



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS

COMPUTACIONALES

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN SISTEMAS COMPUTACIONALES**

TEMA:

**“DESARROLLO DE UNA APLICACIÓN PARA DIFUNDIR LA OFERTA
TURÍSTICA DEL CANTÓN IBARRA MEDIANTE DISPOSITIVOS MÓVILES”**

AUTOR: LUIS ALBERTO CISNEROS GÓMEZ

DIRECTOR: ING. DIEGO TREJO

IBARRA - ECUADOR

2015



**UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA**

AUTORIZACIÓN DE USO Y PUBLICACIÓN

A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1 IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital Institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

| DATOS DE CONTACTO | | | |
|-------------------------------|---|------------------------------------|------------|
| CÉDULA DE IDENTIDAD | | 10034443296 | |
| APELLIDOS Y NOMBRES | | CISNEROS GÓMEZ LUIS ALBERTO | |
| DIRECCIÓN | | Ibarra, Los Ceibos, Río Chimbo 556 | |
| E-MAIL | | tiko.cisneros@gmail.com | |
| TELÉFONO FIJO: | 2 951-607 | TELÉFONO CELULAR | 0987956044 |
| DATOS DE LA OBRA | | | |
| TÍTULO | "DESARROLLO DE UNA APLICACIÓN PARA DIFUNDIR LA OFERTA TURÍSTICA DEL CANTÓN IBARRA MEDIANTE DISPOSITIVOS MÓVILES." | | |
| AUTOR | CISNEROS GÓMEZ LUIS ALBERTO | | |
| FECHA | JUNIO, 2015 | | |
| PROGRAMA | <input type="checkbox"/> PREGRADO | <input type="checkbox"/> POSTGRADO | |
| TÍTULO POR EL QUE OPTA | INGENIERO EN SISTEMAS COMPUTACIONALES | | |
| DIRECTOR | ING. DIEGO TREJO | | |

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Luis Alberto Cisneros Gómez, portador de cédula de ciudadanía N 100344329-6, en calidad de autor y titular de los derechos patrimoniales de la obra o Trabajo de Grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 144.



Firma

Nombre: Luis Alberto Cisneros Gómez

C.I.: 100344329-6

Ibarra, Junio del 2015



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CONSTANCIA

El autor manifiesta que la obra objeto de la presente autorización es original y se desarrolló, sin violar los derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

A handwritten signature in blue ink, appearing to read "Luis Alberto Cisneros", is written over a horizontal line.

Firma

Nombre: Luis Alberto Cisneros Gómez

C.I.: 100344329-6

Ibarra, Junio del 2015



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, Luis Alberto Cisneros Gómez, portador de cédula de ciudadanía Nro. 100344329-6, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4, 5 y 6 en calidad de autor del Trabajo de Grado denominado: “DESARROLLO DE UNA APLICACIÓN PARA DIFUNDIR LA OFERTA TURÍSTICA DEL CANTÓN IBARRA MEDIANTE DISPOSITIVOS MÓVILES”, que ha sido desarrollada para optar el título de Ingeniería en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo de grado aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica de Norte de la ciudad de Ibarra.

Firma

Nombre: Luis Alberto Cisneros Gómez

C.I.: 100344329-6

Ibarra, Junio del 2015



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

El señor egresado Luis Alberto Cisneros Gómez, ha trabajado en el desarrollo del trabajo de grado: “DESARROLLO DE UNA APLICACIÓN PARA DIFUNDIR LA OFERTA TURÍSTICA DEL CANTÓN IBARRA MEDIANTE DISPOSITIVOS MÓVILES”, previo a la obtención del Título de Ingeniero en Sistemas Computacionales, realizada con interés profesional y responsabilidad, lo cual certifico en honor a la verdad.

Ing. Diego Trejo

Director de Tesis

CERTIFICADO

La Dirección de Gestión de Desarrollo Turístico del Gobierno Autónomo Descentralizado Municipal San Miguel de Ibarra, expresa un sincero agradecimiento a la Universidad Técnica del Norte de la ciudad de Ibarra, por aprobar el desarrollo del presente trabajo de grado, el cual beneficia directamente a nuestro cantón Ibarra.

A la vez certifica que el señor Luis Alberto Cisneros Gómez, estudiante de la Escuela de Ingeniería en Sistemas Computacionales de la U.T.N., ha trabajado en el desarrollo del presente trabajo, apoyado en la información turística y fotografías pertenecientes a esta Dependencia Municipal. Que el desarrollo de la aplicación para difundir la oferta turística del cantón Ibarra mediante dispositivos móviles, está funcionando y disponible para su descarga e instalación en los smartphones; la misma que permitirá una mejor difusión de la oferta turística del cantón Ibarra, a través del Internet.

Ibarra, 10 de abril 2015



Ing. Ana Gabriela Merizalde E.

C.C. 1003628375

DIRECTORA

**GESTIÓN DE DESARROLLO TURÍSTICO
GAD-IBARRA**



DEDICATORIA

Quiero dedicar este trabajo de grado a mi madre *Verónica Gómez*, quién ha sido un ejemplo de estudio, trabajo, esfuerzo y superación durante toda mi vida. Ella me ha enseñado a superar las adversidades sin decaer en el intento. Me ha inculcado con amor todos los valores, virtudes y principios que poseo. Ha confiado siempre en mis capacidades y me ha alentado en el proceso de ejecución de mis metas.

Gracias mamá por confiar en mí, darme fortaleza y brindarme la oportunidad de alcanzar un peldaño más dentro de la escalera de mi vida.

Luis Alberto Cisneros Gómez

AGRADECIMIENTO

Quiero agradecer primeramente a Dios por brindarme inteligencia, fortaleza e iluminar el camino de mi carrera.

A mi madre, Kevin, mi abuelita y mis hermanas; quienes me han brindado su afecto incondicional, y siempre han estado apoyándome para continuar adelante en los momentos difíciles de mi vida.

A la Universidad Técnica del Norte por ofrecerme un espacio donde pude realizar mis estudios universitarios.

Al director de mi tesis, el Ing. Diego Trejo, quién durante me ha colaborado en el desarrollo de este trabajo.

A todos los docentes que han sido partícipes de toda mi formación académica, entregándome sus conocimientos y experiencias.

A mis amigos/as con quienes compartí experiencias, opiniones, confidencias, festejos y la carrera universitaria.

A todos ellos, muchísimas gracias.

Luis Alberto Cisneros Gómez

RESUMEN

El presente trabajo de grado consiste en el desarrollo e implementación de una aplicación móvil para la difusión de atractivos turísticos de Ibarra, brindando al turista la información suficiente para incitarlo a visitar la ciudad.

En el primer capítulo habla sobre la problemática presente, con la intención de definir la solución más viable.

En el segundo capítulo pretende informar al lector de los temas informáticos relacionados con el desarrollo del trabajo de grado, con la finalidad de comprender el mismo.

El tercer capítulo explica la oferta turística que posee la ciudad de Ibarra, además da a conocer la información de los procesos que realiza el Departamento de turismo del GAD-I.

En el cuarto capítulo es donde se ejecuta la metodología de desarrollo de software XP y se pone en práctica cada una de sus fases.

En el quinto capítulo se redactan las conclusiones y recomendaciones del aplicativo desarrollado, y también se realiza el análisis de impacto dentro de varias áreas.

SUMMARY

This thesis project is about the development and implementation of a mobile app for the dissemination of the tourist attractions in Ibarra, providing tourist complete information to invite him to visit the city.

The first chapter discusses the issue with the intention of defining the most viable solution.

The second chapter informs the reader of all issues related to development of the project, in order to understand it.

The third chapter explains the tourism of Ibarra and also discloses all the processes executed by the Department of Tourism of GAD-I.

The fourth chapter is where the methodology of software XP runs and is put into practice for each of its phases.

