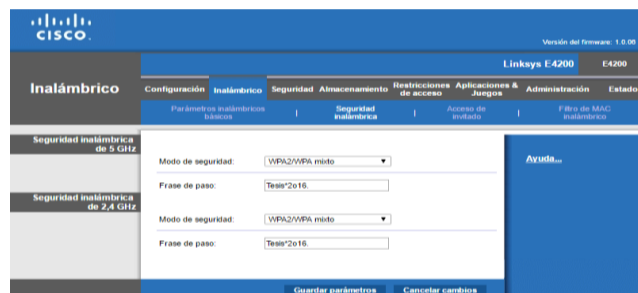


Figura 38. Asignación de la contraseña de acceso a la red
Fuente: Autores

2.8.3.3 Ubicación en los puertos de los Routers inalámbricos.

El edificio de la Facultad de Ingeniería consta de una organización de red para los diferentes puntos de acceso a la Red, optimizado por múltiples Racks ubicados estratégicamente en diferentes lugares de la edificación para su mejor distribución y fácil acceso a los switches.

Dando como definición una Rack, es un soporte metálico con el objetivo de alojar diferentes equipos electrónicos como puede ser informáticos y de comunicación, sus medidas estándar son normalizadas para que pueda ser ocupada equipos de diferentes marcas.

Observando los puertos libres de cada switch ubicado en diferentes racks, ha dado la oportunidad de ocupar esos dichos puertos para la ubicación de nuestros routers inalámbricos conjunto con la ubicación de cada punto de acceso.

Para poder localizar cada uno de los puertos que se ha conectado se utilizó el equipo testador para redes LAN, el cual permite localizar cada cable dependiendo a que cable se conectó la otra parte del testador que emite una señal para cada cable y así poder determinar si el cable se encuentra en buen estado y está llegando al otro extremo donde se conectara al punto del rack en donde se configura las respectivas vlans que permitirá el acceso a la red creada STAPD.

Según la necesidad de los puntos de acceso de la red inalámbrica STAPD se ubicó en diferentes racks, dependiendo el puerto que se va a utilizar para de esta forma conocer la vlan que se debe configurar, así se podrá tener una mejor distribución de esta forma se muestra en la Tabla 4.

Tabla 4. Identificación de Routers con los racks en la UNACH .

RACKS	ROUTERS
BACKBONE	Router 1, Router 4
LABORATORIO DE COMPUTO IV	Router 2
LABORATORIO DE CONTROL AUTOMÁTICO	Router 3
ADMINISTRATIVO	Router 5 Router 8 Router 9
CONSEJO ADMINISTRATIVO	Router 6
LABORATORIO DE CENTRO DE IDIOMAS	Router 7

Fuente: Autores

2.8.3.4 Topología de la red inalámbrica STAPD

Por la ubicación de los racks con los routers se adoptó una topología de forma de árbol el cual es necesaria para la conexión entre los diferentes pisos, reconociendo una estructura de topología de tipo árbol tanto físico como lógico.

En la siguiente figura se muestra la topología de acuerdo a los racks instaladas en la facultad de Ingeniera de la Universidad Nacional de Chimborazo, como se muestra en la figura 39.

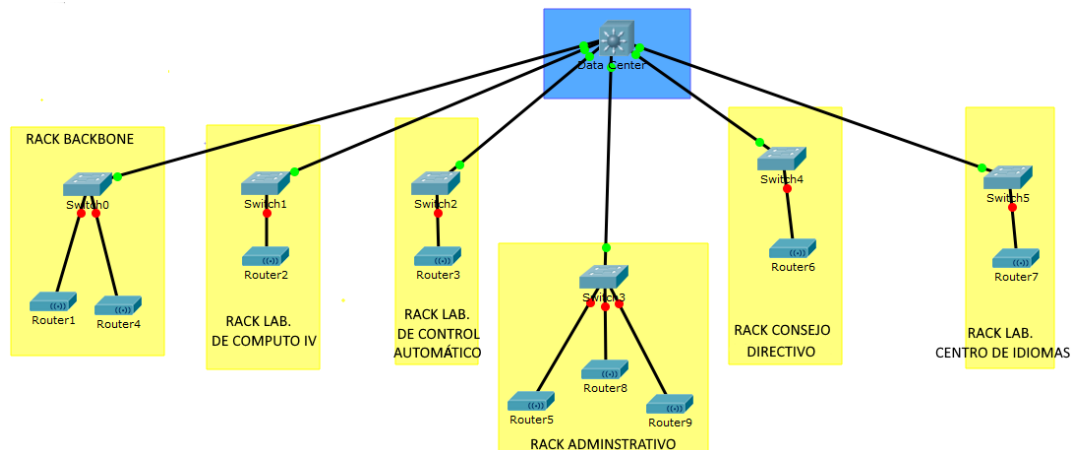


Figura 39. Topología de árbol de la red inalámbrica STAPD.
Fuente: Autores.

2.8.3.5 Ubicación de los puntos de acceso de la red STAPD.

Para una mejor sectorización de nuestro proyecto, se ha instalado tres routers para cada piso, siendo tres pisos en la facultad de ingeniería de la Universidad Nacional de Chimborazo sale un total de nueve routers configurados en diferentes rangos de direcciones IPs, con esto se identificó según la dirección IP en que piso y en qué sector de piso se encuentra la persona quien pida ayuda o pueda ayudar.

Para el primer piso se ha congruado las direcciones IPs en diferentes rangos como lo muestra la siguiente Tabla 5.

PRIMER PISO

Tabla 5. Identificación de los routers en el primer piso.

	Inicio IP	Hasta IP	Ubicación
Router 7	192.168.1.71	192.168.1.79	Ala A
Router 8	192.168.1.81	192.168.1.89	Centro
Router 9	192.168.1.91	192.168.1.99	Ala B

Fuente: Autores

Como se puede ver en la figura 40 se muestra la distribución de los routers en el primer piso.

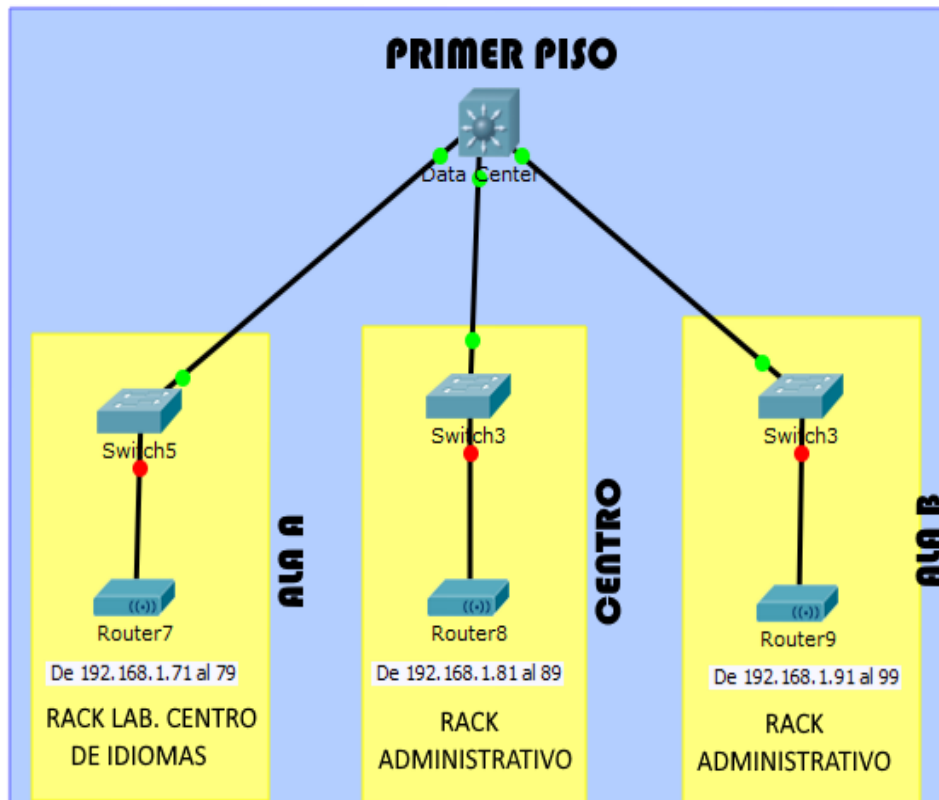


Figura 40. Topología de árbol del primer piso junto al rango de direcciones IP.
Fuente: Autores

Para el segundo piso se ha configurado las direcciones IPs en diferentes rangos como lo muestra la siguiente tabla 6:

SEGUNDO PISO

Tabla 6. Identificación de los routers en el segundo piso.

	Inicio IP	Hasta IP	Ubicación
Router 4	192.168.1.41	192.168.1.49	Ala A
Router 5	192.168.1.51	192.168.1.59	Centro
Router 6	192.168.1.61	192.168.1.69	Ala B

Fuente: Autores

Como se puede ver en la figura 41 se muestra la distribución de los routers en el primer piso.

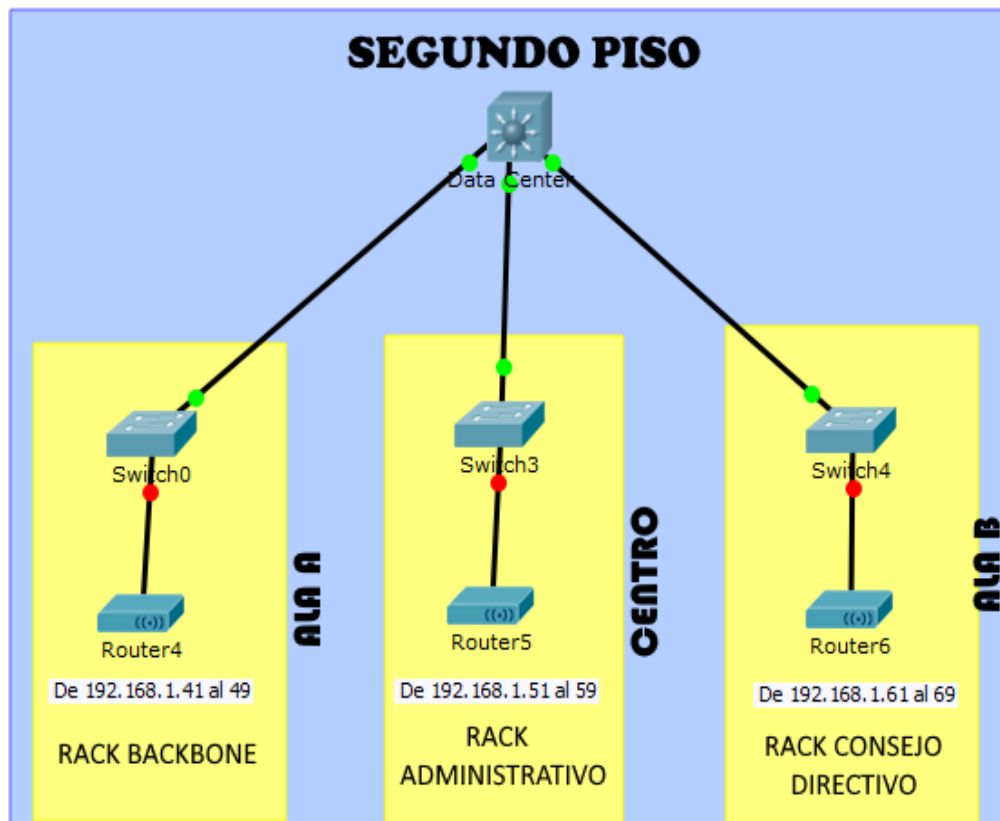


Figura 41. Topología árbol del segundo piso junto al rango de direcciones IP.
Fuente: Autores

Para el segundo piso se ha configurado las direcciones IPs en diferentes rangos como lo muestra la siguiente Tabla 7.

