

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**DESARROLLO DE UN PROTOTIPO DE APLICACIÓN MÓVIL
TURÍSTICA CON LA INFORMACIÓN DEL CANTÓN BAÑOS**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
“INGENIERO EN ELECTRÓNICA Y REDES DE INFORMACIÓN”**

LUIS IVÁN MEZA MORALES

luis.meza@epn.edu.ec

DIRECTOR: Dr. LUIS FELIPE URQUIZA AGUIAR

luis.urquiza@epn.edu.ec

Quito, Julio 2019

AVAL

Certifico que el presente trabajo fue desarrollado por Luis Iván Meza Morales, bajo mi supervisión.

Dr. Luis Felipe Urquiza
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Luis Iván Meza Morales, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Luis Iván Meza Morales

DEDICATORIA

A mi familia, a mis Padres Sra. Rebeca Morales y el Sr. Iván Meza que a pesar del tiempo han estado siempre junto a mi apoyándome, siendo mi fortaleza para seguir adelante, siendo mi soporte y el pilar fundamental para cumplir con esta meta.

A mis hermanos, Ivone y Stalin que día tras día me motivan a continuar, siendo parte importante de mi vida, inyectado siempre su fuerza y amor para salir adelante.

A mi novia Bri que ahora comparte mi vida y es quien me acompaña durante todo este proceso, siendo también ese motor que me impulsa a seguir adelante.

A mi Ñaño Calo siempre me ha apoyado y ha estado a mi lado, Karlita que con sus ocurrencias siempre me saca una sonrisa me alegra y me da fuerzas para seguir.

A todos mis amigos que he cosechado durante toda esta larga estancia en la universidad.

AGRADECIMIENTO

Primero a Dios, por ser quien ha guiado mis pasos durante todo este largo camino y hoy me ha traído hasta aquí.

A mi Madre Sra. Rebeca Morales que siempre ha estado junto a mí no a desmayado en impulsarme a seguir adelante por todos los días llamarme y hablar conmigo por todos los días orar por mí.

A mi Padre Sr. Iván Meza por siempre está conmigo y por ser un gran padre y demostrarnos que debemos ser personas de bien y seguir adelante a pesar de las adversidades.

A mis hermanos Ivone, Stalin que siempre me dan la mano en todo y que están conmigo, en las buenas y en las malas.

A mi novia Bri por ser un pilar fundamental para lograr esta meta.

A mi Ñaño Calo siempre me ha apoyado y ha estado a mi lado, Karlita que con sus ocurrencias siempre me saca una sonrisa me alegra y me da fuerzas para seguir.

A mi director de tesis Luis por ser paciente y apoyarme durante los semestres que fue mi tutor académico y durante todo este proceso de titulación.

A todos mis amigos que ahora son mi nueva familia Gracias, Garbis, Fausto, Big Edgar, Fernando, Gabo, Eli, André, José, Pato, Ángel, Sebas, David, Vini, Diego.

Para mis amigos con los que compartí los últimos semestres y que formaron parte importante para conseguir este logro, Cris Vivar, Johan, Carlos, Alejo, Diego, Christian, Cleo, José y Edu.

Y a todos los amigos con los que he compartido durante esta vida estudiantil que son muchos que faltara hoja para poner los nombres. Gracias por siempre estar en los buenos, malos momentos, principalmente en los momentos de estudio y de “sano esparcimiento”.

ÍNDICE DE CONTENIDO

AVAL	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS	XI
ÍNDICE DE CÓDIGOS	XIII
RESUMEN	XIV
ABSTRACT	XV
1. INTRODUCCIÓN.....	1
1.1 OBJETIVOS	2
1.2 ALCANCE.....	2
1.3 MARCO TEÓRICO.....	4
1.3.1 DISPOSITIVOS MÓVILES.....	4
1.3.2 APLICACIONES MÓVILES.....	4
1.3.2.1 Aplicaciones web	5
1.3.2.2 Aplicaciones Nativas.....	6
1.3.2.3 Aplicaciones Híbridas	6
1.3.3 PHP.....	7
1.3.4 ANDROID.....	7
1.3.4.1 Características de Android.....	7
1.3.4.2 Inconvenientes de Android.....	8
1.3.5 ANDROID STUDIO	9
1.3.6 WAMPSERVER	10
1.3.7 MySQL GESTOR DE BASE DE DATOS	10
1.3.8 AMAZON WEB SERVICE	11
1.3.9 SERVICIO WEB.....	12
1.3.9.1 REST	12
1.3.9.2 JSON	13
1.3.10 METODOLOGÍA DE DESARROLLO DE SOFTWARE KANBAN	13

2. METODOLOGÍA.....	15
2.1 DISEÑO.....	15
2.1.1 SITUACIÓN ACTUAL DE COMO SE MANEJA LA INFORMACIÓN TURÍSTICA DEL CANTÓN BAÑOS.....	15
2.1.2 TABLERO DE ACTIVIDADES KANBAN.....	16
2.1.3 ANÁLISIS DE REQUERIMIENTOS.....	18
2.1.3.1 Requerimientos Funcionales (RF).....	18
2.1.3.2 Requerimientos No Funcionales (RNF).....	19
2.1.4 MÓDULOS DEL PROTOTIPO.....	19
2.1.5 DISEÑO DE LA CAPA DE DATOS.....	21
2.1.5.1 Diagrama Entidad Relación.....	21
2.1.6 DISEÑO DE LA CAPA DE NEGOCIO.....	23
2.1.6.1 Diagrama de Casos de Uso (DCU).....	23
2.1.6.2 Diseño del Diagrama de Clases.....	28
2.1.6.3 Diseño de los Servicios Web.....	29
2.1.7 DISEÑO DE LA CAPA DE PRESENTACIÓN.....	29
2.1.7.1 Diseño de las Vistas del prototipo de Aplicación Usuario Turista.....	30
2.1.7.2 Diseño de las Vistas del prototipo de Aplicación Usuario Administrador.....	33
2.2 IMPLEMENTACIÓN.....	37
2.2.1 TABLERO DE ACTIVIDADES KANBAN.....	37
2.2.2 INSTALACIÓN DE LAS HERRAMIENTAS.....	39
2.2.2.1 Creación de una instancia de Amazon EC2 en AWS.....	39
2.2.2.2 WampServer.....	42
2.2.2.3 Android Studio.....	43
2.2.3 CODIFICACIÓN DE LA CAPA DE BASE DE DATOS.....	45
2.2.4 CODIFICACIÓN DE LA CAPA DE NEGOCIO.....	47
2.2.4.1 Servicio web.....	47
2.2.4.2 Consultas en Android.....	51
2.2.5 CODIFICACIÓN DE LA CAPA DE PRESENTACIÓN.....	53
2.2.5.1 Codificación de la Clase Sitio.....	53
2.2.5.2 Codificación de las vistas del Prototipo de aplicación.....	54
3. RESULTADOS Y DISCUSIÓN.....	61
3.1 TABLERO DE ACTIVIDADES KANBAN.....	62

3.2 PRUEBAS DE VALIDACIÓN DE LOS REQUERIMIENTOS FUNCIONALES.....	63
3.2.1 PRUEBAS DE VALIDACIÓN DE RFT01: VISUALIZAR CLASIFICACIÓN DE SITIO O ACTIVIDAD POR CATEGORÍAS	63
3.2.2 PRUEBAS DE VALIDACIÓN DE RFT02: VISUALIZAR UNA FOTO Y DESCRIPCIÓN DEL SITIO O ACTIVIDAD	64
3.2.3 PRUEBAS DE VALIDACIÓN DE RFT03: VISUALIZAR EN UN MAPA COMO LLEGAR AL SITIO SELECCIONADO	65
3.2.4 PRUEBAS DE VALIDACIÓN DE RFT04: BUSCAR SITIOS O ACTIVIDADES	66
3.2.5 PRUEBAS DE VALIDACIÓN DE RFT05: ORDENAR LOS SITIOS SEGÚN SUS PRECIOS ASCENDENTE O DESCENDENTE	67
3.2.6 PRUEBAS DE VALIDACIÓN DE RFT06: ORDENAR LOS SITIOS SIGUIENDO UN ORDEN ALFABÉTICO A-Z O Z-A.....	68
3.2.7 PRUEBAS DE VALIDACIÓN DE RFT07: CALIFICAR UN SITIO O ACTIVIDAD	70
3.2.8 PRUEBAS DE VALIDACIÓN DE RFT08: ALMACENAR IMÁGENES DEL SITIO O ACTIVIDAD.....	71
3.2.9 PRUEBAS DE VALIDACIÓN DE RFT09: ALMACENAR LA INFORMACIÓN CONCERNIENTE ADMINISTRADORES.....	72
3.2.10 PRUEBAS DE VALIDACIÓN DE RFT10: ALMACENAR LA INFORMACIÓN DE LOS SITIOS TURÍSTICOS.	73
3.2.11 PRUEBAS DE VALIDACIÓN DE RFT11: AUTENTICAR UN ADMINISTRADOR.....	73
3.2.12 PRUEBAS DE VALIDACIÓN DE RFT12: CREAR NUEVA CATEGORÍA.....	74
3.2.13 PRUEBAS DE VALIDACIÓN DE RFT13: EDITAR CATEGORÍA.....	75
3.2.14 PRUEBAS DE VALIDACIÓN DE RFT14: ELIMINAR CATEGORÍA. ...	75
3.2.15 PRUEBAS DE VALIDACIÓN DE RFT15: CREAR NUEVO SITIO O ACTIVIDAD.....	76
3.2.16 PRUEBAS DE VALIDACIÓN DE RFT16: EDITAR SITIO O ACTIVIDAD.....	77
3.2.17 PRUEBAS DE VALIDACIÓN DE RFT17: ELIMINAR SITIO O ACTIVIDAD.....	78
3.2.18 PRUEBAS DE VALIDACIÓN DE RFT18: CREAR NUEVO USUARIO.....	79
3.2.19 PRUEBAS DE VALIDACIÓN DE RFT19: EDITAR USUARIO.....	80
3.2.20 PRUEBAS DE VALIDACIÓN DE RFT20: ELIMINAR USUARIO.....	81
3.3 PRUEBAS DE VALIDACIÓN DE LOS REQUERIMIENTOS NO FUNCIONALES.....	82

3.3.1	REQUERIMIENTO NO FUNCIONAL RNFT01: PRESENTACIÓN	83
3.3.2	REQUERIMIENTO NO FUNCIONAL RNFT02: DISPONIBILIDAD	84
3.3.3	REQUERIMIENTO NO FUNCIONAL RNFT03: ESCALABILIDAD.....	84
3.3.4	REQUERIMIENTO NO FUNCIONAL RNFT04: AUTENTICACIÓN	84
3.4	PRUEBAS DE FUNCIONAMIENTO DEL PROTOTIPO.....	86
3.4.1	RESPUESTAS A ENCUESTAS USUARIOS ADMINISTRADORES.....	89
3.4.2	RESPUESTAS A ENCUESTAS USUARIOS TURISTAS.....	90
3.5	TABLERO FINAL KANBAN.....	92
4.	CONCLUSIONES Y RECOMENDACIONES.....	94
4.1	CONCLUSIONES.....	94
4.2	RECOMENDACIONES	95
5.	REFERENCIAS BIBLIOGRÁFICAS	97
6.	ANEXOS.....	104
	ANEXO A. ENCUESTAS OBTENCIÓN DE REQUERIMIENTOS.....	104
	ANEXO B. CÓDIGO BASE DE DATOS	104
	ANEXO C. CÓDIGO SERVICIO WEB	104
	ANEXO D. CÓDIGO APLICACIÓN.....	104
	ANEXO E. ENCUESTAS DE PRUEBAS	104
	ORDEN DE EMPASTADO.....	105

ÍNDICE DE FIGURAS

Figura 1.1 Esquema de Diseño del Prototipo.	3
Figura 2.1 Diagrama Entidad-Relación de la Capa de Base de Datos.	23
Figura 2.2 Usuarios Turista y Administrador.....	24
Figura 2.3 Caso de Uso del Subsistema Módulo Principal.	26
Figura 2.4 Caso de Uso del Subsistema Módulo Almacenamiento.	26
Figura 2.5 Caso de Uso del Subsistema Módulo Administración.	27
Figura 2.6 Caso de Uso Buscar Sitio o Actividad.	27
Figura 2.7 Diagrama de Clases del Prototipo de Aplicación.....	28
Figura 2.8 Arquitectura del Servicio Web.	29
Figura 2.9 Vista de Pantalla de Inicio de la Aplicación	30
Figura 2.10 Vista Sitios Dentro de Categoría	31
Figura 2.11 Vista que Muestra la Información del Sitio o Actividad.	32
Figura 2.12 Vista que Muestra la Ubicación y la Ruta para Llegar al Sitio o Actividad.....	32
Figura 2.13 Vista de la Barra de Tareas o Barra de Acceso Rápido.	33
Figura 2.14 Vista de Autenticación del Usuario Administrador.....	34
Figura 2.15 Vista de Recuperar Contraseña	34
Figura 2.16 Vista de las opciones de administrador.....	35
Figura 2.17 Vista de Administración de Usuarios.....	35
Figura 2.18 Vista del Formulario de un Nuevo Usuario.....	36
Figura 2.19 Vista de Pantalla para Editar o Borrar un Usuario Existente.	37
Figura 2.20 Arquitectura del Servidor.....	39
Figura 2.21 Crear Cuenta de AWS.....	39
Figura 2.22 Selección de EC2.....	40
Figura 2.23 Crear Instancia AWS.....	40
Figura 2.24 Selección de Imagen de AWS.....	41
Figura 2.25 Características de la Instancia Escogida.....	41
Figura 2.26 Características del Tipo de Servidor Escogido.....	41
Figura 2.27 Interfaz para Conectar al Servidor Remoto en la Nube AWS.....	42
Figura 2.28 Interfaz del Servidor en AWS.....	42
Figura 2.29 Descarga del WampServer 64 bits.....	43
Figura 2.30 WampServer instalado y todos los servicios activos.	43
Figura 2.31 Descarga de Android studio for Windows 64-bit.....	44
Figura 2.32 Instalación de Android Studio.....	44
Figura 2.33 Creación de Nuevo Proyecto de Android.....	45
Figura 2.34 Lista de Categorías.....	55
Figura 2.35 Sitios Asociados a una Categoría.....	57
Figura 2.36 Información de Sitio.....	58
Figura 2.37 Muestra Ruta Para Llegar al Sitio.....	60
Figura 3.1 Visualizar Clasificación de Sitio o Actividad por Categorías.....	63
Figura 3.2 Visualización de Imagen e Información de Sitio.....	64
Figura 3.3 Visualización de la Ruta a un Sitio.....	65

Figura 3.4 Búsqueda de Sitio en Submenú General	66
Figura 3.5 Búsqueda de sitio dentro de Categoría	67
Figura 3.6 Ordenar Sitios por Precio Descendente y Ascendente en Submenú General	68
Figura 3.7 Ordenar Sitios por Precio Descendente y Ascendente dentro de Categoría.....	68
Figura 3.8 Ordenar Sitios Orden Alfabético AZ o ZA en Submenú General.....	69
Figura 3.9 Ordenar Sitios Orden Alfabético AZ o ZA dentro de Categorías	70
Figura 3.10 Calificar un Sitio o Actividad	71
Figura 3.11 Almacenar Imágenes de un Sitio o Actividad	72
Figura 3.12 Autenticar un Administrador	74
Figura 3.13 Crear Categoría.....	74
Figura 3.14 Editar Categoría	75
Figura 3.15 Eliminar Categoría.....	76
Figura 3.16 Crear Sitio o Actividad	77
Figura 3.17 Editar Sitio o Actividad.....	78
Figura 3.18 Eliminar Sitio o Actividad	79
Figura 3.19 Crear Usuario	80
Figura 3.20 Editar Usuario.....	80
Figura 3.21 Eliminar Usuario	81
Figura 3.22 Autenticación de Usuario.....	85

ÍNDICE DE TABLAS

Tabla 1.1	Características De Android	8
Tabla 1.2	Funcionalidades Adicionales de Android Studio	9
Tabla 1.3	Características de MySQL	11
Tabla 2.1	Tablero Kanban	17
Tabla 2.2	Requerimientos Funcionales.....	18
Tabla 2.3	Requerimientos No Funcionales	19
Tabla 2.4	Estimación de Ejecución de los RF del Módulo Principal.....	20
Tabla 2.5	Estimación de Ejecución de los RF del Módulo de Almacenamiento...	20
Tabla 2.6	Estimación de Ejecución de los RF del Módulo de Administración.....	21
Tabla 2.7	Estimación de ejecución de los RF del Módulo Cliente.....	21
Tabla 2.8	Descripción de los RF, Casos de Uso y Subsistema Asociado.	25
Tabla 2.9	Tablero Kanban de la Fase de Implementación.....	38
Tabla 3.1	Tablero de Actividades Kanban	62
Tabla 3.2	Descripción de RFT01	63
Tabla 3.3	Descripción de RFT02	64
Tabla 3.4	Descripción de RFT03	65
Tabla 3.5	Descripción de RFT04	66
Tabla 3.6	Descripción de RFT05	67
Tabla 3.7	Descripción de RFT06	69
Tabla 3.8	Descripción de RFT07	70
Tabla 3.9	Descripción de RFT08	71
Tabla 3.10	Descripción de RFT09	72
Tabla 3.11	Descripción de RFT10	73
Tabla 3.12	Descripción de RFT11	73
Tabla 3.13	Descripción de RFT12	74
Tabla 3.14	Descripción de RFT13	75
Tabla 3.15	Descripción de RFT14	76
Tabla 3.16	Descripción de RFT15	77
Tabla 3.17	Descripción de RFT16	77
Tabla 3.18	Descripción de RFT17	78
Tabla 3.19	Descripción de RFT18	79
Tabla 3.20	Descripción de RFT19	80
Tabla 3.21	Descripción de RFT20	81
Tabla 3.22	Validación y Cumplimiento de Todos los RF	82
Tabla 3.23	Requerimientos No Funcionales	83
Tabla 3.24	Resultados Pregunta Presentación.....	83
Tabla 3.25	Respuesta Pregunta Autenticación	85
Tabla 3.26	Validación / Cumplimiento RNFT	85
Tabla 3.27	Respuestas a la Pregunta Uno Administradores	89
Tabla 3.28	Resultados Pregunta Dos Administradores.	90
Tabla 3.29	Respuesta Preguntas Usuario Administrador	90
Tabla 3.30	Respuesta Pregunta Uno Usuario Turista.....	91
Tabla 3.31	Respuestas a las Preguntas del Usuario Turista	91

Tabla 3.32 Respuesta Pregunta Usuario Turista y Administrador	92
Tabla 3.33 Tablero Final Kanban.....	93

ÍNDICE DE CÓDIGOS

Código 2.1 Codificación de la Tabla Sitio.....	46
Código 2.2 Consulta a la Tabla Sitio.....	47
Código 2.3 Código de conexión con la Base de Datos.....	47
Código 2.4 Código del Archivo Index.php.....	48
Código 2.5 Método para consultar todos los sitios desde A hasta la Z.....	49
Código 2.6 Código eliminarSitiold.....	50
Código 2.7 Código consultarUsuariold.....	51
Código 2.8 Código consultarTodosSitios.....	52
Código 2.9 Código insertarSitio.....	52
Código 2.10 Código Clase Sitio.....	53
Código 2.11 Código de Clase TipoSito.....	54
Código 2.12 Generar Lista de Categorías.....	56
Código 2.13 Búsqueda de Sitio dentro de Categoría.....	58
Código 2.14 Descripción e Información del Sitio.....	59
Código 2.15 Selección Geográfica del Sitio.....	61

RESUMEN

Baños se ha convertido en el uno de los cantones más visitados por quienes quieren practicar deportes de aventura, relajarse en sus piscinas de aguas termales, disfrutar de una gran biodiversidad. Eso se debe a que Baños tiene una ubicación privilegiada a la entrada de la Amazonía, contando así con muchos atractivos turísticos naturales.

En el presente Proyecto de Titulación se desarrolla un prototipo de aplicación móvil turística la cual utiliza la información del cantón Baños. En el capítulo uno se desarrolla el marco teórico, La aplicación está desarrollada bajo la plataforma Android y permite acceder desde cualquier lugar a la información turística de Baños. Toda la información está clasificada en diferentes categorías y por cada punto de interés, el usuario visualiza la información general del sitio o actividad escogida. Entre la información que se visualiza por cada elemento incluido en la base de datos se tiene: descripción, teléfonos, mail, página web, dirección, el horario, etc.

Una de las funciones implementadas es que, al dar clic en la dirección del punto de interés del usuario, se muestra su ubicación y una ruta de cómo llegar a dicho sitio. Además, de la función de búsqueda, se han especificado una función para ordenar los sitios de acuerdo al precio, desde el más barato al más caro y viceversa; también se incluye ordenar de forma alfabética desde la A hasta la Z y de la Z hasta la A, permitiendo así que el usuario pueda encontrar de forma fácil lo que necesita.

En el Capítulo 1, se describe el objetivo general y el alcance que tiene el presente Proyecto de Titulación, además, se menciona los fundamentos teóricos de Servicios Web, Android Studio y Base de Datos.

En el Capítulo 2, se describe la metodología para el desarrollo del presente Proyecto de Titulación, la cual es dividida en dos partes: diseño e implementación.

En el Capítulo 3, se verifica el correcto funcionamiento del Prototipo de Aplicación Móvil Turística con la Información del cantón Baños y se muestra los resultados de las pruebas realizadas.

En el Capítulo 4, se describe las respectivas conclusiones y recomendaciones.

PALABRAS CLAVE: Aplicación Móvil, Base de Datos, Servicios en la Nube, Cantón Baños, Lugares Turísticos.

ABSTRACT

The city of Baños has been converted into one of the sectors most visited for those who want to practice adventure sports, relax in one of the many thermal pools or enjoy the immense biodiversity. All of this is due to Baños' location at edge of the Amazon creating these natural tourist attractions.

A mobile tourist application prototype, centered around the city of Baños, has been created for this project. The application is developed for the use solely on Android platforms and will allow the access of information about any of the tourist spots in Baños, from any location in the world. All of the information is classified into different categories and for every point of interest the user will be able to visualize the general information about the location or the chosen activity. The database includes a wide range of information such as description, telephone number, email, webpage, address and schedule for each tourist attraction.

One of the application's functions, provided to users, is the access to directions to the chosen spot of interest based on the user's current location. Also included is the option to search for specific tourist spots as well as order search results according to price, from the least expensive to the most expensive and vice versa, alphabetically or the reverse allowing the user to find what they need quickly and easily.

In Chapter 1, the general objective and reach of this thesis project is described, as well as the theoretical foundations of Web Services, Android Studio and Database.

In Chapter 2, the methodology for the development of this thesis project is described, which will be divided into two parts: design and implementation.

In Chapter 3, the proper functioning of the mobile tourist application prototype with the information of the city of Baños will be verified and the results of the tests carried out will be shown.

In Chapter 4, the respective conclusions and recommendations will be described.

KEY WORDS: Mobile Application, Database, Services in the Cloud, City of Baños, Tourist Spots.

1. INTRODUCCIÓN

Baños de Agua Santa se encuentra ubicada en la provincia de Tungurahua, en los flancos externos de la cordillera oriental de los Andes, entre el río Pastaza, el río Bascún y las faldas del volcán Tungurahua, a una altitud de 1820 msnm y con un clima lluvioso tropical de 19°C en promedio. Es un cantón turístico con una buena reputación en el trato al turista y opciones de ocio variadas, por lo que se encuentra entre las principales ciudades turísticas como Quito, Guayaquil, Cuenca. Más aun siendo una de las primeras ciudades visitadas por quienes quieren realizar la práctica de deportes de aventura, descansar en sus piscinas de aguas termales y está dotada de una gran biodiversidad. Debido a su majestuosidad natural, el cantón obtuvo el galardón denominado “Baños, un regalo para la tierra” otorgado por la WWF (World Wild Fund for Nature; ‘Fondo Mundial para la Naturaleza’), por esta razón Baños ha estado en constante cambio e innovación en el ámbito turístico [1] [2].

A través de los años el cantón Baños ha ido incrementado notablemente su oferta turística y gracias al apoyo del gobierno nacional por medio del Ministerio de Turismo y su campaña turística "All You Need Is Ecuador"[3].

El Internet es hoy una de las principales herramientas de búsqueda de cualquier información y en especial la turística. Del cantón Baños se tiene información en Internet, pero en diferentes páginas, muy dispersa, enfocada talvez solo a una empresa en particular haciendo que la experiencia de búsqueda del usuario se limite a esa empresa. Por otro lado, disponer de un dispositivo inteligente se ha convertido en una necesidad del día a día, porque son instrumentos de gestión para la vida cotidiana gracias a las diferentes aplicaciones móviles que se pueden instalarle, de esta forma, por ejemplo: si se quiere viajar se podrá planificar con anticipación los lugares por visitar, localizarlos, saber sus números de contacto, horarios, etc.

Para facilitar la búsqueda de información del cantón Baños, se plantea el desarrollo de un prototipo de aplicación móvil turística la cual funciona sobre el sistema operativo Android, de esta manera los turistas pueden acceder desde cualquier lugar a la información turística del Cantón Baños; también puede ser usada para mostrar sitios turísticos no conocidos y promocionar los espacios públicos remodelados.

1.1 OBJETIVOS

El objetivo general de este Trabajo Técnico es: Desarrollar un prototipo de aplicación móvil turística con la información del cantón Baños.

Los objetivos específicos de este Trabajo Técnico son:

- Analizar los conceptos fundamentales de Android, gestor de base de datos, servicio web de tipo REST.
- Diseñar los módulos del proyecto que son necesarios para su funcionamiento.
- Implementar el prototipo de aplicación móvil.
- Analizar el funcionamiento del prototipo de aplicación móvil una vez implementada mediante la realización de las pruebas correspondientes.

1.2 ALCANCE

Se pretende desarrollar un prototipo de aplicación móvil sobre el sistema operativo Android el cual permita dar a conocer los atractivos turísticos que tiene el cantón Baños. Esta aplicación facilita la búsqueda de información sobre: ¿qué hacer?, ¿dónde comer?, ¿dónde alojarse?, los deportes extremos que se pueden practicar y los diferentes balnearios que se pueden visitar.

Además, ya que el cantón Baños está ubicado en las faldas del volcán Tungurahua, existiendo así un riesgo latente de erupción por lo que se ha visto necesario incluir en el prototipo de aplicación la información de los lugares que han sido identificados como sitios seguros, donde la población puede acudir en caso de una emergencia de esa índole; permitiéndole así al usuario de la aplicación conocer previamente estos sitios, teniendo una idea clara de cómo llegar a los mismos.

La metodología Kanban es utilizada como mecanismo de gestión del proceso de desarrollo del prototipo de aplicación móvil [4].

En la Figura 1.1 se presenta el esquema propuesto del prototipo el cual cuenta con el prototipo de aplicación móvil, una base de datos y un servicio web de tipo REST.

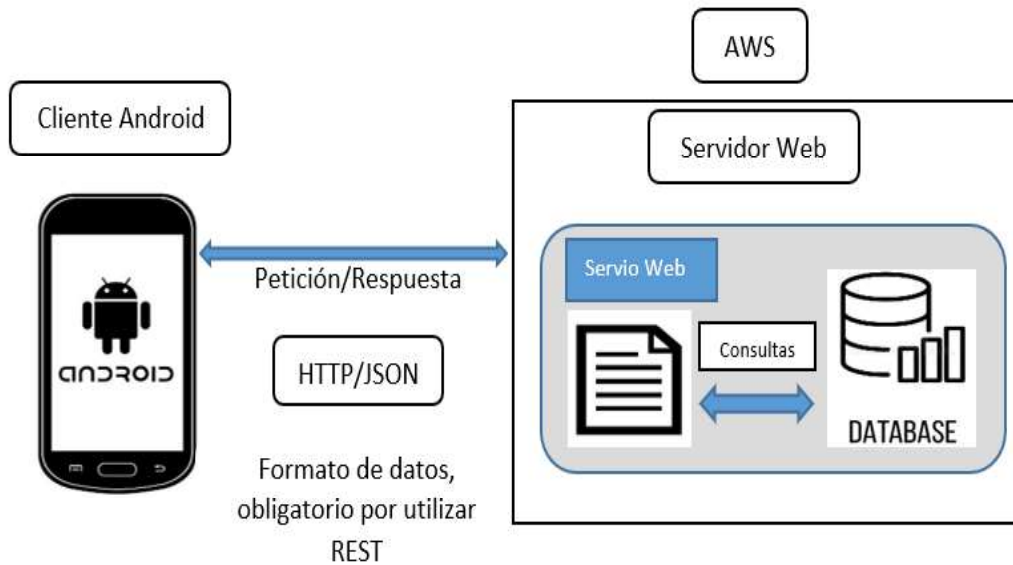


Figura 1.1 Esquema de Diseño del Prototipo.

El prototipo de aplicación móvil tiene los siguientes módulos:

Módulo principal: este módulo implementa un servicio REST, el mismo que se encarga de establecer la comunicación entre los dispositivos móviles y el servidor de datos [5]. De acuerdo con la petición realizada se procede a presentar la información pertinente a los clientes. Este servicio web brinda la posibilidad de consultar los sitios turísticos ordenados alfabéticamente; ordenados ascendente y descendente de acuerdo con el precio del servicio del sitio; búsqueda exacta de sitios, es decir, por ejemplo, el sitio exacto “pailón del diablo”; clasificación de los lugares turísticos de acuerdo con su categoría, por ejemplo, “Deportes Extremos”; y consultas del detalle de la información general del sitio turísticos.

Módulo de almacenamiento: este módulo implementa una base de datos [6] para almacenar la información concerniente de los lugares turísticos por visitar en la ciudad de Baños, además de la información que se necesite para llevar a cabo la implementación del prototipo final.

Módulo de administración: permite ingresar, visualizar y modificar información turística dependiendo si es una actividad o un sitio.

Módulo cliente: permite visualizar toda la información turística que se ha cargado acerca del cantón Baños, además, permite ver la ubicación de cada uno de los lugares por visitar y al seleccionar un sitio turístico en la interfaz del usuario se muestre la ruta en un mapa para llegar a dicho sitio.

Finalmente, al prototipo se le realizan las pruebas de funcionamiento para comprobar que se realizó lo especificado, para esto se facilitará al menos a diez personas el prototipo de aplicación para que la instalen y utilicen, luego de eso se les aplica una encuesta a estos usuarios y según los resultados, se realizan las correcciones de la aplicación en caso de ser necesario.

1.3 MARCO TEÓRICO

El desarrollo del turismo en Baños se ha ido incrementado de manera considerable, es por tal motivo que el turismo es una de las principales actividades económicas del cantón [7].