In the fifth chapter are the conclusions and recommendations about development which are written and also the impact analysis that is performed in several areas.

ÍNDICE DE CONTENIDO

| | |
|--|------|
| AUTORIZACIÓN DE USO Y PUBLICACIÓN | i |
| CONSTANCIA | iii |
| CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE | iv |
| CERTIFICACIÓN..... | v |
| DEDICATORIA | vii |
| AGRADECIMIENTO | viii |
| RESUMEN..... | ix |
| SUMMARY | x |
| ÍNDICE DE CONTENIDO | xi |
| ÍNDICE DE TABLAS..... | xvi |
| ÍNDICE DE FIGURAS..... | xxi |
| CAPÍTULO I..... | 1 |
| 1 EL PROBLEMA..... | 1 |
| 1.1 ANTECEDENTES | 1 |
| 1.2 DEFINICIÓN DEL PROBLEMA..... | 2 |
| 1.3 OBJETIVOS | 3 |
| 1.3.1 Objetivo General | 3 |
| 1.3.2 Objetivos Específicos..... | 3 |
| 1.4 JUSTIFICACIÓN | 3 |
| 1.5 ALCANCE | 5 |
| 1.5.1 Metodología de desarrollo..... | 5 |
| 1.5.2 Sistema Administrador | 5 |
| 1.5.3 Contenido de la App..... | 6 |
| 1.5.4 Arquitectura de la App..... | 6 |

| | |
|---|----|
| CAPÍTULO II..... | 9 |
| 2 MARCO TEÓRICO | 9 |
| 2.1 SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVILES | 9 |
| 2.1.1 Android OS..... | 10 |
| 2.1.2 iOS..... | 18 |
| 2.1.3 Windows Phone | 23 |
| 2.1.4 Symbian | 27 |
| 2.1.5 Firefox O.S..... | 31 |
| 2.1.6 Ubuntu Touch..... | 37 |
| 2.1.7 BlackBerry OS..... | 38 |
| 2.2 APLICACIONES MÓVILES | 40 |
| 2.2.1 Definición | 40 |
| 2.2.2 Utilidades de las aplicaciones | 41 |
| 2.2.3 Tipos de soluciones | 44 |
| 2.2.4 Tipos de aplicaciones según su desarrollo..... | 46 |
| 2.2.5 Distribución | 50 |
| 2.2.6 Aplicaciones de pago y gratuitas..... | 51 |
| 2.3 SERVICIOS WEB..... | 53 |
| 2.3.1 Representational State Transfer (REST)..... | 53 |
| 2.3.2 SOAP y REST | 55 |
| 2.4 JSON..... | 56 |
| 2.4.1 Sintaxis..... | 57 |
| 2.4.2 Diferencias con XML | 60 |
| 2.5 FRAMEWORK JQUERY MOBILE..... | 60 |
| 2.5.1 ¿Por qué usar jQuery Mobile? | 61 |
| 2.5.2 Funcionamiento..... | 62 |

| | | |
|-------|--|----|
| 2.5.3 | Características | 63 |
| 2.5.4 | Contenido de la librería | 63 |
| 2.5.5 | Soporte de plataformas | 64 |
| 2.5.6 | Documentación | 66 |
| 2.5.7 | Estructura básica | 68 |
| 2.6 | PLATAFORMA APACHE CORDOVA | 69 |
| 2.6.1 | Funcionamiento..... | 70 |
| 2.6.2 | Arquitectura..... | 71 |
| 2.6.3 | Phonegap y Apache Cordova | 72 |
| 2.6.4 | Componentes Básicos | 73 |
| 2.6.5 | Plugins | 74 |
| 2.6.6 | Soporte de la plataforma | 75 |
| 2.7 | OPENSIFT | 76 |
| 2.7.1 | OpenShift Runtime Environments | 77 |
| 2.7.2 | Bases de datos de OpenShift..... | 77 |
| 2.7.3 | Versiones de OpenShift | 78 |
| 2.7.4 | Planes de Openshift Online..... | 79 |
| 2.8 | METODOLOGÍA XP | 80 |
| 2.8.1 | Agile Manifesto..... | 81 |
| 2.8.2 | Bases de la programación extrema | 84 |
| 2.8.3 | Prácticas | 85 |
| 2.8.4 | Roles..... | 89 |
| 2.8.5 | Ciclo de vida..... | 90 |
| | CAPÍTULO III..... | 93 |
| 3 | TURISMO EN IBARRA | 93 |
| 3.1 | ANTECEDENTES | 93 |

| | | |
|-------|--|-----|
| 3.1.1 | Ibarra..... | 93 |
| 3.1.2 | Historia de Ibarra..... | 96 |
| 3.2 | TURISMO EN EL CANTÓN | 98 |
| 3.2.1 | Concepto de turismo | 98 |
| 3.2.2 | Desarrollo turístico | 98 |
| 3.2.3 | Turismo de Aventura..... | 99 |
| 3.2.4 | Turismo comunitario..... | 100 |
| 3.2.5 | Servicios Turísticos | 102 |
| 3.3 | DEPARTAMENTO DE TURISMO GAD-I | 103 |
| 3.3.1 | Misión y Visión | 103 |
| 3.3.2 | Misión..... | 103 |
| 3.3.3 | Visión | 104 |
| 3.3.4 | Objetivos departamentales..... | 104 |
| 3.3.5 | Organigrama | 105 |
| 3.3.6 | Procesos | 105 |
| | CAPÍTULO IV | 115 |
| 4 | METODOLOGÍA DE DESARROLLO | 115 |
| 4.1 | FASE DE EXPLORACIÓN | 115 |
| 4.1.1 | Historias de Usuario..... | 115 |
| 4.2 | FASE DE PLANIFICACIÓN..... | 122 |
| 4.2.1 | Plan de entregas | 122 |
| 4.2.2 | Módulos del aplicativo | 124 |
| 4.2.3 | Planificación: Iteración I | 125 |
| 4.2.4 | Planificación: Iteración II | 141 |
| 4.3 | FASE DE DISEÑO | 150 |
| 4.3.1 | Arquitectura del sistema administrador | 150 |

| | |
|--|------------|
| 4.3.2 Diagrama Entidad-Relación | 151 |
| 4.3.3 Funcionalidad del aplicativo | 152 |
| 4.3.4 Diseño de la aplicación móvil | 153 |
| 4.4 FASE DE PRUEBAS | 155 |
| 4.4.1 Pruebas Iteración I | 155 |
| 4.4.2 Pruebas Iteración II | 180 |
| CAPÍTULO V | 187 |
| 5 CONCLUSIONES, RECOMENDACIONES Y ANÁLISIS DE IMPACTO | 187 |
| 5.1 Conclusiones..... | 187 |
| 5.2 Recomendaciones..... | 188 |
| 5.3 Análisis de impacto | 190 |
| 5.3.1 Impacto productivo..... | 190 |
| 5.3.2 Impacto económico..... | 191 |
| 5.3.3 Impacto de innovación..... | 192 |
| 5.3.4 Impacto humano..... | 192 |
| 5.3.5 Impacto social | 193 |
| 5.3.6 Impacto turístico..... | 193 |
| 5.3.7 Impacto ambiental..... | 194 |
| BIBLIOGRAFÍA..... | 195 |
| ANEXOS..... | 199 |