TERCER PISO

Tabla 7. Identificación de los routers en el tercer piso.

	Inicio IP	Hasta IP	Ubicación
Router 1	192.168.1.11	192.168.1.19	Ala A
Router 2	192.168.1.21	192.168.1.29	Centro
Router 3	192.168.1.31	192.168.1.39	Ala B

Fuente: Autores

Como se puede ver en la figura 42 se muestra la distribución de los routers en el primer piso.

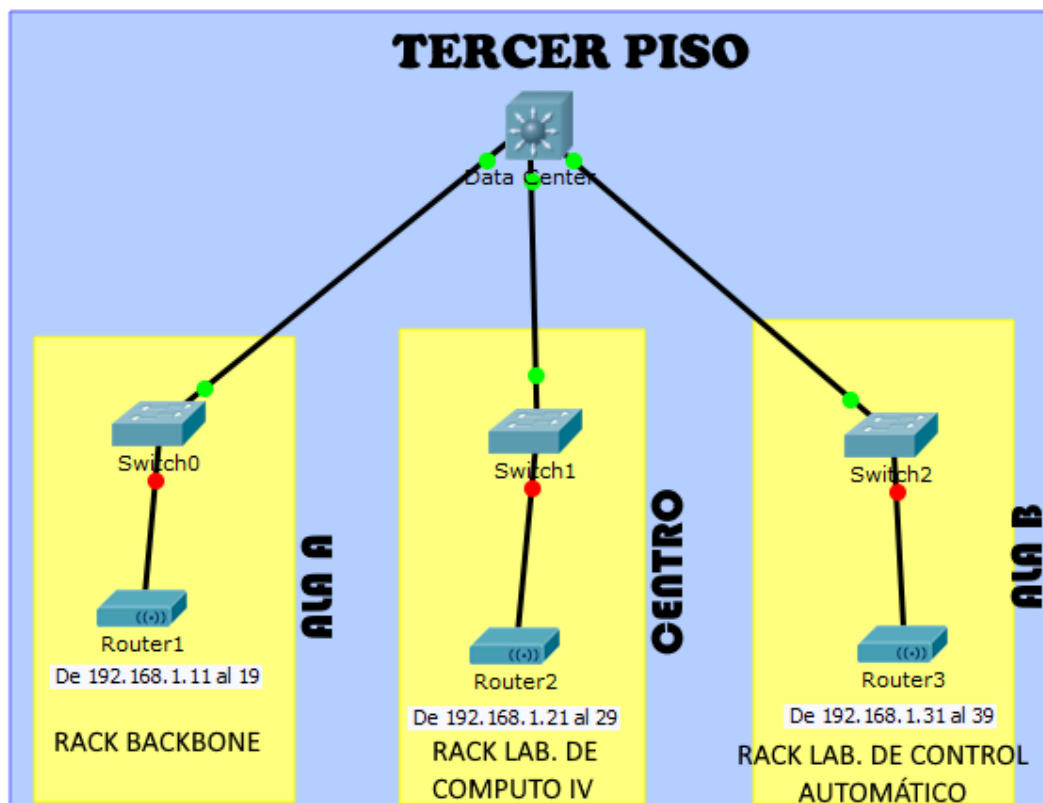


Figura 42. Topología árbol del tercer piso junto al rango de direcciones IP.
Fuente: Autores

2.8.4 Ubicación de los routers de la Red Inalámbrica STAPD

2.8.4.1 Ubicación de los routers en el edificio de la facultad Ingeniería

Con la ayuda de los planos arquitectónicos de la facultad de ingeniería facilitados por la Universidad Nacional de Chimbilaco, se indicó la ubicación exacta de los puntos de acceso (AP).

Para la ubicación de cada uno de los aps se estudió de una manera estratégica para que la radiación de cada uno de los routers emane la señal en las áreas estratégicas y se pueda tomar como base para poder obtener la dirección ip de cada uno de los aps y así determinar el área en el que se encuentra cada persona conectada.

A continuación se presenta el mapa donde se puede observar los puntos estratégicos en los cuales se procedió a ubicar cada uno de los aps en los diferentes pisos del edificio principal de la Facultad de Ingeniería de la Universidad Nacional de Chimborazo.

En el primer piso se ubican tres puntos de acceso en el área del Ala A, Centro y en el Ala B, como se muestra en la figura 43.

PLANO ARQUITECTÓNICO PRIMER PISO

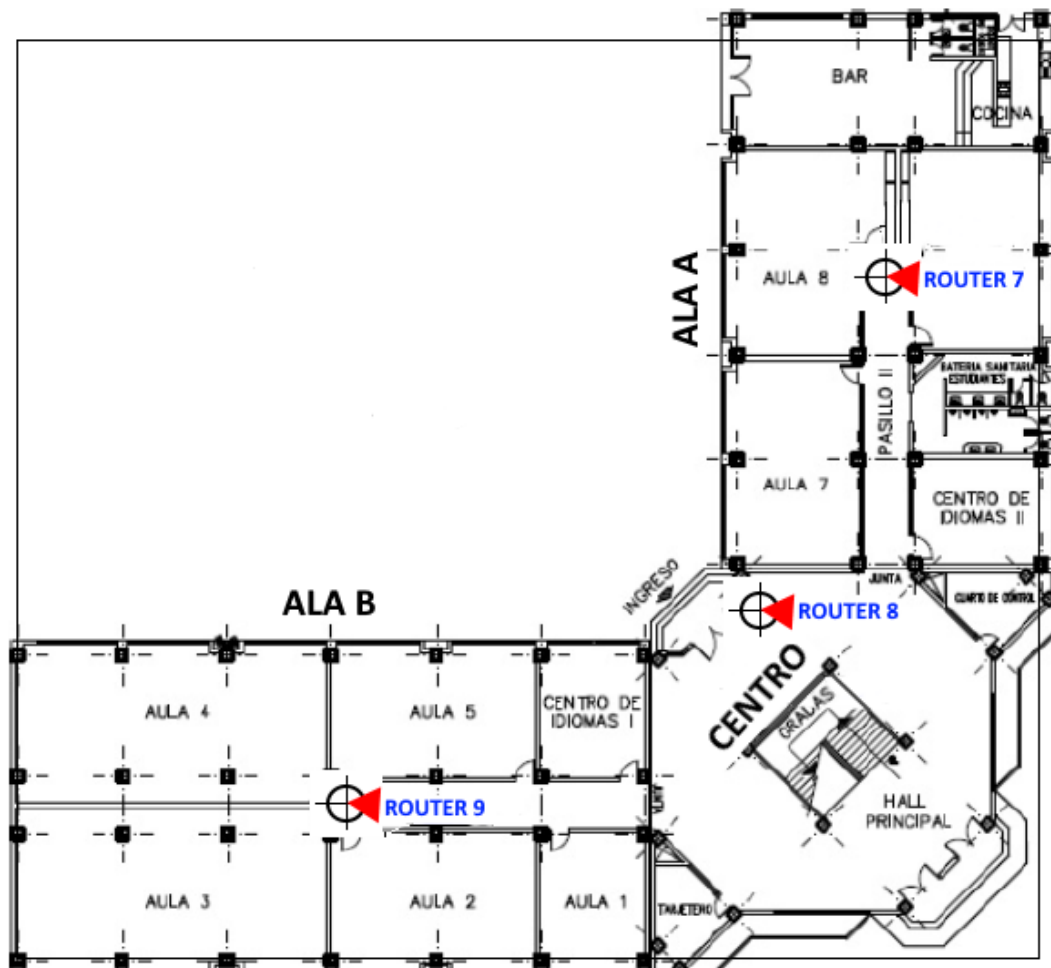


Figura 43. Ubicación de los routers en el plano arquitectónico del primero piso
Fuente: Plano arquitectónico facilitado por la Universidad Nacional de Chimborazo

En el segundo piso se ubican tres puntos de acceso en el área del Ala A, Centro y en el Ala B, como se muestra en la figura 44.

PLANO ARQUITECTÓNICO SEGUNDO PISO

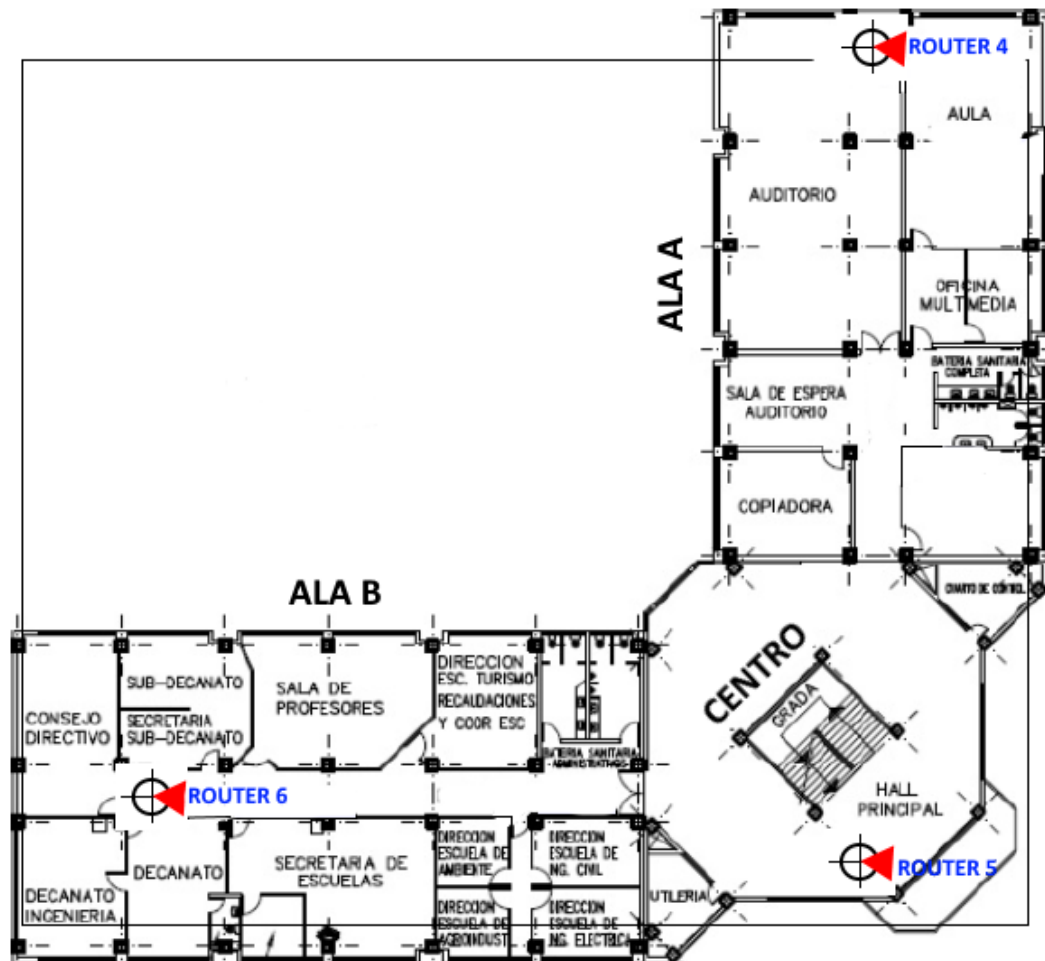


Figura 44. Ubicación de los routers en el plano arquitectónico del segundo piso
Fuente: Plano arquitectónico facilitado por la Universidad Nacional de Chimborazo

En el segundo piso se tomaron las medidas pertinentes para la ubicación del AP que se localiza en el ala A, debido a que el lugar donde se decidió instalar estaba copado por dos routers que son de la facultad, es por esta razón que se decidió instalar el AP dentro del auditorio de la facultad de ingeniería para que el router trabaje de la mejor forma y diera las direcciones IPs adecuadas para conocer la posición de cada persona y poder enviar la solicitud a la base de datos y que pueda responder de mejor forma.