Los dispositivos móviles en los últimos años han experimentado un gran crecimiento, provocando un incremento notable en su uso. Los dispositivos móviles permitieron que el acceso a la información en cualquier lugar y momento se convierta en una tarea cotidiana, creando la necesidad del desarrollo de aplicaciones móviles que satisfagan los requisitos de información de los usuarios.

Teniendo en cuenta que el turismo representa una gran fuente de ingresos para los Baños [8], se plantea el desarrollo de un prototipo de aplicación móvil turística, la cual permite mostrar la información de los sitios turísticos que pueden ser visitados, facilitando al turista el conocer previamente los servicios que se brindan o las actividades que pueden realizar.

1.3.1 DISPOSITIVOS MÓVILES

Los dispositivos móviles han evolucionado de manera notable en la última década hoy los fabricantes compiten entre ellos para ofrecer a los clientes, dispositivos con una gran capacidad, velocidad y de muy buenas características. Esto hace que los dispositivos móviles sean una herramienta útil, dejando de ser simples dispositivos de comunicación convirtiéndose en instrumentos de gestión de la vida diaria, gracias a las aplicaciones que se puede instalar en estos [9].

Una buena parte del mercado utiliza un sistema operativo gratuito diseñado por Google denominado Android, el mismo que los fabricantes pueden utilizar en sus productos, pudiendo incluir capas personalizadas de software para diferenciarse de otras empresas de la misma línea.

1.3.2 APLICACIONES MÓVILES

Las aplicaciones móviles son desarrolladas para ser ejecutadas en los dispositivos móviles, permitiendo así poder acceder a los datos desde cualquier lugar y hora del día. Cuando se

va a desarrollar una aplicación móvil se debe tener en cuenta ciertas restricciones con respecto al hardware de estos dispositivos móviles, como: las dimensiones reducidas, bajo poder de cómputo, ancho de banda limitado, escasa capacidad de almacenamiento, etc. Entre las aplicaciones móviles más desarrolladas se tiene: las de juegos, mensajería, fotografía, mapas y navegación, entrenamiento físico entre otras [10].

Las aplicaciones suelen estar disponibles en las plataformas de distribución las cuales comenzaron a aparecer en el 2008 y que generalmente son operadas por los propietarios del sistema operativo para móviles como: Apple App Store, Google Play, Windows Store, se tiene aplicaciones que son gratuitas y otras que son pagadas [11].

Las aplicaciones móviles generalmente, están destinadas a un gran número de usuarios finales y las mismas son liberadas en versiones rápidas para así poder satisfacer las demandas del mercado que cada día es más exigente y va desarrollándose aceleradamente.

El mercado de los dispositivos móviles en los últimos años ha mostrado un gran crecimiento siendo el nuevo reto para los para los proveedores de aplicaciones el proporcionar soluciones para todas las plataformas existentes, lo cual resulta de alto costo [12]. Los proveedores están solucionando este problema creando y manteniendo una única aplicación para todas las plataformas a lo que se denomina desarrollo multiplataforma.

Mantener la misma base de código para diferentes plataformas el objetivo de los desarrollares, permitiendo así la reducción del esfuerzo y costo de desarrollo. a continuación, se describe tres formas de desarrollo de aplicaciones móviles: un enfoque nativo y dos enfoques multiplataforma (web e híbrido).

1.3.2.1 Aplicaciones web

Las aplicaciones web que son diseñadas para los dispositivos móviles deben ser ejecutadas en el navegador del mismo. La tecnología utilizada para crear los sitios web es la misma que es utilizada para el desarrollo de las aplicaciones siendo estas: HTML, CSS y JavaScript [13].

Entre las ventajas que se tiene es que en el dispositivo móvil no necesita la instalación de ningún componente en particular, ni la aprobación de algún fabricante para que las aplicaciones sean utilizadas, lo único que se necesita es acceso a internet. Por otra parte, las actualizaciones de la aplicación se reflejan directamente en el dispositivo y están disponibles inmediatamente ya que los cambios realizados son hechos sobre el servidor.

Para este tipo de aplicaciones solo es necesario un navegador, siendo independiente de la plataforma.

Entre sus desventajas se tiene: bajo desempeño por problemas de conectividad, no utiliza todos los elementos de hardware del dispositivo, como: la cámara, el GPS, entre otros [14].

1.3.2.2 Aplicaciones Nativas

Las aplicaciones nativas son las que se ejecutan en una plataforma específica, ya que para su uso se considera el tipo de dispositivo, el sistema operativo y su versión. El código fuente de este tipo de aplicaciones es compilado para así obtener código ejecutable, siendo este proceso similar al utilizado en las aplicaciones de escritorio. Para la distribución de las aplicaciones luego de estar listas deben ser transferidas a las tiendas de aplicaciones (app stores) las que son específicas de cada sistema operativo [15]. Una vez ahí las mismas tendrán un proceso de auditoría donde se evaluará si la aplicación se adecúa a los requerimientos de la plataforma en la cual va a operar después de este proceso está disponible para los usuarios.

Las ventajas de este tipo de aplicaciones son: puede utilizar todos los elementos del dispositivo como: cámara, GPS, acelerómetro, agenda, entre otras, algunas trabajan sin acceso a internet, ejecución rápida, ejecución en modo de segundo plano (background) notificando al usuario cuando ocurra un evento que necesite su atención.

La desventaja es que se tiene un mayor costo de desarrollo, debido a que se debe utilizar un lenguaje de programación diferente para cada una de las plataformas en donde se quiere transferir la aplicación móvil causando mayores costos de actualización y distribución de nuevas versiones [16].

1.3.2.3 Aplicaciones Híbridas

Las aplicaciones híbridas son las que combinan lo mejor de los dos tipos de aplicaciones descritas anteriormente. Permiten acceder a una buena parte de las capacidades específicas de los dispositivos y utilizan tecnologías multiplataforma como HTML, Javascript y CSS, es decir, son desarrolladas utilizando tecnología web y son ejecutadas dentro de un contenedor web sobre el dispositivo móvil [17]. Las ventajas de esta metodología son: distribución de la aplicación a través de las tiendas de aplicaciones, la reutilización de código para múltiples plataformas y la posibilidad de utilizar las características de hardware del dispositivo.

La desventaja es que, al utilizar la misma interfaz para todas las plataformas, la apariencia de la aplicación no será como la de una aplicación nativa y la ejecución será más lenta que la ejecución en una aplicación nativa [18].

1.3.3 PHP

PHP es un lenguaje de programación de propósito general que es especialmente adecuado para el desarrollo web y también es un lenguaje totalmente gratuito, rápido, flexible y pragmático [19].

PHP es originalmente diseñado para el desarrollo web de contenido dinámico. PHP fue uno de los primeros lenguajes de programación que estaban del lado del servidor, que en lugar de llamar un archivo externo que procese los datos, este se podía incorporar directamente en el documento HTML. Para interpretar el código se necesita un servidor web con un módulo de procesador de PHP que genera la página web resultante. PHP puede ser usado en la mayoría de los servidores web y siendo el caso en sistemas operativos y plataformas sin ningún costo [20]. También incluye una interfaz de línea de comandos la cual puede ser usada en aplicaciones gráficas independientes.

1.3.4 ANDROID

Android es un sistema operativo, el cual es diseñado para teléfonos móviles al igual que los sistemas operativos iOS (Apple), Windows Mobile, Symbian OS, pero ahora Android ya no solo se instala en teléfonos móviles, sino también en: tabletas, GPS, televisores, electrodomésticos, juguetes, etc. Está basado en Linux y es un sistema operativo libre, gratuito y multiplataforma [21].

A partir de la versión 5.0 de Android, se utiliza una variación de Java llamada ART la cual permite programar aplicaciones, que acceden a todas las funciones del teléfono.

Una de las causas para que existan miles de aplicaciones disponibles en las tiendas de cada una de las plataformas es, que los dispositivos extienden su funcionalidad, mejoran la experiencia del usuario, su sencillez y que existan herramientas de programación gratuitas [22].

1.3.4.1 Características de Android

Android es un sistema operativo potente con grandes características. Algunas de ellas se muestran en la Tabla 1.1.

Tabla 1.1 Características De Android

Característica	Descripción
Conectividad	GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, WiFi, LTE, NFC y WiMAX
UI Amigable	Las pantallas básicas de Android OS proporcionan una interfaz de usuario intuitiva y amigable.
Soporte Multimedia	H.263, H.264, MPEG - 4 SP, AMR, AMR - WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, y BM.
Almacenamiento	SQLite, una base de datos relacional ligera, es usada para propósitos de almacenamiento.
Navegador Web	Basado en el motor de código abierto WebKit layout, junto con el motor JavaScript de Chrome V8 soportando HTML5 y CSS3.
Mensajería	SMS y MMS
Multitarea	El usuario puede saltar de una tarea a otra al mismo tiempo ejecutando simultáneamente varias aplicaciones.
Soporte Táctil	Android tiene soporte nativo multitáctil disponible en los teléfonos celulares inteligentes (smartphone)
WiFi Directo	Una tecnología que permite a las aplicaciones se sincronice directamente, a través de un ancho de banda con conexión de punto a punto.

1.3.4.2 Inconvenientes de Android

Los inconvenientes que ha presentado Android son por la fragmentación que sufren sus terminales, ya que cada fabricante crea su propia versión de Android y cuando Google lanza una nueva versión las actualizaciones no se despliegan automáticamente en los dispositivos móviles. Esta situación pretende ser cambiada ya que las actualizaciones serán aplicadas al menos durante los 18 meses siguientes a partir desde que los fabricantes saquen a la venta sus dispositivos móviles en los diferentes mercados [23]. Por otro lado, Google ha unificado las versiones del sistema operativo que es utilizado en tabletas y móviles a partir de la versión 4.0.

Tener la última versión de Android en un determinado dispositivo móvil no es posible, a pesar de tener el código fuente del sistema operativo, ya que el código para soportar el hardware de cada fabricante normalmente no es público y para hacer que funcione en un dispositivo móvil faltaría una parte del firmware [24].

Las nuevas versiones de Android que van saliendo cada vez requieren de más recursos, por eso no se puede instalar la última versión en los dispositivos antiguos, debido a la insuficiencia de memoria RAM, velocidad de procesador, etc.

1.3.5 ANDROID STUDIO

En sus inicios, Eclipse (denominado ADT – Android Development Tools) fue el entorno de desarrollo software (IDE - Integrated development environment), Google lo ha dejado de mantener y el entorno Android Studio se ha convertido en la herramienta oficial para crear aplicaciones para Android y se basa en IntelliJ IDEA. Además del editor de códigos y las herramientas para desarrolladores de IntelliJ, Android Studio, la Tabla 1.2 muestra funciones que aumentan su productividad durante la compilación de las aplicaciones [25].

Tabla 1.2 Funcionalidades Adicionales de Android Studio

Funcionalidades adicionales de Android Studio	
1	Un sistema de compilación basado en Gradle flexible
2	Un emulador rápido con varias funciones
3	Un entorno unificado en el que puedes realizar desarrollos para todos los dispositivos Android
4	Instant Run para aplicar cambios mientras tu app se ejecuta sin la necesidad de compilar un nuevo APK
5	Integración de plantillas de código y GitHub para ayudarte a compilar funciones comunes de las apps e importar ejemplos de código
6	Gran cantidad de herramientas y frameworks de prueba
7	Herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versión, etc.
8	Compatibilidad con C++ y NDK
9	Soporte incorporado para Google Cloud Platform, lo que facilita la integración de Google Cloud Messaging y App Engine.

1.3.6 WAMPSEVER

WampServer es un entorno de desarrollo web para Windows con el que se puede desarrollar aplicaciones web con Apache, PHP, y base de datos MySQL Database. El mismo que incluye PHPMyAdmin y SQLiteManager para manejar las bases de datos.

WampServer provee los cuatro elementos necesarios para un servicio web como son: Sistema Operativo (Windows), base de datos (MySQL), software para el servidor web (Apache) y el software de programación script web (PHP), debiendo su nombre a dichas herramientas [26].

Entre las funcionalidades que brinda son:

- Gestionar los servicios de Apache y MySQL
- Instalar y cambiar de Apache, MySQL y PHP
- Gestión de la configuración de sus servidores
- Acceder a sus registros
- Acceder a sus archivos de configuración

1.3.7 MySQL GESTOR DE BASE DE DATOS

MySQL es un sistema gestor de bases de datos, muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, siempre será una opción atractiva con usuarios que desarrollan tanto aplicaciones comerciales, como de entretenimiento gracias a su facilidad de uso, tiempo reducido de puesta en marcha, alto grado de estabilidad y un rápido desarrollo, estas características sumadas a su libre distribución en Internet bajo licencia GPL y que está disponible para múltiples plataformas como: Linux, UNIX y Windows, hacen que MySQL se posicione como uno de los principales gestores de base de datos [27].

La Tabla 1.3 se indica las características de MySQL.

Tabla 1.3 Características de MySQL

Características de MySQL	
1	Está desarrollado en C/C++.
2	Se ha optimizado para equipos de múltiples procesadores.
3	La API está disponible en diferentes lenguajes de programación C, C++, Eiffel, Java, Perl, PHP, Python, Ruby y TCL.
4	Se puede utilizar como cliente-servidor o incrustado en aplicaciones.
5	La conectividad contempla TCP/IP, sockets UNIX y sockets NT
6	Cuenta con un conjunto grande de tipos de datos.
7	Su velocidad de respuesta es muy destacable.
8	Soporta múltiples métodos de almacenamiento de las tablas, con prestaciones y rendimiento diferentes así se puede optimizar la base de datos a cada caso concreto.
9	Los mensajes de error también pueden estar en español y permite las palabras acentuadas o con la letra 'ñ'.
10	Se ha llegado a comprobar que puede manejar cincuenta millones de registros, sesenta mil tablas y cinco millones de columnas.
11	Es altamente confiable cuando se refiere a su estabilidad.
12	Su administración se basa en usuarios y privilegios.

Por otro lado, las limitaciones de MySQL se pueden explicar de mejor manera comprendiendo sus principios de diseño. Por ejemplo, no soporta procedimientos almacenados, no incluye disparadores, no incluye vistas, no incluye características de objetos como tipos de datos estructurados definidos por el usuario, herencia, etc [28].

1.3.8 AMAZON WEB SERVICE

Amazon Web Services (AWS) es una empresa proveedora de servicios de nube (cloud), que inició sus actividades en el 2006, y hoy es la empresa líder del mercado en ofertar estos servicios. Se destaca por su participación dominante en el mercado, diversas capacidades de productos, precios razonables y facilidad para la integración [29].

Entre los servicios de AWS se destaca el Amazon Elastic Compute Cloud (Amazon EC2), que es capacidad de computación con posibilidad de escalamiento almacenado en la nube, la cual está disponible para sus usuarios. Este servicio permite al usuario configurar su

propio servidor con las características que sean requeridas, ya sea de tipo Red Hat, Ubuntu, SUSE Linux, Windows, entre otros, permitiendo establecer características como memoria RAM, desempeño de la red, soporte de IPv6, número de CPU y capacidad de almacenamiento [30].

AWS permite realizar instancias de este servicio en nivel libre (gratuitamente), este servicio es conocido como “EC2 Free-Tier” el cual tiene recursos limitados y su duración máxima es de 12 meses gratuitos siempre y cuando no se exceda de los valores establecidos. El EC2 permite la implementación de servicios como infraestructura, teniendo la posibilidad de tomar el control del servidor por completo e implementar el software que se necesite [31]. Todas estas ventajas serán aprovechadas para el desarrollo del prototipo de aplicación móvil turística.

1.3.9 SERVICIO WEB

Se define al Servicio Web como un sistema software diseñado para soportar una interacción interoperable máquina a máquina sobre una red.

Los Servicios Web suelen ser APIs Web que pueden ser accedidas dentro de una red (principalmente Internet) ejecutándose en el dispositivo que los aloja, permitiendo la integración con otros componentes y cumpliendo una función determinada [32].

SOAP Y REST son las implementaciones comunes de los servicios Web.

En este caso SOAP se refiere al protocolo de comunicación que se usa entre cliente y servidor, intercambiando mensajes basados en XML (WSDL), mientras que REST intercambia mensajes que por lo general son en formato JSON. En los dos casos tanto el cliente como el servidor deberán conocer el formato y lenguaje de los mensajes para así poder encapsular y des encapsular las peticiones y respuestas [33].

1.3.9.1 REST

El protocolo REST (Representational State Transfer), se utiliza en una arquitectura cliente-servidor, en la cual un servicio es visto como un recurso el mismo que es identificado a través de una dirección URL, por la cual puede ser accedido o consumido [34].

La utilización de mensajes en formato simple, que se intercambian entre el cliente y el servidor es la forma de acceder a estos servicios web. Los cuatro métodos definidos por REST a partir de HTTP son: GET, PUT, DELETE y POST, siendo los más utilizados el GET y PUT. El método GET es usado para enviar una representación de un recurso o servicio al cliente y PUT es utilizado para la transferencia del estado de un cliente al recurso [35].

A través de REST para la comunicación e intercambio de información entre el cliente y servidor, se usará diferentes formatos y lenguajes como: XML, HTML, JSON. JSON ya que son sus mensajes muy sencillos, es el más difundido y utilizado por personas según las redes sociales y comunidades en Internet. Cada mensaje que es intercambiado debe contener la información para el funcionamiento del servicio, de esta forma para cada servicio el cliente y servidor conocerán el formato o protocolo interno de los mensajes [36].

1.3.9.2 JSON

JSON tiene un formato ligero el cual está basado en texto, su sintaxis se basa en un subconjunto de funcionalidades y palabras reservadas del lenguaje de script web JavaScript y tiene una definición de lenguaje completamente independiente. Generar y procesar un documento con el formato JSON es fácil debido a su sencillez. JSON es un excelente lenguaje para el intercambio de datos entre las aplicaciones cliente servidor o las aplicaciones que se ejecutan en diferentes escenarios, debido a que utiliza convenciones para el manejo de datos, que son comunes en los lenguajes: C, C++, C#, Java, Perl, Python, etc [37].

Un mensaje JSON está constituido por dos estructuras básicas: colección de parejas o estructuras nombre-valor y lista ordenada de valores.

Colección de parejas o estructuras nombre-valor son conocidas en otros lenguajes como: registros, diccionarios, tablas, hash, listas de claves o arreglos asociativos [38].

Lista ordenada de valores, se presen en la mayoría de los lenguajes como: arreglos, vectores, listas, etc.

Para conformar los mensajes se usan las anteriores estructuras, en los mensajes se define un protocolo interno representado como: palabra clave - valor o un conjunto de valores ordenados con significado, los mensajes son interpretados por el cliente y servidor, pero decodificados de acuerdo al protocolo interno que se ha definido, siendo necesario el uso de librerías que permitirán interpretar la sintaxis propia del lenguaje JSON [39].

1.3.10 METODOLOGÍA DE DESARROLLO DE SOFTWARE KANBAN

La palabra Kanban se define como la combinación de dos palabras japonesas, kan, que quiere decir 'visual', y ban, que quiere decir 'tarjeta', siendo denominada una metodología de producción u organización del trabajo que se basa en señales visuales ayudadas de tarjetas para gestionar el esfuerzo y dedicación del equipo de trabajo [40].

La idea de Kanban surge en el corazón de la metodología Lean, la misma que desarrollo Toyota, con el fin de mejorar su producción y estuvo basado en la técnica denominada justo a tiempo (JIT).

Los principios que se promueven en la metodología Kanban son [41]:

- **Calidad perfecta a la primera:** esto consiste en que todo lo que se hace se lo debe hacer bien, no rápido, debido a que costara más tiempo hacer algo rápido y después arreglarlo, definitivamente hacerlo bien desde el principio.
- **Minimización del despilfarro:** se debe hacer lo que es justo y necesario, concentrándose en lo que hace y sin entretenerse en otras tareas innecesarias.
- **Mejora continua:** según los objetivos que se pretende lograr o alcanzar, se debe ir mejorando los desarrollos continuamente.
- **Flexibilidad:** debido a faltantes o pendientes se tomará la decisión de que tareas realizar, aquí las tareas entrantes se las puede priorizar o condicionar según lo que se necesite.
- La relación con los proveedores debe ser construida y mantenida a largo plazo.

Las reglas en las que se basa la metodología Kanban [42]:

1. No se debe mandar producto defectuoso a los procesos subsecuentes.
2. Los procesos subsecuentes requerirán solo lo que es necesario.
3. Producir solamente la cantidad exacta requerida por el proceso subsecuente.
4. Balancear la producción.
5. Kanban es un medio para evitar especulaciones.
6. Estabilizar y racionalizar el proceso.

En el desarrollo de Kanban, lo común es ver tarjetas con las tareas que se ubique en un tablero o en una pared, las tarjetas son llamadas "Tareas Kanban".

Kanban es un sistema de gestión donde se produce exactamente aquella cantidad de trabajo que el sistema es capaz de asumir, lo cual principalmente permite asegurar una producción continua y sin sobrecargas en el equipo de trabajo [43].

Kanban permite la reducción del sistema de información, debido a que no se requiere la elaboración de un plan detallado para la subdivisión productiva de cada proceso.

Simplemente se debe informar a la línea de montaje final o de procesos terminados, lo que quiere decir que cada proceso retirara el trabajo realizado en el proceso anterior [44].

2. METODOLOGÍA

El presente capítulo se lo divide en dos apartados, el Diseño y la Implementación del prototipo. En la fase de diseño se definen los requerimientos que debe cumplir el prototipo, los mismos establecen el comportamiento del sistema. A continuación, se tiene la implementación, donde se muestra la instalación de las herramientas y la codificación de todos los módulos especificados en el diseño. Tanto para la etapa del diseño como para la implementación los componentes que integran el prototipo son separados en 3 Capas: Datos, Negocio y Presentación.

2.1 DISEÑO

En esta parte, se detalla el diseño a realizar para el prototipo de aplicación móvil turística. Como punto inicial según la metodología Kanban [45], se debe definir un tablero de actividades que sirve como guía para todo este desarrollo. Como siguiente paso, se definen los requerimientos tanto funcionales como no funcionales del sistema. Con estos requerimientos después de clasificarlos se realizan los respectivos diagramas de casos de uso. A continuación, se realiza el diagrama de clases, mismo que describe la estructura del prototipo por desarrollar; este diagrama describe las clases del sistema, sus atributos, operaciones y las relaciones entre los objetos de los módulos cliente, administración y principal. Además, se elabora el diagrama relacional de la base de datos, el cual permite implementar el módulo de almacenamiento, y finalmente se define las vistas del prototipo de aplicación móvil.

2.1.1 SITUACIÓN ACTUAL DE COMO SE MANEJA LA INFORMACIÓN TURÍSTICA DEL CANTÓN BAÑOS

La información turística del cantón Baños esta manejada de manera general por la Dirección de Turismo y Recreación de la municipalidad (DTR). La DTR tiene como función principal dar a conocer a los visitantes y público en general todos los atractivos turísticos que tiene el cantón, para ello se ha valido de trípticos y mapas los cuales son entregados a las personas que visitan los puntos de información turística.

Los trípticos y mapas contienen información no muy detallada. Para tener toda la información completa la DTR facilita varios trípticos o los números telefónicos de los representantes del sitio escogido para visitar, es decir que el turista sale del punto de información con un mapa y varios trípticos en la mano.

De esta forma se ha venido manejando la entrega de información turística, llenándole de trípticos y mapas al turista, el cual no puede planificar de forma adecuada su viaje ya que no tiene la información previamente.

Otro inconveniente que se tiene es que en Internet existe mucha información del cantón Baños gracias a los blogs de personas, páginas web de empresas, revistas, periódicos, etc, pero esta información muchas veces esta desactualizada y también está enfocada a una sola empresa limitando el abanico de opciones que puede tener el turista [46].

La DRT desea organizar de mejor manera la información turística, quiere que este en un solo lugar, que sea fácil de encontrar, este detallada, no enfocada a una sola empresa y que ayude a mostrar lugares que son pocos conocidos o que han sido remodelados. Para ello se plantea como solución el desarrollo del prototipo de aplicación móvil turística.

2.1.2 TABLERO DE ACTIVIDADES KANBAN

En la Tabla 2.1 se presenta el tablero Kanban, el mismo que proporciona la información sintetizada en tareas, permitiendo así una visibilidad total de cada una de las actividades que permiten realizar el prototipo de aplicación móvil.

El tablero consta de 3 columnas, la primera que muestra las actividades que se van a realizar con el título de “Tareas por Realizar”, la segunda muestra las actividades que se están realizando con el título “Tareas en Proceso” y la tercera contiene las actividades que fueron terminadas con el título “Tareas Finalizadas” [47]. Primero se selecciona la tarea que se llevará a cabo, se la mueve a la segunda columna “Tareas en Proceso”, cuando ya se haya terminado con la tarea se la mueve a la tercera columna de “Tareas Finalizadas”, siendo este el trato que se da a cada una de las tareas.

Tabla 2.1 Tablero Kanban

Tablero Kanban			
Fases	Tareas por Realizar	Tareas en Proceso	Tareas Finalizadas
Fase de Diseño		Entrevista con los funcionarios de la DTR para especificar la situación actual del manejo de la información turística del cantón Baños	
		Obtener los requerimientos funcionales	
		Obtener los requerimientos no funcionales	
		Elaborar los casos de uso de todos los módulos	
		Elaborar el diagrama de clases de los todos los módulos	
		Elaborar diagrama relacional de la base de datos	
		Definir las vistas del prototipo de aplicación móvil	
Fase de Implementación		Creación de una instancia EC2 en AWS	
		Instalación y creación de la base de datos	
		Se programará el módulo principal el mismo que implementará el servicio web	
		Se codificará el módulo cliente el cual consta del prototipo de aplicación móvil de acuerdo a los requerimientos obtenidos	
		Se codificará el módulo de administración	
Fase de Pruebas		Se realizará la validación de los requerimientos funcionales y no funcionales	
		Se realizará pruebas de funcionamiento del prototipo	

En este caso las tareas las fueron separadas por fases y por colores para así tener de mejor manera organizado el tablero Kanban. Las fases son: fase de diseño, fase de implementación y fase de pruebas cada una con sus respectivas tareas, que conforme se las vaya ejecutando irán cambiando de posición en el tablero.

2.1.3 ANÁLISIS DE REQUERIMIENTOS

Los requerimientos del prototipo son definidos en base a encuestas, las mismas que se encuentran en el ANEXO A. Estas encuestas fueron realizadas a los personeros de la dirección de turismo y recreación de la municipalidad de Baños los mismos que definieron todo lo que requerían en el prototipo de aplicación.

2.1.3.1 Requerimientos Funcionales (RF)

En relación a las encuestas y en concordancia con el plan aprobado, se definen los requerimientos funcionales (RF) los mismos que indican las funciones del sistema del prototipo de aplicación móvil turística [48]. En la Tabla 2.2 se describe los requerimientos funcionales que tiene el prototipo teniendo como nomenclatura RFT (Requerimiento Funcional Turismo).

Tabla 2.2 Requerimientos Funcionales.

Número RFT	Descripción
RFT01	Visualizar clasificación de sitio o actividad por categorías
RFT02	Visualizar una foto y descripción del sitio o actividad
RFT03	Visualizar en un mapa como poder llegar al sitio seleccionado
RFT04	Buscar sitios o actividades
RFT05	Ordenar los sitios según sus precios ascendente o descendente
RFT06	Ordenar los sitios siguiendo un orden alfabético A-Z o Z-A
RFT07	Calificar un sitio o actividad
RFT08	Almacenar Imágenes del sitio o actividad
RFT09	Almacenar la información concerniente administradores
RFT10	Almacenar la información de los sitios turísticos
RFT11	Autenticar un administrador
RFT12	Crear nueva categoría
RFT13	Editar categoría
RFT14	Eliminar categoría
RFT15	Crear nuevo sitio o actividad
RFT16	Editar sitio o actividad
RFT17	Eliminar sitio o actividad
RFT18	Crear nuevo usuario
RFT19	Editar usuario
RFT20	Eliminar usuario

2.1.3.2 Requerimientos No Funcionales (RNF)

Los RNF no describen funciones a realizar, si no son características del prototipo que establecen restricciones en el diseño e implementación del software [49], como, por ejemplo, la disponibilidad, presentación, escalabilidad, etc. Estos requerimientos proveen las características en el comportamiento del software completando así las funcionalidades. A continuación, en la Tabla 2.3 se tiene los RNFT (Requerimientos No Funcionales Turismo) identificados para el desarrollo del prototipo de aplicación móvil turística.

Tabla 2.3 Requerimientos No Funcionales

Etiqueta	Número RNFT	Descripción
Presentación	RNFT01	El prototipo de aplicación debe tener una interfaz que se amigable e intuitiva de manejar.
Disponibilidad	RNFT02	El Prototipo deberá estar disponible al menos el 99% del tiempo.
Escalabilidad	RNFT03	El prototipo debe ser construido en una plataforma la cual debe permitir el incremento de los recursos tales como: memoria RAM, Procesador, Almacenamiento.
Autenticación	RNFT04	El acceso al módulo de administración va a estar restringido, los usuarios administradores tendrán su usuario y contraseña para acceder.

2.1.4 MÓDULOS DEL PROTOTIPO

Los requerimientos funcionales son agrupados dependiendo a que modulo pertenecen, permitiendo así que cada módulo cumpla con su función específica y cumpliendo con todas las características que requiere el prototipo de aplicación móvil turística.

- **Módulo principal:** este módulo implementa un servicio REST, el mismo que se encarga de establecer la comunicación entre los dispositivos móviles y el servidor de datos. De acuerdo con la petición realizada se procede a presentar la información pertinente a los clientes. Este servicio web brinda la posibilidad de consultar los sitios turísticos ordenados alfabéticamente; ordenados ascendente y descendente de acuerdo con el precio del servicio del lugar; búsqueda exacta de sitios, es decir, por ejemplo, el sitio exacto “pailón del diablo”; clasificación de los lugares turísticos de acuerdo con su categoría, por ejemplo, “Deportes Extremos”; y consultas del detalle de la información general del sitio turísticos.
- **Módulo de almacenamiento:** este módulo implementa una base de datos para almacenar la información concerniente de los lugares turísticos por visitar en la

ciudad de Baños, además de la información que se necesite para llevar a cabo la implementación del prototipo final.

- **Módulo de administración:** permite ingresar, visualizar y modificar información turística dependiendo si es una actividad o un sitio.
- **Módulo cliente:** permite visualizar toda la información turística que se ha cargado acerca del cantón Baños, además, permite ver la ubicación de cada uno de los lugares por visitar y al seleccionar un sitio turístico en la interfaz del usuario se muestra la ruta en un mapa para llegar a dicho sitio.

En la Tabla 2.4 se muestra la clasificación de los requerimientos funcionales (RF) por módulo principal.

Tabla 2.4 Estimación de Ejecución de los RF del Módulo Principal.