ÍNDICE DE TABLAS

| | |
|---|-----|
| Tabla 2-1. Recopilación de versiones del sistema operativo Andriod | 18 |
| Tabla 2-2. Recopilación de versiones del sistema operativo iOS | 23 |
| Tabla 2-3. Versiones del sistema operativo Windows Phone | 26 |
| Tabla 2-4. Comparativa entre XML y JSON..... | 60 |
| Tabla 2-5. OpenShift Online Pricing..... | 80 |
| Tabla 3-1. Roles y responsabilidades para el proceso de actualización y control de catastro turístico | 105 |
| Tabla 3-2. Roles y responsabilidades para el proceso de entrega de información a los ciudadanos..... | 107 |
| Tabla 3-4. Roles y responsabilidades para el proceso de señalética turística externa e interna | 108 |
| Tabla 3-5. Roles y responsabilidades para el proceso de gestión de promoción y marketing turístico | 109 |
| Tabla 3-6. Roles y responsabilidades para el proceso de capacitación turística | 110 |
| Tabla 3-7. Roles y responsabilidades para el proceso de administración de la guianza en museo Atahualpa..... | 111 |
| Tabla 3-8. Roles y responsabilidades para el proceso de ferias turísticas.. | 112 |
| Tabla 3-3. Roles y responsabilidades para el proceso de entrega de licencia anual de funcionamiento turístico | 113 |
| Tabla 4-1. Historia Ingreso al Sistema | 116 |
| Tabla 4-2. Historia Gestión de Usuarios | 117 |
| Tabla 4-3. Historia Gestión de Categorías | 117 |

| | |
|---|-----|
| Tabla 4-4 Historia Gestión de Información general | 118 |
| Tabla 4-5 Historia Gestión de Atractivos..... | 119 |
| Tabla 4-6 Historia Acceso a Datos..... | 119 |
| Tabla 4-7 Historia Ingreso a la aplicación | 120 |
| Tabla 4-8 Historia Visualización de categorías | 120 |
| Tabla 4-9 Historia Visualización de Información de la ciudad | 121 |
| Tabla 4-10 Historia Visualización de Atractivos | 122 |
| Tabla 4-11 Plan de Entregas | 123 |
| Tabla 4-12 Cronograma Iteración I | 125 |
| Tabla 4-13 Tareas Historia 1..... | 125 |
| Tabla 4-14 Tarea: Configuración del entorno de programación..... | 126 |
| Tabla 4-15 Tarea: Creación del script de la base de datos..... | 126 |
| Tabla 4-16 Tarea: Diseño de la interfaz de login y ventanas de usuario | 127 |
| Tabla 4-17 Tarea: Mapeo de entidades | 127 |
| Tabla 4-18 Tarea Creación de clases JAVA | 128 |
| Tabla 4-19 Tarea Implementación de funcionalidad | 128 |
| Tabla 4-20 Tareas Historia 2..... | 129 |
| Tabla 4-21 Tarea Historia Diseño de la interfaz de gestión de usuarios..... | 129 |
| Tabla 4-22 Tarea Creación de clases JAVA | 130 |
| Tabla 4-23 Tarea Implementación de funcionalidad | 130 |
| Tabla 4-24 Tareas Historia 3..... | 131 |
| Tabla 4-25 Tarea: Creación del script de la base de datos..... | 131 |
| Tabla 4-26 Tarea: Diseño de la interfaz de gestión de categorías..... | 132 |
| Tabla 4-27 Tarea: Mapeo de entidades | 132 |

| | |
|---|-----|
| Tabla 4-28 Tarea Creación de clases JAVA | 133 |
| Tabla 4-29 Tarea Implementación de funcionalidad | 133 |
| Tabla 4-30 Tareas Historia 4..... | 133 |
| Tabla 4-31 Tarea: Creación del script de la base de datos..... | 134 |
| Tabla 4-32 Tarea: Diseño de la interfaz de gestión de información general | 134 |
| Tabla 4-33 Tarea: Mapeo de entidades | 135 |
| Tabla 4-34 Tarea Creación de clases JAVA | 135 |
| Tabla 4-35 Tarea Implementación de funcionalidad | 136 |
| Tabla 4-36 Tareas Historia 5..... | 136 |
| Tabla 4-37 Tarea: Creación del script de la base de datos..... | 137 |
| Tabla 4-38 Tarea: Diseño de la interfaz de gestión de atractivos | 137 |
| Tabla 4-39 Tarea: Mapeo de entidades | 138 |
| Tabla 4-40 Tarea Creación de clases JAVA | 138 |
| Tabla 4-41 Tarea Implementación de funcionalidad | 139 |
| Tabla 4-42 Tareas Historia 6..... | 139 |
| Tabla 4-43 Tarea Configuración del entorno en la nube | 139 |
| Tabla 4-44 Tarea Estructurado del objeto JSON | 140 |
| Tabla 4-45 Creación del servicio REST | 140 |
| Tabla 4-46 Cronograma Iteración II | 141 |
| Tabla 4-47 Tareas Historia 7..... | 141 |
| Tabla 4-48 Tarea Configuración del entorno de programación..... | 142 |
| Tabla 4-49 Tarea Diseño de la interfaz de inicio y selección de idioma..... | 142 |
| Tabla 4-50 Tarea Implementación de funcionalidad en la App | 143 |
| Tabla 4-51 Tareas Historia 8..... | 143 |

| | |
|---|-----|
| Tabla 4-52 Tarea Diseño del menú de categorías y pantalla de lista de sitios..... | 144 |
| Tabla 4-53 Tarea Creación de métodos JavaScript..... | 144 |
| Tabla 4-54 Tarea Implementación de funcionalidad en la App | 145 |
| Tabla 4-55 Tareas Historia 9..... | 145 |
| Tabla 4-56 Tarea Diseño de pantalla de home y visualización de la información general | 146 |
| Tabla 4-57 Tarea Creación de métodos JavaScript..... | 147 |
| Tabla 4-58 Tarea Implementación de funcionalidad en la App | 147 |
| Tabla 4-59 Tareas Historia 10..... | 147 |
| Tabla 4-60 Tarea Diseño de la visualización de la información de los atractivos..... | 148 |
| Tabla 4-61 Tarea Creación de métodos JavaScript..... | 148 |
| Tabla 4-62 Tarea Implementación de Google Maps | 149 |
| Tabla 4-63 Tarea Implementación de funcionalidad en la App | 149 |
| Tabla 4-64 Prueba: Ingreso correcto de datos..... | 155 |
| Tabla 4-65 Prueba: Ingreso incorrecto de datos | 156 |
| Tabla 4-66 Prueba: Registro correcto de usuario | 158 |
| Tabla 4-67 Prueba: Registro incorrecto de usuario..... | 159 |
| Tabla 4-68 Prueba: Modificación de usuario..... | 160 |
| Tabla 4-69 Prueba: Desactivación de usuario | 161 |
| Tabla 4-70 Prueba: Ingreso correcto de categoría..... | 163 |
| Tabla 4-71 Prueba: Ingreso incorrecto de categoría..... | 164 |
| Tabla 4-72 Prueba: Modificación de categoría..... | 165 |

| | |
|--|-----|
| Tabla 4-73 Prueba: Eliminación de categoría | 166 |
| Tabla 4-74 Prueba: Ingreso correcto de información general | 168 |
| Tabla 4-75 Prueba: Ingreso incorrecto de información general | 169 |
| Tabla 4-76 Prueba: Modificación de información general | 170 |
| Tabla 4-77 Prueba: Eliminación de información general..... | 171 |
| Tabla 4-78 Prueba: Ingreso correcto de atractivo | 173 |
| Tabla 4-79 Prueba: Ingreso incorrecto de atractivo | 175 |
| Tabla 4-80 Prueba: Modificación de atractivo | 176 |
| Tabla 4-81 Prueba: Eliminación de atractivo..... | 178 |
| Tabla 4-82 Prueba: Perfil de usuario | 179 |
| Tabla 4-83 Prueba: Funcionamiento del servicio REST | 180 |
| Tabla 4-84 Prueba: Carga y verificación de internet | 181 |
| Tabla 4-85 Prueba: Visualización de categorías..... | 182 |
| Tabla 4-86 Prueba: Visualización de Información General | 183 |
| Tabla 4-87 Prueba: Visualización de Atractivos..... | 184 |
| Tabla 4-88 Prueba: Visualización de ubicación de atractivo | 185 |
| Tabla 5-1. Niveles de impacto..... | 190 |
| Tabla 5-2. Impacto productivo..... | 191 |
| Tabla 5-3. Impacto económico..... | 191 |
| Tabla 5-4. Impacto de innovación | 192 |
| Tabla 5-5 Impacto humano | 192 |
| Tabla 5-6 Impacto Social | 193 |
| Tabla 5-7 Impacto Turístico | 193 |
| Tabla 5-8 Impacto Ambiental | 194 |