En el tercer piso se ubican tres puntos de acceso en el área del Ala A, Centro y en el Ala B, como se muestra en la figura 45.

PLANO ARQUITECTÓNICO TERCER PISO

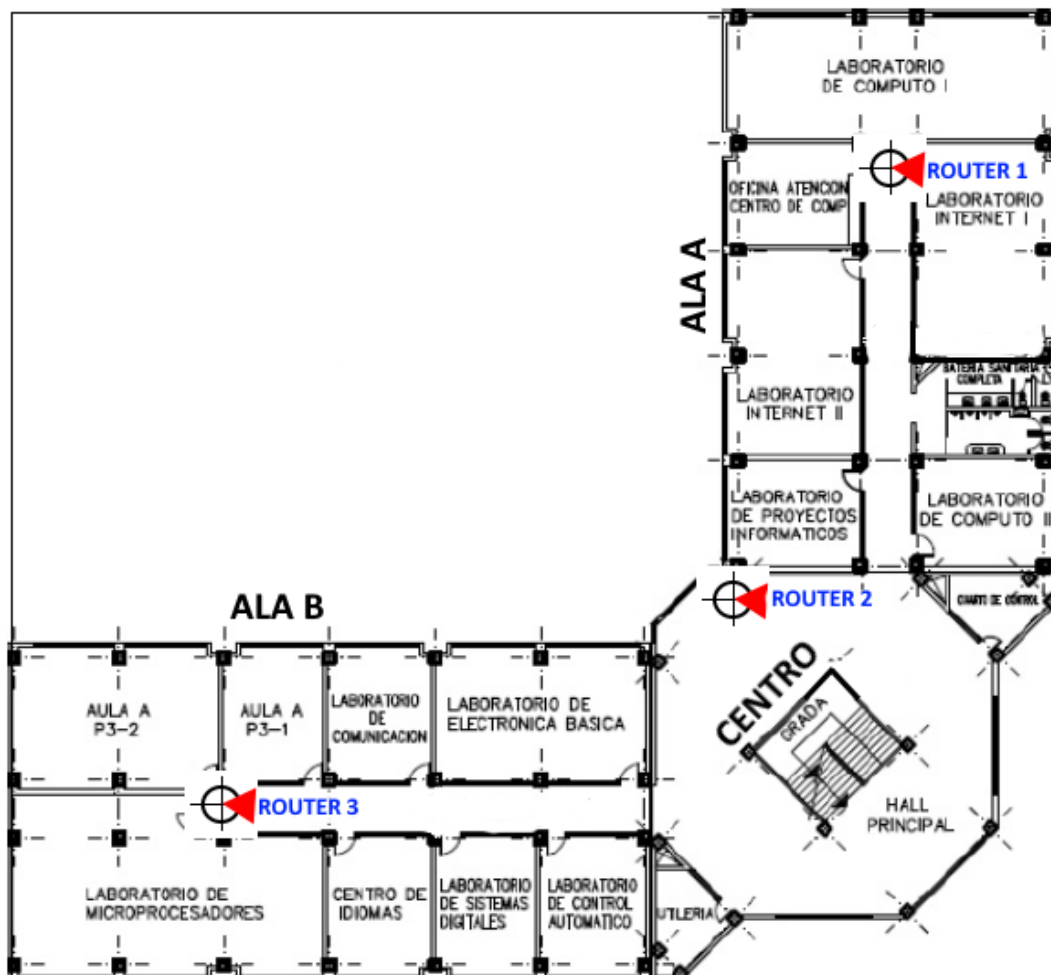


Figura 45. Ubicación de los routers en el plano arquitectónico del tercer piso.
Fuente: Plano arquitectónico facilitado por la Universidad Nacional de Chimborazo.

2.8.5 Configuración para los Routers de la Red Inalámbrica STAPD

El proveedor de la red inalámbrica es la Universidad Nacional de Chimborazo, ya que tiene una red inalámbrica instalada en la facultad de Ingeniería y aprovechando de los switches ya instalados en diferentes racks ubicadas en la facultad, le facilita al sistema STAPD la conexión por medio de puertos libres.

De acuerdo con eso, la configuración de los routers para que tengan conexión a internet se asignado una Vlan para cada puerto, la configuración de un puerto con la Vlan 252 habilitada por CTE (Centro de Tecnología Estudiantil de la Universidad Nacional de Chimborazo).

Para la configuración de los switch es necesario tener instalada PuTTY el cual permite ingresar en modo privilegiado a la interfaz de configuración, para este switch se asignado la Vlan 252 para los puertos 9, 11, 13, la configuración se detallará en Anexo 2.

2.8.6 Pruebas de interactividad de la aplicación Android (STAPD)

Las pruebas de interactividad realizadas con la aplicación STAPD y el sistema de Base de Datos, se basa en determinar el tiempo de respuesta del servidor con el usuario. Como se muestra en las siguientes Tablas 8,9.

Tabla 8. Tiempos de la conexión a la base de datos

BASE DE DATOS	INICIO DE CONEXIÓN (Segundo)	CONEXIÓN (Segundo)
1	0.1	0,781
2	0.2	0,654
3	0.1	0,869
4	0.1	0,771
5	0.2	0,973
6	0.2	0,788
7	0.2	0,754
8	0.2	0,874
9	0.2	0,965
10	0.2	0,785
11	0.1	0,829
12	0.1	0,929
13	0.0	0,831
14	0.2	0,732
15	0.1	0,933
16	0.2	0,834
17	0.1	0,736
18	0.2	0,937

19	0.0	0,838
20	0.1	0,739
21	0.2	0,740
22	0.2	0,842
23	0.1	0,743
24	0.2	0,744
25	0.1	0,945
26	0.1	0,947
27	0.1	0,748
28	0.2	0,849
29	0.1	0,750
30	0.1	0,951
31	0.2	0,953
32	0.2	0,854
33	0.2	0,855
34	0.0	0,756
35	0.0	0,934
36	0.2	0,823
37	0.1	0,812
38	0.2	0,865
39	0.1	0,778
40	0.1	0,933

Fuente: Autores

Tabla 9. Tiempo de repuesta entre el servidor al usuario

SERVIDOR	INICIO DE RESPUESTA (Segundo)	RESPUESTA (Segundo)
1	0,2	0,745
2	0,1	0,906
3	0,3	0,767
4	0,1	0,864
5	0,2	0,827
6	0,0	0,897
7	0,2	0,753
8	0,1	0,916
9	0,2	0,879
10	0,1	0,742
11	0,2	0,905
12	0,1	0,968
13	0,1	0,731

14	0,2	0,994
15	0,2	0,757
16	0,0	0,892
17	0,0	0,783
18	0,0	0,946
19	0,1	0,809
20	0,1	0,772
21	0,2	0,835
22	0,1	0,898
23	0,2	0,961
24	0,2	0,724
25	0,2	0,787
26	0,1	0,985
27	0,2	0,813
28	0,2	0,976
29	0,1	0,939
30	0,2	0,802
31	0,1	0,765
32	0,1	0,728
33	0,2	0,991
34	0,2	0,954
35	0,1	0,817
36	0,0	0,734
37	0,0	0,876
38	0,0	0,834
39	0,0	0,987
40	0,0	0,745

Fuente: Autores

2.8.7 Comprobación de hipótesis

Para la comprobar la hipótesis se utiliza el método estadístico CHI-CUADRADO, que accede a dos grados de posibilidad, alternativa la que se quiere comprobar y nula (rechaza la hipótesis alternativa).

2.8.8 Planteamiento de la hipótesis estadística

Hipótesis nula (Ho): La aplicación móvil de Android (STAPD) no permitirá la comunicación y ubicación de las personas necesitadas de ayuda con las personas

que desean ayudar, dentro del edificio principal de la facultad de ingeniería de la Universidad Nacional de Chimborazo.

Hipótesis alternativa (HI): La aplicación móvil de Android (STAPD) permitirá la comunicación y ubicación de las personas necesitadas de ayuda con las personas que desean ayudar, dentro del edificio principal de la facultad de ingeniería de la Universidad Nacional de Chimborazo.

2.8.9 Establecimiento del nivel de significancia

Las pruebas se realizaron con un 95% de confiabilidad, es decir, se trabajó con un nivel de significancia de $\alpha=0.05$.

2.8.10 Determinación del valor estadístico de prueba

Si el valor de CHI-CUADRADO es menor o igual que el CHI-CUADRADO crítico entonces se acepta la hipótesis nula, caso contrario se la rechaza.

$$X^2 \leq \text{Valor Crítico}$$

Para aceptar o rechazar esta hipótesis se tomaron en cuenta 2 escenarios, un escenario A, medición de tiempos de la conexión a la Base de Datos y un escenario B, medición de tiempos de respuesta del Servidor al Usuario

En la Tabla 7 se muestran los valores obtenidos en cada uno de los escenarios en los cuales se realizaron las pruebas, a continuación se muestra en la Tabla 10, donde están propuestos los dos escenarios.

Tabla 10. Valores Obtenidos

	Conexión(s)	Inicio (s)	Total
Escenario A	0.781	0.1375	0.9185
Escenario B	0.735	0.1243	0.8593
Total	1.516	0.2618	1.7778

Fuente: Autores

Para obtener las frecuencias esperadas se multiplica el total de cada columna, por el total de cada fila, y se divide entre el total de cada fila y columna, puede

apreciarse en la Tabla 8 y la Tabla 9 representa valores críticos definidos por este método, como se muestra en la Tabla 11 y 12.

Tabla 11. Valores de frecuencias esperadas

	Conexión(s)	Inicio (s)	Proporción de Muestra
Escenario A	0.7832	0.1352	45.92x10E-2
Escenario B	0.7327	0.1265	42.96x10E-2

Fuente: Autores

Tabla 12. Valores Críticos Método Chi-Cuadrado

		PROBABILIDAD			
		0,995	0,990	0,975	0,950
GRADOS DE LIBERTAD	1	0,000	0,000	0,001	0,004
	2	0,010	0,020	0,051	0,103
	3	0,720	0,115	0,216	0,352
	4	0,207	0,297	0,484	0,711
	5	0,412	0,554	0,831	1,145
	6	0,676	0,872	1,237	1,635
	7	0,989	1,239	1,690	2,167
	8	1,344	1,646	2,180	2,733
	9	1,735	2,088	2,700	3,325
	10	2,156	2,558	3,247	3,940
	11	2,603	3,053	3,816	4,575
	12	3,074	3,571	4,404	5,226

Fuente: Autores

Los rangos de probabilidad obtenidos de acuerdo al número de grados de libertad que se obtuvo detallan de una forma eficiente que cada conexión medida se demora entre 0,01 la más veloz y la más demorada tarda entre 3,07 segundos lo cual permite establecer un número adecuado para poder obtener de forma matemática y se obtiene todos los valores requeridos para sacar el nivel de satisfacción de nuestro proyecto.