Módulo Principal / Requerimientos Funcionales Turismo	Horas Estimadas
RFT04	8
RFT05	7
RFT06	6
RFT07	8
Total	29

En la Tabla 2.5 se muestra la clasificación de los requerimientos funcionales (RF) por módulo de almacenamiento.

Tabla 2.5 Estimación de Ejecución de los RF del Módulo de Almacenamiento.

Módulo de Almacenamiento/ Requerimientos Funcionales Turismo	Horas Estimadas
RFT08	7
RFT09	8
RFT10	6
Total	21

En la Tabla 2.6 se muestra la clasificación de los requerimientos funcionales (RF) por módulo de administración.

Tabla 2.6 Estimación de Ejecución de los RF del Módulo de Administración.

Módulo de Administración / Requerimientos Funcionales Turismo	Horas Estimadas
RFT11	8
RFT12	7
RFT13	8
RFT14	8
RFT15	7
RFT16	9
RFT17	8
RFT18	8
RFT19	8
RFT20	8
Total	79

En la Tabla 2.7 se muestra la clasificación de los requerimientos funcionales (RF) por módulo cliente.

Tabla 2.7 Estimación de ejecución de los RF del Módulo Cliente.

Módulo Cliente / Requerimientos Funcionales Turismo	Horas Estimadas
RFT01	8
RFT02	7
RFT03	10
Total	25

2.1.5 DISEÑO DE LA CAPA DE DATOS

En esta parte se realiza un diseño del diagrama entidad relación en donde están las tablas con sus diferentes campos (entidad) y la forma en que se relacionan unas con otras (relaciones) [50]. El mismo diseño se emplea para la creación de la base de datos.

2.1.5.1 Diagrama Entidad Relación

Se describe el diagrama Entidad-Relación que va a dar paso a la creación de la base de datos del prototipo de aplicación móvil turística, este diagrama, es un esquema de flujo que ilustra como las entidades, personas, objetos o conceptos se relacionan entre si dentro de un sistema. Es el diagrama se muestra las diferentes tablas creadas y la relación que existe entre las mismas.

La tabla sitio: permite guardar la información como nombre, descripción, horario, teléfonos, etc, de los sitios o actividades que pueden ser visitados o realizados en el cantón Baños.

La tabla tipo sitio (categoría): Está directamente relacionada con la tabla sitio ya que esta tabla clasifica a todos los sitios en categorías teniendo también una descripción de la categoría para que sea fácil encontrar lo que se busca.

La tabla fotografía: También está relacionada con la tabla sitio ya que como su nombre lo dice aquí se guardan las fotos o imágenes que van a pertenecer a cada una de los sitios o actividades ingresadas en la tabla correspondiente.

La tabla calificación: Permite dejar una calificación con estrellas a cada sitio o actividad que ha sido visitado, esta calificación al final es promediada y visualizada por los usuarios del prototipo de aplicación móvil turística, estando relacionada con la tabla sitio.

La tabla sitio tiene usuario (dueño o administrador del sitio): Tiene la información de quien es el responsable del sitio (el dueño o administrador del negocio) la misma que sirve para tener un contacto directo con los responsables de cada sitio y tener actualizada la información de los mismos. Se ha creado esta tabla debido a que uno de los problemas que tenía la DTR es la comunicación con los dueños o administradores de negocios, para la actualización de la información de los sitios.

La tabla usuario: Tiene la información de los usuarios administradores los cuales van a poder crear, eliminar, modificar, la información de un sitio, de un tipo sitio (categoría) de esta forma se administra toda la información del prototipo de aplicación móvil turística.

En la Figura 2.1 se indica, el diagrama con las diferentes tablas creadas y la relación que existe entre las mismas, diagrama Entidad-Relación de la capa de base de datos

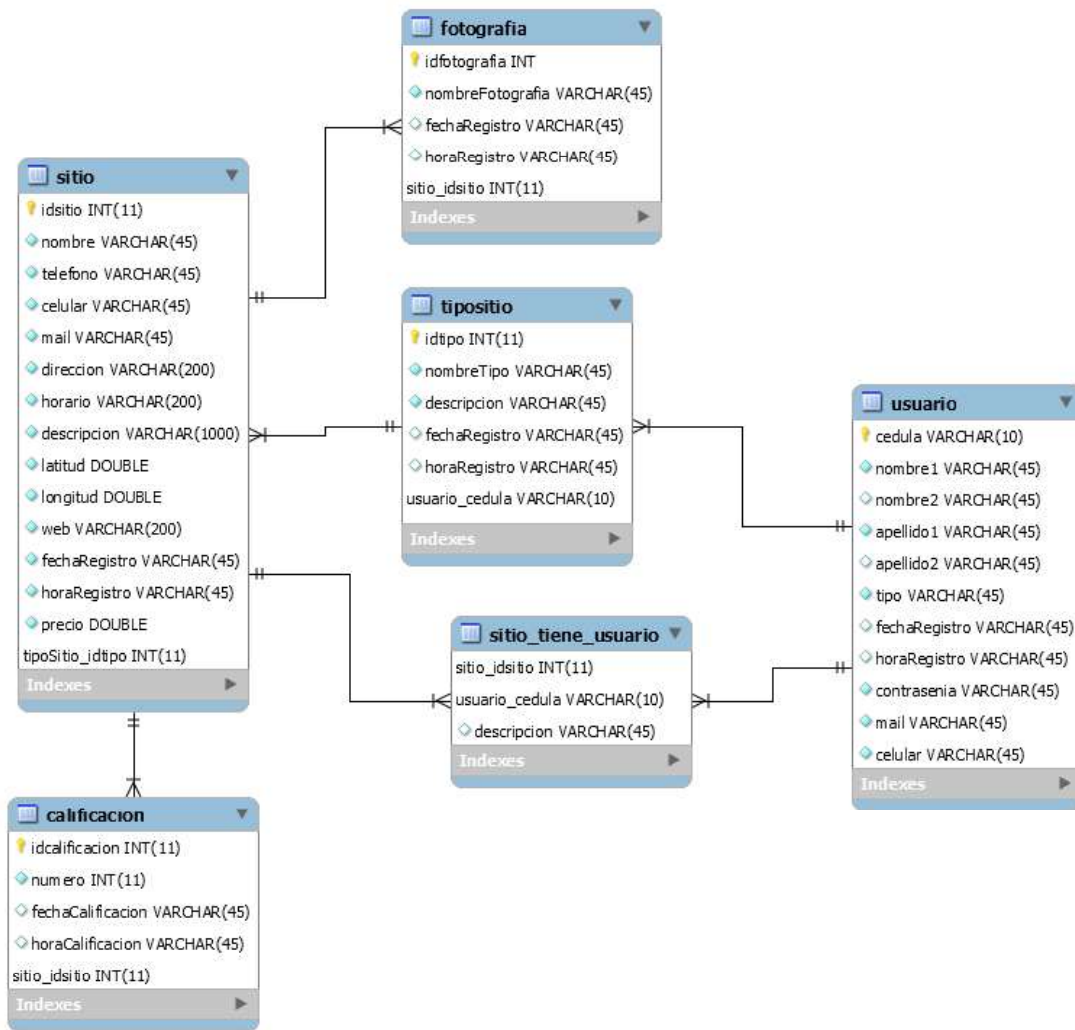


Figura 2.1 Diagrama Entidad-Relación de la Capa de Base de Datos.

2.1.6 DISEÑO DE LA CAPA DE NEGOCIO

Esta capa presenta la lógica de los procesos que permiten implementar el prototipo, entre los cuales está el diagrama de casos de uso, el diagrama de clases, el diseño del servicio web.

2.1.6.1 Diagrama de Casos de Uso (DCU)

Los diagramas de casos de uso son aquellos que permiten la descripción de los escenarios, incluye el escenario principal y los posibles escenarios alternativos [51]. Estos diagramas muestran las características del software que satisfacen las necesidades de los usuarios, sirven también para especificar la comunicación y el comportamiento del sistema y su interacción con los usuarios.

En el presente prototipo se definen los usuarios de tipo turista y tipo administrador.

- **Usuario Turista:** En el prototipo final este usuario corresponde a las personas que utilizaran la aplicación, siendo capaces de ver la información turística del cantón Baños en general. También, pueden hacer una búsqueda por el nombre del sitio, sino ordenar los sitios desde el más caro al más barato, en orden alfabético o viceversa. Dentro de la información se puede acceder a la web oficial del sitio escogido o a sus números de teléfono para hacer preguntas o reservas.
- **Usuario Administrador:** En el prototipo final este usuario corresponde a las personas que pueden administrar la aplicación, los mismos que tienen una identificación y contraseña para poder ingresar a las opciones de administrador. Entre las opciones de administrador se tiene: crear, editar y borrar las opciones, usuarios, sitios y categorías.

La Figura 2.2 muestra, a los usuarios turista y administrador que serán utilizados para crear los casos de uso.

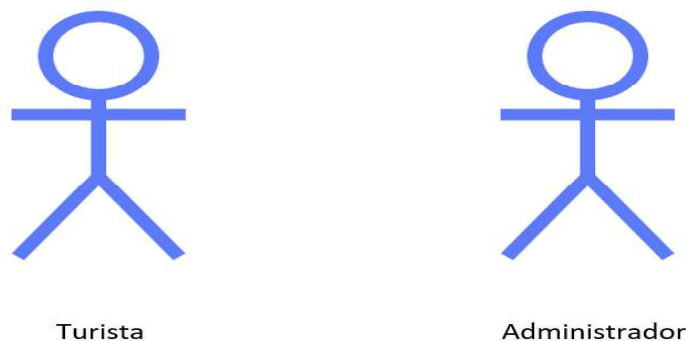


Figura 2.2 Usuarios Turista y Administrador.

Con el fin de ordenar adecuadamente los RF, se los procede agrupar y con estos se crea un subsistema, para cada uno de los actores de esta forma se sabe específicamente que es lo que va a realizar cada uno de ellos, facilitando así la elaboración de los casos de uso correspondientes.

La Tabla 2.8 muestra que existe 4 subsistemas el SBS01 que pertenece a las funcionalidades de módulo principal, el subsistema SBS02 son las funcionalidades del módulo almacenamiento, el subsistema SBS03 son las funcionalidades del módulo de Administración, el subsistema SBS04 son las funcionalidades del módulo cliente. Los casos de uso están enumerados con la nomenclatura CU00 lo que permite relacionar el caso de uso con el requerimiento funcional y con el subsistema creado.

Tabla 2.8 Descripción de los RF, Casos de Uso y Subsistema Asociado.

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS04	Turista	CU04	RFT01	Visualizar clasificación de sitio o actividad por categorías
SBS04	Turista	CU04	RFT02	Visualizar una foto y descripción del sitio o actividad
SBS04	Turista	CU04	RFT03	Visualizar en un mapa como poder llegar al sitio seleccionado
SBS01	Turista	CU01	RFT04	Buscar sitios o actividades
SBS01	Turista	CU01	RFT05	Ordenar los sitios según sus precios ascendente o descendente
SBS01	Turista	CU01	RFT06	Ordenar los sitios siguiendo un orden alfabético A-Z o Z-A
SBS01	Turista	CU01	RFT07	Calificar un sitio o actividad
SBS02	Administrador	CU02	RFT08	Almacenar Imágenes del sitio o actividad
SBS02	Administrador	CU02	RFT09	Almacenar la información concerniente administradores
SBS02	Administrador	CU02	RFT10	Almacenar la Información del sitio o actividad
SBS03	Administrador	CU03	RFT11	Autenticar un administrador
SBS03	Administrador	CU03	RFT12	Crear nueva categoría
SBS03	Administrador	CU03	RFT13	Editar categoría
SBS03	Administrador	CU03	RFT14	Eliminar categoría
SBS03	Administrador	CU03	RFT15	Crear nuevo sitio o actividad
SBS03	Administrador	CU03	RFT16	Editar sitio o actividad
SBS03	Administrador	CU03	RFT17	Eliminar sitio o actividad
SBS03	Administrador	CU03	RFT18	Crear nuevo usuario
SBS03	Administrador	CU03	RFT19	Editar usuario
SBS03	Administrador	CU03	RFT20	Eliminar usuario

El caso de uso CU01 asignado al usuario Turista y relacionado al subsistema módulo principal SBS01. La Figura 2.3 describe el caso de uso, que permite buscar sitio o actividad, ordenar los sitios de acuerdo al precio en forma descendente o ascendente, ordenar los sitios en forma alfabético A - Z o Z – A y calificar con estrellas el sitio o actividad.

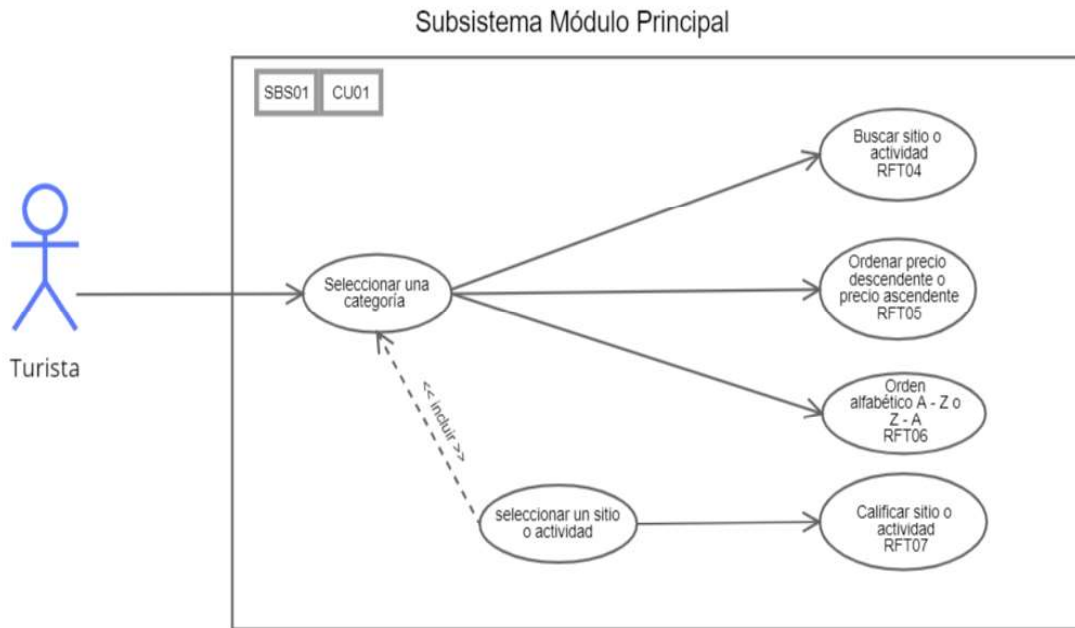


Figura 2.3 Caso de Uso del Subsistema Módulo Principal.

El caso de uso CU02 asignado al usuario Administrador y relacionado al subsistema módulo almacenamiento SBS02. La Figura 2.4 describe el caso de uso, que permite almacenar las imágenes del sitio o actividad, almacenar la información del sitio o actividad, almacenar información concerniente al administrador.

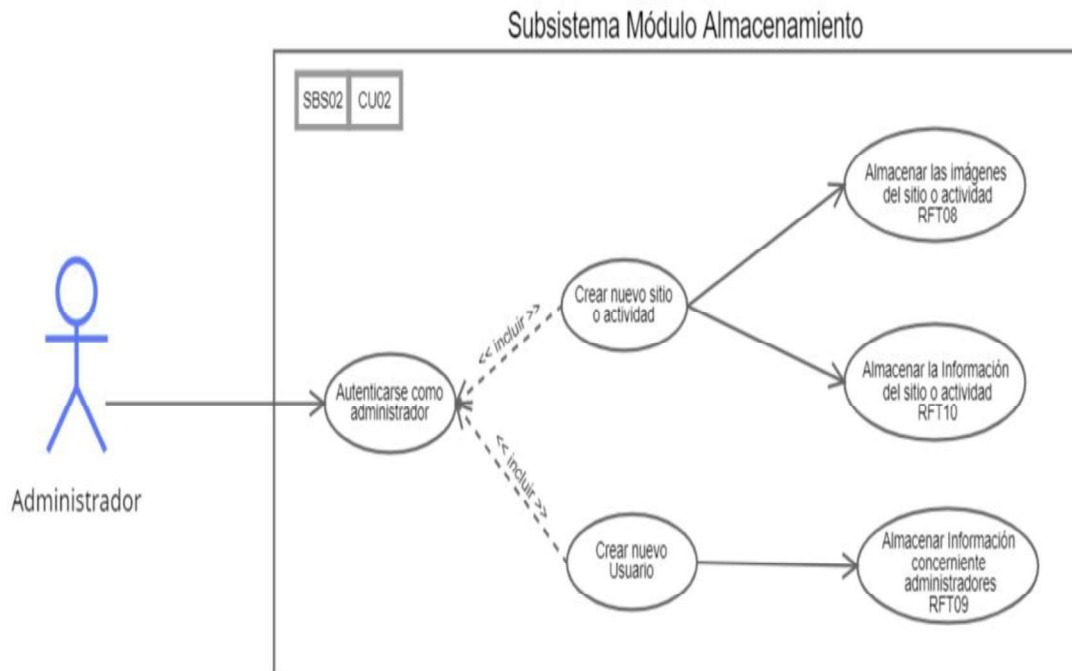


Figura 2.4 Caso de Uso del Subsistema Módulo Almacenamiento.

El caso de uso CU03 asignado al usuario administrador y relacionado al subsistema módulo de administración SBS03. La Figura 2.5 describe el caso de uso, que permite al administrador crear, editar, eliminar las opciones categoría, sitio o actividad y usuarios.

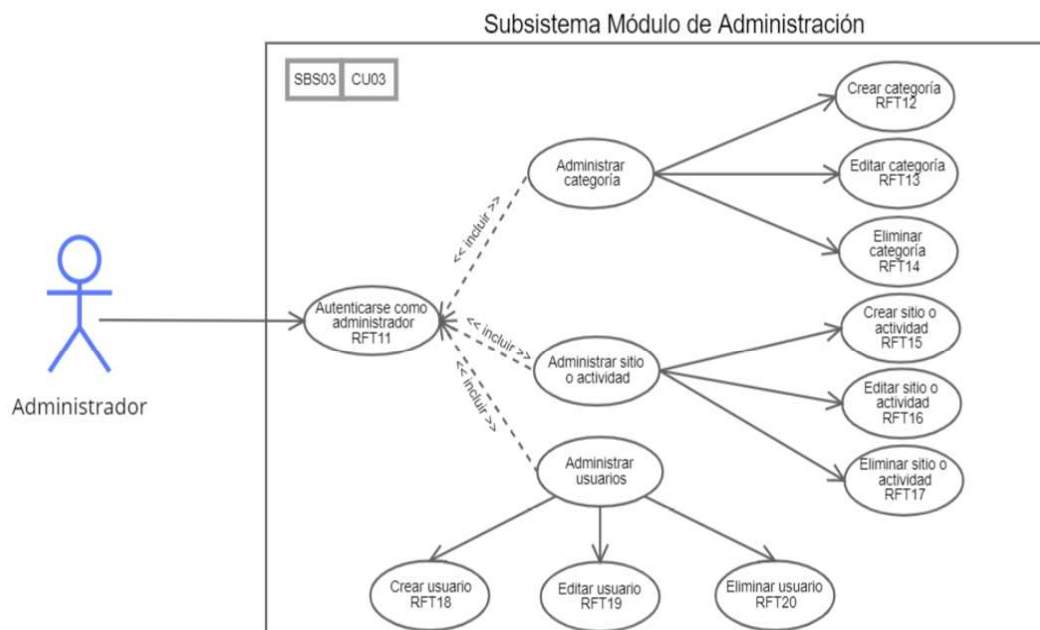


Figura 2.5 Caso de Uso del Subsistema Módulo Administración.

El caso de uso CU04 asignado al usuario turista y relacionado al subsistema módulo cliente SBS04. La Figura 2.6 describe el caso de uso, que permite al turista visualizar la foto y descripción del sitio o actividad, visualizar en un mapa como poder llegar al sitio seleccionado.

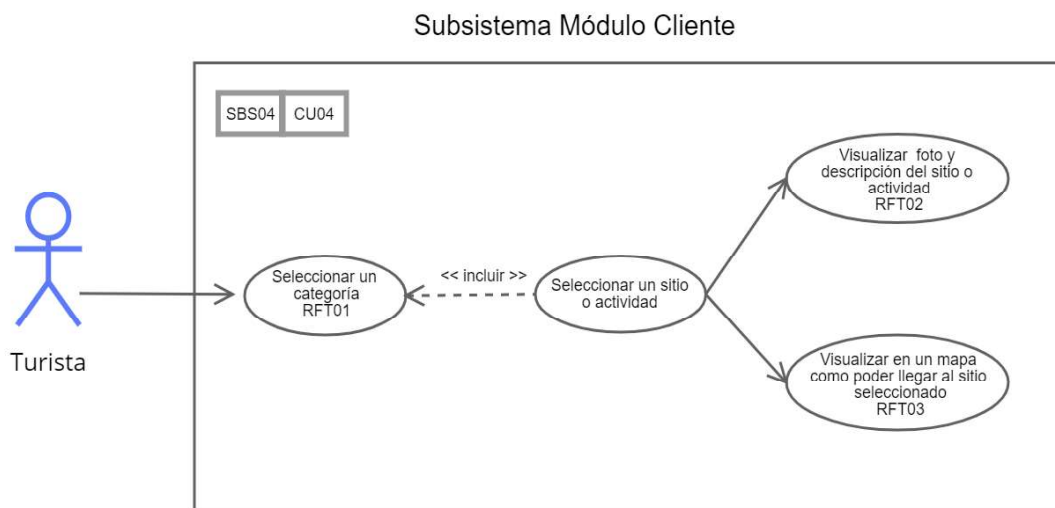


Figura 2.6 Caso de Uso Buscar Sitio o Actividad.

2.1.6.2 Diseño del Diagrama de Clases

Un diagrama de clases presenta la estructura del sistema programado mediante la utilización del lenguaje orientado a objetos de manera gráfica [52]. Las clases presentadas son las que el sistema usa para que los servicios permitan el intercambio de datos entre la capa de base de datos y la capa de presentación.

Los atributos de las clases son de tipo privado, es así que para acceder a sus atributos y modificarlos es necesario definir métodos públicos “get ()” y “set ()”, de esta manera se tiene que agregar encapsulamiento. Las clases “sitio”, “fotografía”, “calificación”, “sitioTieneUsuario”, tienen una cardinalidad de “uno” a “n”, lo que representa que una instancia de la clase “sitio” puede tener varias instancias de las clases “fotografía”, “calificación”, “sitioTieneUsuario”. En cambio, entre la clase “sitio” y “tipositio” existe una cardinalidad “n” a “uno”, lo que representa que la clase “sitio” solo tiene una instancia de la clase “tipositio”.

Por otro lado, se tiene entre las clases “tipositio” y “usuario”, una cardinalidad de “n” a “uno”, lo que representa que la clase “tipositio” solo tiene una instancia de la clase “usuario”. Finalmente se tiene las clases “sitioTieneUsuario” y “usuario” con la misma cardinalidad de “n” a “uno”, lo que representa que la clase “sitioTieneUsuario” solo tiene una instancia de la clase “usuario”.

En la Figura 2.7 se muestra el diagrama de clases, donde se observa las clases que definen el funcionamiento de prototipo de aplicación móvil turística.

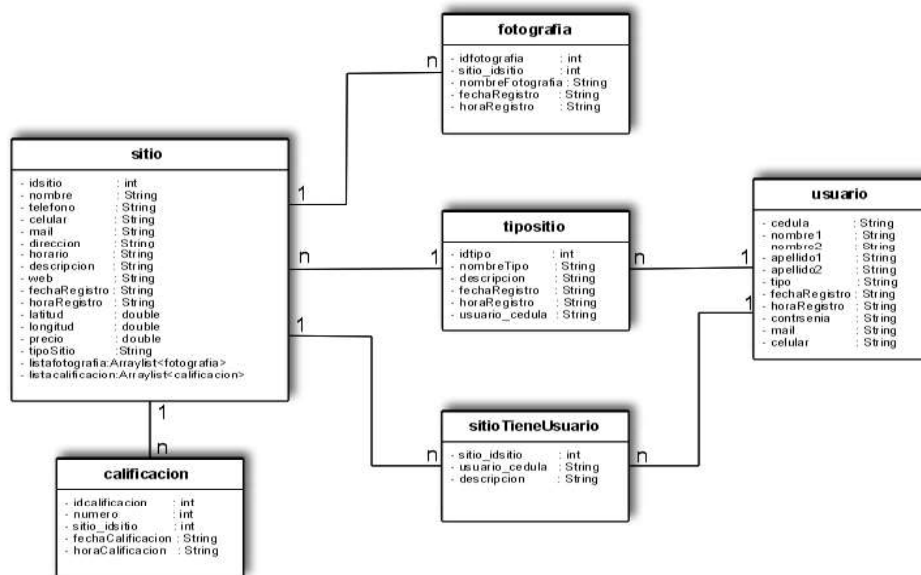


Figura 2.7 Diagrama de Clases del Prototipo de Aplicación.

2.1.6.3 Diseño de los Servicios Web

Para la programación de un servicio web se utiliza el lenguaje de programación PHP [53] con el objetivo de ofertar a los clientes Android varias funcionalidades. El servicio web sigue la lógica de la Figura 2.8.

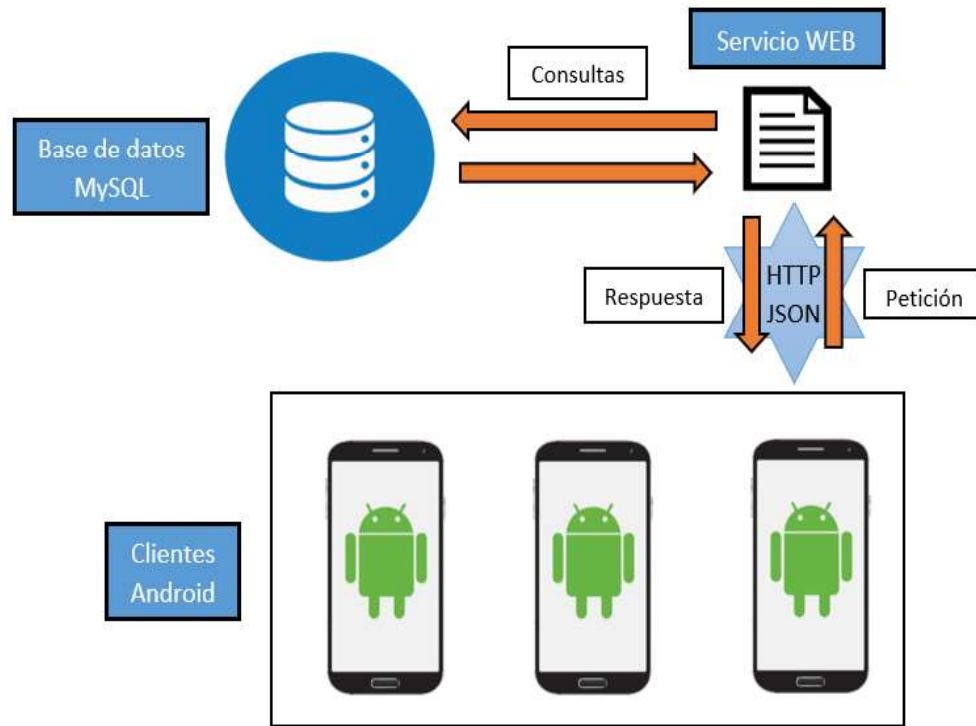


Figura 2.8 Arquitectura del Servicio Web.

La Figura 2.8 muestra que el usuario del prototipo de aplicación móvil turística realizada sobre el sistema operativo Android, se realiza las peticiones al servidor mediante los mensajes JSON [54]. Las peticiones son procesadas por el servicio web, que tiene todas las consultas codificadas en archivos PHP perteneciente a las funcionalidades del sistema y también la respectiva base de datos MySQL. El servidor devuelve las respuestas de igual manera en formato JSON que son decodificadas en el lado del cliente para mostrar la información solicitada. El servicio web contiene la lógica para ejecutar la acción pedida por el cliente Android.

2.1.7 DISEÑO DE LA CAPA DE PRESENTACIÓN

Esta capa muestra las interfaces que interactúan con el usuario final. Estas interfaces corresponden a la del prototipo de aplicación móvil turística, las mismas que permiten al usuario final la información turística del cantón Baños y al usuario administrador poder cargar, editar, borrar dicha información.

El diseño de la capa presentación es realizado en base a la herramienta online cacao [55] que permite realizar diferentes diagramas.

2.1.7.1 Diseño de las Vistas del prototipo de Aplicación Usuario Turista

El prototipo de aplicación cuenta con diferentes interfaces graficas. Los diferentes bosquejos de las vistas realizados para el usuario turista se muestran a continuación. Las vistas han sido diseñadas tomando en cuenta la opinión reflejada en las encuestas realizadas, por ejemplo, puntos como: que sea fácil de manejar, contenga imágenes, tamaño de los botones, distribución del menú. El diseño planteado también cumple con los requerimientos funcionales ya definidos anteriormente.

En la Figura 2.9 se muestra el prototipo de aplicación al momento de abrirla, donde se observa todas las categorías existentes, tienen el nombre de categoría con letras mayúsculas y una pequeña descripción haciendo referencia a la categoría.



Figura 2.9 Vista de Pantalla de Inicio de la Aplicación

En la Figura 2.10 se muestra la vista cuando se ingresa a una categoría en este caso a "¿Dónde Comer?", aquí se despliegan todos los lugares donde se puede ir a comer, tanto como restaurantes, cafeterías y heladerías. También la opción de búsqueda se muestra,

aquí se puede ingresar el nombre del sitio y dando un clic en la lupa se busca el sitio o se tendrá un anuncio vacío lo que quiere decir que el sitio no está en la base de datos.

Las opciones de ordenar los lugares de forma alfabética de AZ y de ZA como también de ordenarlos por el precio en este caso de menor a mayor precio o viceversa, todo esto solo dando clic sobre los botones asignados a esas tareas. Cabe recalcar que todas las otras categorías van a tener las mismas características descritas anteriormente.



Figura 2.10 Vista Sitios Dentro de Categoría

Figura 2.11 se muestra la información del sitio o actividad seleccionada. Para el ejemplo se tiene el restaurante "Swiss Bistro", en la parte superior aparece una o varias fotografías, luego se tiene las estrellas que sirven para calificar el sitio o la actividad. También se tiene la información del sitio o actividad seleccionada entre la que destaca: Nombre, Descripción, Horario, Precio, Teléfono, Mail entre otros. En la parte final se tiene el botón de "Ubicación Geográfica", el cual abre la siguiente pantalla. Los demás sitios o actividades de las otras categorías contarán con las mismas características que aquí se han descrito.



Figura 2.11 Vista que Muestra la Información del Sitio o Actividad.