ÍNDICE DE FIGURAS

| | |
|--|----|
| Figura 1-1. Arquitectura de la App móvil híbrida | 6 |
| Figura 2-1. Arquitectura del sistema operativo Android | 13 |
| Figura 2-2. iPhone con iOS 7..... | 18 |
| Figura 2-3. Arquitectura del sistema operativo iOS..... | 20 |
| Figura 2-4. Windows Phone en dispositivos móviles | 23 |
| Figura 2-5. Diagrama de alto nivel de Windows Phone 8 | 25 |
| Figura 2-6. Captura de pantalla de Symbian OS | 27 |
| Figura 2-7. Estructura Tecnológica de Symbian OS | 30 |
| Figura 2-8. Firefox OS en dispositivo móvil..... | 31 |
| Figura 2-9. Arquitectura de Firefox OS | 35 |
| Figura 2-10. Firefox OS bootstrapping process | 37 |
| Figura 2-11. Sistema operativo Ubuntu Touch..... | 38 |
| Figura 2-12. BlackBerry Z30 | 39 |
| Figura 2-13. Aplicación móvil de video2brain..... | 40 |
| Figura 2-14. Aprovechamiento del sistema de notificaciones. | 47 |
| Figura 2-15. Aplicación nativa y web de Facebook..... | 48 |
| Figura 2-16. Aplicación híbrida de Netflix con visualización similar en iOS y Android..... | 49 |
| Figura 2-17. Sintaxis de objetos..... | 58 |
| Figura 2-18. Sintaxis de arreglos | 58 |
| Figura 2-19. Sintaxis de valor | 59 |
| Figura 2-20. Sintaxis de cadena de caracteres..... | 59 |
| Figura 2-21. Sintaxis de arreglos | 60 |

| | |
|--|-----|
| Figura 2-22. JQuery Mobile logo..... | 61 |
| Figura 2-23. Sitio de documentación oficial de JQuery Mobile | 67 |
| Figura 2-24. Sitio del tutorial de JQuery Mobile de la W3C..... | 67 |
| Figura 2-25. Estructura inicial de JQuery Mobile | 68 |
| Figura 2-26. Estructura básica multi-página de JQuery Mobile..... | 69 |
| Figura 2-27. Logo de Apache Cordova | 69 |
| Figura 2-28. Proceso de empaquetamiento de Cordova | 70 |
| Figura 2-29. Arquitectura de una aplicación Apache Cordova | 71 |
| Figura 2-30. Soporte de Apache Cordova..... | 76 |
| Figura 2-31. OpenShift logo..... | 76 |
| Figura 2-32. Relación entre las tres versiones de OpenShift..... | 78 |
| Figura 2-33. Agile Manifesto | 82 |
| Figura 2-34. Los 12 principios creados con el manifiesto | 82 |
| Figura 2-35. Prácticas de XP | 86 |
| Figura 2-36. Ciclo de vida de la metodología XP | 91 |
| Figura 3-1. Arte de la ciudad de Ibarra 2014 - 2019 | 93 |
| Figura 3-2. Principales elevaciones montañosas..... | 95 |
| Figura 3-3. Símbolos de Ibarra | 95 |
| Figura 3-4. Retorno a Ibarra..... | 97 |
| Figura 3-5. Práctica de rafting..... | 99 |
| Figura 3-6. Comunidad indígena de Zuleta..... | 101 |
| Figura 3-7. Vista nocturna hostería “La Estelita” | 102 |
| Figura 3-8. Organigrama del Departamento de Turismo GAD-I..... | 105 |
| Figura 3-9. Diagrama de Flujo: Actualización y control de catastro turístico | 106 |

| | |
|---|-----|
| Figura 3-10. Diagrama de Flujo: Entrega de información a los ciudadanos | 107 |
| Figura 3-12. Diagrama de Flujo: Señalética turística externa e interna | 108 |
| Figura 3-13. Diagrama de Flujo: Gestión de promoción y marketing turístico | 109 |
| Figura 3-14. Diagrama de Flujo: Capacitación turística | 110 |
| Figura 3-11. Diagrama de Flujo: Entrega de licencia anual de funcionamiento turístico | 113 |
| Figura 4-1. Arquitectura del Sistema Administrador | 151 |
| Figura 4-2 Diagrama Entidad-Relación | 151 |
| Figura 4-3 Funcionalidad del aplicativo | 152 |
| Figura 4-4 Diseño de la App móvil | 154 |
| Figura 4-5. Prueba ingreso correcto de datos | 156 |
| Figura 4-6. Prueba ingreso incorrecto de datos | 157 |
| Figura 4-7. Prueba registro correcto de usuario | 158 |
| Figura 4-8. Prueba registro incorrecto de usuario | 159 |
| Figura 4-9. Prueba modificación de usuario | 160 |
| Figura 4-10. Prueba desactivación de usuario | 161 |
| Figura 4-11. Desactivación de usuario exitosa | 162 |
| Figura 4-12. Prueba ingreso correcto de categoría | 163 |
| Figura 4-13. Prueba ingreso incorrecto de categoría | 164 |
| Figura 4-14. Prueba modificación de categoría | 165 |
| Figura 4-15. Prueba eliminación de categoría | 166 |
| Figura 4-16. Mensaje de eliminación para categoría | 167 |
| Figura 4-17. Prueba ingreso correcto de información general | 168 |

| | |
|---|-----|
| Figura 4-18. Prueba ingreso incorrecto de información general | 169 |
| Figura 4-19. Prueba modificación de información general | 170 |
| Figura 4-20. Prueba eliminación de información general | 172 |
| Figura 4-21. Mensaje de eliminación para información general..... | 172 |
| Figura 4-22. Prueba ingreso correcto de atractivo | 174 |
| Figura 4-23. Prueba ingreso incorrecto de atractivo | 175 |
| Figura 4-24. Prueba modificación de atractivo | 177 |
| Figura 4-25. Prueba eliminación de atractivo | 178 |
| Figura 4-26. Prueba perfil de usuario..... | 179 |
| Figura 4-27. Prueba carga y verificación de internet..... | 181 |
| Figura 4-28. Prueba visualización de categorías | 182 |
| Figura 4-29. Prueba visualización de Información General..... | 183 |
| Figura 4-30. Prueba visualización de Atractivos | 184 |
| Figura 4-31. Prueba visualización de ubicación de atractivo | 185 |

CAPÍTULO I

1 EL PROBLEMA

Dentro de este capítulo trataremos sobre el problema presente, con la intención de definir la solución más viable para este. Este problema está relacionado con la ciudad de Ibarra dentro del ámbito del turismo.

1.1 ANTECEDENTES

Ibarra es la capital de la provincia de Imbabura, ubicada en la región andina al norte del Ecuador. Es conocida como "La Ciudad Blanca" por sus fachadas y por los asentamientos españoles. Con más de cuatrocientos años de historia y en donde coexisten variedad de culturas que la convierten en única en la provincia de Imbabura.

La ciudad de Ibarra está formada por cinco parroquias urbanas: El Sagrario, San Francisco, Caranqui, Alpachaca y la Dolorosa de Priorato; y ocho parroquias rurales: Ambuquí, Angochagua, La Carolina, La Esperanza, Lita, Salinas, San Antonio y Cochapamba; llegando a tener una superficie total de 162.22 kilómetros cuadrados.