La tabla 10 muestra los valores obtenidos para nuestra comprobación de hipótesis a través del método CHI-CUADRADO, la tabla 13 muestra los valores para realizar el cálculo necesario.

Tabla 13. Resultados del método estadístico del CHI-CUADRADO

# de filas	2
# de columnas	2
$X^2=$	0.093x10E-2
Grados de Libertad	1
Nivel de significación	0,05
Probabilidad	0,95
VALOR CRITICO	0.004

Fuente: Autores

En la tabla 13 se muestra la prueba Chi-Cuadrado, requiere la comparación del Valor Crítico (X^2_{tabla}) con $X^2(X^2_{prueba})$.

De acuerdo con el resultado se obtiene que X^2 es mayor que el valor crítico lo cual lleva a rechazar la hipótesis nula y aceptar la hipótesis alternativa, es decir:

“HI: La aplicación móvil de Android (STAPD) permitirá la comunicación y ubicación de las personas necesitadas de ayuda con las personas que desean ayudar, dentro del edificio principal de la facultad de ingeniería de la Universidad Nacional de Chimborazo.”

CAPITULO III

2. Resultados

El diseño e implementación de un sistema de telecomunicación para los dispositivos electrónicos de personas con discapacidades de la Universidad Nacional de Chimborazo, permitió ayudar en la orientación, ubicación e interacción entre los usuarios que solicitan ayuda, en nuestro caso son las personas con discapacidades que visitan el edificio principal de la facultad de ingeniería, los cuales mediante la aplicación creada envían una petición a la base de datos que desean movilizarse y al otro lado el sistema verifica que alguien está dispuesto a ayudar por lo cual emite una alerta indicando la ubicación, el piso y el lugar exacto dentro del edificio de la facultad de ingeniería.

La confiabilidad del sistema es de un 90% de estabilidad al momento de conectarse con el Sistema de Telecomunicaciones para Ayuda a Personas Discapacitadas ya que el servidor creado siempre está activo y posee una respuesta casi inmediata al momento de interactuar entre usuarios y base de datos.

Los resultados de velocidad obtenidos por medio de las pruebas adecuadas efectuadas en dos ambientes diferentes, el primer método a experimentar fue la conexión entre el dispositivo del usuario y la base de datos, el segundo método fue entre el servidor y la aplicación en el usuario receptor, lo que permite determinar la diferencia de tiempo de conexión del dispositivo con el sistema de base de datos, en el sentido de cargar y descarga de información adecuada, esto ayuda a obtener información del sistema de base de datos necesaria para determinar de forma eficiente y directa si nuestra aplicación es viable para el edificio principal de la Facultad de Ingeniería de la Universidad Nacional de Chimborazo.

A continuación, se detalla el resultado obtenido utilizando las diferentes mediciones que se tomó al momento de realizar las pruebas de carga y descarga de información de la base de datos y del servidor, para poder analizar de una mejor manera y matemáticamente explicar la eficacia de nuestro proyecto se tomó en consideración el método CHI-CUADRADO lo cual permite determinar la viabilidad del proyecto.

- ❖ La conexión que tiene el dispositivo con el sistema de base de datos es de 0.781 segundos, es decir que es menor a 1 segundo.

- ❖ De la misma forma el tiempo de respuesta que existe entre el servidor y el usuario del sistema (STAPD) es de 0.735 segundos, es decir que es menor a 1 segundo

CAPITULO IV

3. Discusión

El proyecto del “Sistema de Telecomunicaciones para Ayuda a Personas con Discapacidad” nombrado (STAPD) por sus siglas, tiene una diferente forma de dar la ubicación de una persona por medio de su teléfono inteligente (Smart phone) que debe estar conectada a la red inalámbrica STAPD, a comparación de otros proyectos como: “Servicio de localización a través de una red UMTS” o como el proyecto de “Ubicación y localización de personas por medio de la potencia del router conectado”, cada uno de ellos tiene su diferente forma de trabajo.

En el caso del proyecto de “Servicio de localización a través de una red UTMS” su funcionamiento está basado como su tema lo dice en redes UTMS el cual se basa en frecuencias CDMA para su ubicación a través de enlaces con satélites.

En el caso del proyecto de “Ubicación y localización de personas por medio de la potencia del ruter conectado” su funcionamiento está basado en calcular la potencia en dBm, referenciando su alcance de potencia con lo sectorizado a esa potencia y tener una ubicación clara.

Como de forma contraria el proyecto STAPD da la ubicación por sectorización por medio de una red inalámbrica interna, en el sentido de que por cada router asignó un rango de dirección IP, con esto el dispositivo se conecta a la red dependiendo donde se encuentre, esta aplicación extrae la dirección IP del dispositivo conectado a la red, a su vez compara y manda la ubicación donde se encuentra el dispositivo.

Dando una comparación el proyecto STAPD tiene una red interna a los otros proyectos, mientras que el proyecto que trabaja con la potencia funciona con la red en la que se encuentre conectado, pero a su vez necesita sectorizar mientras que el proyecto STAPD esta sectorizado, ahora en el caso del proyecto que utiliza UTMS es necesario el enlazamiento del satélite

De esta forma el proyecto STAPD es eficiente para espacios equivalentemente pequeños o mejor dicho para infraestructuras ya sea de un piso o de varios pisos, a comparación de los dos proyectos mencionados que necesitan para el uno sectorizar donde va a trabajar y el otro en espacios libres con punto de vista al satélite.

CAPITULO V

4. Conclusiones y Recomendaciones

5.1 Conclusiones

- ❖ El sistema de telecomunicaciones para ayuda a personas con discapacidad STAPD permite la interacción entre personas con discapacidades que solicitan ayuda para poder movilizarse dentro del edificio principal de la facultad de ingeniería de la universidad nacional de Chimborazo y las personas naturales que deseen brindar la ayuda en un momento adecuado para poder facilitar de esta forma la orientación y la ubicación de a qué lugar se desean movilizar las personas que poseen alguna discapacidad y utilizan el sistema como una ayuda oportuna.
- ❖ La tecnología que se encuentra vigente en este momento ha hecho posible que las personas que poseen alguna capacidad especial estén muy al tanto de todos los sistemas que les pueden ayudar a mejor de una u otra forma su estilo de vida, un claro ejemplo de esto es la presencia de nuestro sistema de telecomunicaciones para ayuda a personas con discapacidades, que como proyecto de la universidad ayuda a las personas que deseen registrarse en nuestro sistema a ubicarse y movilizarse de una manera más eficiente dentro del edificio de ingeniería de la universidad nacional de Chimborazo.
- ❖ Nuestro controlador web posee una característica muy importante que permite al sistema seleccionar un dato exclusivo que el teléfono celular ingresa a la base de datos, en nuestro caso especial es la dirección IP con

la cual está trabajando nuestro dispositivo móvil es decir el celular y la eficacia que el usuario posea y que registre en la interface adecuado cuando la persona solicite ayuda.

- ❖ La conectividad existente entre el dispositivo que desee utilizar el sistema y el controlador de nuestro sistema es eficiente y se obtiene una respuesta inmediata sin necesidad de retraso ya que la red permite interactuar casi sin retrasos en las transmisiones y recepciones de datos

- ❖ El desarrollo de la aplicación móvil se la obtuvo utilizando el software Android Studio que es un entorno de desarrollo muy exclusivo para Smartphone y que permite utilizar todos los recursos que los teléfonos móviles utilizan en la actualidad, es muy importante mencionar que la aplicación posee una interface muy adecuada para las personas que van a utilizar nuestro servicio debido a que es muy básico pero esencial al momento de realizar todas las funciones que necesita en nuestro sistema.

5.2 Recomendaciones

- ❖ Algo muy importante que se debería tener en cuenta es que los conocimientos adquiridos durante toda la carrera deberían ser un poco más profundos en ciertos temas como por ejemplo lenguajes de programación como son Java, Netbeans, Android Studio, PHP, etc. Esto ayudara a que se tenga un conocimiento adecuado y así saber cómo afrontar los problemas que se generan cuando se realiza el trabajo de graduación.

- ❖ Se debería establecer un tiempo adecuado para que las personas que se encuentren en el tribunal de revisión de los proyectos de grado revisen como está avanzando dichos proyectos y no descuidar hasta el final cuando se acumula demasiado trabajo y la revisión toma más tiempo del estimado.

- ❖ Brindar una información oportuna acerca del sistema de telecomunicaciones para ayuda a personas con discapacidades para que puedan hacer uso del mismo y que de esta forma puedan movilizarse con más eficiencia dentro del edición de ingeniería de la universidad nacional de Chimborazo

- ❖ Le sistema de telecomunicaciones de ayuda a personas con discapacidad (STAPD), es un prototipo que esta implementado en el edificio principal de la facultad de Ingeniería, el cual puede ser utilizado para más desarrollo en el ámbito del sistema operativo Android y también en sistema de base de datos conjunto de servidores virtuales o físicos.

CAPITULO VI

5. PROPUESTA

6.1 Título de la propuesta

SISTEMA DE TELECOMUNICACIÓN PARA LOS DISPOSITIVOS ELECTRÓNICOS DE PERSONAS CON DISCAPACIDADES DE LA UNIVERSIDAD NACIONAL DE CHIMBORAZO, QUE PERMITA AYUDAR EN LA ORIENTACIÓN, UBICACIÓN E INTERACCIÓN ENTRE LOS USUARIOS Y LA SOCIEDAD.

6.2 Introducción

El Gobierno Nacional con la Secretaria Nacional de Planificación y Desarrollo crea en el año 2013 el Plan Nacional del Buen Vivir el cual indica en sus objetivos la igualdad, la inclusión y la equidad social y territorial en la diversidad, por lo cual la Universidad Nacional de Chimborazo ha realizado diferentes tipos de proyectos de investigación el cual ayuda a las personas con discapacidad a la inclusión social siendo como base de prueba en el edificio principal de la facultad de ingeniería, teniendo la idea de darle un plus extra a estos proyectos de investigación se tuvo la iniciativa de realizar un prototipo de un sistema de telecomunicaciones para ayuda a personas con discapacidad (STAPD), con el fin de dar la ubicación actual de alguna persona con discapacidad o sin discapacidad que tengan la necesidad de una ayuda o a su vez de ayudar, para esto le va a llegar un aviso a su teléfono celular con la ayuda de una aplicación móvil y con el sistema (STAPD).

Este proyecto de inclusión a la sociedad podría cumplir los objetivos del plan nacional del buen vivir obteniendo resultados positivos y solidificando las ideas presentados en este estudio.