La Figura 2.12 muestra la vista después de a ver dado el clic en ver ubicación geográfica, aquí se despliega un mapa indicando donde está ubicado el sitio o actividad selecciona. También se tiene la vista cuando se da clic en el botón “Ver Ruta”, de este modo muestra la ruta que se debe seguir para llegar al sitio o actividad seleccionado desde el punto que el turista se encuentra. Cabe recalcar que para que esto funcione el teléfono celular debe tener activado la función del GPS. Todos los sitios y actividades de las demás categorías tienen su ubicación y se también se muestra la ruta al mismo.

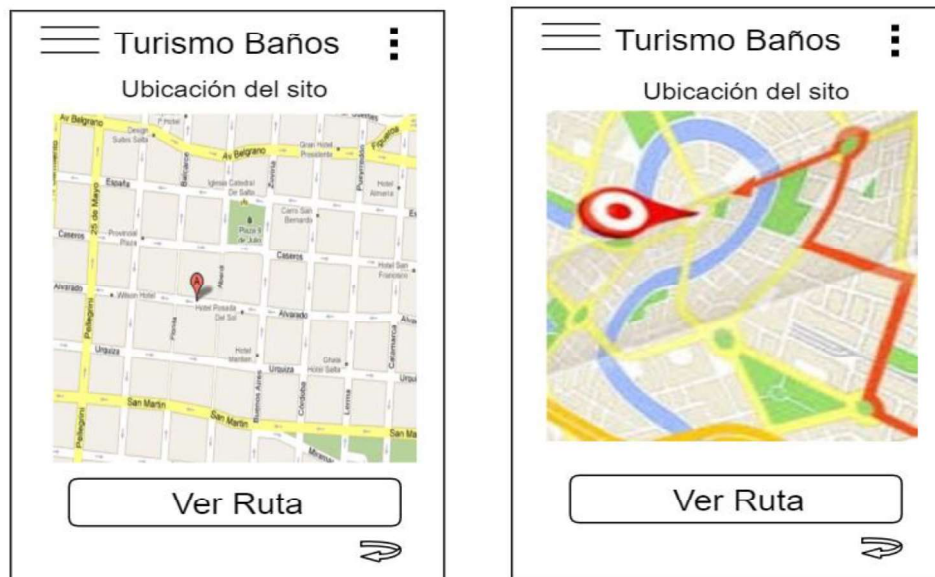


Figura 2.12 Vista que Muestra la Ubicación y la Ruta para Llegar al Sitio o Actividad.

La Figura 2.13 muestra que al aplicar clic en cualquier momento sobre la barra de tareas o barra de acceso rápido (tres barras de la parte superior izquierda), se puede regresar desde donde se esté a las categorías dando un clic sobre el botón con el mismo nombre. La opción “General” muestra todos los sitios y actividades que están en la base de datos del prototipo de aplicación. En “Historia” se muestra una reseña histórica del cantón Baños. La opción “Emergencia” muestra los números telefónicos de algunos organismos de socorro como: la policía, bomberos, hospital, cruz roja, entre otros, cabe mencionar que también se incluye la línea única de emergencia 911. Finalmente se tiene la información de contacto del desarrollador.



Figura 2.13 Vista de la Barra de Tareas o Barra de Acceso Rápido.

2.1.7.2 Diseño de las Vistas del prototipo de Aplicación Usuario Administrador

Los diferentes bosquejos de las vistas realizados para el usuario Administrador se muestran a continuación. Donde se puede ver todas las tareas que puede realizar el usuario administrador.

En la Figura 2.14 se muestra que el primer paso para entrar a la interfaz de administrador es dar un clic sobre los tres puntos de acceso rápido (tres puntos que se encuentran en la parte superior derecha del prototipo de aplicación), ahí se desplegara un botón con la palabra “Administrar”, se da clic, aparece una pantalla donde se tiene que escribir la identificación y la contraseña para poder autenticarse, se da clic en el botón “Ingresar” y se

puede ir al menú de administrador. También se tiene el botón que ayuda a recuperar la contraseña en caso de que haya sido olvidada.

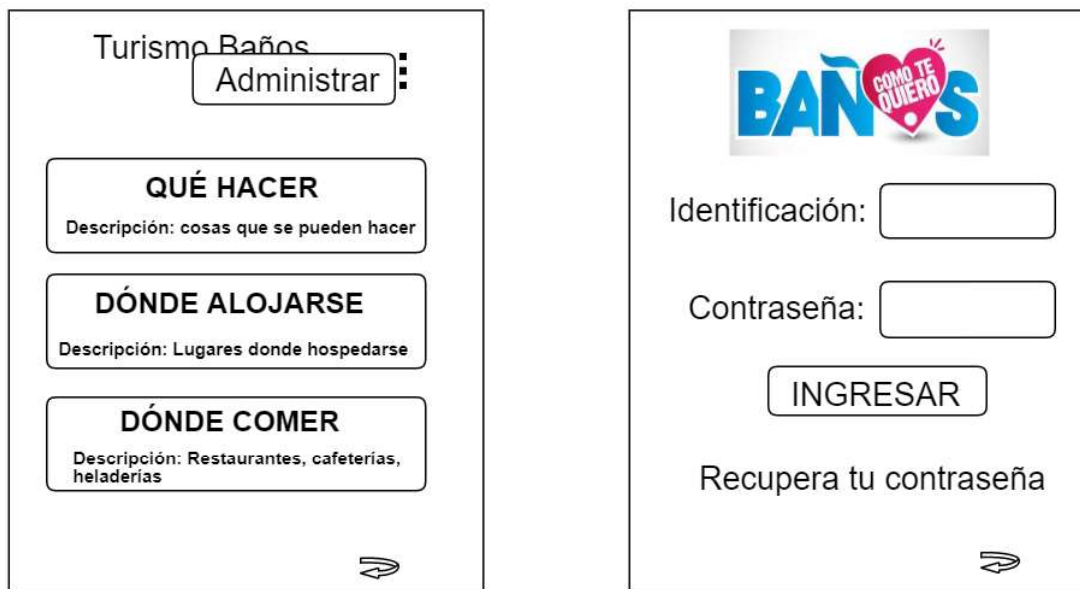


Figura 2.14 Vista de Autenticación del Usuario Administrador

La Figura 2.15 muestra la vista después de haber dado un clic sobre “Recupera tu contraseña” aquí simplemente se procede a ingresar la identificación personal del usuario administrador que ha olvidado su contraseña, se presiona en el botón “Recuperar” y su contraseña se envía al mail que tiene registrado el usuario.



Figura 2.15 Vista de Recuperar Contraseña

Luego de autenticarse se ingresa a la pantalla de la Figura 2.16 donde se tiene tres opciones de administrador, que se encuentran tres botones con las siguientes opciones “Usuarios”, “Sitios”, “Categorías”.

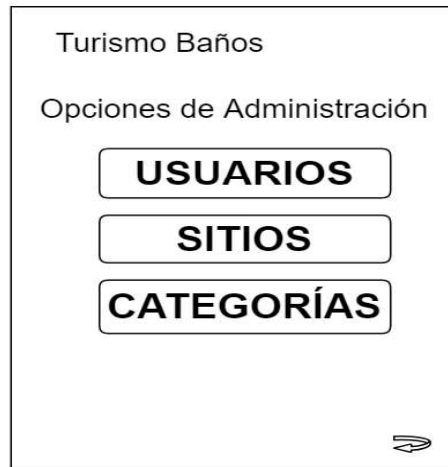


Figura 2.16 Vista de las opciones de administrador

En la Figura 2.17 se muestra que se ingresó a la opción de administración de usuarios en este caso se muestra una lista con los nombres de los usuarios que se tiene agregados como administradores. La opción de búsqueda ingresando el id del usuario requerido luego se da clic en el botón con la lupa para empezar la búsqueda. Se tiene el botón “nuevo” donde luego de darle clic se despliega la nueva pantalla. Las vistas de administración de sitios y categorías tienen las mismas características, la lupa de búsqueda, el botón nuevo y una lista de los sitios o categorías.



Figura 2.17 Vista de Administración de Usuarios.

En la Figura 2.18 se puede Observar la pantalla que se generó luego de haber dado clic en el botón nuevo, aquí aparece un formulario vacío el cual debe ser llenado con la información básica como: nombre, apellido, cedula, contraseña, mail, entre otros, que son requeridos. Al terminar de llenar el formulario se da clic en el botón guardar y se crea para este caso el nuevo usuario administrador. Para crear un nuevo sitio lo primero que se realiza es elegir en el mapa la ubicación correspondiente, también se ingresa las imágenes o fotografías del sitio, y tiene las mismas características anteriormente mencionadas.

The figure consists of two side-by-side screenshots of a web application interface. Both screenshots have a title 'Turismo Baños' at the top. The left screenshot is titled 'ADMINISTRACIÓN DE USUARIOS' and features a search bar labeled 'Id del usuario' with a magnifying glass icon, a 'NUEVO' button, and a list of user names: 'Luis Iván Meza', 'Stalin Morales', and 'Brianna Douglass'. The right screenshot is a form for creating a new user, with fields for 'Nombre', 'Apellido', 'Cédula', 'Tipo', 'Fecha Registro', 'Hora Registro', 'Contraseña', 'Celular', and 'Mail'. A large 'GUARDAR' button is at the bottom of the form. Both screenshots have a small circular arrow icon in the bottom right corner.

Figura 2.18 Vista del Formulario de un Nuevo Usuario

En la Figura 2.19 se muestra la pantalla cuando se quiere editar o borrar un usuario administrador existente. Para ello se da clic en un nombre de la lista de usuarios administradores existente, desplegándose así el formulario ya lleno, listo para ser editado o borrado. En este punto cuando se requiera borrar o editar la información de un sitio se la realiza de forma similar.

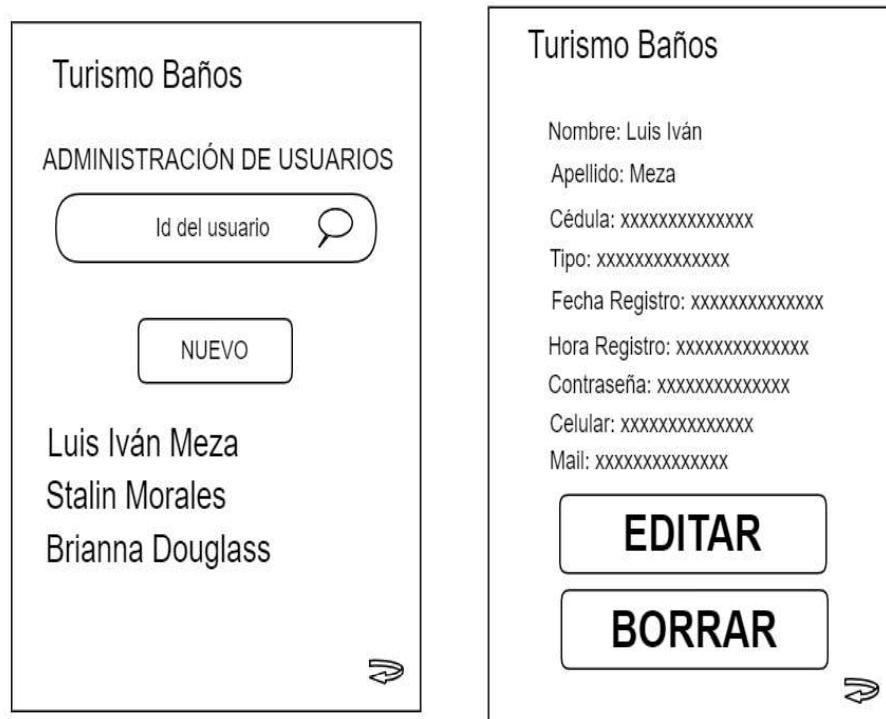


Figura 2.19 Vista de Pantalla para Editar o Borrar un Usuario Existente.

2.2 IMPLEMENTACIÓN

En esta parte se muestra el proceso a seguir para la implementación del prototipo de aplicación móvil turística. Teniendo en primera instancia la actualización del tablero Kanban con las tareas necesarias para realizar en esta etapa, luego se presenta la instalación y configuración de las herramientas de software necesarias, además la codificación del prototipo de aplicación siguiendo el diseño que se planteó en el Apartado 2.1. Al final se muestra un ejemplo de la codificación de las interfaces del prototipo de aplicación móvil a desarrollar.

2.2.1 TABLERO DE ACTIVIDADES KANBAN

En la Tabla 2.9 se muestra la actualización del tablero Kanban en este caso las actividades de la fase de implementación que estaban ubicadas en la columna de "Tareas por Realizar" pasaran a la columna de "Tareas en Proceso". Las actividades que pertenecen a la fase de diseño se encuentran todas terminadas y están en la columna de "Tareas Finalizadas", las actividades que no se mueven en esta fase son las de la fase de pruebas y permanecen en la columna de "Tareas por Realizar".

Tabla 2.9 Tablero Kanban de la Fase de Implementación.

Tablero Kanban			
Fases	Tareas por Realizar	Tareas en Proceso	Tareas Finalizadas
Fase de Diseño			Entrevista con los funcionarios de la DTR para especificar la situación actual del manejo de la información turística del cantón Baños
			Obtener los requerimientos funcionales
			Obtener los requerimientos no funcionales
			Elaborar los casos de uso de todos los módulos
			Elaborar el diagrama de clases de los todos los módulos
			Elaborar diagrama relacional de la base de datos
			Definir las vistas del prototipo de aplicación móvil
Fase de Implementación		Creación de una instancia EC2 en AWS	
		Instalación y creación de la base de datos	
		Se programará el módulo principal el mismo que implementará el servicio web	
	Se codificará el módulo cliente el cual consta del prototipo de aplicación móvil de acuerdo a los requerimientos obtenidos		
	Se codificará el módulo de administración		
Fase de Pruebas	Se realizará la validación de los requerimientos funcionales y no funcionales		
	Se realizará pruebas de funcionamiento del prototipo		

2.2.2 INSTALACIÓN DE LAS HERRAMIENTAS

En este punto se presenta la instalación de todas las herramientas las cuales permiten desarrollar el prototipo de aplicación móvil, mostrando la instalación y su configuración.

En la Figura 2.20 se muestra el esquema del servidor el mismo que es un servidor Microsoft Windows Server en AWS y las herramientas necesarias que se deben instalar. WampServer es utilizado como ya se indicó en el Capítulo 1 en el Apartado 1.3.8 ya que este dispone de algunos paquetes importantes entre ellos se tiene: el sistema de gestión de base de datos MySQL y phpMyAdmin [56] que almacena la información turística y el servicio web que se utiliza para las funcionalidades de la aplicación.



Figura 2.20 Arquitectura del Servidor

2.2.2.1 Creación de una instancia de Amazon EC2 en AWS

El primer paso para la creación de instancia de Amazon EC2 se debe crear una cuenta AWS [57] incluye 12 meses de acceso a capa gratuita, en esta parte se ingresa un correo electrónico, contraseña y nombre de la cuenta de AWS como se muestra en la Figura 2.21.

aws

Crear una cuenta de AWS

Las cuentas de AWS incluyen 12 meses de acceso a capa gratuita

Incluye uso de Amazon EC2, Amazon S3 y Amazon DynamoDB
Visite aws.amazon.com/free para consultar las condiciones completas de la oferta

Dirección de correo electrónico

Contraseña

Confirmar contraseña

Nombre de la cuenta de AWS ⓘ

Continuar

Figura 2.21 Crear Cuenta de AWS

Después de haber creado la cuenta se procede a crear una instancia de la máquina virtual en la plataforma AWS. En la página de AWS se escoge la opción “Compute” y se selecciona EC2 [58]. Esta opción es la que más se acopla al sistema permitiendo controlar y administrar las funciones del servidor, el sistema operativo y el software de implementación. La Figura 2.22 muestra la selección del EC2.

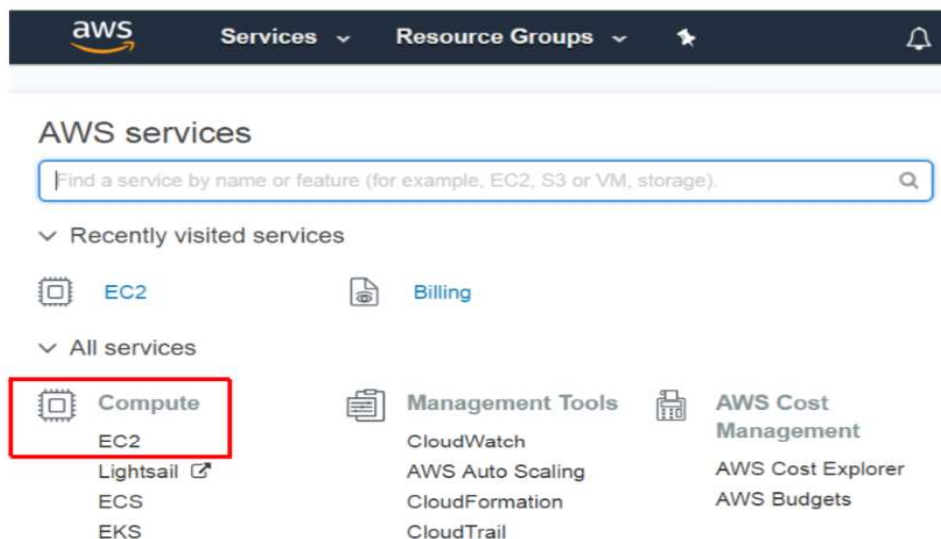


Figura 2.22 Selección de EC2

El siguiente punto, es seleccionar “ejecutar instancia” dentro de la consola de administración como se muestra en la Figura 2.23.

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.



Note: Your instances will launch in the US East (N. Virginia) region

Figura 2.23 Crear Instancia AWS

En este punto con AWS, se procede a detallar el software y las especificaciones de la instancia que se desea utilizar. Existen varias opciones para elegir como una imagen de MV de Amazon. Para el desarrollo de este Proyecto de Titulación se utiliza Microsoft Windows Server 2012 R2 Base, el cual se puede mostrar en la Figura 2.24.

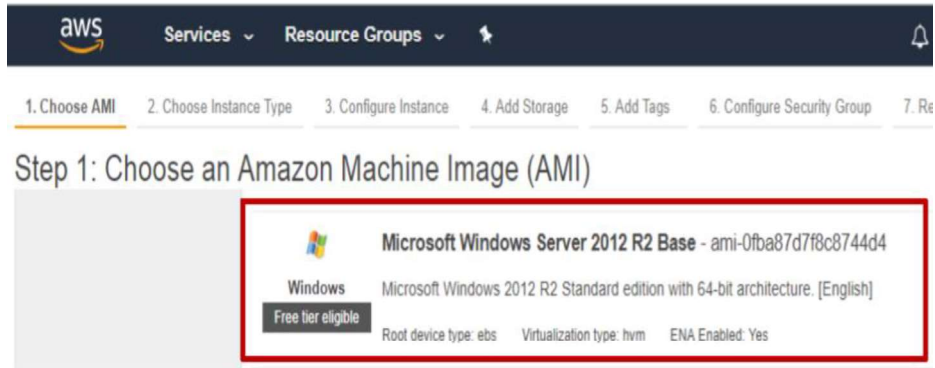


Figura 2.24 Selección de Imagen de AWS

Luego se mira los tipos de instancias que dispone AWS donde, cada tipo dispone de diferentes configuraciones en este caso como: CPU, memoria, almacenamiento y capacidad de red. Así se muestra en la Figura 2.25

Step 2: Choose an Instance Type

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes

Figura 2.25 Características de la Instancia Escogida

En la Figura 2.26 se muestra los detalles de la máquina virtual, con el sistema operativo escogido (tipo de instancia), el tamaño de almacenamiento y tamaño de memoria.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details [Edit AMI](#)

Microsoft Windows Server 2012 R2 Base - ami-2a9a1655
Free tier eligible Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. [English]
 Root Device Type: ebs Virtualization type: hvm

Instance Type [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Storage [Edit storage](#)

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-004c50f90619d0efe	30	gp2	100 / 3000	N/A	Yes	Not Encrypted

Figura 2.26 Características del Tipo de Servidor Escogido

Una vez comprobados los parámetros principales, se procede con la conexión al servidor a través del botón "Conectar" como se muestra en la Figura 2.27

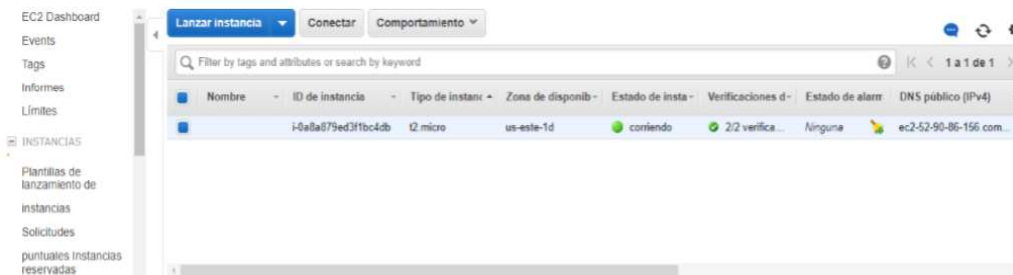


Figura 2.27 Interfaz para Conectar al Servidor Remoto en la Nube AWS.

Finalmente, se muestra en la Figura 2.28 la interfaz del servidor que esta alojada en AWS. Aquí se puede ver en la parte superior derecha el detalle de las respectivas IPs a las cuales se redirecciona el prototipo de aplicación móvil turística para así que pueda acceder a los servicios del proveedor.

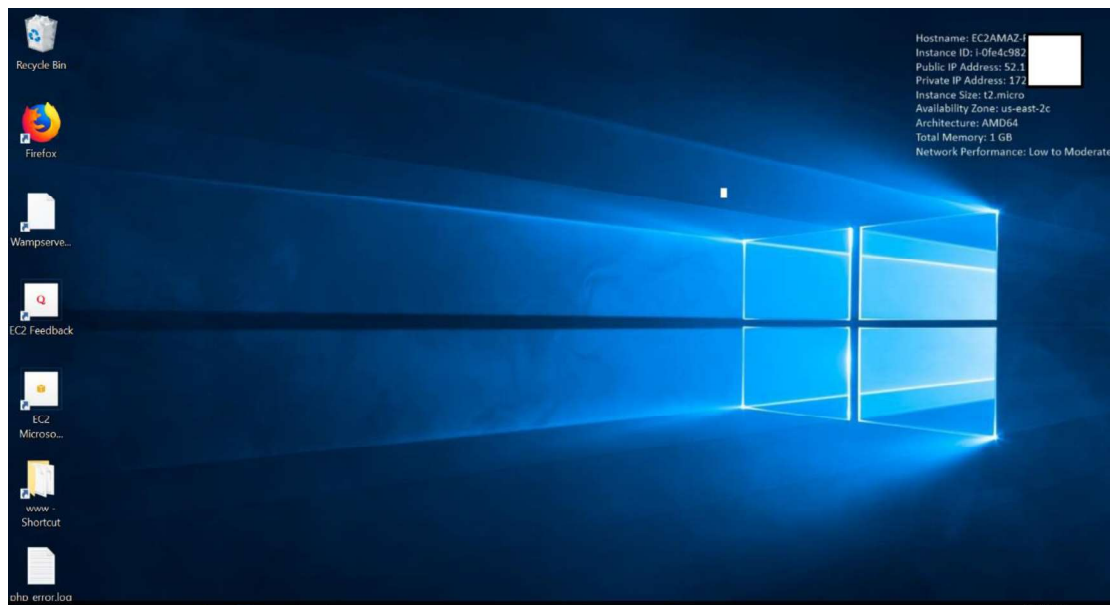


Figura 2.28 Interfaz del Servidor en AWS

2.2.2.2 WampServer

Dentro de la máquina virtual instalada y configurada, se procede a la instalación del software WampServer. El proceso de instalación de esta herramienta se describe, teniendo como primer paso ingresar al sitio oficial de WampServer [59] y se procede a la descarga dependiendo del sistema operativo del servidor. En la Figura 2.29 se muestra que se descarga el WampServer de 64 bits.

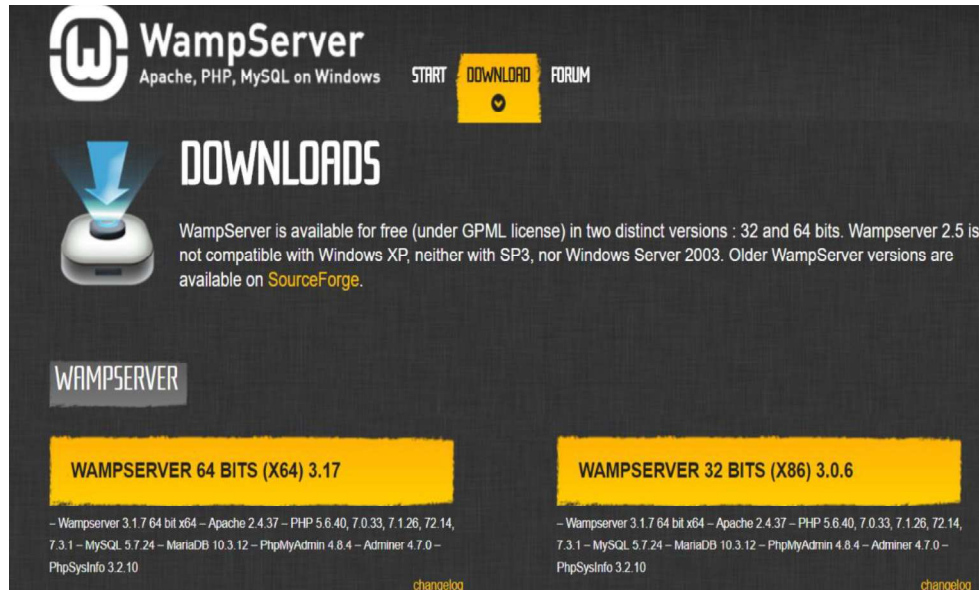


Figura 2.29 Descarga del WampServer 64 bits

Cuando se termina con la descarga del programa se procede a ejecutarlo para instalarlo en el servidor. En la Figura 2.30 se muestra el programa instalado y que están activados todos los servicios pertenecientes al WampServer.

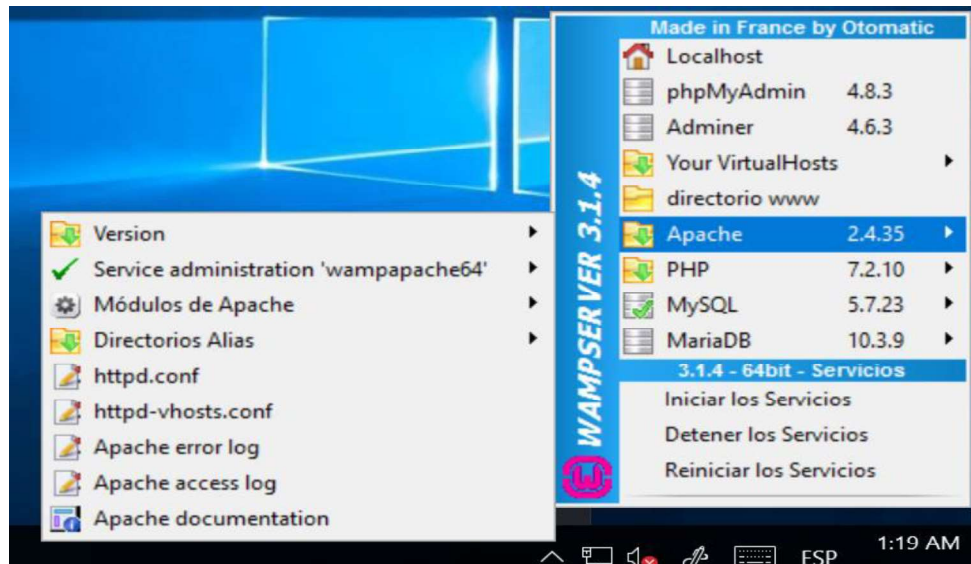


Figura 2.30 WampServer instalado y todos los servicios activos.

2.2.2.3 Android Studio

Android Studio es instalada en la computadora de desarrollo. Android Studio es la herramienta principal para el desarrollo del prototipo de aplicación móvil turística. Como primer paso se descarga la herramienta de la página oficial, el Android Studio que se

descarga es el Android Studio 3.4 for Windows 64-bit [60]. En la Figura 2.31 se indica la descarga de Android Studio.

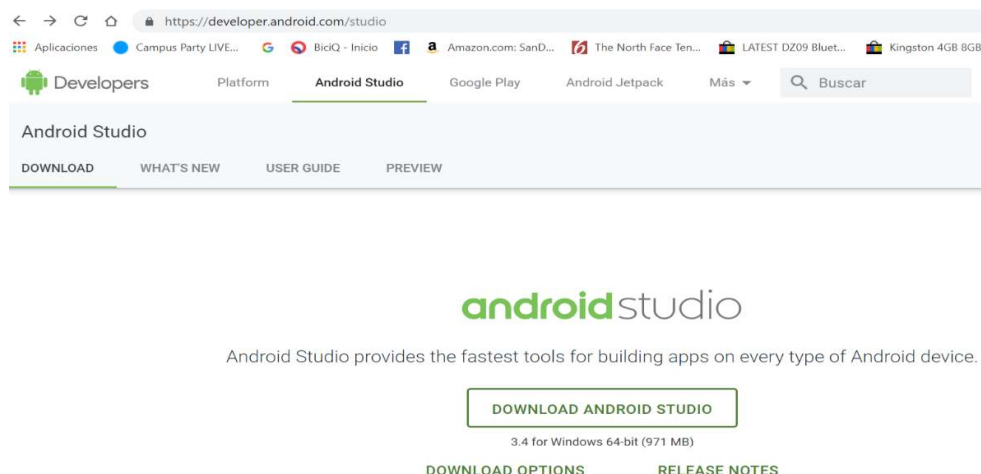


Figura 2.31 Descarga de Android studio for Windows 64-bit

La Figura 2.32 indica que una vez descargado el programa se procede a la instalación, se lo ejecuta como administrador y simplemente se siguen las instrucciones presentadas dándole solo clic en siguiente hasta que quede totalmente instalado.