Quién visite Ibarra disfrutará de un ambiente plácido y paisajes ensoñadores, podrá recrear la mente y la vista en sus parques, plazas, iglesias, museos, lugares tradicionales, lagunas, cascadas, cerros y deleitar su paladar con su deliciosa gastronomía.

“El Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra es la entidad que planifica, regula, ejecuta y promueve el desarrollo integral sostenible del cantón, a través de servicios eficientes, transparentes y de calidad; con la participación activa de la ciudadanía socialmente responsable a fin de lograr el buen vivir.” (GAD-Ibarra, 2014)

La Dirección de Desarrollo y Planificación Turística Municipal de Ibarra tiene la responsabilidad de impulsar el desarrollo turístico integral, como una de las actividades económicas prioritarias del cantón, para posicionarlo como un destino turístico de calidad.

1.2 DEFINICIÓN DEL PROBLEMA

Los cambios tecnológicos suscitados en estos últimos años han llevado a las personas al uso de dispositivos móviles como teléfonos inteligentes, tabletas, entre otros. El incremento de estos dispositivos permite a los usuarios acceder a un sinnúmero de características y aplicativos que estos poseen, facilitando tareas cotidianas, acceso a información y brindando momentos de relajación y diversión.

A pesar del auge de la tecnología móvil y su desarrollo dentro del país, no se ha logrado poner mucho énfasis a la investigación, para la construcción de aplicaciones informáticas en el área turística.

La Dirección de Desarrollo y Planificación Turística Municipal posee un sitio web (www.touribarra.gob.ec) por medio del cual se da a conocer la oferta

turística existente. Además dentro de sus instalaciones brinda información turística bilingüe a los visitantes y se entrega material promocional.

El Cantón Ibarra ***no dispone de una aplicación para dispositivos móviles, que le permita promocionar el turismo***; desaprovechando un recurso informático muy importante en la actualidad. Al contar con una aplicación de este tipo se logrará ampliar la oferta turística del cantón, llamando la atención de los turistas nacionales y especialmente a los extranjeros para visitar la ciudad.

1.3 OBJETIVOS

1.3.1 Objetivo General

Desarrollar una aplicación para difundir la oferta turística del cantón Ibarra mediante dispositivos móviles.

1.3.2 Objetivos Específicos

- Recopilar información sobre las aplicaciones móviles, los archivos Json, la plataforma Apache Cordova, el framework JQuery Mobile y la metodología XP.
- Conocer el desarrollo turístico del cantón Ibarra.
- Plantear la arquitectura de la App.
- Desarrollar la aplicación usando la metodología XP.

1.4 JUSTIFICACIÓN

Siendo Ibarra parte de la Provincia de Imbabura, es un cantón en el cual la actividad turística genera ingresos económicos a sus habitantes, hecho que les permite disponer de un sustento para sus familias.

La sociedad ha cambiado la forma de conectarse a la red en los últimos años, debido a la adquisición de los llamados teléfonos inteligentes o smartphones, que se encuentran disponibles a nivel mundial. El uso de los mismos ha dado lugar a un mejoramiento de la calidad de conexiones a Internet y la disminución de sus precios.

Esto ha convertido a los dispositivos móviles en uno de los principales medios de conexión a la red. En este sentido, el contar con una aplicación móvil personalizada es fundamental ya que ello implica disponer de herramientas adecuadas de promoción y venta, que permitirán conseguir un gran número de posibles clientes. Una aplicación móvil de calidad garantizará la máxima difusión de sus productos o servicios.

En el ámbito del turismo el mercado dispone de aplicaciones que contienen información relevante de los principales sitios turísticos de una ciudad o lugar específico. La mayoría de estas aplicaciones han sido realizadas por empresas reconocidas como: Despegar.com o TripAdvisor; las cuales han logrado orientar su uso en beneficio de las mismas.

La necesidad de implementar una aplicación móvil sobre la actividad turística radica en la importancia de promocionar los atractivos del Cantón Ibarra y conseguir posicionarlo como un destino turístico de calidad, a nivel nacional como internacional; esto gracias al aumento significativo del uso de teléfonos inteligentes y tabletas alrededor del mundo.

1.5 ALCANCE

El presente trabajo de grado se encuentra orientado al desarrollo tecnológico, por medio de su aplicación se contribuirá al posicionamiento de Ibarra como destino turístico de calidad y se realizará una estratégica difusión de todos los atractivos naturales y culturales que el cantón posee. Esto se logrará con la construcción de una aplicación móvil híbrida diseñada para dispositivos Android y publicada en Google Play.

1.5.1 Metodología de desarrollo

La aplicación a desarrollar se realizará mediante la metodología XP, ya que esta es una metodología ágil que se centra en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, impulsando un trabajo en equipo y orientando a un buen clima de trabajo.

Las características fundamentales de esta metodología de desarrollo de software son:

- Posee un desarrollo interactivo incremental.
- Está basada en la prueba y error.
- Tiene una integración continua del equipo de trabajo con el cliente.
- Combina las mejores prácticas para desarrollar software.

1.5.2 Sistema Administrador

Para realizar cambios de contenidos de la aplicación móvil, se creará un sencillo sistema para su administración. Este contendrá los siguientes módulos:

- **Control y acceso al sistema:** Permitirá el ingreso y control de la aplicación multiusuario.
- **Manejo de información turística:** Manejará la información perteneciente a la ciudad de Ibarra, como: clima, historia y transporte. Además permitirá el ingreso de categorías con los respectivos atractivos de la ciudad.

1.5.3 Contenido de la App

La aplicación constará de la promoción de 50 atractivos turísticos, los cuales estarán en 2 idiomas (inglés y español) para su mayor comprensión por parte de los visitantes extranjeros. Además presentará información de la descripción del atractivo, los servicios que presta y la ubicación de los mismos.

1.5.4 Arquitectura de la App

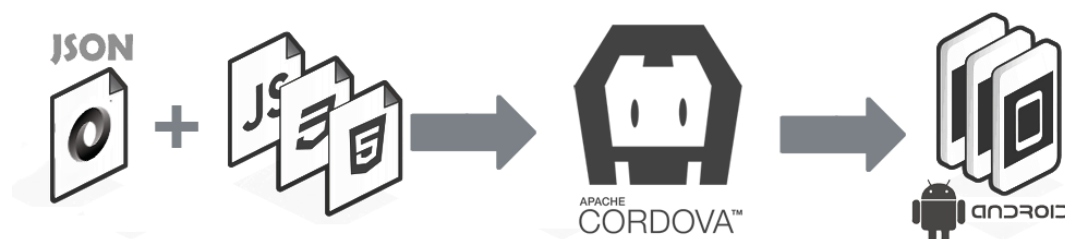


Figura 1-1. Arquitectura de la App móvil híbrida

Fuente: Propia

La aplicación móvil híbrida gestionará la lógica de negocio para comunicarse con un servicio web que devolverá una estructura JSON con la información respectiva de los atractivos turísticos. Mediante el código JavaScript de JQuery Mobile se pretende la lectura y visualización de los datos contenidos

en el archivo JSON. Además se usará la plataforma APACHE CORDOVA para lograr que la App se comporte y se vea como una aplicación nativa.

Las razones para usar CORDOVA son las siguientes:

- La plataforma proporciona una API que permite acceder a las funcionalidades nativas de los dispositivos móviles utilizando JavaScript.
- La interfaz de una aplicación desarrollada con CORDOVA se crea utilizando HTML, CSS y JavaScript; y se muestra a través de un navegador que toma el 100% del ancho y el 100% del alto del dispositivo.
- Además de la serie de funcionalidades que ofrece, CORDOVA permite crear nuevos “plugins nativos”.
- El producto final de las aplicaciones CORDOVA son un archivo binario (IPA, APK, XAP...) listo para ser distribuido en los correspondientes marketplaces.
- La arquitectura de la aplicación suele utilizar un modelo de página única donde toda la lógica de la aplicación está en la misma.