- Estudio y análisis de un sistema de base de datos el cual permita el almacenamiento de registros realizados por los usuarios del sistema de telecomunicaciones para ayuda a personas discapacitadas (STAPD). Con la plataforma XAMPP y MySQL se realizó un servidor independiente, que consiste en el registro y comparación de datos registrados por medio de la aplicación móvil Android. (Xampp, s.f.)
- El desarrollador NetBeans es el software el cual se ha desarrollado la aplicación móvil con las funciones de dar la ubicación mediante a la red inalámbrica wifi, obteniendo la dirección IP (Internet Protocol), dando rangos prioritarios para la sectorización del edificio de la Facultad de Ingeniería de la Universidad Nacional de Chimborazo.
- Diseño de la interface y estética de la aplicación utilizando Android Studio, software que permitió la compilación del programa realizado en NetBeans, dando como formato APK, formato necesario para la instalación del sistema de telecomunicaciones para ayuda a personas con discapacidad (STAPD) en los dispositivos móvil con sistema operativo Android.
- Implementación de una red inalámbrica de internet exclusiva para el sistema de telecomunicaciones para ayuda a personas con discapacidad (STAPD) con subnetes de direcciones IPs estáticas ubicadas estratégicamente junto a los routers con rangos diferentes de IPs. (Dordoigne, 2014)

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el servidor

web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. Desde la versión "5.6.15", XAMPP cambió la base de datos de MySQL A MariaDB. El cual es un fork de MySQL con licencia GPL.

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos. (esepestudio, s.f.)

NETBEANS permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. un módulo es un archivo java que contiene clases de java escritas para interactuar con las apis de netbeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma netbeans pueden ser extendidas fácilmente por otros desarrolladores de software. (NetBeans, s.f.)

ANDROID STUDIO es un entorno de desarrollo integrado (IDE), basado en IntelliJ IDEA de la compañía JetBrains, que proporciona varias **mejoras con respecto al plugin ADT** (Android Developer Tools) para Eclipse. Android Studio utiliza una licencia de software libre Apache 2.0, está programado en Java y es multiplataforma. (Android, s.f.)

6.3 Objetivos

6.3.1 Objetivo General

- Diseñar e implementar un sistema de telecomunicación para los dispositivos electrónicos de personas con discapacidades de la Universidad

Nacional de Chimborazo, que permita ayudar en la orientación, ubicación e interacción entre los usuarios y la sociedad.

6.3.2 Objetivos Específicos

- Estudiar el estado del arte sobre el internet de las cosas para personas con discapacidades.
- Determinar las características y funcionalidades del controlador web del sistema de telecomunicaciones para ayuda a personas discapacitadas STAPD
- Determinar la conectividad de dispositivos de personas discapacitadas con el controlador del sistema STAPD.
- Desarrollar la aplicación móvil para el sistema STAPD.

6.4 Fundamentación Científico-Técnico

La aplicación Android STAPD” está encaminada a las personas con discapacidad que se encuentren en la Universidad Nacional de Chimborazo, mediante esta tecnología de telecomunicación contribuye al Plan Nacional del Buen Vivir con su fin de la inclusión de estas personas a la sociedad.

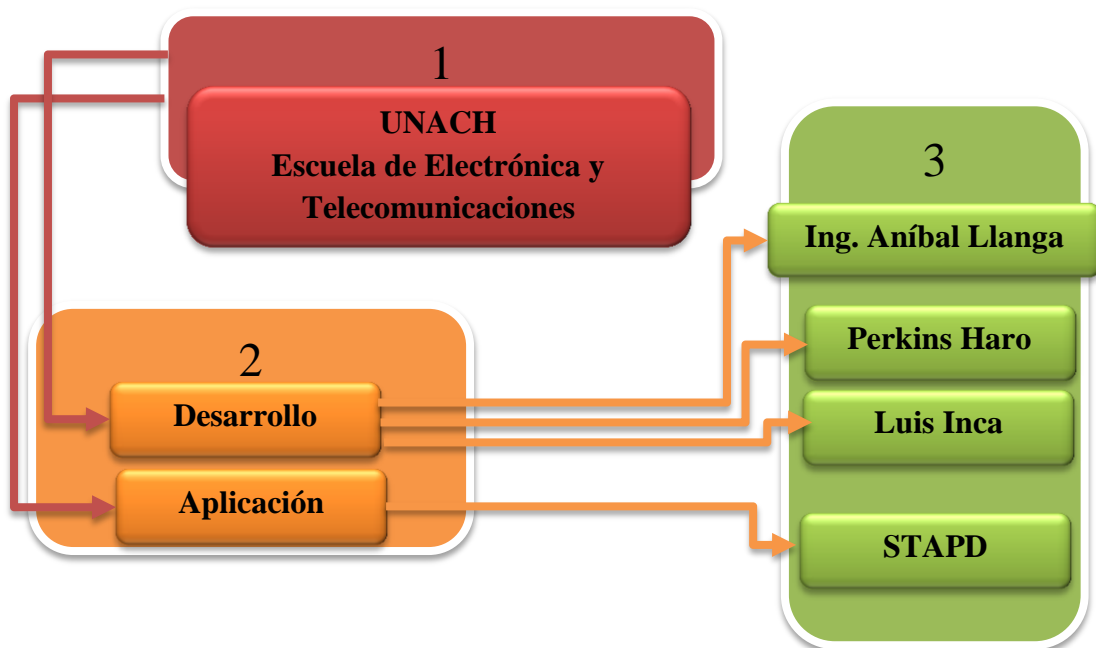
6.5 Descripción de la propuesta

El Sistema de telecomunicación de ayuda a personas con discapacidad, cuenta con una aplicación móvil gratuita compatible con el sistema operativo Android, esta aplicación llamada STAPD por las siglas del nombre del sistema, esto permite la interactividad en entre otros dispositivos que tengan la aplicación instalada y de igual forma con el servidor conectado mediante el internet, con el fin de dar la ubicación donde se encuentre la persona con discapacidad o a su vez a las personas que deseen ayudar.

Con esta aplicación se tiene la interacción entre las personas que se deseen ayudar ya sea de forma de personas sin discapacidad pueden ayudar a personas

con discapacidad o también entre personas con discapacidades diferentes que puedan complementarse, por ejemplo, una persona con discapacidad parapléjica puede ubicarle por medio del sistema STAPD y guiarle a una persona con discapacidad visual o también viceversa.

6.6 Diseño organizacional



6.7 Monitoreo y Evaluación de la propuesta

Para tener un buen monitoreo se realizará una base de pruebas, se enviará tiempos de respuesta entre usuarios y usuarios a servidor mediante la comprobación de conexión realizando ping entre ellos.

Para la evaluación de la propuesta se realizó tiempos de respuesta de conectividad con la red inalámbrica, con el servidor y entre usuarios dando así una medida de velocidad y ver su rendimiento del mensaje de ubicación si llega de una forma inmediata a los usuarios.

6. BIBLIOGRAFÍA

- Android, A. (s.f.). *Academia Android*. Obtenido de academiaandroid.com
- CCM, E. (2016). *CCM*. Obtenido de <http://es.ccm.net/contents/789-introduccion-a-wi-fi-802-11-o-wifi>
- Cisco. (2013). *Cisco Linksys E4200*.
- Copyright. (2008 - 2016). *Definición*. Obtenido de <http://definicion.de/cable-utp/>
- De:, D. (s.f.). *Definicion De:.* Obtenido de Definición de discapacidad - Qué es, Significado y Concepto <http://definicion.de/discapacidad/#ixzz4Enkosej0>
- DecTek. (2007 de Marzo de 20). *DecTek*. Obtenido de DecTek: <http://www.bjpace.com.cn/DEKTEC/english/DTA-115.pdf>
- Definiciones. (s.f.). *Definicion De:.* Obtenido de <http://definicion.de/discapacidad/#ixzz4Enkosej0>
- Dmitriev, A. (2015). *Stackoverflow*. Obtenido de <http://stackoverflow.com/questions/28957283/android-studio-create-java-project-with-no-android-dependencies>
- Dordoigne, J. (2014). Redes Inalambricas. En J. Dordoigne, *Redes Inalambricas 5ta Edicion*.
- esepestudio. (s.f.). *esepestudio*. Obtenido de esepestudio.com
- Gabriel, V. M. (2013). *Repositorio ESPE*. Obtenido de Repositorio ESPE: <http://repositorio.espe.edu.ec/bitstream/21000/6482/3/AC-ELE-ESPE-047120.pdf>
- García, R. O. (2014). *Instituto de Microelectronica Aplicada*. Obtenido de Instituto de Microelectronica Aplicada: <http://www.iuma.ulpgc.es/~nunez/clases-micros-para-com/mpc0809-trabajos/mpc0809RuymanOjedaSTBs.pdf>
- Guevara, J. M. (2011). *Fundamentos de Programación en Java*. Madrid.
- Guido Ovaco, Y. P. (2014). *Dspace Espol*. Obtenido de Dspace Espol: <https://www.dspace.espol.edu.ec/bitstream/123456789/25417/1/Resumen%20de%20tesis%20GOvaco%20y%20YPilco,%20director%20de%20tesis%20%20%20M.Sc.%20C%3%A9sar%20Yepez%20F.%2023%20dic%202013.pdf>
- Información, I. (2010). *Cbles UTP*.
- ISO/IEC 14496-3. (1 de Septiembre de 2009). *webstore*. Obtenido de https://webstore.iec.ch/preview/info_isoiec14496-3%7Bed4.0%7Den.pdf

- Martínez, C. (2013). *HTLM5facil*. Obtenido de <http://html5facil.com/tips/brackets-io-un-editor-de-texto-open-source-por-adobe/>
- MINISTERIO DE INDUSTRIA, E. Y. (3 de Abril de 2010). *Ministerio de Industrio, Energia y Turismo*. Obtenido de <http://www.televisiandigital.gob.es/tecnologias/Interactividad/Paginas/interactividad.aspx>
- MINTEL. (29 de Mayo de 2015). *MINTEL*. Obtenido de MINTEL: <http://www.telecomunicaciones.gob.ec/mintel-propondra-politica-publica-sobre-la-emision-de-alertas-de-emergencia-a-traves-de-la-senal-de-tdt/>
- Narváez, L. (2009). *La Gestión de Riesgos*. Lima: PUL CREATIVO S.R.L.
- National Geographic. (2013). *National Geographic*. Obtenido de National Geographic: <http://www.nationalgeographic.es/medio-ambiente/desastres-naturales/volcanes-definicion>
- NetBeans. (s.f.). *bligoo*. Obtenido de investigacion1javaep.bligoo.ec
- Paredes, D. (2014). *La TVdi: concepto y factor de impulso de los Sistemas de Alertas Tempranas. Un caso práctico: Proyecto Remediando*. Mérida - Venezuela.
- Parreño, J. (2014). *CREACIÓN DE NUEVOS SERVICIOS DE TELEVISIÓN DIGITAL TERRESTRE BAJO EL ESTÁNDAR ISDB-Tb PARA LA PLATAFORMA VILLAGEFLOW MEDIANTE EL ANÁLISIS DE SU ESTRUCTURA*. Sangolquí.
- Pisciotta, N. O. (2010). Sistema ISDB-Tb. *UBP SERIE MATERIALES DE INVESTIGACION*, 45.
- Recoleta, I. L. (2011). *Redes*. Obtenido de <http://ual.dyndns.org/Biblioteca/Redes/Pdf/Unidad%2006.pdf>
- Rouse, M. (2010). *Scribd*. Obtenido de www.es.scribd.com/doc/52719732/MySQL-estructura-y-forma-de-trabajar
- Tv Interactica GINGA. (s.f.). *Tv Interactica GINGA*. Obtenido de Tv Interactica GINGA: <http://www.ginga.org.br/es>
- Tv, E. (2016). *Ei Tv Entretenimiento e Interactividad para Televisión*. Obtenido de Ei Tv Entretenimiento e Interactividad para Televisión: <http://www.eitv.com.br/es/produtos/eitv-smartbox/>
- Valencia, J. (2013). *Desarrollo de Aplicaciones Interactivas para TV Digital orientadas a formar a la población en Desastres Naturales*. Quito-Ecuador.