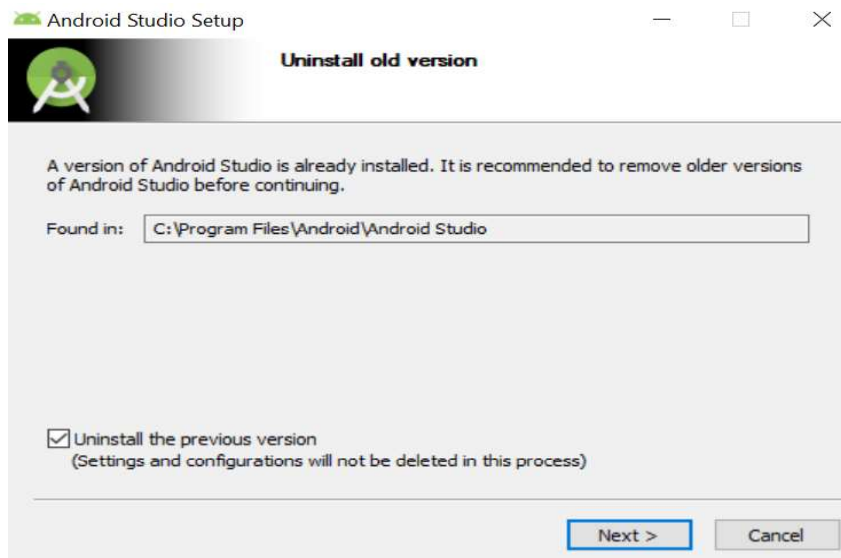


Figura 2.32 Instalación de Android Studio

En la figura se puede ver que se ha instalado correctamente Android Studio y se puede empezar a desarrollar el prototipo de aplicación móvil de acuerdo a los requerimientos establecidos. En la Figura 2.33 se observa la ventana abierta para crear un nuevo proyecto de Android.

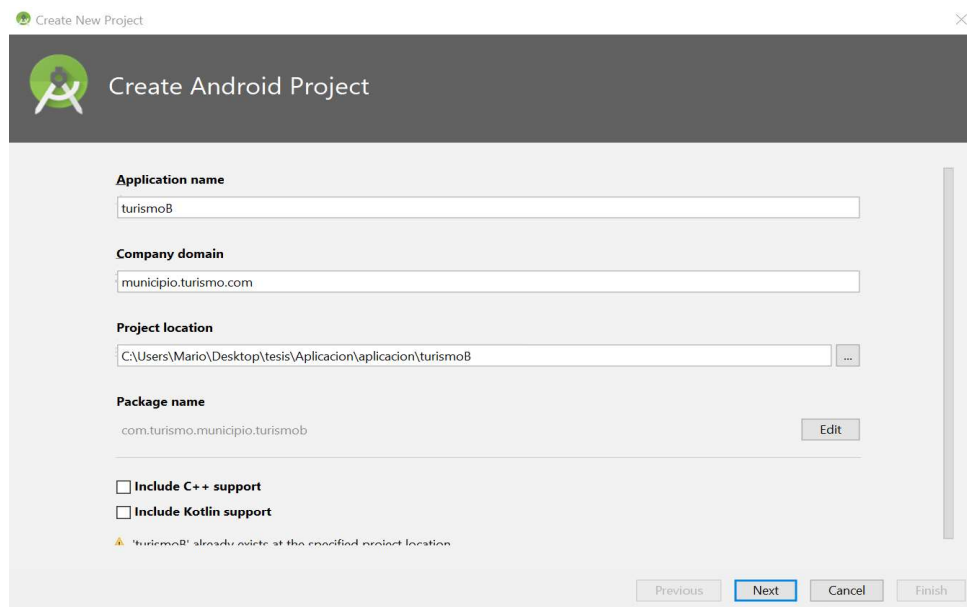


Figura 2.33 Creación de Nuevo Proyecto de Android

2.2.3 CODIFICACIÓN DE LA CAPA DE BASE DE DATOS

En esta parte se describe la implementación de la capa de Base de Datos, la misma que esta compuesta por diferentes tablas como se ha descrito en la etapa de diseño en el Apartado 2.1.5.

También se debe considerar que WampServer proporciona la base de datos MySQL y su administrador phpMyAdmin, teniendo la ventaja de no instalarlas por separado. De esta forma la creación de la base de datos con sus respectivas tablas de acuerdo a lo establecido en la etapa de diseño se lo realizara con este software.

La codificación de la base de datos se muestra en el Código 2.1, siendo este un ejemplo de código de una tabla requerida siguiendo el modelo relacional. En este caso el código permite la creación de la tabla sitio la misma que contiene los siguientes campos.

- **Idsitio:** el mismo que es único para cada sitio y se genera automáticamente cada vez que un sitio ha sido creado.
- **nombre:** nombre del sitio que va a ser cargado en el prototipo de aplicación.
- **teléfono:** teléfono del sitio que va a ser cargado en el prototipo de aplicación.
- **celular:** celular del sitio que va a ser cargado en el prototipo de aplicación.
- **mail:** mail del sitio que va a ser cargado en el prototipo de aplicación.

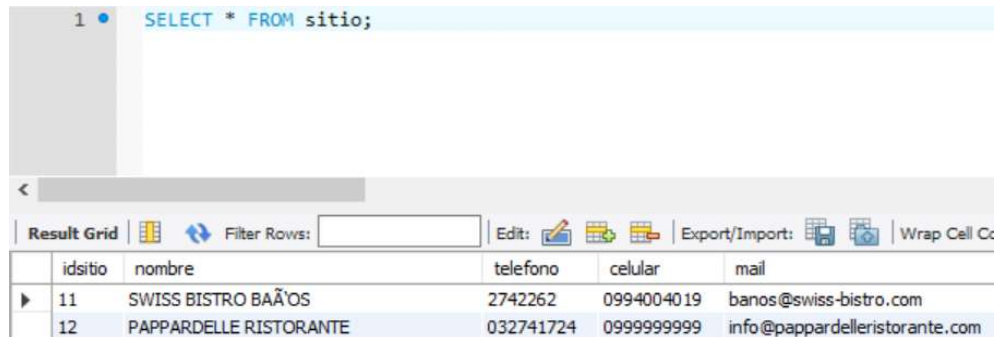
- ***dirección:*** dirección del sitio que va a ser cargado en el prototipo de aplicación.
- ***horario:*** horario del sitio que va a ser cargado en el prototipo de aplicación.
- ***descripción:*** texto que describa el sitio que va a ser cargado en el prototipo de aplicación.
- ***latitud:*** ayuda para la ubicación en el mapa del sitio que va a ser cargado en el prototipo de aplicación.
- ***longitud:*** ayuda para la ubicación en el mapa del sitio que va a ser cargado en el prototipo de aplicación.
- ***web:*** sitio web del sitio que va a ser cargado en el prototipo de aplicación.
- ***fechaRegistro:*** ayuda para saber la fecha en que se registró el sitio y a saber cuándo caducara su suscripción en el prototipo de aplicación.
- ***horaRegistro:*** ayuda para saber la hora que fue registrado el sitio.
- ***precio:*** costos del sitio que va a ser cargado en el prototipo de aplicación.

En el Código 2.1 se indica, que con la primera línea de código se va a crear la tabla con el nombre sitio en la base de datos, de la línea dos a la línea quince se definen los campos que va a contener la tabla sitio. Los campos tienen una propiedad NOT NULL, lo que quiere decir que no va a aceptar valores nulos en otras palabras todos los campos deben estar llenos. En la línea dieciséis se tiene el campo tipoSitio_idtipo que es un campo que representa el identificador de la categoría, este valor es igual auto generado. En la línea diecisiete, se establece como clave primaria para esta tabla, idsitio y tipoSitio_idtipo. El script completo de todas las tablas se presenta en el ANEXO B.

```
CREATE TABLE IF NOT EXISTS `dbturismo`.`sitio` (
  `idsitio` INT(11) NOT NULL AUTO_INCREMENT,
  `nombre` VARCHAR(45) NOT NULL,
  `telefono` VARCHAR(45) NOT NULL,
  `celular` VARCHAR(45) NOT NULL,
  `mail` VARCHAR(45) NOT NULL,
  `direccion` VARCHAR(200) NOT NULL,
  `horario` VARCHAR(200) NOT NULL,
  `descripcion` VARCHAR(1000) NOT NULL,
  `latitud` DOUBLE NOT NULL,
  `longitud` DOUBLE NOT NULL,
  `web` VARCHAR(200) NOT NULL,
  `fechaRegistro` VARCHAR(45) NOT NULL,
  `horaRegistro` VARCHAR(45) NOT NULL,
  `precio` DOUBLE NOT NULL,
  `tipoSitio_idtipo` INT(11) NOT NULL,
  PRIMARY KEY (`idsitio`, `tipoSitio_idtipo`),
```

Código 2.1 Codificación de la Tabla Sitio.

En el Código 2.2 se muestra la consulta hecha a la base de datos de los sitios ingresados en la tabla sitio.



```
1 • SELECT * FROM sitio;
```

	idsitio	nombre	telefono	celular	mail
▶	11	SWISS BISTRO BAÑOS	2742262	0994004019	banos@swiss-bistro.com
	12	PAPPARDELLE RISTORANTE	032741724	0999999999	info@pappardelleristorante.com

Código 2.2 Consulta a la Tabla Sitio.

2.2.4 CODIFICACIÓN DE LA CAPA DE NEGOCIO

La codificación de la capa de negocios se divide en 2 partes la codificación de los servicios web y la codificación de las consultas en Android. Se presenta en esta parte ejemplos del código realizado en la implementación de las consultas, todo el script está en el ANEXO C.

2.2.4.1 Servicio web

Para la codificación del servicio web se utiliza el programa Notepad ++ [Administrator] [59]. Dentro del mismo se tiene un documento denominado consultas.php el mismo que contiene la conexión con la base de datos y los métodos de las consultas realizadas.

En el Código 2.3 se muestra la conexión con la base de datos, en la línea nueve de código se muestran los datos de conexión, en este caso los datos son el localhost debido a que el MySQL se encuentra en el mismo servidor, el nombre de usuario, la contraseña y el nombre de la base de datos.

```
1  <?php
2  require('mail.php');
3
4  class consultas{
5
6      public static function conexionbd(){
7          $conexion = mysqli_connect("localhost", "root", "0rdRgEVwZE4krw41", "dbturismo");
8          if (mysqli_connect_errno()) {
9              $rsp = array("errorMySQL" => mysqli_connect_error());
10             echo json_encode($rsp);
11             exit();
12         }
13         return $conexion;
14     }
15 }
```

Código 2.3 Código de conexión con la Base de Datos.

Como se puede observar en la Código 2.4 se presenta la codificación del archivo index.php, el cual permite el procesamiento de los mensajes enviados por los clientes remotos. Los mensajes son de tipo JSON y cada uno contienen la identificación del tipo de consulta. Por ejemplo, “consultarTodosTipoSitio” este tipo de consulta permite obtener todas las categorías (tipos sitios) configuradas en el sistema, esta consulta es realizada por los turistas el momento que ingresan al prototipo de aplicación móvil. Otro ejemplo es “InsertarTipoSitio” empleada para agregar una nueva categoría (tipo sitio) al sistema. De este modo se clasifica los sitios y se logra organizar las consultas enviadas por los usuarios finales.

```

13     case "consultarTodosTipoSitio":
14         consultas::consultarTodosTipoSitio();
15         break;
16     case "consultarTipoSitiopPorId":
17         consultas::consultarTipoSitiopPorId($datos['idtipo']);
18         break;
19     case "consultarTipoSitiopPorNombre":
20         consultas::consultarTipoSitiopPorNombre($datos['nombreTipo']);
21         break;
22     case "insertarTipoSitio":
23         consultas::insertarTipoSitio($datos['datostipositio']);
24         break;
25     case "actualizarTipoSitioId":
26         consultas::actualizarTipoSitioId($datos['datostipositio']);
27         break;
28     case "eliminarTipoSitioId":
29         consultas::eliminarTipoSitioId($datos['idtipo']);
30         break;
31     case "consultarTodoSitioPorTipo":
32         consultas::consultarTodoSitioPorTipo($datos['tipoSitio_idtipo']);
33         break;
34     case "consultarTodosSitioAZPorTipo":
35         consultas::consultarTodosSitioAZPorTipo($datos['tipoSitio_idtipo']);
36         break;
37     case "consultarTodosSitioZAPorTipo":
38         consultas::consultarTodosSitioZAPorTipo($datos['tipoSitio_idtipo']);
39         break;

```

Código 2.4 Código del Archivo Index.php.

El Código 2.5 muestra el método para consultar todos los sitios desde la A hasta la Z. en donde se puede ver que lo primero que se realiza es el establecimiento de la conexión con la base de datos, se prepara la consulta a la base de datos.

Luego se reemplaza el símbolo de “?” por la variable que se transfiere, se ejecuta la sentencia, se tiene una respuesta, y se obtiene el json que se va a enviar al cliente, este es un array que luego se convierte en json.

También hay que tomar en cuenta que la sentencia o la consulta puede ser errónea, para lo cual se capturara el error y se enviara el error.

Otra opción es que la sentencia sea correcta pero que no exista registros, ahí se envía un mensaje informativo, el cual indica que no se hace referencia a ningún registro.

Si la sentencia es correcta no hay error y existe el registro toma los datos y genera un nuevo array con estos datos. En este caso se agrega el array sitio al array respuesta, se codifica en json el array respuesta, al final se libera la memoria y se cierra la conexión.

```
116     public static function consultarTodosSitioAZPorTipo($tipoSitio_idtipo){
117         $conexion=consultas::conexionbd();
118         $sentencia =$conexion->prepare("SELECT * FROM sitio where tipoSitio_idtipo = ? ORDER BY nombre ASC ");
119
120         if(!$sentencia) {
121             $rsp = array("errorMySQL" => $conexion->error, E_USER_ERROR);
122             echo json_encode($rsp);
123         }
124
125         $sentencia->bind_param('i',$tipoSitio_idtipo);
126         $sentencia->execute();
127         $rsp = $sentencia->get_result();
128         $jsonRsp = array();
129
130         if($sentencia->affected_rows == -1) {
131             $rsp = array("confirmacion" => $sentencia->error);
132             echo json_encode($rsp);
133         }else if($sentencia->affected_rows == 0){
134             $rsp = array("confirmacion" => "no se hace referencnia a ningun registro");
135             echo json_encode($rsp);
136         }else if($sentencia->affected_rows > 0){
137             while ($fila = mysqli_fetch_assoc($rsp)) {
138                 $sitio = array(
139                     "idsitio" => $fila["idsitio"],
140                     "nombre" => $fila["nombre"],
141                     "telefono" => $fila["telefono"],
142                     "celular" => $fila["celular"],
143                     "mail"=> $fila["mail"],
144                     "direccion"=> $fila["direccion"],
145                     "horario"=> $fila["horario"],
146                     "descripcion"=> $fila["descripcion"],
```

Código 2.5 Método para consultar todos los sitios desde A hasta la Z

El Código 2.6 muestra el método que permite la eliminación de un sitio mediante su identificación única (Id). El usuario administrador, a través de la aplicación móvil, selecciona eliminar un sitio en específico, y el dispositivo envía al servidor el mensaje tipo JSON con la identificación de mensaje “eliminarSitiold”. Este mensaje es procesado por el archivo index.php para invocar el método con el nombre “eliminarSitiold”, como se observa en el fragmento de código, posterior a esto el método ejecuta la sentencia que corresponda en la base de datos.

```

879 public static function eliminarSitiold($idsitio){
880     $conexion=consultas::conexionbd();
881     $sentencia = $conexion->prepare("DELETE FROM sitio WHERE idsitio = ?");
882     if(!$sentencia) {
883         $rsp = array("errorMySQL" => $conexion->error, E_USER_ERROR);
884         echo json_encode($rsp);
885     }
886     $sentencia->bind_param('i',$idsitio);
887     $sentencia->execute();
888     if($sentencia->affected_rows == -1) {
889         $rsp = array("confirmacion" => $sentencia->error);
890         echo json_encode($rsp);
891     }else if($sentencia->affected_rows == 0){
892         $rsp = array("confirmacion" => "no se hace referencia a ningun registro");
893         echo json_encode($rsp);
894     }else if($sentencia->affected_rows > 0){
895         $dir_subida = $_SERVER['DOCUMENT_ROOT']."/img/";
896         $carpeta = $dir_subida .$idsitio;
897         $files = glob($carpeta . '/*');
898         foreach($files as $file){
899             if(is_file($file)){
900                 unlink($file);
901             }
902         }
903         if(rmdir($carpeta)){
904             $rsp = array("confirmacion" => "sitio eliminado correctamente");
905             echo json_encode($rsp);
906         }
907     }
908 }

```

Código 2.6 Código eliminarSitiold.

El Código 2.7 muestra la petición de consulta de un usuario por identificación “consultarUsuariold”. Este método permite obtener un usuario en particular de la base de datos. Este método es utilizado por la aplicación móvil para identificar al usuario que ha iniciado sesión dentro de la aplicación móvil. Como se observa en el código presentado la identificación única del usuario es la cédula de ciudadanía. La sentencia a la base de datos es completada con la cédula y posterior a ello se busca en la base de datos el usuario de

interés. Finalmente, se responde al dispositivo con la información del usuario que fue encontrada. El código de las demás consultas está en el ANEXO D.

```
1318     public static function consultarUsuarioId($cedula){
1319         $conexion=consultas::conexionbd();
1320         $sentencia = $conexion->prepare("SELECT * FROM usuario WHERE cedula = ?");
1321         if(!$sentencia) {
1322             $rsp = array("errorMySQL" => $conexion->error, E_USER_ERROR);
1323             echo json_encode($rsp);
1324         }
1325         $sentencia->bind_param('s',$cedula);
1326         $sentencia->execute();
1327         $resultado = $sentencia->get_result();
1328         $jsonRsp = array();
1329         if($sentencia->affected_rows == -1) {
1330             $rsp = array("confirmacion" => $sentencia->error);
1331             echo json_encode($rsp);
1332         }else if($sentencia->affected_rows == 0){
1333             $rsp = array("confirmacion" => "no se hace referencia a ningun registro");
1334             echo json_encode($rsp);
1335         }else if($sentencia->affected_rows > 0){
1336             while ($fila = $resultado->fetch_assoc()) {
1337                 $usuario = array("cedula" => $fila["cedula"],
1338                                 "nombre1" => $fila["nombre1"],
1339                                 "nombre2" => $fila["nombre2"],
1340                                 "apellido1" => $fila["apellido1"],
1341                                 "apellido2" => $fila["apellido2"],
1342                                 "tipo"=> $fila["tipo"],
1343                                 "fechaRegistro"=> $fila["fechaRegistro"],
1344                                 "horaRegistro"=> $fila["horaRegistro"],
1345                                 "mail"=> $fila["mail"],
1346                                 "celular"=> $fila["celular"],
1347                                 "contrasenia"=> $fila["contrasenia"]);
```

Código 2.7 Código consultarUsuarioId.

2.2.4.2 Consultas en Android

En la aplicación móvil se han codificado las consultas que se llevan a cabo en el servidor de comunicaciones, por ejemplo, en el Código 2.8 se observa la implementación de la consulta “consultarTodosSitio”, esta consulta es empleada para listar todos los sitios registrados en la base de datos. Como se observa el tipo de consulta es “consultarTodosSitio”, la cual corresponde a un mensaje tipo JSON. Este mensaje es procesado por el archivo index.php es cual identifica el tipo de mensaje enviado e invoca al método el cual contiene la consulta a la base de datos.

```

33     public static String consultarTodosSitio(){
34         JSONObject jsonObject = new JSONObject();
35         try {
36             jsonObject.put("tipo","consultarTodosSitio");
37         } catch (JSONException e) {
38             e.printStackTrace();
39         }
40         return jsonObject.toString();
41     }

```

Código 2.8 Código consultarTodosSitios.

El Código 2.9 muestra la codificación del mensaje JSON que permite la creación de un nuevo sitio en la base de datos. Este mensaje contiene todos los campos que corresponde a un sitio como el nombre, el teléfono, celular, entre otros. Se recalca que el tipo de mensaje corresponde a un “insertarSitio”. Después de ensamblar el mensaje, éste es enviado al servidor de comunicaciones para su respectivo procesamiento. Todo el código realizado esta en el ANEXO D, correspondiente a la aplicación móvil

```

203     public static String insertarSitio(String nombre,String telefono,String celular, String mail,
204                                     String direccion, String horario,String descripcion,
205                                     double latitud,double longitud,String web,String fechaRegistro,
206                                     String horaRegistro,double precio,String fotografia,
207                                     int tipoSitio_idtipo){
208         JSONObject jsonObject = new JSONObject();
209         JSONObject jsonDatos = new JSONObject();
210         try {
211             jsonDatos.put("nombre",nombre);
212             jsonDatos.put("telefono",telefono);
213             jsonDatos.put("celular",celular);
214             jsonDatos.put("mail",mail);
215             jsonDatos.put("direccion",direccion);
216             jsonDatos.put("horario",horario);
217             jsonDatos.put("descripcion",descripcion);
218             jsonDatos.put("latitud",latitud);
219             jsonDatos.put("longitud",longitud);
220             jsonDatos.put("web",web);
221             jsonDatos.put("fechaRegistro", fechaRegistro);
222             jsonDatos.put("horaRegistro", horaRegistro);
223             jsonDatos.put("precio", precio);
224             jsonDatos.put("fotografia", fotografia);
225             jsonDatos.put("tipoSitio_idtipo", tipoSitio_idtipo);
226         } catch (Exception e){
227         }
228         try {
229             jsonObject.put("tipo","insertarSitio");
230             jsonObject.put("datosSitio",jsonDatos);
231
232         } catch (JSONException e) {
233             e.printStackTrace();

```

Código 2.9 Código insertarSitio.

2.2.5 CODIFICACIÓN DE LA CAPA DE PRESENTACIÓN

En esta parte se presenta la implementación de la capa de Presentación siguiendo el diseño establecido en el Capítulo 2 Apartado 2.1.7. Para la codificación de la interfaz gráfica y de las clases se muestra ejemplos del código ya que el script completo está en el ANEXO D. Además, en este anexo se encuentra el código completo desarrollado en Android Studio.

2.2.5.1 Codificación de la Clase Sitio

En esta parte la codificación de la clase sitio, se crea a la clase sitio todo esto en Android Studio, la clase sitio tiene todos los atributos mencionados anteriormente.

Se tiene todos los nombres de los atributos y el tipo de acceso de los mismos, que en este caso son privados. En el Código 2.10 se indica cómo se implementa el método get() para cada uno de los atributos, lo cual es utilizado para mostrar o modificar los valores de los atributos privados. A manera de ejemplo se presenta un extracto de todo el código empleado.

```
1 public class Sitio {
2     private int idSitio;
3     private String nombre;
4     private String telefono;
5     private String celular;
6     private String mail;
7     private String direccion;
8
9     public int getIdSitio() {
10         return idSitio;
11     }
12
13     public String getNombre() {
14         return nombre;
15     }
16
17     public String getTelefono() {
18         return telefono;
19     }
20
21     public String getCelular() {
22         return celular;
23     }
24
25     public String getMail() {
26         return mail;
27     }
28 }
```

Código 2.10 Código Clase Sitio

De la misma forma el Código 2.11 muestra el código que representa la creación de la clase TipoSitio sus atributos que son privados y los métodos get() y set() de cada uno de ellos.

```
1 public class TipoSitio {
2     private Integer idtipo;
3     private String nombreTipo;
4     private String descripcion;
5     private String fechaRegistro;
6     private String horaRegistro;
7
8     public Integer getIdtipo() {
9         return idtipo;
10    }
11
12    public void setIdtipo(Integer idtipo) {
13        this.idtipo = idtipo;
14    }
15
16    public String getNombreTipo() {
17        return nombreTipo;
18    }
19
20    public void setNombreTipo(String nombreTipo) {
21        this.nombreTipo = nombreTipo;
22    }
23
24    public String getDescripcion() {
25        return descripcion;
26    }
27
28    public void setDescripcion(String descripcion) {
29        this.descripcion = descripcion;
30    }
}
```

Código 2.11 Código de Clase TipoSito

2.2.5.2 Codificación de las vistas del Prototipo de aplicación

Como se puede ver en la Figura 2.34 se encuentra la lista de categorías que proporciona el prototipo de sistema de turismo desarrollado. Esta es la pantalla principal que el usuario turista mira al ingresar a la aplicación móvil. Para este ejemplo las categorías son: ¿Qué Hacer?, ¿Dónde Alojarse?, “¿Dónde Comer?”, entre otras. Se debe notar que estas categorías son administrables, es decir es posible agregar nuevas categorías, así como también modificarlas y eliminarlas.

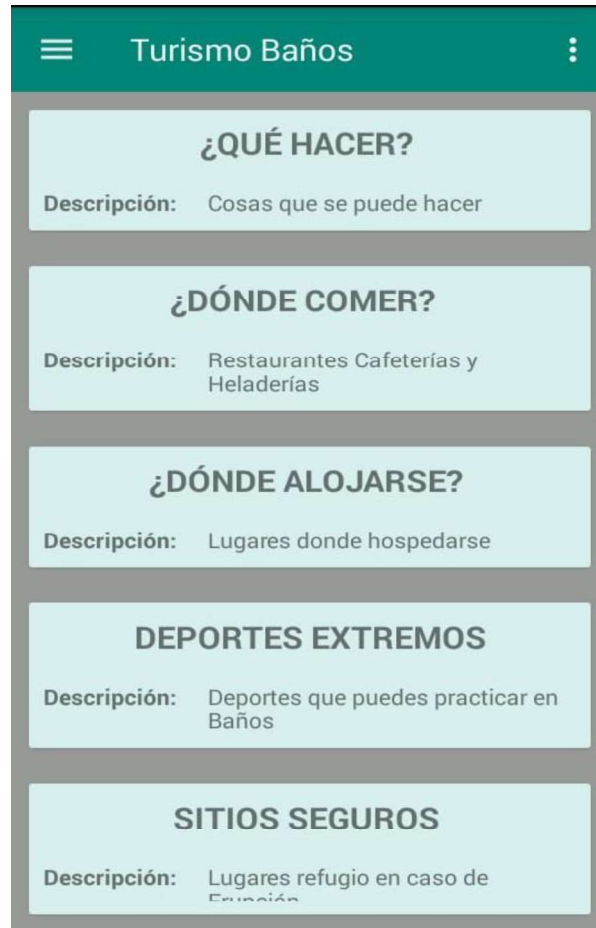


Figura 2.34 Lista de Categorías

En el código 2.12 se puede observar la codificación empleada para generar la Figura 2.35. Por efectos explicativos se presenta una porción de todo el código empleado. Para generar la lista de categorías en la pantalla principal se ha empleado el componente `LinearLayout`, mismo que permite organizar componentes en la pantalla. Como se observa la organización es vertical, lo cual indica que los componentes que pertenecen a este `LinearLayout` se organizan uno debajo del otro. En este caso las categorías corresponden a un `CardView`, mismas que se ubican una debajo de la otra.

Las etiquetas corresponden a un `TextView`, como se aprecia en el Código 2.12, la etiqueta corresponde al nombre de la categoría. Es así como se ha organizado la información que se llega como mensaje desde el servidor de comunicaciones. Se ha diseñado una interfaz amigable al usuario final, de tal manera que la información muestre información relevante para el usuario final.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="match_parent"
5      xmlns:app="http://schemas.android.com/apk/res-auto"
6      android:orientation="vertical">
7
8      <android.support.v7.widget.CardView
9          android:layout_width="match_parent"
10         android:layout_height="wrap_content"
11         app:cardBackgroundColor="#d7efd9"
12         android:layout_margin="10dp">
13
14         <LinearLayout
15             android:layout_width="match_parent"
16             android:layout_height="wrap_content"
17             android:orientation="vertical"
18             android:layout_margin="10dp">
19
20             <LinearLayout
21                 android:layout_width="match_parent"
22                 android:layout_height="wrap_content"
23                 android:orientation="horizontal">
24
25                 <TextView
26                     android:layout_width="0dp"
27                     android:layout_weight="1"
28                     android:layout_height="wrap_content"
29                     android:textSize="19dp"
30                     android:textStyle="bold"
31                     android:text="Nombre:"/>

```

Código 2.12 Generar Lista de Categorías

En la Figura 2.35 se observa la lista de sitios asociados a una categoría en particular. Para la figura estos sitios corresponden a la categoría “¿Dónde Comer?”, de esta forma se ha agrupado sitios como restaurantes, cafeterías, heladerías entre otras. En esta figura se observa que existe botones que permiten organizar la presentación de los sitios. Por ejemplo, es posible reordenar los sitios según una disposición ascendente o descendente alfabéticamente. Además, es posible organizar los sitios de acuerdo con el precio del mismo, es decir de mayor a menor el precio, o de menor a mayor del precio según como prefiera el usuario final.

Por otro lado, se provee una búsqueda de sitios. Esto indica que el usuario puede escribir una palabra en el cuadro de texto y buscar. Si existen coincidencias, en la lista de resultados

se reflejan estos resultados. Por ejemplo, es posible buscar la palabra “pailón” y en los resultados se puede observar el sitio “Pailón del Diablo”.



Figura 2.35 Sitios Asociados a una Categoría

Para codificar esta vista se ha hecho uso de “LinearLayouts”, mismos que organizan los componentes dentro de la pantalla del usuario final. En el Código 2.13 se observa a manera de ejemplo la codificación para la barra de búsqueda de un sitio, como se observa se utiliza el componente “EditText”, mismo que es utilizado para capturar el nombre ingresado por el usuario final; también se incluye un “ImageButton” el cual es el controlador que permite enviar este dato al servidor de comunicaciones. El resultado se refleja en la lista de sitios como se puede observar en la figura 2.35.

```

16 <LinearLayout
17     android:layout_width="match_parent"
18     android:layout_height="wrap_content"
19     android:orientation="horizontal">
20
21     <EditText
22         android:layout_width="0dp"
23         android:layout_height="wrap_content"
24         android:layout_weight="4"
25         android:id="@+id/edt_nombre_exacto"/>
26
27     <ImageButton
28         android:layout_width="0dp"
29         android:layout_height="wrap_content"
30         android:layout_weight="1"
31         android:id="@+id/btn_exacta"
32         android:src="@drawable/ic_lupa"/>
33 </LinearLayout>
34

```

Código 2.13 Búsqueda de Sitio dentro de Categoría

En la Figura 2.36 se observa el detalle de un sitio en particular. Se ha tomado como ejemplo el sitio “Swiss Bistro Baños”. Los componentes que integran este sitio son el nombre, la descripción, el precio, el teléfono, entre otros. Como se observa todos estos datos son descriptivos y le dan al turista información suficiente para determinar la visita. Además, se observa que existe la posibilidad de ver la ruta para llegar sitio.



Figura 2.36 Información de Sitio

En el Código 2.14 se puede observar la codificación de la descripción del sitio. Para ello, se ha incluido el controlador “ScrollView” mismo que permite desplazar la información de abajo hacia arriba cuando la información es muy amplia y no cabe en la pantalla del dispositivo. Se ha agregado un LinearLayout que permite organizar los elementos dentro de la pantalla.

Los elementos corresponden a cada uno de los elementos que integran la descripción del sitio como el nombre, precio, entre otros. Como se observa al final del Código 2.14 se ha agregado el componente TextView que identifica el nombre del sitio, a este componente le corresponde otro TextView en el cual se escribe la información que llega desde el servidor.