CAPÍTULO II

2 MARCO TEÓRICO

Este capítulo contiene la teoría de este trabajo de grado, y pretende informar al lector de los temas informáticos relacionados con la finalidad de comprender el desarrollo del mismo.

2.1 SISTEMAS OPERATIVOS PARA DISPOSITIVOS MÓVILES

Las computadoras, los teléfonos inteligentes y las tabletas poseen un sistema operativo, el cual es un conjunto de programas que se encarga de la gestión de todos los recursos de hardware y provee servicios a las aplicaciones que se ejecutan en el mismo.

Los sistemas operativos móviles son muy simples y se encuentran orientados a la conectividad a internet, los formatos multimedia y las diferentes maneras de introducir información dentro de ellos. La mayoría de este tipo de sistemas operativos está basada en el modelo de capas.

El modelo mencionado anteriormente posee las siguientes capas:

- **Kernel o núcleo:** Es el corazón del sistema operativo y proporciona acceso seguro a los distintos elementos que conforman el hardware del dispositivo.
- **Middleware:** Es el software encargado de asistir a una aplicación para que esta pueda interactuar con otras aplicaciones, software, redes y hardware.

- **Entorno de ejecución de aplicaciones:** Consiste en un gestor de aplicaciones y un conjunto de interfaces programables para facilitar la creación de software.
- **Interfaz de usuario:** Es el medio por el cual el usuario puede interactuar con el dispositivo mediante componentes gráficos.

Con el crecimiento y popularidad del mercado de dispositivos móviles, los sistemas operativos que funcionan dentro de ellos adquieren mucha importancia y logran funcionalidades muy similares a las que son utilizados en las computadoras.

2.1.1 Android OS

Android es un sistema operativo libre, gratuito y multiplataforma, basado en el kernel de Linux y diseñado para ser usado en dispositivos móviles como teléfonos inteligentes o tabletas, y también para relojes inteligentes, televisores, entre otros.

Este sistema operativo permite realizar aplicaciones mediante una variación del lenguaje de programación JAVA llamada DALVIK. Además proporciona las interfaces necesarias para el acceso a las funcionalidades del dispositivo de una forma sencilla.

La diferencia Android con los otros sistemas operativos es la capacidad de brindar al programador la facilidad de crear de sus propios widgets, aplicaciones, o incluso la modificación del propio sistema operativo ya que

este es de código libre y está construido en JAVA que es un lenguaje muy simple a la hora del desarrollo.

A) Reseña histórica

Según (Gironés, 2013) los orígenes de este sistema operativo son los siguientes:

Google adquiere Android Inc. en el año 2005. Se trataba de una pequeña compañía que acababa de ser creada, orientada a la producción de aplicaciones para terminales móviles. Ese mismo año empiezan a trabajar en la creación de una máquina virtual Java optimizada para móviles (Dalvik VM).

En el año 2007 se crea el consorcio Handset Alliance con el objetivo de desarrollar estándares abiertos para móviles. Está formado por Google, Intel, Texas Instruments, Motorola, T-Mobile, Samsung, Ericsson, Toshiba, Vodafone, NTT DoCoMo, Sprint Nextel y otros. Una pieza clave de los objetivos de esta alianza es promover el diseño y difusión de la plataforma Android. Sus miembros se han comprometido a publicar una parte importante de su propiedad intelectual como código abierto bajo la licencia Apache v2.0.

En noviembre del 2007 se lanza una primera versión del Android SDK. Al año siguiente aparece el primer móvil con Android (T-Mobile G1). En octubre, Google libera el código fuente de Android, principalmente bajo licencia de código abierto Apache (licencia GPL v2 para el núcleo). Ese mismo mes, se abre Android Market para la descarga de aplicaciones. En

abril del 2009, Google lanza la versión 1.5 del SDK que incorpora nuevas características como el teclado en pantalla. A finales del 2009 se lanza la versión 2.0 y durante el 2010 las versiones 2.1, 2.2 y 2.3.

Durante el año 2010, Android se consolida como uno de los sistemas operativos para móviles más utilizados, con resultados cercanos al iPhone, e incluso superando al sistema de Apple en EE.UU.

En el 2011 se lanzan la versión 3.x específica para tabletas y 4.x tanto para móviles como para tabletas. Durante este año, Android se consolida como la plataforma para móviles más importante, alcanzando una cuota de mercado superior al 50%. En 2012, Google cambia su estrategia en su tienda de descargas online, reemplazando Android Market por Google Play Store, donde en un solo portal unifica la descarga de aplicaciones como de contenidos.

B) Arquitectura

La arquitectura de Android está conformada por capas, donde cada una de ellas puede hacer uso de servicios de las capas inferiores a esta. La característica más importante es que todas las capas que conforman el sistema operativo son basadas en software libre.

El sistema operativo usa como base el kernel de Linux; pero cabe recalcar que los dos son muy diferentes ya que Android no cuenta con un sistema nativo de ventanas de Linux y no posee soporte para glibc (librería estándar de

C). Además Android agrega algunas funciones específicas para plataformas móviles.



Figura 2-1. Arquitectura del sistema operativo Android

Fuente: <http://bit.ly/1A9XA7g>

Capa del kernel (roja)

Esta capa es el corazón del sistema operativo, permite el manejo de memoria, el multiproceso, la pila de protocolos y el soporte de drivers para el dispositivo. Esto es de mucha utilidad para que no existan inconvenientes con cada uno de los fabricantes de móviles respecto a las funcionalidades de cada dispositivo.

Capa de librerías (verde)

Esta capa incluye un conjunto de librerías nativas escritas en C/C++, las cuales son usadas en varios componentes de Android y expuestas a todos los desarrolladores.

Algunas de estas librerías son: (Gironés, 2013)

- **System C library:** una derivación de la librería BSD de C estándar (libc), adaptada para dispositivos embebidos basados en Linux.
- **Media Framework:** librería basada en PacketVideo's OpenCORE; soporta codecs de reproducción y grabación de multitud de formatos de audio, vídeo e imágenes MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.
- **Surface Manager:** maneja el acceso al subsistema de representación gráfica en 2D y 3D.
- **WebKit:** soporta un moderno navegador web utilizado en el navegador Android y en la vista webview. Se trata de la misma librería que utiliza Google Chrome y Safari de Apple.
- **SGL:** motor de gráficos 2D.
- **Librerías 3D:** implementación basada en OpenGL ES 1.0 API. Las librerías utilizan el acelerador hardware 3D si está disponible, o el software altamente optimizado de proyección 3D.
- **FreeType:** fuentes en bitmap y renderizado vectorial.
- **SQLite:** potente y ligero motor de bases de datos relacionales disponible para todas las aplicaciones.
- **SSL:** proporciona servicios de encriptación Secure Socket Layer.

Android Runtime (amarilla)

Android Runtime no se considera totalmente como una capa, está basado en un set de librerías que aportan funcionalidades a las librerías basadas en el

lenguaje de programación JAVA. Además dentro de este se encuentra la máquina virtual Dalvik, que gestiona la ejecución de programas realizados en el lenguaje antes mencionado.

Cada aplicación de Android corre en su propio proceso, con su propia instancia de Dalvik; es por esto que el sistema operativo ha sido escrito de tal forma que dentro de un dispositivo puedan correr múltiples máquinas virtuales.

Entorno de aplicaciones (azul)

Esta capa es la más visible para un programador, aquí es donde se proporciona la plataforma de desarrollo de software libre para las diferentes aplicaciones. El diseño de la misma simplifica la reutilización de componentes.

La gran fortaleza del entorno de desarrollo de Android es el aprovechamiento del lenguaje de programación JAVA. Es importante decir que el SDK de Android no cubre todas las bondades que ofrece el entorno de ejecución de JAVA (JRE).