Xampp. (s.f.). *Xampp apache + MySQL*. Obtenido de <https://www.apachefriends.org/es/index.html>

Zapata Cristian, B. (30 de Abril de 2013). *Zapata Cristian, Blog*. Obtenido de <http://mantenimientosdeunapc.blogspot.com/2011/11/que-es-xampp-y-para-que-sirve.html>

ANEXO 1

- Creación y conexión a la base de datos XAMPP

```
private void jButtonConnectActionPerformed(java.awt.event.ActionEvent
evt) {
```

```
    db.MySQLConnection("root", "", "");
    db.createDB("STAPD");
    db.createTable("Registro");
```

Activacion de desactivaci0n y activacion de los botones

```
db.MySQLConnection("root", "", "STAPD");
```

```
    jButtonConnect.setEnabled(false);
    jButtonDelete.setEnabled(true);
    jButtonGet.setEnabled(true);
    jButtonRegister.setEnabled(true);
    db.closeConnection();
```

```
}
```

```
private void jButtonRegisterActionPerformed(java.awt.event.ActionEvent evt) { void
    Window_Register wr = new Window_Register();
    wr.setVisible(true);
}
```

```
private void jButtonGetActionPerformed(java.awt.event.ActionEvent
evt) {
    db.MySQLConnection("root", "", "STAPD");
    db.getValues("Registro");
    db.closeConnection();
}
```

- Vinculación de Android Studio con la base de datos, programado en Breakets

```
<?php
header( 'Content-Type: text/html;charset=utf-8' );

function ejecutarSQLCommand($commando){

    $mysqli = new mysqli("localhost", "root", "", "STAPD");

    /* check connection */
    if ($mysqli->connect_errno) {
```

```

        printf("Connect failed: %s\n", $mysqli->connect_error);
        exit();
    }

    if ( $mysqli->multi_query($commando)) {
        if ($resultset = $mysqli->store_result()) {
            while ($row = $resultset->fetch_array(MYSQLI_BOTH)) {

            }
            $resultset->free();
        }
    }

    $mysqli->close();
}

function getSQLResultSet($commando){

    $mysqli = new mysqli("localhost", "root", "", "STAPD");
    /* check connection */
    if ($mysqli->connect_errno) {
        printf("Connect failed: %s\n", $mysqli->connect_error);
        exit();
    }

    if ( $mysqli->multi_query($commando)) {
        return $mysqli->store_result();
    }

    $mysqli->close();
}
?>

```

- Programación para registrar informacion de la base de datos, Programado en Breakets

```

<?php include ('functions.php');
$ci=$_GET['ci'];
$nombre=$_GET['nombre'];
$edad=$_GET['edad'];
$telefono=$_GET['tel'];
$ip=$_GET['ip'];

ejecutarSQLCommand("INSERT INTO `registro` (ci,nombre, edad,
telefono,ip)
VALUES (
'$ci','$nombre','$edad',
'$telefono','$ip')

ON DUPLICATE KEY UPDATE `ci`= '$ci',`nombre`= '$nombre',`edad`=
'$edad',

```

```
`telefono`='$telefono',`ip`= '$ip';");
```

```
?>
```

- Programación para consultar informacion de la base de datos, programado en Breakets

```
<?php
include('functions.php');
$ci=$_GET["ci"];

if($resultset=getSQLResultSet("SELECT * FROM `registro` WHERE
ci='$ci'")){
    while ($row = $resultset->fetch_array(MYSQLI_NUM)){
        echo json_encode($row);
    }
}
?>
```

- Programación para recuperar informacion de la base de datos, programado en Breakets

```
<?php
include('functions.php');
$ip=$_GET["ip"];

$tope = $rango + 8;
if($resultset=getSQLResultSet("SELECT discapacidad FROM
`discapitados`")){

    while ($row = $resultset->fetch_array(MYSQLI_NUM)){
        echo = json_encode($row);
    }
}
?>
```

```
<?php
include('functions.php');
$rango=$_GET["rango"];

$tope = $rango + 8;
if($resultset=getSQLResultSet("SELECT direccion, discapacidad FROM
`discapitados`")){

    while ($row = $resultset->fetch_array(MYSQLI_NUM)){
        $ip = json_encode($row);

        $ultimo = $ip[12].$ip[13];
```

```

        if($ultimo >= $rango && $ultimo <= $tope){
            echo $ip;
        }
    }
}
?>

```

- Registros de la aplicación móvil Android programado en Brackets.

```

<?php include ('functions.php');
$ci=$_GET['ci'];
$nombre=$_GET['nombre'];
$edad=$_GET['edad'];
$telefono=$_GET['tel'];
$ip=$_GET['ip'];

ejecutarSQLCommand("INSERT INTO `registro` (ci,nombre, edad,
telefono,ip)
VALUES (
'$ci','$nombre','$edad',
'$telefono','$ip')

ON DUPLICATE KEY UPDATE `ci`= '$ci`,`nombre`= '$nombre`,`edad`=
'$edad',
`telefono`='$telefono`,`ip`= '$ip';");

?>

```

- A continuación, la aplicación consta de una programación el cual registra en la base de datos la dirección IP la cual esté conectado a la red el dispositivo.

```

<?php include ('functions.php');
$ip=$_GET['ip'];
$capacidad=$_GET['capacidad'];

ejecutarSQLCommand("INSERT INTO `discapacitados` (direccion,
discapacidad)
VALUES (
'$ip','$capacidad')

ON DUPLICATE KEY UPDATE `direccion`= '$ip`,`discapacidad`=
'$capacidad';");

?>

```


- Eliminación de la dirección IP del sistema de base de datos programado en Android Studio.

```
<?php
include('functions.php');
$ip=$_GET["ip"];
$resultset=getSQLResultSet("DELETE FROM `discapacitados` WHERE direccion
= '$ip'")
?>
```

- Programación en Android Studio para la extracción de la IP del dispositivo móvil.

```
try {
    Enumeration<NetworkInterface> enumNetworkInterfaces =
NetworkInterface.getNetworkInterfaces();
    while (enumNetworkInterfaces.hasMoreElements()) {
        NetworkInterface networkInterface =
enumNetworkInterfaces.nextElement();
        Enumeration<InetAddress> enumInetAddress =
networkInterface.getInetAddresses();
        while (enumInetAddress.hasMoreElements()) {
            InetAddress inetAddress =
enumInetAddress.nextElement();
            if (inetAddress.isSiteLocalAddress()) { //Saca ip
                String ipH = inetAddress.getHostAddress();
                obtIpSub = ipH.trim();
                obtIpSub =
inetAddress.getHostAddress().substring(10);//cambiar a 10 los 2 ultimos
digitos de la IP
                inicio=Integer.parseInt(obtIpSub);
                inicio/=10;
                inicio=inicio*10+1;
                EditText txtDarAyuda = (EditText)
findViewById(R.id.txtDarAyuda);
                txtDarAyuda.setText(obtIpSub);
                new
ConsultarDatos().execute("http://192.168.1.15:8080/webService/obtenerIp.
php?rango="+inicio);
            }
        }
    }
} catch (SocketException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
    ipH += "Something Wrong! " + e.toString() + "\n";
}
```

- Ubicación en el lugar donde se encuentra a persona con la dirección IP extraída se detalla en la siguiente programación:

```

private class ConsultarDatos extends AsyncTask<String, Void, String> {
    @Override
    protected String doInBackground(String... urls) {
        // params comes from the execute() call: params[0] is the
url.
        try {
            return downloadUrl(urls[0]);
        } catch (IOException e) {
            return "Unable to retrieve web page. URL may be
invalid.";
        }
    }
    // onPostExecute displays the results of the AsyncTask.
    @Override
    protected void onPostExecute(String result) {
        String mensaje="";
        JSONArray ja = null;
        try {
            if(inicio==11)
                mensaje="Necesita ayuda en Ala A Piso 3";
            if(inicio==21)
                mensaje="Necesita ayuda en el centro Piso 3";
            if(inicio==31)
                mensaje="Necesita ayuda en Ala B Piso 3";

            if(inicio==41)
                mensaje="Necesita ayuda en Ala A Piso 2";
            if(inicio==51)
                mensaje="Necesita ayuda en el centro Piso 2";
            if(inicio==61)
                mensaje="Necesita ayuda en Ala B Piso 2";

            if(inicio==71)
                mensaje="Necesita ayuda en Ala A Piso 1";
            if(inicio==81)
                mensaje="Necesita ayuda en el centro Piso 1";
            if(inicio==91)
                mensaje="Necesita ayuda en Ala B Piso 1";

            ja = new JSONArray(result);

            AlertDialog.Builder alertDialog1 = new
AlertDialog.Builder(Listado.this);
            alertDialog1.setTitle("Mensaje de Ayuda");

            if(ja.length()>0)
                alertDialog1.setMessage("Una persona con discapacidad
"+ja.getString(1)+" "+mensaje);
            else

```

```

        alertDialog1.setMessage("Nadie necesita ayuda");
        alertDialog1.show();

    } catch (JSONException e) {
        AlertDialog.Builder alertDialog1 = new
AlertDialog.Builder(Listado.this);
        alertDialog1.setTitle("Mensaje de Ayuda");
        alertDialog1.setMessage("Nadie necesita ayuda");
        alertDialog1.show();
    }
}
}
}

```

- Programación general de la aplicación STAPD

```

package com.example.hp.stapd;

import android.app.Dialog;
import android.content.Context;
import android.content.Intent;
import android.net.Uri;
import android.net.nsd.NsdManager;
import android.net.nsd.NsdServiceInfo;
import android.os.AsyncTask;
import android.support.annotation.MainThread;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import org.json.JSONArray;
import org.json.JSONException;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.Reader;
import java.io.UnsupportedEncodingException;
import java.net.HttpURLConnection;
import java.net.InetAddress;
import java.net.NetworkInterface;
import java.net.ServerSocket;
import java.net.SocketException;
import java.net.URL;
import java.util.Enumeration;
import android.app.AlertDialog;
import com.google.android.gms.appindexing.Action;
import com.google.android.gms.appindexing.AppIndex;
import com.google.android.gms.common.api.GoogleApiClient;
public class stapd extends AppCompatActivity {
    Button btnconsultar, btnGuardar, btnAyuda;
    EditText etCi, etNombre, etEdad, etTelefono;
    String ipBD = "";