```
122 <LinearLayout
123     android:layout_width="match_parent"
124     android:layout_height="wrap_content"
125     android:orientation="horizontal"
126     android:layout_margin="5dp">
127
128     <TextView
129         android:layout_width="0dp"
130         android:layout_height="wrap_content"
131         android:layout_weight="1"
132         android:text="Nombre: "/>
133
134     <EditText
135         android:layout_width="0dp"
136         android:layout_height="wrap_content"
137         android:layout_weight="1"
138         android:inputType="text"
139         android:id="@+id/edtNombre_sitio"/>
140
141     <TextView
142         android:layout_width="0dp"
143         android:layout_height="wrap_content"
144         android:layout_weight="2"
145         android:text="Text"
146         android:id="@+id/txtNombre"/>
147
148 </LinearLayout>
```

Código 2.14 Descripción e Información del Sitio

Como se puede ver en la Figura 2.37, existe la posibilidad de que los usuarios puedan observar la ruta hacia el sitio en cuestión. Se ha agregado esta funcionalidad para facilitar

la visita a los lugares turísticos, debido a que en su mayoría los turistas no son de la ciudad. Este sitio corresponde a “Swiss Bistro”, y la ubicación de la persona, para este ejemplo, es cercano al terminal terrestre de Baños.

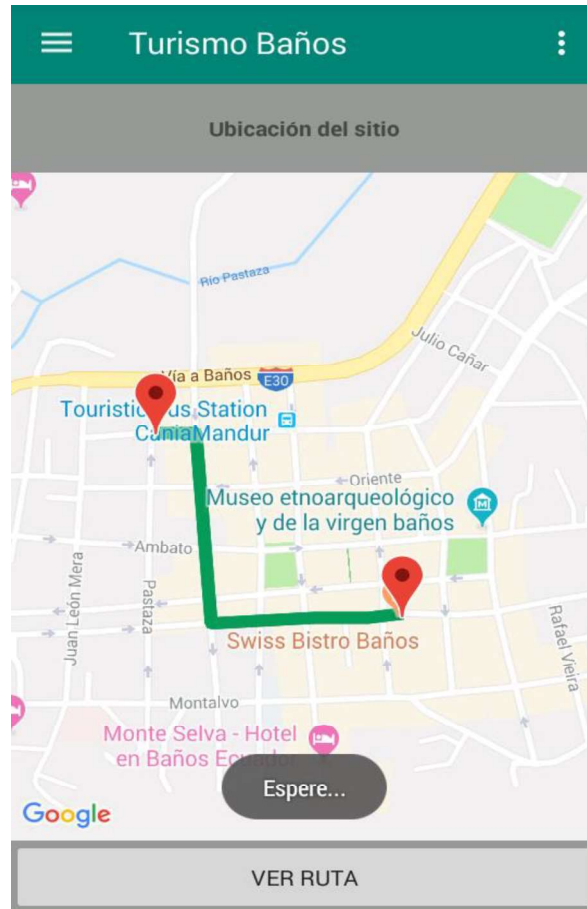


Figura 2.37 Muestra Ruta Para Llegar al Sitio

La codificación de esta pantalla se lo ha realizado como se puede observar en el Código 2.15, mismo que se observa que se utiliza el componente `LinearLayout` para organizar los elementos. Este componente es fundamental en la organización de la información por observar, para este ejemplo la disposición de estos componentes es “@vertical”. Se ha utilizado el componente “`TextView`” para informar al usuario final sobre la pantalla que observa. Se ha empleado un componente que soporta la carga de un mapa de `GoogleMaps`. Se lo realiza de esta forma para mostrar al usuario final la ruta que debe tomar para llegar al sitio de su elección.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      android:orientation="vertical"
7      tools:context=".UbicacionFragment">
8
9      <TextView
10         android:layout_width="match_parent"
11         android:layout_height="wrap_content"
12         android:textAlignment="center"
13         android:textStyle="bold"
14         android:layout_margin="20dp"
15         android:id="@+id/txt_mensaje_ubicacion"
16         android:text="Seleccione la ubicación geográfica del sitio"/>
17
18     <fragment xmlns:android="http://schemas.android.com/apk/res/android"
19         xmlns:map="http://schemas.android.com/apk/res-auto"
20         xmlns:tools="http://schemas.android.com/tools"
21         android:id="@+id/frg_ubicacion"
22         android:name="com.google.android.gms.maps.SupportMapFragment"
23         android:layout_width="match_parent"
24         android:layout_height="0dp"
25         android:layout_weight="9"
26         tools:context=".MapsActivity" />
27
28     <Button
29         android:layout_width="match_parent"
30         android:layout_height="0dp"
31         android:layout_weight="1"
32         android:id="@+id/btn_establecer_ubicacion"
33         android:text="establecer"/>
34 </LinearLayout>

```

Código 2.15 Selección Geográfica del Sitio

3. RESULTADOS Y DISCUSIÓN

En este capítulo se muestra los resultados de las pruebas de funcionamiento realizadas al prototipo de aplicación móvil. Como primer paso se tiene la actualización de las actividades en el Tablero Kanban, luego se procede a la validación de los requerimientos tanto funcionales como no funcionales, según lo planteado en el Capítulo 2 Apartado 2.1.3 al final se analizan los resultados obtenidos.

3.1 TABLERO DE ACTIVIDADES KANBAN

La Tabla 3.1 muestra el tablero Kanban actualizado para este capítulo. Las tareas de la fase de pruebas que están en la columna de “Tareas por realizar” pasaran a la columna de “Tareas en Proceso”. Todas las tareas de la Fase de Diseño y de la Fase de Implementación pasaran a la columna de “Tareas Finalizadas” lo que indica que ya se ha cumplido con todo el proceso.

Tabla 3.1 Tablero de Actividades Kanban

Tablero Kanban			
Fases	Tareas por Realizar	Tareas en Proceso	Tareas Finalizadas
Fase de Diseño			Entrevista con los funcionarios de la DTR para especificar la situación actual del manejo de la información turística del cantón Baños
			Obtener los requerimientos funcionales
			Obtener los requerimientos no funcionales
			Elaborar los casos de uso de todos los módulos
			Elaborar el diagrama de clases de los todos los módulos
			Elaborar diagrama relacional de la base de datos
			Definir las vistas del prototipo de aplicación móvil
	Fase de Implementación		
			Instalación y creación de la base de datos
			Se programará el módulo principal el mismo que implementará el servicio web
			Se codificará el módulo cliente el cual consta del prototipo de aplicación móvil de acuerdo a los requerimientos obtenidos
			Se codificará el módulo de administración
Fase de Pruebas		Se realizará la validación de los requerimientos funcionales y no funcionales	
		Se realizará pruebas de validación de funcionamiento del prototipo	

3.2 PRUEBAS DE VALIDACIÓN DE LOS REQUERIMIENTOS FUNCIONALES

En este apartado se realiza la validación, para probar que se han cumplido con todos los requerimientos funcionales descritos en el Capítulo 2 Apartado 2.1.3

3.2.1 PRUEBAS DE VALIDACIÓN DE RFT01: VISUALIZAR CLASIFICACIÓN DE SITIO O ACTIVIDAD POR CATEGORÍAS

Se procede a verificar el cumplimiento del requerimiento funcional RFT01, esta tarea es realizada por el usuario turista. La Tabla 3.2 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla3.2 Descripción de RFT01

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS04	Turista	CU04	RFT01	Visualizar clasificación de sitio o actividad por categorías

La Figura 3.1 muestra la pantalla principal de la aplicación, permitiendo ver que los sitios o actividades se encuentran clasificados en diferentes categorías, para el ejemplo se selecciona la categoría “¿Dónde Comer?”, donde la siguiente pantalla muestra los restaurantes que se encuentra en la categoría seleccionada cumpliendo así con el primer requerimiento funcional.

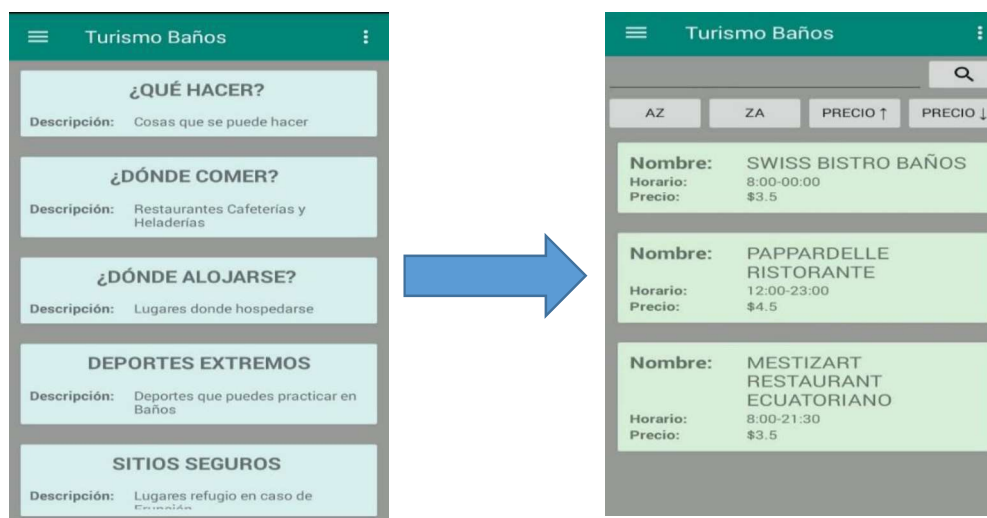


Figura 3.1 Visualizar Clasificación de Sitio o Actividad por Categorías

3.2.2 PRUEBAS DE VALIDACIÓN DE RFT02: VISUALIZAR UNA FOTO Y DESCRIPCIÓN DEL SITIO O ACTIVIDAD

Se procede a verificar el cumplimiento del requerimiento funcional RFT02, esta tarea es realizada por el turista que tiene la aplicación instalada en su celular con sistema operativo Android. La Tabla 3.3 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.3 Descripción de RFT02

Subsistema	Usurario	Caso de Uso	Número RFT	Descripción
SBS04	Turista	CU04	RFT02	Visualizar una foto y descripción del sitio o actividad

La Figura 3.2 muestra que para cumplir este requerimiento se ingresa a la aplicación la pantalla de inicio muestra todas las categorías existentes, se da clic en la categoría que el usuario elija, para el ejemplo es “¿Dónde Comer?”. Dentro de la categoría se muestra los restaurantes, cafeterías, heladerías, se selecciona uno, se visualiza la foto y la descripción del mismo, así cumpliendo con el segundo requerimiento funcional.

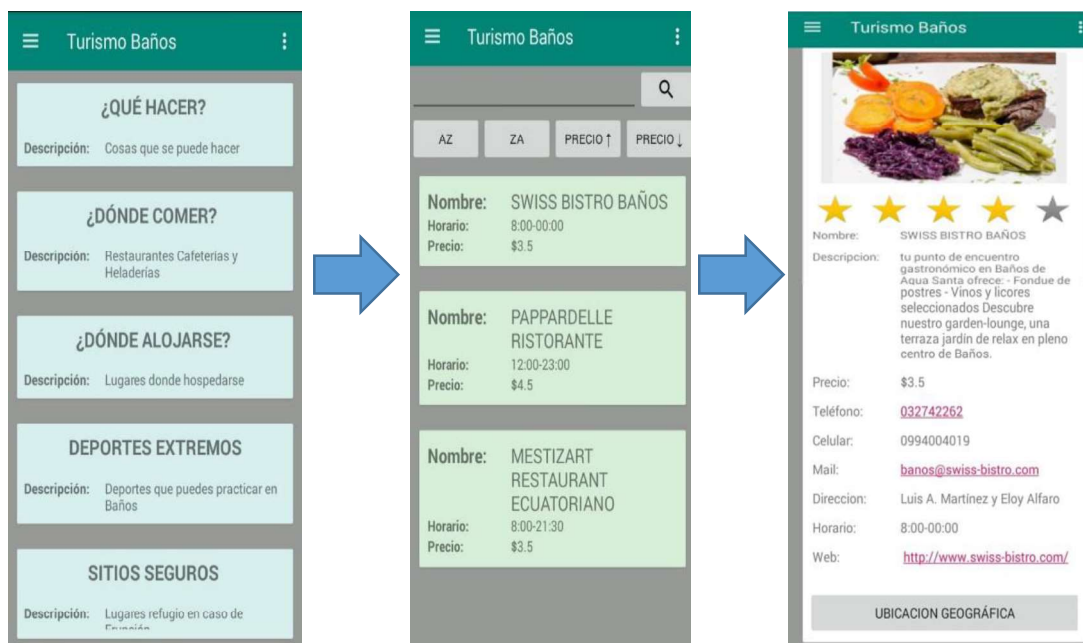


Figura 3.2 Visualización de Imagen e Información de Sitio

3.2.3 PRUEBAS DE VALIDACIÓN DE RFT03: VISUALIZAR EN UN MAPA COMO LLEGAR AL SITIO SELECCIONADO

Se procede a verificar el cumplimiento del requerimiento funcional RFT03, esta tarea es realizada por el turista que tiene la aplicación instalada en su celular con sistema operativo Android. La Tabla 3.4 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.4 Descripción de RFT03

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS04	Turista	CU04	RFT03	Visualizar en un mapa como poder llegar al sitio seleccionado

La Figura 3.3 muestra, el primer paso es ingresar a la aplicación, la pantalla de inicio muestra todas las categorías existentes, se da clic en la categoría que el usuario elija, para el ejemplo es “¿Dónde Comer?”. Dentro de la categoría se muestra los restaurantes, cafeterías, heladerías, se selecciona uno, se puede visualizar la foto y la descripción del mismo. En la parte inferior se tiene un botón con el nombre “Ubicación Geográfica”, se da clic y se muestra la nueva pantalla con un mapa, la ubicación del sitio. Una vez en el mapa en la parte inferior se encuentra un botón denominado “Ver Ruta”, se da clic sobre este y se muestra la ruta a seguir para llegar a dicho sitio.

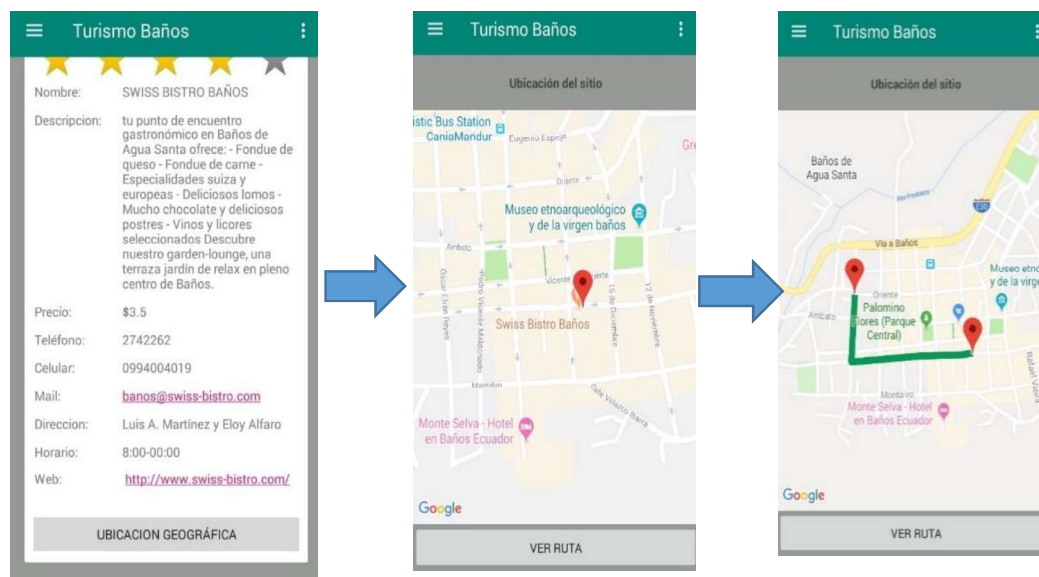


Figura 3.3 Visualización de la Ruta a un Sitio

3.2.4 PRUEBAS DE VALIDACIÓN DE RFT04: BUSCAR SITIOS O ACTIVIDADES

Se procede a verificar el cumplimiento del requerimiento funcional RFT04, esta tarea es realizada por el turista que tiene la aplicación instalada en su celular con sistema operativo Android. La Tabla 3.5 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.5 Descripción de RFT04

Subsistema	UsUARIO	Caso de Uso	Número RFT	Descripción
SBS01	Turista	CU01	RFT04	Buscar sitios o actividades

La Figura 3.4 muestra, que el primer paso, es ingresar a la aplicación, dar clic sobre la barra de tareas o barra de acceso rápido (tres barras de la parte superior izquierda), mostrándose un submenú dentro del mismo se da clic en “General”. Dentro de general aparecen todos los sitios y actividades ingresados en el prototipo de aplicación y se tiene en la parte superior la opción para buscar por nombre. En esta parte se ingresa el nombre del sitio que se desea buscar al terminar se da clic en la lupa mostrando el resultado de la búsqueda y de esta forma se cumple el cuarto requerimiento funcional, La Figura 3.5 muestra la búsqueda de sitio dentro del menú categoría. La búsqueda no solo se la puede realizar dentro del submenú General sino también dentro de cada categoría.

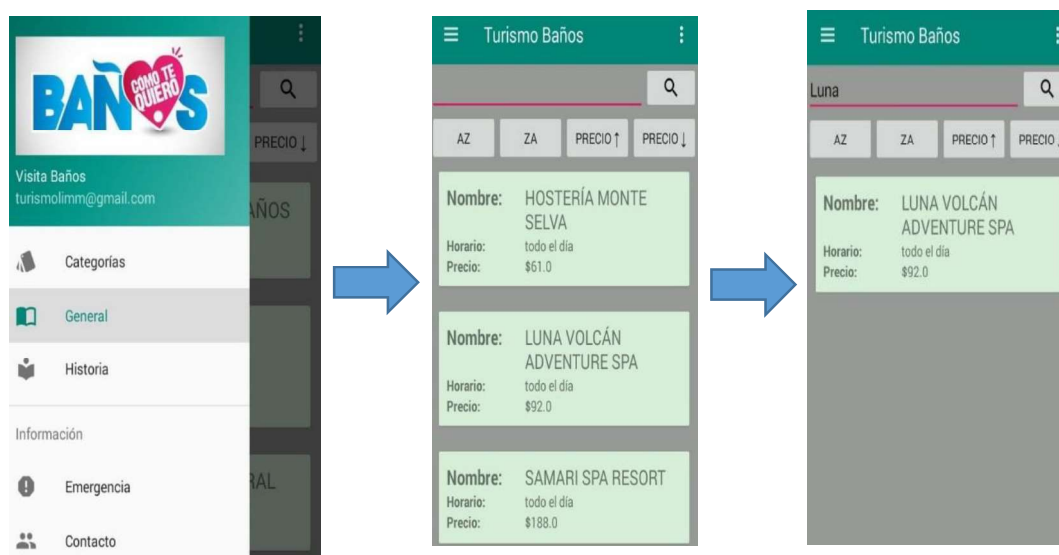


Figura 3.4 Búsqueda de Sitio en Submenú General

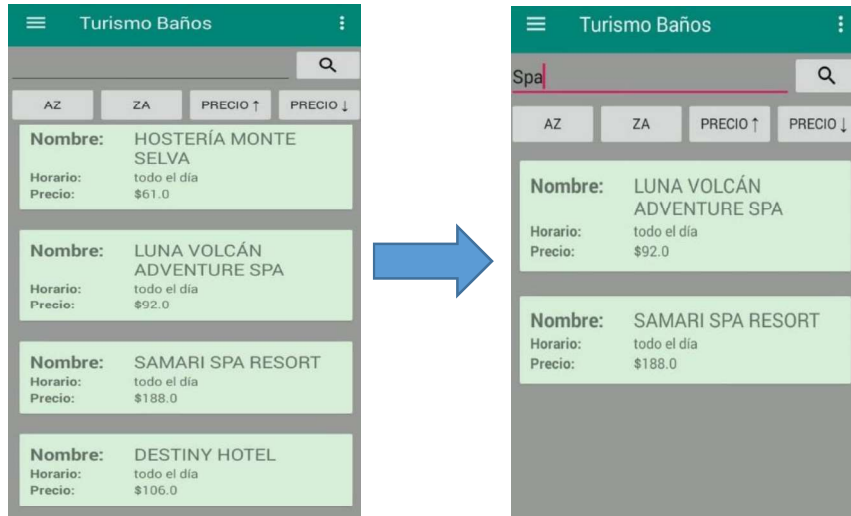


Figura 3.5 Búsqueda de sitio dentro de Categoría

3.2.5 PRUEBAS DE VALIDACIÓN DE RFT05: ORDENAR LOS SITIOS SEGÚN SUS PRECIOS ASCENDENTE O DESCENDENTE

Se procede a verificar el cumplimiento del requerimiento funcional RFT05, esta tarea es realizada por el turista que tiene la aplicación instalada en su celular con sistema operativo Android. La Tabla 3.6 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.6 Descripción de RFT05

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS01	Turista	CU01	RFT05	Ordenar los sitios según sus precios ascendente o descendente

La Figura 3.6 muestra, que el primer paso es, ingresar a la aplicación, dar clic sobre la barra de tareas o barra de acceso rápido (tres barras de la parte superior izquierda), mostrándose un submenú dentro del mismo, se da clic en General. Dentro de general aparecen todas los sitios y actividades ingresados en el prototipo de aplicación y se tiene en la parte superior los botones con el nombre precio. Los botones precio, están acompañados de una flecha la cual está dirigida hacia abajo significa que es precio descendente y la flecha dirigida hacia arriba que es precio ascendente. Pues bien, se da clic en estos botones y los sitios de ordenan según sus precios de forma ascendente o descendente, esto no solo lo puedes realizar dentro del submenú General sino también

dentro de cada categoría, la Figura 3.7 muestra como ordenar sitios por precio descendente y ascendente dentro del submenú categoría.

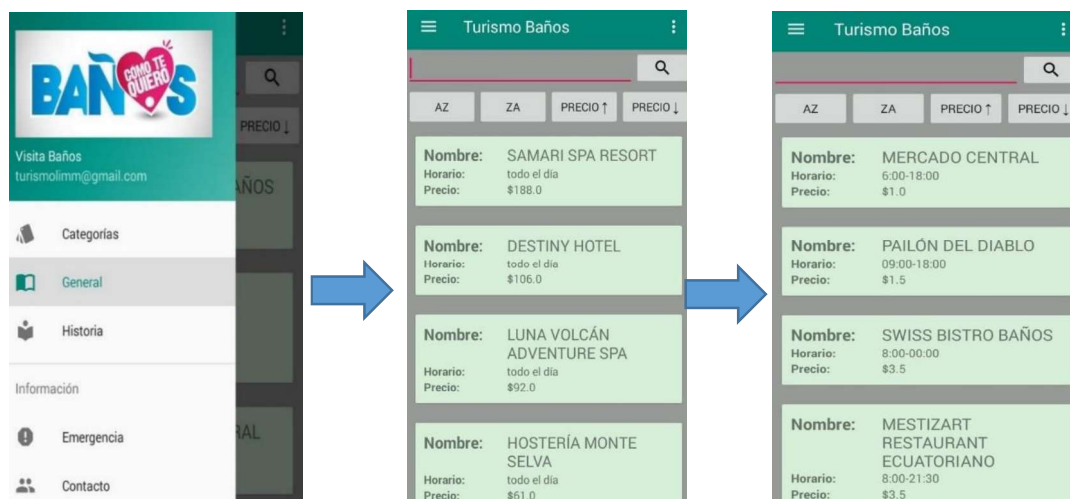


Figura 3.6 Ordenar Sitios por Precio Descendente y Ascendente en Submenú General

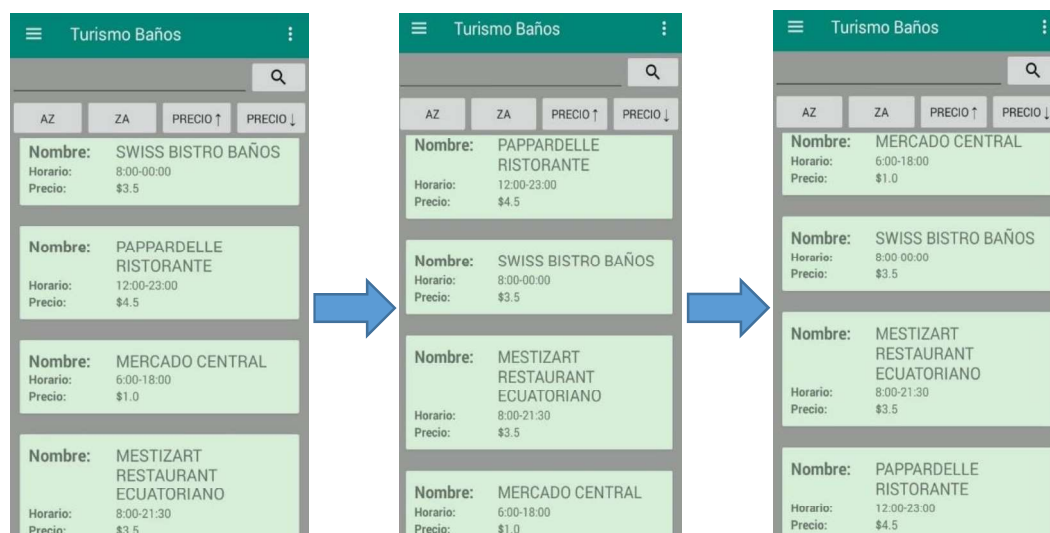


Figura 3.7 Ordenar Sitios por Precio Descendente y Ascendente dentro de Categoría

3.2.6 PRUEBAS DE VALIDACIÓN DE RFT06: ORDENAR LOS SITIOS SIGUIENDO UN ORDEN ALFABÉTICO A-Z O Z-A

Se procede a verificar el cumplimiento del requerimiento funcional RFT06, esta tarea es realizada por el turista que tiene la aplicación instalada en su celular con sistema operativo Android. La Tabla 3.7 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.7 Descripción de RFT06

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS01	Turista	CU01	RFT06	Ordenar los sitios siguiendo un orden alfabético A-Z o Z-A

La Figura 3.8 muestra, el sexto requerimiento, donde el turista tiene que ingresar a la aplicación, se da clic sobre la barra de tareas o barra de acceso rápido (tres barras de la parte superior izquierda), se muestra un submenú dentro del mismo, se da clic en General. Dentro de general aparecen todos los sitios y actividades ingresados en el prototipo de aplicación y se tiene en la parte superior los botones con los nombres AZ y ZA. Al dar clic en estos botones los sitios se ordenarán de forma alfabética desde la AZ o desde la ZA. De esta forma se cumple el requerimiento funcional seis y esta forma de ordenar también se puede realizar dentro de cada categoría, la Figura 3.9 muestra como ordenar sitios de forma alfabética de la AZ o ZA dentro del submenú de Categorías

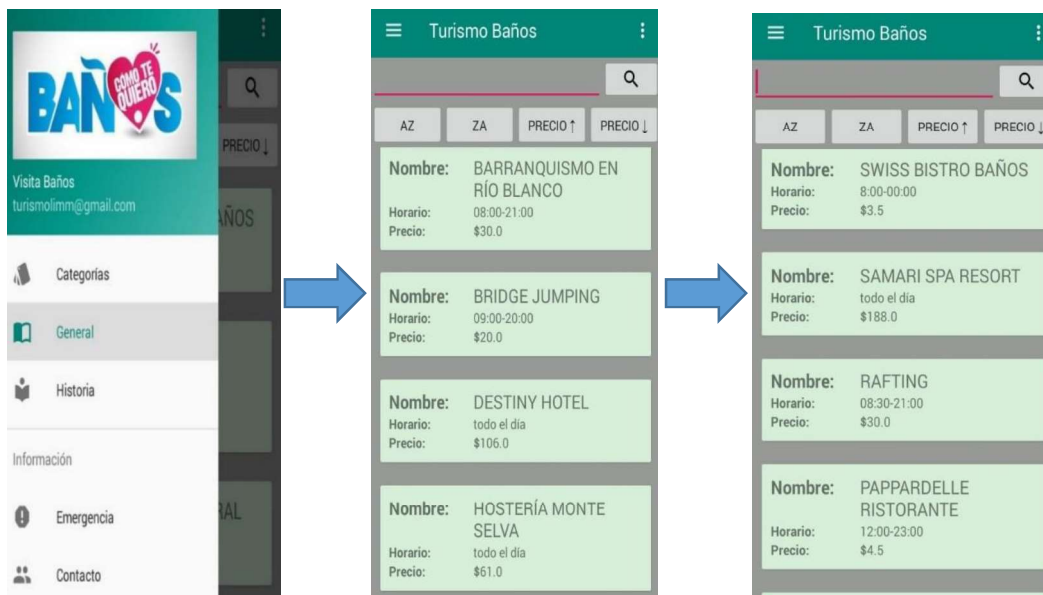


Figura 3.8 Ordenar Sitios Orden Alfabético AZ o ZA en Submenú General

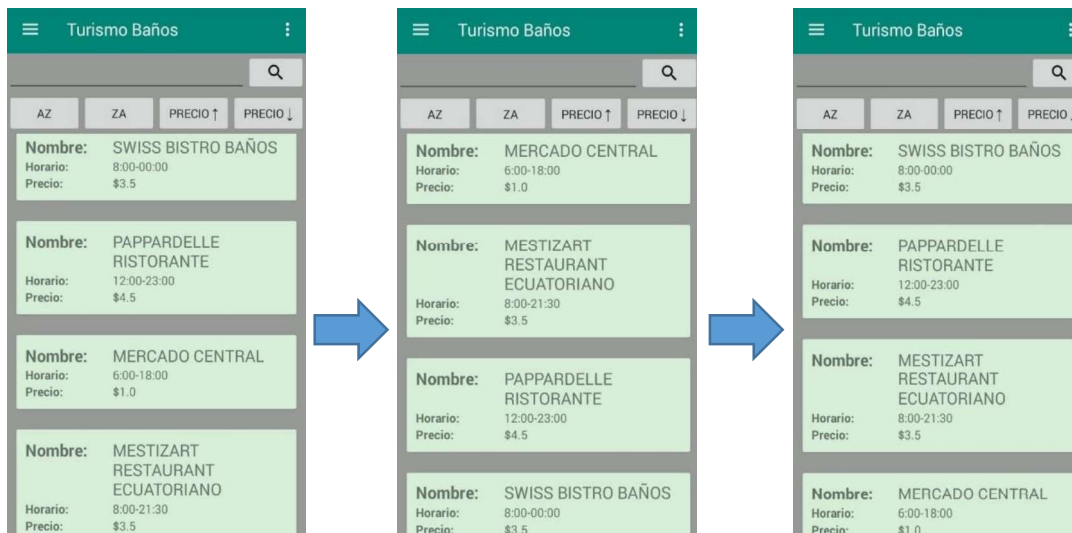


Figura 3.9 Ordenar Sitios Orden Alfabético AZ o ZA dentro de Categorías

3.2.7 PRUEBAS DE VALIDACIÓN DE RFT07: CALIFICAR UN SITIO O ACTIVIDAD

Se procede a verificar el cumplimiento del requerimiento funcional RFT07, esta tarea es realizada por el turista que tiene la aplicación instalada en su celular con sistema operativo Android. La Tabla 3.8 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.8 Descripción de RFT07

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS01	Turista	CU01	RFT07	Calificar un sitio o actividad

La Figura 3.10 muestra, que el primer paso es ingresar a la aplicación, la pantalla de inicio muestra todas las categorías existentes, se da clic, en la categoría que se desee, para el ejemplo es “¿Dónde Comer?”. Dentro de la categoría se muestra los restaurantes, cafeterías, heladerías, se selecciona uno y se puede visualizar las fotos del mismo, a continuación de eso se tiene cinco estrellas para calificar el sitio o actividad, designado como una estrella muy mala y cinco estrellas excelente.