Los servicios que podemos destacar de esta capa son: (Gironés, 2013)

- **Views:** extenso conjunto de vistas (parte visual de los componentes).
- **Resource Manager:** proporciona acceso a recursos que no son en el código.
- **Activity Manager:** maneja el ciclo de vida de las aplicaciones y proporciona un sistema de navegación entre ellas.

- **Notification Manager:** permite a las aplicaciones mostrar alertas personalizadas en la barra de estado.
- **Content Providers:** mecanismo sencillo para acceder a datos de otras aplicaciones (como los contactos).

Aplicaciones (azul superior)

Este último nivel es donde se encuentran todo el conjunto de aplicaciones instaladas en un dispositivo con Android.

C) Versiones

A continuación se muestra una tabla con la información resumida de las versiones de este sistema operativo.

| Versión | Descripción |
|---|--|
| Android 1.0 Nivel de API 1 (septiembre de 2008) | Primera versión de Android. Nunca se usó correctamente. |
| Android 1.1 Nivel de API 2 (febrero 2009) | No se añadieron funcionalidades, pero se compusieron errores de la versión anterior. |
| Android 1.5 Nivel de API 3 (abril 2009, Cupcake) | Es la primera versión con algún usuario. Se incorporaron novedades como la incorporación del teclado en la pantalla, y la capacidad de grabación de audio y video. |
| Android 1.6 Nivel de API 4 (diciembre 2009, Donut) | Permite capacidades de búsqueda avanzada en todo el dispositivo. También incorpora <i>gestures</i> y <i>multi-touch</i> . Facilita que una aplicación pueda trabajar con diferentes densidades de pantalla. Existen mejoras en la aplicación cámara. |
| Android 2.0 Nivel de API 5 (octubre 2009, Éclair) | Apenas cuenta con usuarios, dado que la mayoría de fabricantes pasaron directamente de la versión 1.6 a la 2.1. Como novedades destaca la incorporación del manejo de bluetooth. |
| Android 2.1 Nivel de API 7 (enero 2010, | Se considera una actualización menor. Se destaca el reconocimiento de voz, fondos de pantalla animados, y mejora el uso del video, historial de navegación, entre |

| | |
|---|--|
| Éclair) | otros. En el paquete Webkit se incluyen métodos para la manipulación de bases de datos. |
| Android 2.2 Nivel de API 8 (mayo 2010, Froyo) | Como característica más destacada se puede indicar la mejora de velocidad de ejecución de aplicaciones. Se añaden varias mejoras con el navegador Web, soporte para Adobe Flash 10.1, la incorporación del motor de JavaScript V8 utilizado en Chrome y la conectividad Wi-Fi. |
| Android 2.3 Nivel de API 9 (diciembre 2010, Gingerbread) | Debido al éxito de Android en las nuevas tabletas, ahora soporta mayores tamaños de pantalla y resoluciones. Incorpora una nueva interfaz de usuario. Incluye soporte nativo para varias cámaras, pensando en la segunda cámara para videoconferencias. En otras novedades se destaca el soporte nativo para la telefonía sobre internet VoIP/SIP. |
| Android 3.0 Nivel de API 11 (febrero 2011, Honeycomb) | Para la mejorar la experiencia de Android se lanza esta versión 3.0 optimizada para dispositivos con pantallas grandes. La nueva interfaz de usuario ha sido rediseñada con paradigmas para la interacción, navegación y personalización. |
| Android 3.1 Nivel de API 12 (mayo 2011) | Permite el manejo de dispositivos conectados por USB (tanto host como dispositivo). Protocolo de transferencia de fotos y videos (PTP/MTP) y de tiempo real (RTP). |
| Android 3.2 Nivel de API 13 (julio 2011) | Optimizaciones para distintos tipos de tableta. Zoom compatible para aplicaciones de tamaño fijo. Sincronización multimedia desde SD. |
| Android 4.0 Nivel de API 14 (octubre 2011, Ice Cream Sandwich) | La característica más importante es que se unifican las dos versiones anteriores (2.x para teléfonos y 3.x para tabletas) en una sola compatible con cualquier tipo de dispositivo. Se introduce una nueva interfaz totalmente renovada. |
| Android 4.0.3 Nivel de API 15 (diciembre 2011) | Se introducen mejoras ligeras en algunas API's incluyendo el de redes sociales, calendario, revisor ortográfico, bases de datos, entre otras. En marzo del 2012 aparece la actualización 4.0.4. |
| Android 4.1 Nivel de API 16 (julio 2012, Jelly Bean) | Se hace hincapié en mejorar la fluidez de la interfaz de usuario, logrando incorporar varias técnicas como: vertical, triple búfer y aumentar la velocidad del procesador al tocar la pantalla. |
| Android 4.2 Nivel de API 17 (noviembre 2012, Jelly Bean - Gummy Bear) | La novedad más importante es la de crear varias cuentas de usuario dentro del mismo dispositivo. Los widgets del escritorio pueden aparecer en la pantalla de bloqueo. Mejoras en las notificaciones. Incorpora una nueva aplicación para la cámara. |
| Android 4.3 Nivel de API 18 | Introduce mejoras en los perfiles restringidos para ejecutar aplicaciones específicas y tener acceso a |

| | |
|--|--|
| (julio 2013) | datos específicos. Da soporte para Bluetooth Low Energy (BLE) y OpenGL ES 3.0. |
| Android 4.4 Nivel de API 19 (octubre 2013, KitKat) | El nombre es debido a la golosina de la empresa Nestlé. El objetivo principal es alcanzar una gama de dispositivos con tamaños reducidos de RAM. La nueva interfaz de usuario posee nuevas características como el modo de inmersión en pantalla completa. Se facilita el acceso a aplicaciones en la nube. Se hace la introducción de la nueva máquina virtual ART. |

Tabla 2-1. Recopilación de versiones del sistema operativo Android

Fuente: (Gironés, 2013)

2.1.2 iOS

iOS es un sistema operativo móvil de tipo Unix, derivado del sistema operativo para escritorio Mac OS X y usado por los diferentes dispositivos pertenecientes a la compañía Apple Inc. En sus principios fue desarrollado para el dispositivo iPhone, pero actualmente es usado en otros como iPod Touch, iPad y el Apple TV.



Figura 2-2. iPhone con iOS 7

Fuente: <http://apple.co/1B3MAN2>

La interfaz de este sistema operativo es amigable y fluida para el usuario al ser construida en el concepto de manipulación directa (gestos multitáctiles).

Además está diseñado para ser potente en el aprovechamiento del hardware y visualizarse de forma elegante e intuitiva.

Dentro de iOS existen un sinnúmero de aplicaciones por defecto, incluyendo el navegador web Safari, un cliente de correo electrónico, un reproductor multimedia y la aplicación de teléfono.

Para los desarrolladores existe el kit iOS (SDK), con el cual se puede crear aplicaciones para los dispositivos móviles pertenecientes a Apple. Las aplicaciones nativas son escritas en el lenguaje de programación Objective-C mediante el IDE de desarrollo Xcode.

A) Reseña Histórica

El sistema operativo OS X se dio a conocer junto con el dispositivo iPhone por Steve Jobs en la Macworld Conference & Expo del 9 de enero del 2007, este fue referenciado a su afín Mac OS X usado en las computadoras de escritorio. Originalmente fue nombrado como iPhone OS el 6 de marzo del 2008, donde también se hizo el lanzamiento de la primera versión beta del iPhone SDK.

Durante el año 2007 hubo un aumento en el crecimiento de la plataforma iPhone debido a la aparición del iPod Touch, un dispositivo con similares características del iPhone excepto la capacidad de realizar llamadas telefónicas.