```

```

/**
 * ATTENTION: This was auto-generated to implement the App Indexing
API.
 * See https://g.co/AppIndexing/AndroidStudio for more information.
 */
private GoogleApiClient client;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_stapd);
    btnconsultar = (Button) findViewById(R.id.btnConsultar);
    btnGuardar = (Button) findViewById(R.id.btnGuardar);
    btnAyuda = (Button) findViewById(R.id.btnAyuda);
    etCi = (EditText) findViewById(R.id.etCi);
    etNombre = (EditText) findViewById(R.id.etNombre);
    etEdad = (EditText) findViewById(R.id.etEdad);
    etTelefono = (EditText) findViewById(R.id.etTelefono);

    btnconsultar.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            new
ConsultarDatos().execute("http://192.168.1.15:8080/webService/consulta.p
hp?ci=" + etCi.getText().toString());
        }
    });
    btnGuardar.setOnClickListener(new View.OnClickListener() {
        @Override

        public void onClick(View v) {
            //obtener direccion ip para almacenar en la base de
datos

            try {
                Enumeration<NetworkInterface> enumNetworkInterfaces
= NetworkInterface.getNetworkInterfaces();
                while (enumNetworkInterfaces.hasMoreElements()) {
                    NetworkInterface networkInterface =
enumNetworkInterfaces.nextElement();
                    Enumeration<InetAddress> enumInetAddress =
networkInterface.getInetAddresses();
                    while (enumInetAddress.hasMoreElements()) {
                        InetAddress inetAddress =
enumInetAddress.nextElement();
                        if (inetAddress.isSiteLocalAddress()) {
//Saca ip

                            String obtIp =
inetAddress.getHostAddress();
                            ipBD = obtIp.trim();
                            Log.d("#Final_Ip:", ipBD);
                        }
                    }
                }
            }
        }
    });
}

```

```

    }
} catch (SocketException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
    ipBD += "Something Wrong! " + e.toString() + "\n";
}
//sentencia que carga el php para la base de datos
new
CargarDatos().execute("http://192.168.1.15:8080/webService/registro.php?
ci=" + etCi.getText().toString() + "&nombre=" +
etNombre.getText().toString()
+ "&edad=" + etEdad.getText().toString() +
"&tel=" + etTelefono.getText().toString() + "&ip=" + ipBD);
}
});
btnAyuda.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

    startActivity(new Intent(stapd.this,pedirayuda.class));
    /*try {
        Enumeration<NetworkInterface> enumNetworkInterfaces
= NetworkInterface.getNetworkInterfaces();
        while (enumNetworkInterfaces.hasMoreElements()) {
            NetworkInterface networkInterface =
enumNetworkInterfaces.nextElement();
            Enumeration<InetAddress> enumInetAddress =
networkInterface.getInetAddresses();
            while (enumInetAddress.hasMoreElements()) {
                InetAddress inetAddress =
enumInetAddress.nextElement();
                if (inetAddress.isSiteLocalAddress()) {
//Saca ip
String obtIp =
inetAddress.getHostAddress();
                ipBD = obtIp.trim();
                Log.d("#Final_Ip:", ipBD);
            }
        }
    }
    new
CargarDatos().execute("http://192.168.1.19:8080/webService/registroIP.ph
p?ip=" + ipBD);

    AlertDialog.Builder alertDialog11 = new
AlertDialog.Builder(stapd.this);
    alertDialog11.setTitle("Mensaje de Ayuda");
    alertDialog11.setMessage("Su solicitud de ayuda fue
enviada con exito");
    alertDialog11.show();

} catch (Exception e) {

}*/
}
}

```

```

    });
    findViewById(R.id.btnDarAyuda).setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {

        startActivity(new Intent(stapd.this,Listado.class));
    }
});
// ATTENTION: This was auto-generated to implement the App
Indexing API.
// See https://g.co/AppIndexing/AndroidStudio for more
information.
    client = new
GoogleApiClient.Builder(this).addApi(AppIndex.API).build();
}
@Override
public void onStart() {
    super.onStart();
    // ATTENTION: This was auto-generated to implement the App
Indexing API.
    // See https://g.co/AppIndexing/AndroidStudio for more
information.
    client.connect();
    Action viewAction = Action.newAction(
        Action.TYPE_VIEW, // TODO: choose an action type.
        "stapd Page", // TODO: Define a title for the content
shown.
        // TODO: If you have web page content that matches this
app activity's content,
// make sure this auto-generated web page URL is
correct.
        // Otherwise, set the URL to null.
        Uri.parse("http://host/path"),
        // TODO: Make sure this auto-generated app URL is
correct.
        Uri.parse("android-
app://com.example.hp.stapd/http/host/path")
    );
    AppIndex.AppIndexApi.start(client, viewAction);
}
@Override
public void onStop() {
    super.onStop();
    // ATTENTION: This was auto-generated to implement the App
Indexing API.
    // See https://g.co/AppIndexing/AndroidStudio for more
information.
    Action viewAction = Action.newAction(
        Action.TYPE_VIEW, // TODO: choose an action type.
        "stapd Page", // TODO: Define a title for the content
shown.
        // TODO: If you have web page content that matches this
app activity's content,
// make sure this auto-generated web page URL is
correct.

```

```

        // Otherwise, set the URL to null.
        Uri.parse("http://host/path"),
        // TODO: Make sure this auto-generated app URL is
correct.
        Uri.parse("android-
app://com.example.hp.stapd/http/host/path")
    );
    AppIndex.AppIndexApi.end(client, viewAction);
    client.disconnect();
}
private class CargarDatos extends AsyncTask<String, Void, String> {
    @Override
    protected String doInBackground(String... urls) {
url.
        // params comes from the execute() call: params[0] is the
        try {
            return downloadUrl(urls[0]);
        } catch (IOException e) {
            return "Unable to retrieve web page. URL may be
invalid.";
        }
    }
    // onPostExecute displays the results of the AsyncTask.
    @Override
    protected void onPostExecute(String result) {
        Toast.makeText(getApplicationContext(), "Se almacenaron los
datos correctamente", Toast.LENGTH_LONG).show();
    }
}
private class ConsultarDatos extends AsyncTask<String, Void, String>
{
    @Override
    protected String doInBackground(String... urls) {
url.
        // params comes from the execute() call: params[0] is the
        try {
            return downloadUrl(urls[0]);
        } catch (IOException e) {
            return "Unable to retrieve web page. URL may be
invalid.";
        }
    }
    // onPostExecute displays the results of the AsyncTask.
    @Override
    protected void onPostExecute(String result) {
        JSONArray ja = null;
        try {
            ja = new JSONArray(result);
            etNombre.setText(ja.getString(1));
            etEdad.setText(ja.getString(2));
            etTelefono.setText(ja.getString(3));
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}
}

```

```

private String downloadUrl(String myurl) throws IOException {
    Log.i("URL", "" + myurl);
    myurl = myurl.replace(" ", "%20");
    InputStream is = null;
    // Only display the first 500 characters of the retrieved
    // web page content.
    int len = 500;
    try {
        URL url = new URL(myurl);
        HttpURLConnection conn = (HttpURLConnection)
url.openConnection();
        conn.setReadTimeout(10000 /* milliseconds */);
        conn.setConnectTimeout(15000 /* milliseconds */);
        conn.setRequestMethod("GET");
        conn.setDoInput(true);
        // Starts the query
        conn.connect();
        int response = conn.getResponseCode();
        Log.d("respuesta", "The response is: " + response);
        is = conn.getInputStream();
        // Convert the InputStream into a string
        String contentAsString = readIt(is, len);
        return contentAsString;
        // Makes sure that the InputStream is closed after the app
is
        // finished using it.
    } finally {
        if (is != null) {
            is.close();
        }
    }
}

private static final String TAG = "#intFinal_Ip";
private static final String TAG2 = "#El nivel es";

public String readIt(InputStream stream, int len) throws
IOException, UnsupportedEncodingException {
    Reader reader = null;
    reader = new InputStreamReader(stream, "UTF-8");
    char[] buffer = new char[len];
    reader.read(buffer);
    return new String(buffer);
}
}

```


- Programación de ubicación de la aplicación STAPD

```
package com.example.hp.stapd;

import android.app.Activity;
import android.app.AlertDialog;
import android.app.ProgressDialog;
import android.net.Uri;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.view.ViewGroupOverlay;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;

import com.google.android.gms.appindexing.Action;
import com.google.android.gms.appindexing.AppIndex;
import com.google.android.gms.common.api.GoogleApiClient;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.Reader;
import java.io.UnsupportedEncodingException;
import java.net.HttpURLConnection;
import java.net.InetAddress;
import java.net.NetworkInterface;
import java.net.SocketException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Enumeration;
import java.util.List;

public class Listado extends AppCompatActivity {

    String ipH = "", piso="", ala="";
    String obtIpSub = "";
    String BdIp2 = "";
    int inicio=0;

    /**
     * ATTENTION: This was auto-generated to implement the App Indexing
API.
     * See https://g.co/AppIndexing/AndroidStudio for more information.
     */
    private GoogleApiClient client;
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_listado);

    try {
        Enumeration<NetworkInterface> enumNetworkInterfaces =
NetworkInterface.getNetworkInterfaces();
        while (enumNetworkInterfaces.hasMoreElements()) {
            NetworkInterface networkInterface =
enumNetworkInterfaces.nextElement();
            Enumeration<InetAddress> enumInetAddress =
networkInterface.getInetAddresses();
            while (enumInetAddress.hasMoreElements()) {
                InetAddress inetAddress =
enumInetAddress.nextElement();
                if (inetAddress.isSiteLocalAddress()) { //Sacar ip
                    String ipH = inetAddress.getHostAddress();
                    obtIpSub = ipH.trim();
                    obtIpSub =
inetAddress.getHostAddress().substring(10);//cambiar a 10 los 2 ultimos
digitos de la IP

                    inicio=Integer.parseInt(obtIpSub);
                    inicio/=10;
                    inicio=inicio*10+1;
                    EditText txtDarAyuda = (EditText)
findViewById(R.id.txtDarAyuda);
                    txtDarAyuda.setText(obtIpSub);
                    new
ConsultarDatos().execute("http://192.168.1.15:8080/webService/obtenerIp.
php?rango="+inicio);
                }
            }
        }
    } catch (SocketException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
        ipH += "Something Wrong! " + e.toString() + "\n";
    }

    // ATTENTION: This was auto-generated to implement the App
Indexing API.
    // See https://g.co/AppIndexing/AndroidStudio for more
information.
    client = new
GoogleApiClient.Builder(this).addApi(AppIndex.API).build();
}

@Override
public void onStart() {
    super.onStart();

    // ATTENTION: This was auto-generated to implement the App
Indexing API.