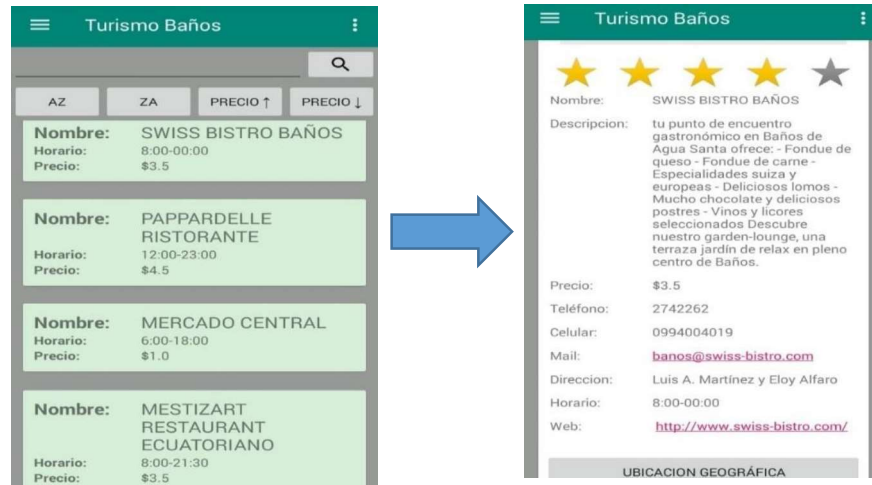


Figura 3.10 Calificar un Sitio o Actividad

3.2.8 PRUEBAS DE VALIDACIÓN DE RFT08: ALMACENAR IMÁGENES DEL SITIO O ACTIVIDAD.

Se procede a verificar el cumplimiento del requerimiento funcional RFT08, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.9 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.9 Descripción de RFT08

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS02	Administrador	CU02	RFT08	Almacenar Imágenes del sitio o actividad

La Figura 3.11 muestra la forma en que se almacena las imágenes del sitio o actividad. Para cumplir con este requerimiento se tiene que crear un nuevo sitio o editar un sitio ya creado, para así ver como las imágenes son guardadas conjuntamente con la información del sitio. Entonces, dentro del menú sitios, se da clic en nuevo para crear un sitio, se muestra una nueva pantalla donde está un formulario vacío, en el cual se ingresa la información del sitio. Se tiene la posibilidad de añadir una fotografía, para ello se debe

presionar en el botón “Añadir Fotografía”, se selecciona la foto esta es cargada y se presiona en el botón de guardar, e inmediatamente la información es enviada al servidor remoto.

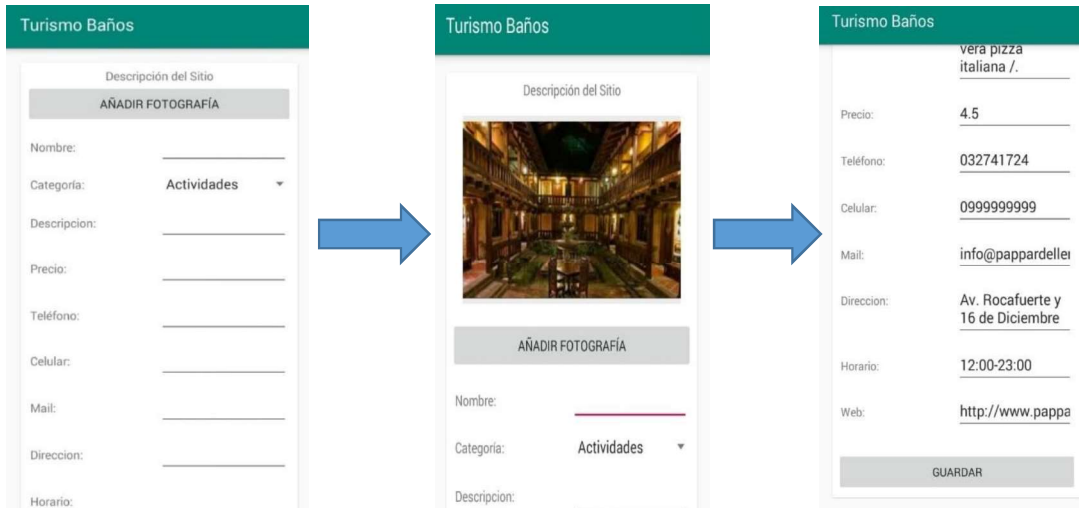


Figura 3.11 Almacenar Imágenes de un Sitio o Actividad

3.2.9 PRUEBAS DE VALIDACIÓN DE RFT09: ALMACENAR LA INFORMACIÓN CONCERNIENTE ADMINISTRADORES.

Se procede a verificar el cumplimiento del requerimiento funcional RFT09, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.10 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.10 Descripción de RFT09

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS02	Administrador	CU02	RFT09	Almacenar la información concerniente administradores

Este proceso es el mismo que se realiza cuando se crea un nuevo usuario, y se detallara en el Apartado 3.2.18. Toda esta información queda almacenada en la base de datos, que está ubicada en el servidor remoto.

3.2.10 PRUEBAS DE VALIDACIÓN DE RFT10: ALMACENAR LA INFORMACIÓN DE LOS SITIOS TURÍSTICOS.

Se procede a verificar el cumplimiento del requerimiento funcional RFT10, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.11 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.11 Descripción de RFT10

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS02	Administrador	CU02	RFT10	Almacenar la información de los sitios turísticos

Este proceso es el mismo que se realiza cuando se crea un nuevo sitio o actividad y se detallara en el Apartado 3.2.15. Toda esta información queda almacenada en la base de datos, que está ubicada en el servidor remoto.

3.2.11 PRUEBAS DE VALIDACIÓN DE RFT11: AUTENTICAR UN ADMINISTRADOR.

Se procede a verificar el cumplimiento del requerimiento funcional RFT11, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.12 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.12 Descripción de RFT11

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT11	Autenticar un administrador

La Figura 3.12 muestra que para autenticar un administrador el primer paso es entrar a la aplicación dar clic sobre los tres puntos de acceso rápido (tres puntos que se encuentran en la parte superior derecha del prototipo de aplicación), se muestra el botón Administrar, se da clic y se visualiza la pantalla donde se puede ingresar las credenciales para autenticarse como administradores



Figura 3.12 Autenticar un Administrador

3.2.12 PRUEBAS DE VALIDACIÓN DE RFT12: CREAR NUEVA CATEGORÍA.

Se procede a verificar el cumplimiento del requerimiento funcional RFT12, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.13 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.13 Descripción de RFT12

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT12	Crear nueva categoría

La Figura 3.13 muestra, que para cumplir con este requerimiento se procede, dentro del menú de administrador, se selecciona categoría, se muestra una nueva pantalla, se selecciona nuevo, se llena el formulario, se guarda y se crea una nueva categoría.



Figura 3.13 Crear Categoría

3.2.13 PRUEBAS DE VALIDACIÓN DE RFT13: EDITAR CATEGORÍA.

Se procede a verificar el cumplimiento del requerimiento funcional RFT13, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.14 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.14 Descripción de RFT13

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT13	Editar categoría

La Figura 3.14 muestra, dentro de administrar categoría, se selecciona una de las categorías creadas, se muestra una nueva pantalla donde se puede ver dos botones, editar y eliminar. Se da clic en Editar, se procede a cambiar lo que se requiera y finalmente se guarda.

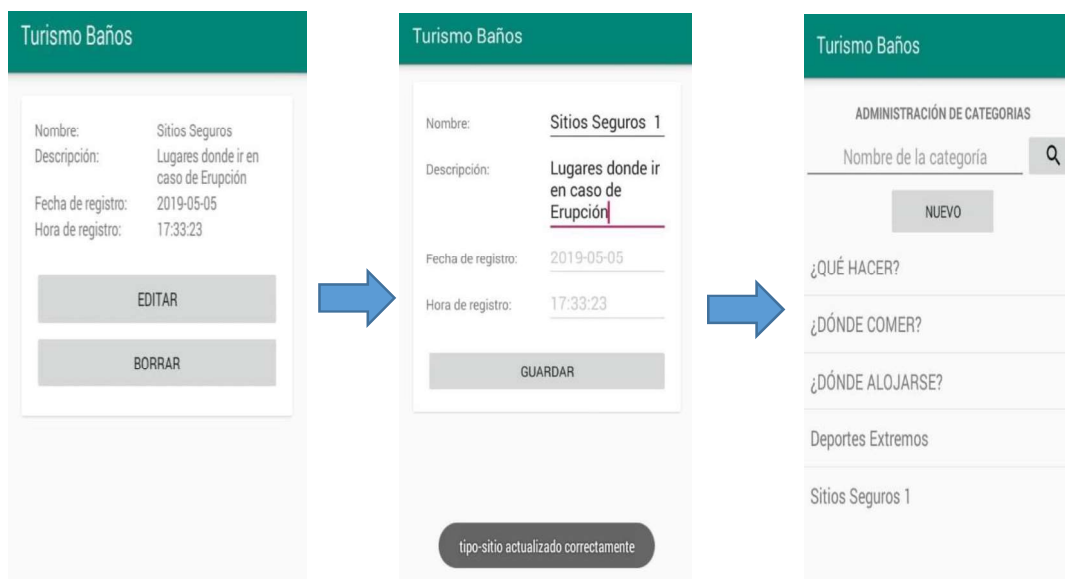


Figura 3.14 Editar Categoría

3.2.14 PRUEBAS DE VALIDACIÓN DE RFT14: ELIMINAR CATEGORÍA.

Se procede a verificar el cumplimiento del requerimiento funcional RFT14, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.15 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.15 Descripción de RFT14

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS01	Administrador	CU14	RFT14	Eliminar categoría

La Figura 3.15 muestra, que para cumplir con este requerimiento se procede, dentro de administrar categoría, se selecciona una de las categorías ya creada, se muestra una nueva pantalla donde se puede ver dos botones, editar y eliminar. Se da clic en Eliminar, de inmediato se elimina la categoría, siempre y cuando esta esté vacía es decir no contenga sitios.



Figura 3.15 Eliminar Categoría

3.2.15 PRUEBAS DE VALIDACIÓN DE RFT15: CREAR NUEVO SITIO O ACTIVIDAD.

Se procede a verificar el cumplimiento del requerimiento funcional RFT15, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.16 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.16 Descripción de RFT15

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT15	Crear nuevo sitio o actividad

La Figura 3.16 muestra, que para cumplir con este requerimiento se procede, dentro del menú de administrador se selecciona un sitio, se da clic en nuevo, se llena el formulario con la información requerida, se da clic en guardar y el sitio ha sido creado.

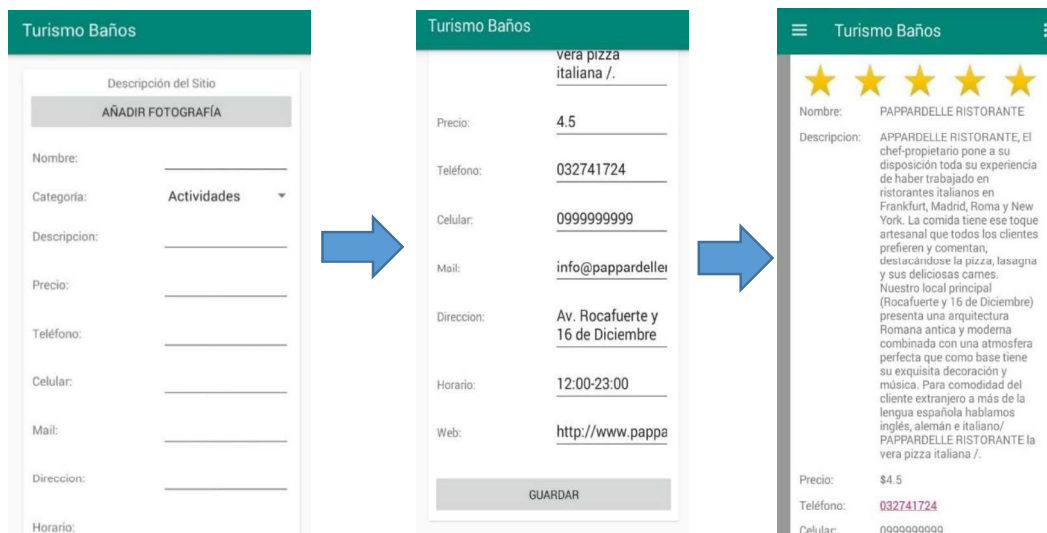


Figura 3.16 Crear Sitio o Actividad

3.2.16 PRUEBAS DE VALIDACIÓN DE RFT16: EDITAR SITIO O ACTIVIDAD.

Se procede a verificar el cumplimiento del requerimiento funcional RFT16, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.17 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.17 Descripción de RFT16

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT16	Editar sitio o actividad

La Figura 3.17 muestra, que para cumplir con este requerimiento se procede, dentro del menú de administrador se selecciona sitio, se da clic sobre el sitio que se desea editar. En la parte inferior se muestra los botones de Editar y Borrar ahí se mira la información luego de pulsar sobre editar, se cambia lo que sea necesario, se da clic en guardar y el sitio ha sido editado.

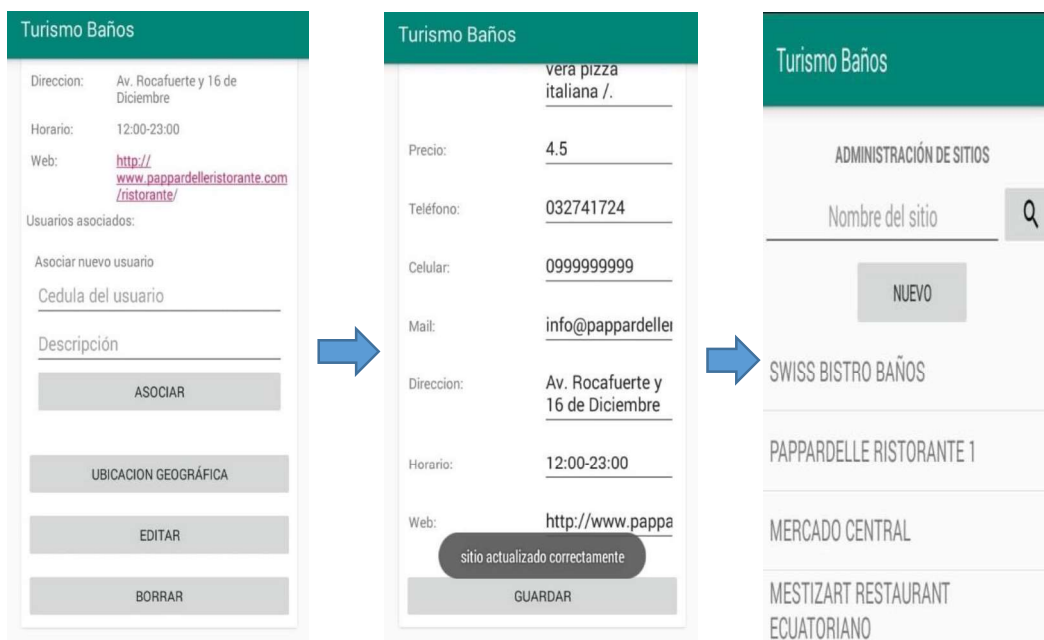


Figura 3.17 Editar Sitio o Actividad

3.2.17 PRUEBAS DE VALIDACIÓN DE RFT17: ELIMINAR SITIO O ACTIVIDAD.

Se procede a verificar el cumplimiento del requerimiento funcional RFT17, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.18 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.18 Descripción de RFT17

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT17	Eliminar sitio o actividad

La Figura 3.18 muestra, que para cumplir con este requerimiento se procede, dentro del menú de administrador se selecciona sitio, se da clic sobre el sitio que se desea eliminar. En la parte inferior se muestra los botones de Editar y Borrar, se da clic, solo el botón borrar, se confirma eliminación y el sitio es borrado.

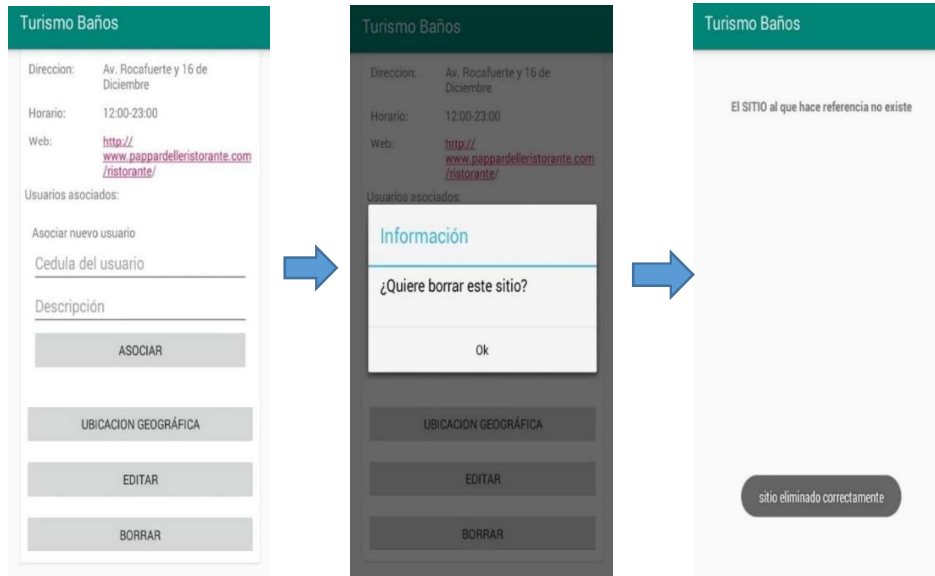


Figura 3.18 Eliminar Sitio o Actividad

3.2.18 PRUEBAS DE VALIDACIÓN DE RFT18: CREAR NUEVO USUARIO.

Se procede a verificar el cumplimiento del requerimiento funcional RFT18, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.19 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.19 Descripción de RFT18

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT18	Crear nuevo usuario

La Figura 3.19 muestra, que para cumplir con este requerimiento se procede, para crear un usuario y así cumplir con el requerimiento dentro el menú de administrador se ingresa a usuarios, dentro se da clic en nuevo, se muestra el formulario vacío, se lo llena, se guardar y se termina de crear el usuario. En esta parte se tiene dos tipos de usuarios un administrador y otro normal, donde el usuario normal se crea para asociarlo a un sitio o actividad.

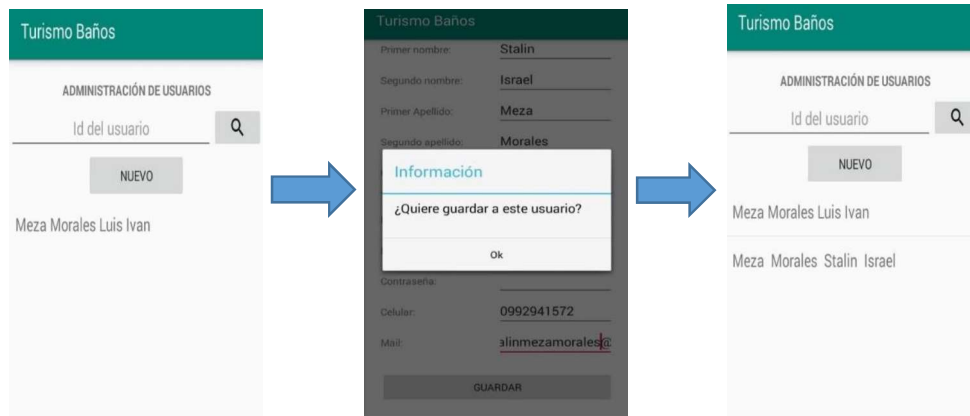


Figura 3.19 Crear Usuario

3.2.19 PRUEBAS DE VALIDACIÓN DE RFT19: EDITAR USUARIO.

Se procede a verificar el cumplimiento del requerimiento funcional RFT19, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.20 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.20 Descripción de RFT19

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT19	Editar usuario

La Figura 3.20 muestra, para editar un usuario y así cumplir con el requerimiento dentro el menú de administrador se ingresa a usuarios, aquí se escoge el usuario a editar su información. En la siguiente pantalla en la parte inferior aparecen dos botones editar y eliminar. Se da clic en editar y se muestra el formulario se modifica la información que se desee, se da clic en guardar y la información queda editada.

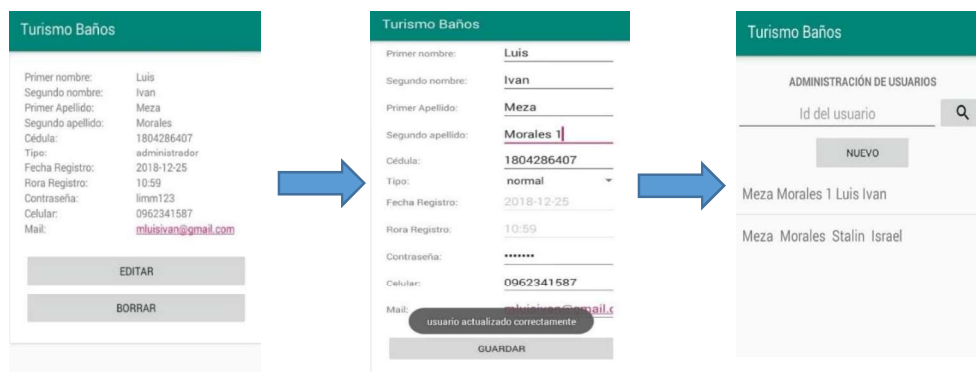


Figura 3.20 Editar Usuario

3.2.20 PRUEBAS DE VALIDACIÓN DE RFT20: ELIMINAR USUARIO.

Se procede a verificar el cumplimiento del requerimiento funcional RFT20, esta tarea es realizada por el administrador de la aplicación. La Tabla 3.21 muestra el requerimiento funcional, el subsistema, el caso de uso al que pertenece.

Tabla 3.21 Descripción de RFT20

Subsistema	Usuario	Caso de Uso	Número RFT	Descripción
SBS03	Administrador	CU03	RFT20	Eliminar usuario

La Figura 3.21 muestra, para Eliminar un usuario y así cumplir con el requerimiento, dentro el menú de administrador se ingresa a usuarios, aquí se escoge el usuario a eliminar. En la siguiente pantalla en la parte inferior aparecen dos botones editar y borrar. Se da clic en Borrar, se confirma que se desea borrar, quedando así borrado el usuario.



Figura 3.21 Eliminar Usuario

Una vez terminada la validación de cada uno de los requerimientos funcionales dependiendo a que subsistema pertenecen sea este al del turista o administrador. Se puede mirar que el prototipo realiza las funciones esperadas y cumple con las especificaciones definidas, en la Tabla 3.22 se muestra los requerimientos funcionales y el cumplimiento de los mismos.

Tabla 3.22 Validación y Cumplimiento de Todos los RF

Número RFT	Descripción	Validación /Cumplimiento
RFT01	Visualizar clasificación de sitio o actividad por categorías	SI
RFT02	Visualizar una foto y descripción del sitio o actividad	SI
RFT03	Visualizar en un mapa como poder llegar al sitio seleccionado	SI
RFT04	Buscar sitios o actividades	SI
RFT05	Ordenar los sitios según sus precios ascendente o descendente	SI
RFT06	Ordenar los sitios siguiendo un orden alfabético A-Z o Z-A	SI
RFT07	Calificar un sitio o actividad	SI
RFT08	Almacenar Imágenes del sitio o actividad	SI
RFT09	Almacenar la información concerniente administradores	SI
RFT10	Almacenar la información de los sitios turísticos	SI
RFT11	Autenticar un administrador	SI
RFT12	Crear nueva categoría	SI
RFT13	Editar categoría	SI
RFT14	Eliminar categoría	SI
RFT15	Crear nuevo sitio o actividad	SI
RFT16	Editar sitio o actividad	SI
RFT17	Eliminar sitio o actividad	SI
RFT18	Crear nuevo usuario	SI
RFT19	Editar usuario	SI
RFT20	Eliminar usuario	SI

3.3 PRUEBAS DE VALIDACIÓN DE LOS REQUERIMIENTOS NO FUNCIONALES

La Tabla 3.23 muestra los Requerimientos No Funcionales, algunos de estos son validados por medio de las encuestas realizadas, tanto al personal de la DTR como a las personas que cumplen con el perfil que requiere el sistema, ya que existen preguntas relacionadas con respecto a la validación de estos requerimientos.

Tabla 3.23 Requerimientos No Funcionales

Etiqueta	Número RNFT	Descripción
Presentación	RNFT01	El prototipo de aplicación debe tener una interfaz que se amigable e intuitiva de manejar.
Disponibilidad	RNFT02	El Prototipo deberá estar disponible al menos el 99% del tiempo.
Escalabilidad	RNFT03	El prototipo debe ser construido en una plataforma la cual debe permitir el incremento de los recursos tales como: memoria RAN, Procesador, Almacenamiento.
Autenticación	RNFT04	El acceso al módulo de administración va a estar restringido, los usuarios administradores tendrán su usuario y contraseña para acceder.

3.3.1 REQUERIMIENTO NO FUNCIONAL RNFT01: PRESENTACIÓN

La presentación debe ser atractiva, intuitiva, funcional y amigable [61], permite así que los usuarios realicen de forma rápida y sencilla las acciones requeridas en el prototipo de aplicación. En las encuestas realizadas se ha incluido una pregunta que permite evaluar el RNFT01 Presentación.

En la Tabla 3.24 se muestra los resultados de la pregunta número dos realizada tanto a los usuarios Administradores como a los usuarios turistas, con un total de 20 personas las que respondieron esta pregunta. Los resultados alcanzados son que un 80% de los encuestados consideran que es excelente, un 15% manifiestan en la encuesta que es muy buena y un 5% expresan que es buena, la presentación.

Tabla 3.24 Resultados Pregunta Presentación.

Usuario Administrador / Turista	Pregunta 2	Calificaciones	Valores	Porcentaje
	¿En la escala del 1 al 5 el prototipo de aplicación tiene una presentación la cual es intuitiva y amigable?	Muy Mala (1)	0	0%
Mala (2)		0	0%	
Buena (3)		1	5%	
Muy Buena (4)		3	15%	
Excelente (5)		16	80%	

3.3.2 REQUERIMIENTO NO FUNCIONAL RNFT02: DISPONIBILIDAD

La disponibilidad se la define como, el grado con que los recursos del sistema están a disposición del usuario, haciendo referencia a la probabilidad de que el servicio funcione adecuadamente [62]. El prototipo de aplicación móvil se desarrolló utilizando el proveedor de servicios AWS, siendo una de las características principales del mismo que la mayor parte del tiempo el servicio está funcionando. En el *Acuerdo de Nivel de Servicios de Amazon*, la disponibilidad que se oferta es la del 99.99% [63]. Las zonas de disponibilidad se han diseñado con la finalidad de brindar redundancia física y proveer resiliencia, lo que permite un rendimiento continuo, incluso si ocurren Interrupciones en el suministro de energía, cortes en el servicio de internet, inundaciones y otras catástrofes naturales [64]. En base a estos datos se cumple el requerimiento no funcional de Disponibilidad.

3.3.3 REQUERIMIENTO NO FUNCIONAL RNFT03: ESCALABILIDAD

La escalabilidad se define como la capacidad de adaptación y respuesta de un sistema con respecto del rendimiento del mismo, cuando de forma significativa crecen el número de usuarios que utilizan el sistema [65]. La escalabilidad es uno de los beneficios ofertados por AWS, permitiendo por su infraestructura global, un nivel de flexibilidad muy alto y los recursos pueden ser usados de forma inmediata cuando lo necesiten [64]. En este caso el crecimiento y la utilización de los recursos tanto de almacenamiento, núcleos de procesador, memoria RAM y velocidad de conexión de datos, pueden ser incrementados de inmediato según la necesidad, Garantizando que el requerimiento de Escalabilidad esta cumplido.

3.3.4 REQUERIMIENTO NO FUNCIONAL RNFT04: AUTENTICACIÓN

La autenticación del prototipo de aplicación móvil turística, se la realiza utilizando las credenciales de acceso en este caso el Usuario que es la cedula y la contraseña seleccionada por el Administrador. Las credenciales son introducidas en el formulario de inicio de sesión de Administrador y así se puede utilizar todas las herramientas para administrar el prototipo de aplicación.

La Tabla 3.25 muestra la pregunta que se realizó a los Usuarios administradores con respecto a la Autenticación, teniendo como resultado diez respuestas positivas que sí pudieron realizar dicho proceso.

Tabla 3.25 Respuesta Pregunta Autenticación

Usuario	Pregunta	Respuestas	
		SI	NO
Administrador	¿Usted pudo Autenticarse como Administrador?	10	0

En la Figura 3.22 se muestra como los usuarios administradores pueden autenticarse.



Figura 3.22 Autenticación de Usuario

Finalmente se presenta la Tabla 3.26 donde se muestra los resultados una vez realizada la validación de los requerimientos no funcionales.

Tabla 3.26 Validación / Cumplimiento RNFT

Etiqueta	Numero RNFT	Descripción	Validación /Cumplimiento
Presentación	RNFT01	El prototipo de aplicación debe tener una interfaz que se amigable e intuitiva de manejar.	SI
Disponibilidad	RNFT02	El Prototipo deberá estar disponible al menos el 99% del tiempo.	SI
Escalabilidad	RNFT03	El prototipo debe ser construido en una plataforma la cual debe permitir el incremento de los recursos tales como: memoria RAN, Procesador, Almacenamiento.	SI
Autenticación	RNFT04	El acceso al módulo de administración va a estar restringido, los usuarios administradores tendrán su usuario y contraseña para acceder.	SI

3.4 PRUEBAS DE FUNCIONAMIENTO DEL PROTOTIPO

En este apartado se realiza las pruebas de funcionamiento del prototipo de aplicación móvil turística, para esto se facilita el prototipo de aplicación móvil a veinte personas entre los que se encuentran los funcionarios de la DTR para que la instalen y la utilicen, luego de eso se realiza una encuesta a estos usuarios. Los usuarios son diez con el perfil de turista y diez con el perfil de administrador. Las encuestas fueron diseñadas a partir de las especificaciones de los requerimientos, teniendo así una encuesta para el usuario administrador y otra para el usuario turista a continuación se muestra el formato de las encuestas realizadas.

Prototipo de Aplicación Móvil Turística con Información del Cantón Baños

(Encuesta Usuario Administrador)

1. En la escala del 1 al 5 que calificación le pondría al prototipo de aplicación Móvil Turística.

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muy Mala	Mala	Buena	Muy Buena	Excelente

2. ¿En la escala del 1 al 5 el prototipo de Aplicación tiene una presentación la cual es intuitiva y amigable?

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muy Mala	Mala	Buena	Muy Buena	Excelente

3. ¿Usted pudo autenticarse como Administrador?

<input type="radio"/>	SI
<input type="radio"/>	NO

4. ¿Usted pudo crear una nueva categoría, sitio, o un nuevo usuario?

<input type="radio"/>	SI
<input type="radio"/>	NO

5. ¿Usted pudo editar una categoría, sitio, o un perfil de usuario?

SI

NO

6. ¿Usted pudo eliminar una categoría, sitio, o perfil de usuario?

SI

NO

7. ¿Si usted pudiera añadir algo a la aplicación que le añadiría?

Realizar reserva de Hoteles

Realizar reserva en Restaurantes

Itinerarios de transporte público

Mostrar sitios o actividades turísticas
cercanos a la ubicación del usuario

Otros

Gracias por su colaboración

Prototipo de Aplicación Móvil Turística con Información del Cantón Baños

(Encuesta Usuario Turista)

1. En la escala del 1 al 5 que calificación le pondría al prototipo de aplicación Móvil Turística.

1

Muy Mala

2

Mala

3

Buena

4

Muy Buena

5

Excelente

2. ¿En la escala del 1 al 5 el prototipo de Aplicación tiene una presentación la cual es intuitiva y amigable?

1

Muy Mala

2

Mala

3

Buena

4

Muy Buena

5

Excelente

3. ¿Usted pudo visualizar la información y fotografía del sitio o actividad?

SI

NO

4. ¿Usted pudo visualizar en un mapa como poder llegar al sitio escogido?

SI

NO

5. ¿Usted pudo buscar un sitio o actividad?

SI

NO

6. ¿Usted pudo ordenar los sitios según sus precios de forma ascendente y descendente?

SI

NO

7. ¿Usted pudo ordenar los sitios de forma alfabética A-Z y Z-A?

SI

NO

8. ¿Usted pudo agregar una calificación al sitio o actividad?

SI

NO

9. ¿Usted pudo acceder a la web del sitio o actividad?

SI

NO

10. ¿Si usted pudiera añadir algo a la aplicación que le añadiría?

- Realizar reserva de Hoteles
- Realizar reserva en Restaurantes
- Itinerarios de transporte público
- Mostrar sitios o actividades turísticas cercanos a la ubicación del usuario
- Otros

Gracias por su colaboración

Las encuestas realizadas están adjuntas en el ANEXO D.

A continuación, se muestra los resultados de cada una de las preguntas realizadas en las encuestas a 10 usuarios administradores y a 10 usuarios turistas.

3.4.1 RESPUESTAS A ENCUESTAS USUARIOS ADMINISTRADORES.

En la Tabla 3.27 se muestra el resultado de la Pregunta 1 de la encuesta realizada a los usuarios administradores. En la cual se puede ver que el 80% de los usuarios Administradores indican que el prototipo de aplicación es excelente

Tabla 3.27 Respuestas a la Pregunta Uno Administradores

	Pregunta 1	Calificaciones	Valores	Porcentaje
Usuario Administrador	En la escala del 1 al 5 que calificación le pondría al prototipo de aplicación móvil turística	Muy Mala (1)	0	0%
		Mala (2)	0	0%
		Buena (3)	0	0%
		Muy Buena (4)	2	20%
		Excelente (5)	8	80%

En la Tabla 3.28 se muestra los resultados de la pregunta dos realizada tanto a los usuarios Administradores como a los usuarios turistas, con un total de 20 personas las que respondieron esta pregunta. Los resultados alcanzados son que un 80% de los encuestados consideran que es excelente, un 15% manifiestan en la encuesta que es muy buena y un 5% expresan que es buena, la presentación.

Tabla 3.28 Resultados Pregunta Dos Administradores.

Usuario Administrador / Turista	Pregunta 2	Calificaciones	Valores	Porcentaje
	¿En la escala del 1 al 5 el prototipo de aplicación tiene una presentación la cual es intuitiva y amigable?	Muy Mala (1)	0	0%
Mala (2)		0	0%	
Buena (3)		1	5%	
Muy Buena (4)		3	15%	
Excelente (5)		16	80%	

La Tabla 3.29 muestra los resultados de las preguntas tres, cuatro, cinco y seis realizadas a los usuarios administradores, donde las respuestas son favorables lo que significa que han podido realizar las operaciones descritas.

Tabla 3.29 Respuesta Preguntas Usuario Administrador

Usuario Administrador	Preguntas	Respuestas	
		SI	No
Usuario Administrador	3. ¿Usted pudo autenticarse como administrador?	10	0
	4. ¿Usted pudo crear una nueva categoría, sitio, o un nuevo usuario?	10	0
	5. ¿Usted pudo editar una categoría, sitio, o un perfil de usuario?	10	0
	6. ¿Usted pudo eliminar una categoría, sitio, o perfil de usuario?	10	0

3.4.2 RESPUESTAS A ENCUESTAS USUARIOS TURISTAS.

Para los usuarios Turistas también se tiene la misma pregunta 1 teniendo una respuesta excelente del 90% de las personas encuestadas. En la Tabla 3.30 se muestra los resultados. Teniendo una conclusión que de veinte personas encuestadas diecisiete dieron una calificación excelente y tres una calificación muy buena lo que en porcentaje es 85% de las personas califican con excelente y el 15% como muy buena.

Tabla 3.30 Respuesta Pregunta Uno Usuario Turista

	Pregunta 1	Calificaciones	Valores	Porcentaje
	Usuario Turista	En la escala del 1 al 5 que calificación le pondría al prototipo de aplicación móvil turística	Muy Mala (1)	0
Mala (2)			0	0%
Buena (3)			0	0%
Muy Buena (4)			1	10%
Excelente (5)			9	90%

En la Tabla 3.31 se muestran los resultados de las demás preguntas realizadas a los usuarios turistas, donde las respuestas han sido positivas lo que refleja que se pudo realizar todas las operaciones descritas.

Tabla 3.31 Respuestas a las Preguntas del Usuario Turista

	Preguntas	Respuestas	
		SI	No
Usuario Turista	3. ¿Usted pudo visualizar la información y fotografía del sitio o actividad?	10	0
	4. ¿Usted pudo visualizar en un mapa como poder llegar al sitio escogido?	10	0
	5. ¿Usted pudo buscar un sitio o actividad?	10	0
	6. ¿Usted pudo ordenar los sitios según sus precios de forma ascendente y descendente?	10	0
	7. ¿Usted pudo ordenar los sitios de forma alfabética A-Z y Z-A?	10	0
	8. ¿Usted pudo agregar una calificación al sitio o lugar?	10	0
	9. ¿Usted pudo acceder a la web del sitio o lugar?	10	0

En la Tabla 3.32 se muestran los resultados de la pregunta diez en la encuesta para los turistas y siete en la encuesta de administradores, la misma que está relacionada con lo que se podría mejorar en la aplicación en trabajos futuros, se ha tabulado las 20 respuestas, siendo la más votada, realizar reserva de hoteles con dieciocho votos que representan el 90%, seguida de, itinerarios de transporte público con dieciséis votos que representan el 80%, luego se tiene, reserva de restaurantes con doce votos que representa el 60%, también se tiene, mostrar sitios o actividades turísticas cercanos a la ubicación del usuario con ocho votos que representa el 40% y por último la opción, otros con 4 votos que representa el 20%.

Tabla 3.32 Respuesta Pregunta Usuario Turista y Administrador

	Pregunta	Opciones	Valores	Porcentaje
Usuario Administrador / Turista	¿Si usted pudiera añadir algo a la aplicación que le añadiría?	Realizar reserva de hoteles	18	90%
		Realizar reserva en restaurantes	12	60%
		Itinerarios de transporte público	16	80%
		Mostrar sitios o actividades turísticas cercanos a la ubicación del usuario	8	40%
		Otros	4	20%

3.5 TABLERO FINAL KANBAN

En la Tabla 3.33 se muestra las tareas realizadas en el tablero Kanban, donde se observa que todas las tareas están finalizadas las mismas que fueron definidas para la realización de este Proyecto de Titulación. Las tareas de todas las fases tanto diseño, implementación y pruebas están en la columna de Tareas finalizadas mostrando el termino de las mismas.

Tabla 3.33 Tablero Final Kanban

Tablero Kanban			
Fases	Tareas por Realizar	Tareas en Proceso	Tareas Finalizadas
Fase de Diseño			Entrevista con los funcionarios de la DTR para especificar la situación actual del manejo de la información turística del cantón Baños
			Obtener los requerimientos funcionales
			Obtener los requerimientos no funcionales
			Elaborar los casos de uso de todos los módulos
			Elaborar el diagrama de clases de los todos los módulos
			Elaborar diagrama relacional de la base de datos
			Definir las vistas del prototipo de aplicación móvil
Fase de Implementación			Creación de una instancia EC2 en AWS
			Instalación y creación de la base de datos
			Se programará el módulo principal el mismo que implementará el servicio web
			Se codificará el módulo cliente el cual consta del prototipo de aplicación móvil de acuerdo a los requerimientos obtenidos
			Se codificará el módulo de administración
Fase de Pruebas			Se realizará la validación de los requerimientos funcionales y no funcionales
			Se realizará pruebas de funcionamiento del prototipo

4. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIONES

- El prototipo de aplicación móvil desarrollado permite al turista que desea visitar el cantón Baños conocer las bondades que ofrece el cantón. Es decir, ayuda al turista a planificar su viaje o su día de estancia según su presupuesto, permitiéndole ver fotos del sitio de interés que quiere visitar, conocer la ubicación del mismo y mostrándole una ruta de cómo llegar.
- El presente Proyecto de Titulación se elaboró utilizando diferentes herramientas del área del diseño y desarrollo de software tales como: Arquitectura de capas, Cloud Computing, Android Studio, MySQL, WampServer, entre otras. Estas tecnologías han permitido fortalecer los conocimientos en programación, base de datos, y servicios web.
- Para la elaboración del presente Proyecto de Titulación se utilizó la metodología Kanban lo que permitió mantener ordenadas las tareas a realizarse. Kanban ha ayudado en el proceso de desarrollo, pues ha delimitado el trabajo en cada fase establecida. Además, esta metodología brinda flexibilidad ya que, se agregaron tareas, no previstas originalmente en el desarrollo del presente prototipo, permitiendo así evitar retrasos y que el trabajo sea ágil.
- En la etapa de diseño, se ideó una solución que permita ofrecer el cumplimiento de todos los requerimientos planteados. En este proceso se agruparon cada uno de estos requerimientos en módulos según el tipo de actividades o funciones que se realizan. Por ejemplo, el módulo de almacenamiento engloba todas las actividades que se relacionan con el manejo de la información de la aplicación dentro de la base datos. De esta manera, se ha logrado organizar las funcionalidades que se deben desarrollar en la etapa de implementación.
- En su implementación el prototipo cumple con todos los requisitos funcionales previstos. De forma concreta, a través de este prototipo el usuario tiene la posibilidad de visualizar la información detallada tanto de sitios o actividades que se encuentran separados en categorías como: “¿Dónde Comer?”, “¿Dónde Alojarse?”, “Deportes Extremos”, entre otros. Además, se permite buscar los sitios, y ordenarlos de acuerdo con el precio u ordenarlos alfabéticamente. Así también la aplicación para ayudar al turismo en el cantón Baños cumple con todos los

requerimientos no funcionales contemplados. Esto es: presentación, disponibilidad, escalabilidad y autenticación.

- Las pruebas realizadas al prototipo de aplicación móvil turística ratifican que se cumplió con todos los requerimientos funcionales y no funcionales establecidos, ya que tanto los usuarios que participaron como turistas y como administradores, contestaron de forma positiva, lo que refleja que 100% de los encuestados pudieron realizar los procesos establecidos.
- Los resultados de las encuestas realizadas muestran que el 80% de los usuarios del prototipo de aplicación están de acuerdo que es intuitiva y amigable, el 15% dijo que era muy buena con respecto a este punto y un 5% que es buena y no se tuvo respuestas negativas. Esto confirma que el prototipo diseñado cumple las expectativas del usuario.
- En las encuestas realizadas se tuvo una pregunta relacionada a mejorar el prototipo en un futuro y el 90% de los encuestados dijeron que se debería implementar una reserva de hoteles directamente desde la aplicación, lo que muestra que los turistas en la actualidad presentan gran interés en el uso de esta clase de aplicaciones que facilitan la planificación de sus viajes.

4.2 RECOMENDACIONES

- Se recomienda tener reuniones periódicas con el cliente que ha solicitado el desarrollo del software, lo cual permite tener una idea clara de lo que el cliente requiere en cada etapa de desarrollo. De esta manera se evita conflictos con el cliente insatisfecho con el software; pues, durante el desarrollo se pueden pedir modificaciones que si no se gestionan retrasarán en la entrega del software. Las modificaciones pueden incluir cambios en la base de datos, cambios en el diseño de las clases o incluso cambios que planteen regresar a las etapas iniciales de desarrollo. Por ello en el desarrollo de este proyecto de titulación se tuvo contacto permanente con los involucrados.
- Es recomendable escoger una buena metodología de desarrollo, ya que esta acompaña durante todo el proceso y guía las tareas por realizar en cada una de las fases. De esta forma, se permite llevar de una manera ordenada el desarrollo del proyecto. El uso de la metodología Kanban en este proyecto de titulación facilito enormemente el proceso de desarrollo.

- El usuario que utiliza el prototipo de aplicación debe ser informado de manera adecuada del éxito o fracaso de los procesos realizados como: buscar, ingresar, editar, eliminar, etc. Por ello, es recomendable que el desarrollador pueda implementar la gestión de errores y excepciones. Una buena práctica es mostrar a los usuarios cuadros de dialogo con opciones que ayuden a la validación de que la información o lo requerido es correcto o erróneo.
- Se recomienda utilizar los servicios de la nube ya que se pueden actualizar de forma inmediata las mejores prestaciones en cuanto a capacidad de disco duro, procesador, memoria RAM, ancho de banda. Gracias a esta escalabilidad, un servicio de consultas como la aplicación turística desarrollada en este proyecto puede soportar un número variable de usuarios sin desperdicio de recursos.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] Sinmapa. (18, Aug,2015). Guía de viaje: Baños de Agua Santa [En línea]. Disponible en: <https://www.sinmapa.net/guia-de-viaje-banos-de-agua-santa/>
- [2] GAD Baños. (24, Sep, 2013). Historia de Baños[En línea]. Disponible en: <http://www.municipiobanos.gob.ec/banos/index.php/es/features/historia>
- [3] Ministerio de Turismo. (27, Jun, 2014). Campaña All You Need is Ecuador presenta cifras record [En línea]. Disponible en: <https://www.turismo.gob.ec/campana-all-you-need-is-ecuador-presenta-cifras-record/>
- [4] Gilibets L. (31, Jul, 2013). Que es la metodología Kanban y como utilizarla. Innovation & Entrepreneurship Business School. [En línea]. Disponible en: <https://www.iebschool.com/blog/metodologia-kanban-agile-scrum/>
- [5] Álvarez Caules C. (14, Jun, 2013). Introducción a Servicios REST. Arquitectura Java. [En línea]. Disponible en: <https://www.arquitecturajava.com/servicios-rest/>
- [6] Lizand V. (6, Jul, 2015). Base de Datos. EcuRed. [En línea]. Disponible en: https://www.ecured.cu/Bases_de_datos
- [7] resGet. La Sociedad Civil de Baños: Datos Generales, Asociatividad, Organizaciones y la Esfera Pública. FLACSO. [En línea]. Disponible en: <https://biblio.flacsoandes.edu.ec/catalog/resGet.php?resId=13627>
- [8] Chávez, R,. Redacción Regional Centro, (25, Ene, 2015). Un millón de turistas al año visitan Baños de Agua Santa (Infografía). El Telégrafo. [En línea]. Disponible en: <https://www.eltelegrafo.com.ec/noticias/regional/1/un-millon-de-turistas-al-ano-visitabanos-de-agua-santa-infografia>
- [9] Jiménez Cordero, M. A., García Coello, E. A. (5, Mar, 2015). Aplicación móvil celular para incentivar el turismo urbano en Guayaquil. Guayaquil. Universidad Politécnica Salesiana. Ingeniería en Sistemas. [En línea]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/10323>
- [10] Enriquez, J. G. ,Casas ,S. I.(2013).Usabilidad en aplicaciones móviles. Revista de informes científicos y técnicos de la universidad nacional de la Patagonia Austral. [En línea]. Disponible en: <http://secyt.unpa.edu.ar/journal/index.php/ICTUNPA/article/view/ICT-UNPA-62-2013/62>

[11] Russo H. (19, Feb, 2012). La historia de las tiendas de aplicaciones móviles. [En línea]. Disponible en: <https://geeksroom.com/2012/02/la-historia-de-las-tiendas-de-aplicaciones-moviles/58834/>

[12] Delía, L., Galdamez, N., Thomas, P., Pesado, P. (2013). Un análisis experimental de tipo de aplicaciones para dispositivos móviles. Argentina. Universidad Nacional de La Plata. Ciencias Informáticas. [En línea]. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/32397/Documento_completo.pdf?sequence=1&isAllowed=y

[13] Raj R., Tolety, S., B. (December, 2012). A study on approaches to build cross-platform mobile applications and criteria to select appropriate approach. India Conference (INDICON), Annual IEEE. [En línea]. Disponible en: https://www.researchgate.net/publication/261053178_A_study_on_approaches_to_build_cross.

[14] Cuenca E. (2017). Diseño y desarrollo de aplicaciones móvil. [En línea]. Disponible en: <https://www.solucionesip.com/servicios/aplicaciones-web-movil>

[15] Soledad B. (21, Jul, 2018). Tipos de aplicaciones móviles: nativas, webs, híbridas. [En línea]. Disponible en: <https://www.solbyte.com/blog/2014/07/21/tipos-de-aplicaciones-moviles-nativas-webs-hibridas/>

[16] Alvarado I. (15, Oct, 2018). Funcionalidades y características de las aplicaciones nativas. [En línea]. Disponible en: <https://ceroideas.es/funcionalidades-y-caracteristicas-de-las-aplicaciones-nativas/>

[17] Martinez G. (12, Sep, 2017). Desarrollo de aplicaciones móviles híbridas. [En línea]. Disponible en: <https://www.northware.mx/2017/09/12/desarrollo-de-aplicaciones-moviles-hibridas/>

[18] Rouse M. (18, Ene, 2017). Desarrollo de aplicaciones móviles híbridas. [En línea]. Disponible en: <https://searchdatacenter.techtarget.com/es/definicion/Desarrollo-de-aplicaciones-moviles-hibridas>

[19] Página Oficial de PHP. Que es PHP. PHP Group. [En línea]. Disponible en: <https://www.php.net/manual/es/intro-what-is.php>

[20] Gomez D. (24, May, 2014). Python vs PHP Aspectos a considerar en programación. BSG Institute. En línea]. Disponible en: <https://bsginstitute.com/bs-campus/blog/Python-vs-PHP-Aspectos-a-considerar-en-Programacion-1120>

- [21] Robledo, D. (2016). Desarrollo de aplicaciones para Android I. España. Secretaria General Técnica, Ministerio de Educación Cultura y Deporte. Enseñanza Informática Programación. [En línea]. Disponible en: <https://sede.educacion.gob.es/publiventa/PdfServlet?pdf=VP18117.pdf&area=E>
- [22] Gobierno de Navarra. Acércate a las TIC. España. Dirección General de Político Económica y Empresarial. [En línea]. Disponible: <https://www.navarra.es/NR/rdonlyres/48F9746B-080C-4DEA-BD95-A5B6E01797E1/315641/7Usodedispositivosmoviles.pdf>
- [23] Santa María F. (26, Feb, 2014). Android: Ventajas y Desventajas. Sataffcreativa. [En línea]. Disponible: <http://www.staffcreativa.pe/blog/android-ventajas-desventajas/>
- [24] Pérez E. (2, Ago, 2017). Android: ventajas, inconvenientes y riesgos. [En línea]. Disponible: <https://www.xatakandroid.com/roms-android/rootear-android-ventajas-inconvenientes-y-riesgos>
- [25] Developers. (25,Abril,2018). Conoce Android Studio. [En línea]. Disponible en: <https://developer.android.com/studio/intro/?hl=es-419>
- [26] Rodríguez, J.,(2016). Manual del Programador. Universidad Distrital Francisco José de Caldas. Colombia. [En línea]. Disponible en: <repository.udistrital.edu.co/bitstream/11349/4695/4/RodriguezMartinezJulyMaribel2016>
- [27]Casillas Santillán, L. A., Gibert Ginesta, M., Pérez Mora, O.(2011) Base de datos en MySQL. [En línea]. Disponible en: http://informatica.gonzalonazareno.org/plataforma/pluginfile.php/243/mod_resource/content/0/Apuntes/UOC_MySQL.pdf
- [28] Pagina oficial de AWS. Problemas conocidos y limitaciones para MySQL en Amazon RDS. [En línea]. Disponible en: https://docs.aws.amazon.com/es_es/AmazonRDS/latest/UserGuide/MySQL.KnownIssuesAndLimitations.html
- [29] Garther. Magic Quadrant for Operational Database Management Systems.[En línea]. Disponible en: <https://www.gartner.com/doc/reprints?id=1-2G2O5FC&ct=150519>.
- [30] AWS. Detalles del producto Amazon EC2 - AWS. Amazon Web Services, Inc.[En línea]. Disponible en: <https://aws.amazon.com/es/ec2/details/>.

- [31] Página Oficial de Amazon. (12, Feb, 2018) Contrato de Nivel de Servicios de Computación de Amazon. [En línea]. Disponible en: https://d1.awsstatic.com/legal/amazon-ec2-sla/Amazon_EC2_Service_Level_Agreement_-_Spanish_Translation_2018_02_12
- [32] Página oficial de IBM. IBM Knowledge Center. Servicio Web. [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SSAW57_9.0.0/com.ibm.websphere.nd.multiplatform.doc/ae/cwbs_wbs2.html
- [33] Chanchi, G.E., Campo,W.Y., Amaya, J.P., Arciniegas, J. L.(2011). Esquema de servicios para televisión digital interactiva, basados en el protocolo REST-JSON. Popayán, Colombia. Universidad del Cauca. Departamento de telemática. [En línea]. Disponible en: <https://seer.ufrgs.br/cadernosdeinformatica/article/view/v6n1p233-240/11807>
- [34] Rosa Moncayo J. (17, May, 2018). Que es REST, conoce su potencia. [En línea]. Disponible en: <https://openwebinars.net/blog/que-es-rest-conoce-su-potencia/>
- [35] Página oficial de IBM. IBM Knowledge Center. Métodos HTTP soportados por las API REST de IBM BPM. [En línea]. Disponible en: https://www.ibm.com/support/knowledgecenter/es/SSV2LR/com.ibm.wbpm.bpc.doc/topics/rdev_restapis_httpmethods.html
- [36] Álvarez C. (18, Nov, 2016). Que es REST. [En línea]. Disponible en: <https://www.arquitecturajava.com/que-es-rest/>
- [37] Página oficial de JSON. Introducción a JSON. [En línea]. Disponible en: <https://www.json.org/json-es.html>
- [38] Kotamraju J. (Jul, 2014). API de Java para procesamiento JSON: Introducción a JSON. [En línea]. Disponible en: <https://www.oracle.com/technetwork/es/articles/java/api-java-para-json-2251318-esa.html>
- [39] Crockford,D. Introducción a JSON. [En línea]. Disponible en: <http://json.org/json-es.html>
- [40] Bermejo,M. El Kanban. Cámara de Comercio de Armenia. Modulo producción multimedia. [En línea]. Disponible en: [https://www.camaraarmenia.org.co/files/Produccion_multimedia_\(Modulo_4\).pdf](https://www.camaraarmenia.org.co/files/Produccion_multimedia_(Modulo_4).pdf)
- [41] Mitra, D. Mitrani, I. Analysis of a Kanban discipline for cell coordination in production lines. Management Science 36,1548 - 1566. Institute for Operations Research and the Management (INFORMS). Linthicum, Maryland, USA.

- [42] Arango Serna, M.D., Campuzano Zapata, L.F., Zapata Cortes, J. A. (05, Nov, 2015). Mejoramiento de procesos de manufactura utilizando kanban. Medellín Colombia. Universidad de Medellín. [En línea]. Disponible en: <http://www.scielo.org.co/pdf/rium/v14n27/v14n27a14.pdf>
- [43] Acevedo, J., Urquiaga, A., Gómez, M., (2001). Gestión de la cadena de suministros. Centro de estudios de avanzada (CETA) y laboratorio de Logística y Gestión de la producción (LOGESPRO). Ciudad de La Habana. [En línea]. Disponible en: https://www.researchgate.net/publication/283054250_Gestion_Cadenas_de_Suministro_en_Cuba
- [44] López Sanchez R. Método de Jalar Flujo de Materiales Kanban. Instituto Tecnológico de Chihuahua. México. Ingeniería Industrial. [En línea]. Disponible en: https://www.academia.edu/29821158/Administracion_de_operaciones_2
- [45] Shore Labs. Kanban Tool. (2019). Metodología Kanban en Pocas Palabras. [En línea]. Disponible en: <https://kanbantool.com/es/metologia-kanban>
- [46] Sancho, A., Buhalis, D., Gallego, J., Mata, J., (2011). OTM Organización Mundial del Turismo. Introducción al Turismo. Madrid España. [En línea]. Disponible en: www.utntyh.com/wp-content/uploads/2011/09/INTRODUCCION-AL-TURISMO-OMT.pdf
- [47] Roche, J., (2018). Deloitte Touche Tohmatsu Limited ("DTTL"). Que es el Tablero Kanban. España. [En línea]. Disponible en: <https://www2.deloitte.com/es/es/pages/technology/articles/que-es-tablero-kanban.html>
- [48] Villaroel, L., (20, Ene, 2017). Requerimientos Funcionales y Requerimiento No Funcionales. [En línea]. Disponible en: <https://ingenieriadesoftwareutmachala.com/2017/01/20/requerimientos-funcionales-y-no-funcionales/>
- [49] Rojo, S., (Abril, 2012). Requerimientos No Funcionales Para Aplicaciones Web. [En línea]. Disponible en: https://www.academia.edu/28559545/Requerimientos_no_funcionales_para_aplicaciones_web
- [50] Rocha, R., (Feb, 2017). El Modelo Entidad-Relación. Universidad Nacional de Colombia. Medellín. [En línea]. Disponible en: www.medellin.unal.edu.co/~fjmoreno/bd1/ModeloERRochav8.pdf

- [51] García, F, J., García, A,. (2018). Fundamentos de la Vista de Casos de Uso. Universidad de Salamanca. Salamanca. [En línea]. Disponible en: <https://repositorio.grial.eu/bitstream/grial/1155/1/UML%20-%20Casos%20de%20uso.pdf>
- [52] Garcia, F,. Aguilar, C,. (2016). Diagramas de Clase en UML. Universidad de Burgos. [En línea]. Disponible en: <https://repositorio.grial.eu/bitstream/grial/353/1/DClase.pdf>
- [53] Arce, A,. (04, Abr,2018). Programación PHP Versión. [En línea]. Disponible en: <https://buildmedia.readthedocs.org/media/pdf/programacion-php/latest/programacion-php.pdf>
- [54] Crockford, D,. Introducción a JSON. [En línea]. Disponible en: <https://www.json.org/json-es.html>
- [55] Página Oficial de Cacao by Nulab,. Herramienta para la Creación de Diagramas. [En línea]. Disponible en: <https://cacao.com/es/signin>
- [56] Página Oficial de phpMyAdmin,. Bringing MySQL to the web. [En línea]. Disponible en: <https://phpmyadmin.net>
- [57] Página Oficial de AWS Crear cuenta. Amazon Web Services, Inc. [En línea]. Disponible en: https://portal.aws.amazon.com/billing/signup?nc2=h_ct&src=header
- [58] Página Oficial de AWS, EC2. Amazon Web Services, Inc. [En línea]. Disponible en: <https://us-east-2.console.aws.amazon.com/ec2/v2/home?region=us-east-2#home:>
- [59] Página Oficial de WampServer, Descarga del Programa. WampServer. [En línea]. Disponible en: <https://www.wampserver.com/en/>
- [60] Página Oficial de Android Studio, Descarga de Programa. Android Studio. [En línea]. Disponible en: <https://developer.android.com/studio#downloads>
- [61] Página Oficial de Notepad ++, Descarga de Programa. Notepad ++. [En línea]. Disponible en: <https://notepad-plus-plus.org/download/v7.6.6.htm>
- [62] Hincapié, L,. (2017). Metodología de gestión de mantenimiento desde una perspectiva de Confiabilidad - Disponibilidad - Mantenibilidad (CDM) para aplicación en equipos de Tecnología de la Información (TI). Universidad Nacional de Colombia. Medellín. [En línea]. Disponible en: bdigital.unal.edu.co/61301/1/98490953.2017.pdf
- [63] Página Oficial de Amazon. (12, Feb, 2018) Contrato de Nivel de Servicios de Computación de Amazon. [En línea]. Disponible en: https://d1.awsstatic.com/legal/amazon-ec2-sla/Amazon_EC2_Service_Level_Agreement_-_Spanish_Translation_2018_02_12

[64] Pagina Oficial de Amazon. Infraestructura Global. [En línea]. Disponible en: <https://aws.amazon.com/es/about-aws/global-infrastructure/>

[65] Calmels, L., Torres, L.,. Infraestructuras de servicios Orientadas a la Web, basadas en Cloud Computing. Universidad Nacional de La Palma. Argentina. [En línea]. Disponible en: www.biblioteca.unlpam.edu.ar/rdata/tesis/i_calesc252.pdf

6. ANEXOS

ANEXO A. ENCUESTAS OBTENCIÓN DE REQUERIMIENTOS

Las encuestas realizadas se encuentran en el CD adjunto a este documento

ANEXO B. CÓDIGO BASE DE DATOS

El código se encuentra en el CD adjunto a este documento

ANEXO C. CÓDIGO SERVICIO WEB

El código se encuentra en el CD adjunto a este documento

ANEXO D. CÓDIGO APLICACIÓN

El código se encuentra en el CD adjunto a este documento

ANEXO E. ENCUESTAS DE PRUEBAS

Las encuestas realizadas se encuentran en el CD adjunto a este documento

ORDEN DE EMPASTADO