```

```

        // See https://g.co/AppIndexing/AndroidStudio for more
information.
        client.connect();
        Action viewAction = Action.newAction(
            Action.TYPE_VIEW, // TODO: choose an action type.
            "Listado Page", // TODO: Define a title for the content
shown.
            // TODO: If you have web page content that matches this
app activity's content,
            // make sure this auto-generated web page URL is
correct.
            // Otherwise, set the URL to null.
            Uri.parse("http://host/path"),
            // TODO: Make sure this auto-generated app URL is
correct.
            Uri.parse("android-
app://com.example.hp.stapd/http/host/path")
        );
        AppIndex.AppIndexApi.start(client, viewAction);
    }

    @Override
    public void onStop() {
        super.onStop();

        // ATTENTION: This was auto-generated to implement the App
Indexing API.
        // See https://g.co/AppIndexing/AndroidStudio for more
information.
        Action viewAction = Action.newAction(
            Action.TYPE_VIEW, // TODO: choose an action type.
            "Listado Page", // TODO: Define a title for the content
shown.
            // TODO: If you have web page content that matches this
app activity's content,
            // make sure this auto-generated web page URL is
correct.
            // Otherwise, set the URL to null.
            Uri.parse("http://host/path"),
            // TODO: Make sure this auto-generated app URL is
correct.
            Uri.parse("android-
app://com.example.hp.stapd/http/host/path")
        );
        AppIndex.AppIndexApi.end(client, viewAction);
        client.disconnect();
    }

    private class ConsultarDatos extends AsyncTask<String, Void, String>
    {
        @Override
        protected String doInBackground(String... urls) {
            // params comes from the execute() call: params[0] is the
url.
            try {

```

```

        return downloadUrl(urls[0]);
    } catch (IOException e) {
        return "Unable to retrieve web page. URL may be
invalid.";
    }
}
// onPostExecute displays the results of the AsyncTask.
@Override
protected void onPostExecute(String result) {
    String mensaje="";
    JSONArray ja = null;
    try {
        if(inicio==11)
            mensaje="Necesita ayuda en Ala A Piso 3";
        if(inicio==21)
            mensaje="Necesita ayuda en el centro Piso 3";
        if(inicio==31)
            mensaje="Necesita ayuda en Ala B Piso 3";

        if(inicio==41)
            mensaje="Necesita ayuda en Ala A Piso 2";
        if(inicio==51)
            mensaje="Necesita ayuda en el centro Piso 2";
        if(inicio==61)
            mensaje="Necesita ayuda en Ala B Piso 2";

        if(inicio==71)
            mensaje="Necesita ayuda en Ala A Piso 1";
        if(inicio==81)
            mensaje="Necesita ayuda en el centro Piso 1";
        if(inicio==91)
            mensaje="Necesita ayuda en Ala B Piso 1";

        ja = new JSONArray(result);

        AlertDialog.Builder alertDialog1 = new
AlertDialog.Builder(Listado.this);
        alertDialog1.setTitle("Mensaje de Ayuda");

        if(ja.length(>0)
alertDialog1.setMessage("Una persona con discapacidad
"+ja.getString(1)+" "+mensaje);
        else
            alertDialog1.setMessage("Nadie necesita ayuda");
        alertDialog1.show();

    } catch (JSONException e) {
        AlertDialog.Builder alertDialog1 = new
AlertDialog.Builder(Listado.this);
        alertDialog1.setTitle("Mensaje de Ayuda");
        alertDialog1.setMessage("Nadie necesita ayuda");
        alertDialog1.show();
    }
}
}

```

```

    }
    private String downloadUrl(String myurl) throws IOException {
        Log.i("URL", "" + myurl);
        myurl = myurl.replace(" ", "%20");
        InputStream is = null;
        // Only display the first 500 characters of the retrieved
        // web page content.
        int len = 500;
        try {
            URL url = new URL(myurl);
            HttpURLConnection conn = (HttpURLConnection)
url.openConnection();
            conn.setReadTimeout(10000 /* milliseconds */);
            conn.setConnectTimeout(15000 /* milliseconds */);
            conn.setRequestMethod("GET");
            conn.setDoInput(true);
            // Starts the query
            conn.connect();
            int response = conn.getResponseCode();
            Log.d("respuesta", "The response is: " + response);
            is = conn.getInputStream();
            // Convert the InputStream into a string
            String contentAsString = readIt(is, len);
            return contentAsString;
            // Makes sure that the InputStream is closed after the app
is
            // finished using it.
        } finally {
            if (is != null) {
                is.close();
            }
        }
    }
    public String readIt(InputStream stream, int len) throws
IOException, UnsupportedEncodingException {
        Reader reader = null;
        reader = new InputStreamReader(stream, "UTF-8");
        char[] buffer = new char[len];
        reader.read(buffer);
        return new String(buffer);
    }
}

```

- Programación de ayuda de la aplicación STAPD

```

package com.example.hp.stapd;

import android.app.AlertDialog;
import android.net.Uri;
import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;

```

```

import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.appindexing.Action;
import com.google.android.gms.appindexing.AppIndex;
import com.google.android.gms.common.api.GoogleApiClient;

import org.json.JSONArray;
import org.json.JSONException;

import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.Reader;
import java.io.UnsupportedEncodingException;
import java.net.HttpURLConnection;
import java.net.InetAddress;
import java.net.NetworkInterface;
import java.net.SocketException;
import java.net.URL;
import java.util.Enumeration;

public class pedirayuda extends AppCompatActivity {
    Button btnGuardar, btnCancelar;
    EditText etCi;
    private GoogleApiClient client;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pedirayuda);

        btnGuardar = (Button) findViewById(R.id.btnPedir);
        btnCancelar = (Button) findViewById(R.id.btnEliminar);
        etCi = (EditText) findViewById(R.id.txtDiscapadad);

        btnGuardar.setOnClickListener(new View.OnClickListener() {

            String ipBD = "";

            public void onClick(View v) {
                //obtener direccion ip para almacenar en la base de
datos

                try {
                    Enumeration<NetworkInterface> enumNetworkInterfaces
= NetworkInterface.getNetworkInterfaces();
                    while (enumNetworkInterfaces.hasMoreElements()) {
                        NetworkInterface networkInterface =
enumNetworkInterfaces.nextElement();
                        Enumeration<InetAddress> enumInetAddress =
networkInterface.getInetAddresses();
                        while (enumInetAddress.hasMoreElements()) {
                            InetAddress inetAddress =
enumInetAddress.nextElement();

```

```

        if (inetAddress.isSiteLocalAddress()) {
//Saca ip
            String obtIp =
inetAddress.getHostAddress();
            ipBD = obtIp.trim();
            Log.d("#Final_Ip:", ipBD);
        }
    }
    }
    new
CargarDatos().execute("http://192.168.1.15:8080/webService/registroIP.php?ip=" + ipBD + "&nombre=" + "&capacidad=" + etCi.getText());

        AlertDialog.Builder alertDialog1 = new
AlertDialog.Builder(pedirayuda.this);
        alertDialog1.setTitle("Mensaje de Ayuda");
        alertDialog1.setMessage("Solicitud de ayuda enviada
con exito");
        alertDialog1.show();
    } catch (SocketException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
        ipBD += "Something Wrong! " + e.toString() + "\n";
    }
    }
});

btnCancelar.setOnClickListener(new View.OnClickListener() {

    String ipBD = "";

    public void onClick(View v) {
//obtener direccion ip para almacenar en la base de
datos

        try {
            Enumeration<NetworkInterface> enumNetworkInterfaces
= NetworkInterface.getNetworkInterfaces();
            while (enumNetworkInterfaces.hasMoreElements()) {
                NetworkInterface networkInterface =
enumNetworkInterfaces.nextElement();
                Enumeration<InetAddress> enumInetAddress =
networkInterface.getInetAddresses();
                while (enumInetAddress.hasMoreElements()) {
                    InetAddress inetAddress =
enumInetAddress.nextElement();
                    if (inetAddress.isSiteLocalAddress()) {
//Saca ip
                        String obtIp =
inetAddress.getHostAddress();
                        ipBD = obtIp.trim();
                        Log.d("#Final_Ip:", ipBD);
                    }
                }
            }
        }
    }
}

```

```

        new
CargarDatos().execute("http://192.168.1.15:8080/webService/cancelar.php?
ip=" + ipBD);

        AlertDialog.Builder alertDialog1 = new
AlertDialog.Builder(pedirayuda.this);
        alertDialog1.setTitle("Mensaje de Ayuda");
        alertDialog1.setMessage("Solicitud eliminada");
        alertDialog1.show();
    } catch (SocketException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
        ipBD += "Something Wrong! " + e.toString() + "\n";
    }
    }
});
}

private class CargarDatos extends AsyncTask<String, Void, String> {
    @Override
    protected String doInBackground(String... urls) {
        // params comes from the execute() call: params[0] is the
url.
        try {
            return downloadUrl(urls[0]);
        } catch (IOException e) {
            return "Unable to retrieve web page. URL may be
invalid.";
        }
    }
    // onPostExecute displays the results of the AsyncTask.
    @Override
    protected void onPostExecute(String result) {
        Toast.makeText(getApplicationContext(), "Se almacenaron los
datos correctamente", Toast.LENGTH_LONG).show();
    }
}
private String downloadUrl(String myurl) throws IOException {
    Log.i("URL", "" + myurl);
    myurl = myurl.replace(" ", "%20");
    InputStream is = null;
    // Only display the first 500 characters of the retrieved
// web page content.
    int len = 500;
    try {
        URL url = new URL(myurl);
        HttpURLConnection conn = (HttpURLConnection)
url.openConnection();
        conn.setReadTimeout(10000 /* milliseconds */);
        conn.setConnectTimeout(15000 /* milliseconds */);
        conn.setRequestMethod("GET");
        conn.setDoInput(true);
        // Starts the query
        conn.connect();

```



```

        int response = conn.getResponseCode();
        Log.d("respuesta", "The response is: " + response);
        is = conn.getInputStream();
        // Convert the InputStream into a string
        String contentAsString = readIt(is, len);
        return contentAsString;
        // Makes sure that the InputStream is closed after the app
is
        // finished using it.
    } finally {
        if (is != null) {
            is.close();
        }
    }
}

public String readIt(InputStream stream, int len) throws
IOException, UnsupportedEncodingException {
    Reader reader = null;
    reader = new InputStreamReader(stream, "UTF-8");
    char[] buffer = new char[len];
    reader.read(buffer);
    return new String(buffer);
}

```

ANEXO 2

- Configuración de los puertos con la Vlan 252.

```
login as: admin
admin@192.168.112.36's password:

SW_ACC_FI_AD2_P1_DW#enable
SW_ACC_FI_AD2_P1_DW#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SW_ACC_FI_AD2_P1_DW(config)#inter
SW_ACC_FI_AD2_P1_DW(config)#interface ra
SW_ACC_FI_AD2_P1_DW(config)#interface range fa
SW_ACC_FI_AD2_P1_DW(config)#interface range fastEthernet 0/9, fas
SW_ACC_FI_AD2_P1_DW(config)#range fastEthernet 0/9, fastEthernet 0/11, fas
SW_ACC_FI_AD2_P1_DW(config)#9, fastEthernet 0/11, fastEthernet 0/13
SW_ACC_FI_AD2_P1_DW(config-if-range)#swit
SW_ACC_FI_AD2_P1_DW(config-if-range)#switchport mode acc
SW_ACC_FI_AD2_P1_DW(config-if-range)#switchport mode access
SW_ACC_FI_AD2_P1_DW(config-if-range)#sw
SW_ACC_FI_AD2_P1_DW(config-if-range)#switchport acc
SW_ACC_FI_AD2_P1_DW(config-if-range)#switchport access vlan 252
SW_ACC_FI_AD2_P1_DW(config-if-range)#exit
SW_ACC_FI_AD2_P1_DW(config)#end
SW_ACC_FI_AD2_P1_DW#wr
Building configuration...

[OK]
SW_ACC_FI_AD2_P1_DW#sh
SW_ACC_FI_AD2_P1_DW#show run
Building configuration...

Current configuration : 6959 bytes
!
version 12.2
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
```

```
!
interface FastEthernet0/8
 switchport mode trunk
!
interface FastEthernet0/9
 switchport access vlan 252
 switchport mode access
 switchport voice vlan 20
 spanning-tree portfast
!
interface FastEthernet0/10
 switchport access vlan 106
 switchport mode access
 switchport voice vlan 20
 spanning-tree portfast
!
interface FastEthernet0/11
 switchport access vlan 252
 switchport mode access
 switchport voice vlan 20
 spanning-tree portfast
!
interface FastEthernet0/12
 switchport access vlan 106
 switchport mode access
 switchport voice vlan 20
 spanning-tree portfast
!
interface FastEthernet0/13
 switchport access vlan 252
 switchport mode access
 switchport voice vlan 20
 spanning-tree portfast
!
--More--
```