El 27 de enero del 2010 Steve Jobs anunció al iPad, el desarrollo de este dispositivo estaba orientado hacia la navegación web y visualización de

contenidos, gozando funciones de un iPod Touch pero con una pantalla más grande. Además durante el mes de junio de este año, durante la presentación del iPhone 4, iPhone OS cambia su nombre oficialmente a “iOS”.

Tim Cook hace la presentación de iOS 6, iPhone 5 y iTunes 11 en septiembre del 2012 en San Francisco.

En junio del 2013 el iOS 7 es presentado como el mayor cambio del sistema operativo desde sus orígenes, ya que la presentación gráfica del mismo es muy diferente teniendo un enfoque plano y con nuevos íconos. Además el nuevo iOS trae nuevas características como AirDrop, filtros de cámara, fondo dinámico, entre otras.

B) Arquitectura

iOS posee una arquitectura basada en capas, donde en las capas superiores se incluye los servicios y tecnologías más importantes, y en las inferiores se manejan los servicios básicos.

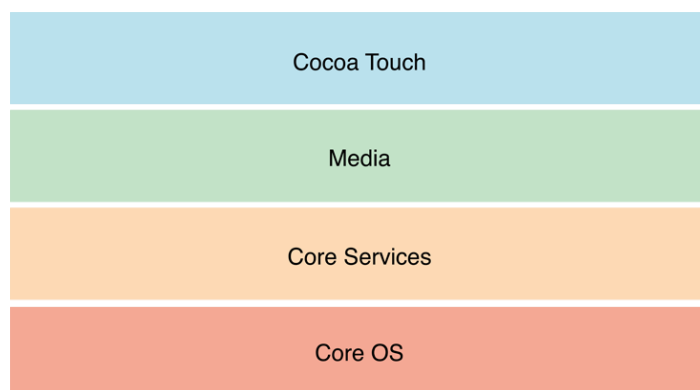


Figura 2-3. Arquitectura del sistema operativo iOS

Fuente: <http://apple.co/1JPvbU6>

Capa “Cocoa Touch”

Es la capa más importante de iOS, ya que posee un API para la creación de las aplicaciones para los dispositivos iPad, iPhone y iPod Touch.

Cocoa Touch está formada por dos frameworks fundamentales que son:

- **UIKit framework:** Proporciona las clases necesarias para la administración de la interfaz de usuario de una aplicación iOS.
- **Foundation framework:** Define un conjunto de clases en el lenguaje Objective-C, que proporcionan acceso y manejo de objetos como servicios del sistema operativo.

Capa “Media”

Conocida como capa de medios de comunicación, es aquella que contiene los gráficos, audio y tecnologías de vídeo; es decir, todos los servicios multimedia que son utilizados por la capa superior.

Capa “Core Services”

Esta capa contiene todos los servicios fundamentales del sistema que son utilizados por todas las aplicaciones. Además posee las tecnologías individuales para admitir características como ubicación, iCloud, medios de comunicación social, y redes.

Capa “Core OS”

La capa Core OS tiene características de bajo nivel como ficheros del sistema, manejo de memoria, gestión de procesos, seguridad y drivers del dispositivo.

C) Versiones

El historial de versiones inició desde el lanzamiento de iPhone OS en el año 2007, y ha ido evolucionando hasta la última versión iOS 7; además está por lanzarse la versión 8 a finales del año 2014.

A continuación se muestra una tabla con la información resumida de las versiones de este sistema operativo.

| Nombre | Última Versión / Build | Estado | Descripción |
|---------------|------------------------|---------------|--|
| iPhone OS 1.0 | 1.1.5 / 4B1 | Discontinuada | Fue la primera versión del sistema operativo. Corría dentro de los iPhone y iPod Touch de primera generación. |
| iPhone OS 2.0 | 2.2.1 / 5H11a | Discontinuada | Instalada en los iPhone 3G, poseía por primera vez el App Store y permitía la descarga de aplicaciones de terceros. |
| iPhone OS 3.0 | 3.1.3 / 7E18 | Discontinuada | Disponible en los iPhone 3GS. Se añadieron funcionalidades como copiar/pegar y MMS. Además esta versión fue adaptada para el iPad. |
| iOS 4 | 4.2.1 / 8C148 | Discontinuada | Usada en el iPhone y iPod Touch (2. ^a , 3. ^a y 4. ^a generación) y iPad (1. ^a y 2. ^a generación). Esta versión no da soporte a dispositivos antiguos. |
| iOS 5 | 5.1.1 / 9B206 | Discontinuada | Es la segunda versión de iOS, no permite actualizarse en los equipos antiguos. Fue usada en iPhone 3GS, iPhone 4, iPhone 4S, iPod touch 3G, iPod touch 4G, iPad, iPad 2 y iPad 3. |
| iOS 6 | 6.1.6 / 10B400 | Discontinuada | Como las anteriores dos versiones no da soporte a dispositivos antiguos, es usada en iPhone 3GS, iPhone 4, iPhone 4S, iPhone 5, iPod Touch (4. ^a y 5. ^a generación) iPad (2. ^a , 3. ^a y 4. ^a generación) y iPad Mini. |
| iOS 7 | 7.1.2 / | Actual | Disponible para el iPhone 4, |

| | | | |
|---------------------|---|------|---|
| 11D201 y 11D201c | | | iPhone 4S, iPhone 5, iPhone 5C, iPhone 5S, iPod Touch (5. ^a generación), iPad (2. ^a , 3. ^a , 4. ^a y 5. ^a generación), iPad Mini y iPad Mini con Pantalla Retina. |
| iOS 8 | - | Beta | Presentada el 2 de julio del 2014, su posible lanzamiento es en septiembre del 2014. |

Tabla 2-2. Recopilación de versiones del sistema operativo iOS

2.1.3 Windows Phone



Figura 2-4. Windows Phone en dispositivos móviles

Fuente: <http://bit.ly/1IEeCNE>

Windows Phone es un sistema operativo para teléfonos inteligentes y dispositivos móviles creado por la empresa estadounidense Microsoft Corporation. Es el sucesor de Windows Mobile y está basado en el núcleo Windows Embedded CE 6.0.

La diferencia con Windows Mobile es su enfoque dirigido solo al mercado de consumo y su nueva interfaz de usuario de diseño “Metro”.

Dentro de la interfaz de usuario Windows Phone se ofrece varios servicios principales como OneDrive, Skype y Xbox Live. Además compite dentro del mercado con los sistemas operativos móviles iOS, Android y Blackberry OS.

A) Reseña Histórica

El sistema operativo Windows Phone, denominado "Photon" durante su desarrollo, llega como sucesor del sistema operativo desarrollado para dispositivos móviles Windows Mobile.

El desarrollo de "Photon" inició en el 2004, pero el trabajo avanzaba lentamente por lo que fue cancelado. Microsoft viendo el gran crecimiento del mercado móvil reorganizó su grupo de trabajo en el año 2008, retomó el proyecto abandonado y logró la creación de un nuevo sistema operativo.

La fecha de aparición de Windows Phone estaba prevista para el año 2009, pero Microsoft tuvo algunos inconvenientes logrando crear Windows Mobile 6.5 que es denominada como una versión de transición.

Windows Phone 7 Series fue lanzado en octubre de 2010, en Europa y en noviembre del mismo año en Estados Unidos. Como su nombre original fue criticado por ser demasiado complejo, el nombre del sistema operativo se redujo oficialmente sólo a Windows Phone 7 el 2 de abril de 2010.

Actualmente el sistema operativo se encuentra en la versión 8.1 que incluye: carpetas en la pantalla de inicio, un Internet Explorer mejorado, entre otras nuevas características.

B) Arquitectura

El sistema operativo está organizado por capas con su propia función dentro de este. El kernel gestiona todas las interacciones con el hardware del dispositivo mediante un grupo de librerías. En las capas superiores al kernel están todos los servicios del sistema y los marcos de programación que las aplicaciones utilizan.

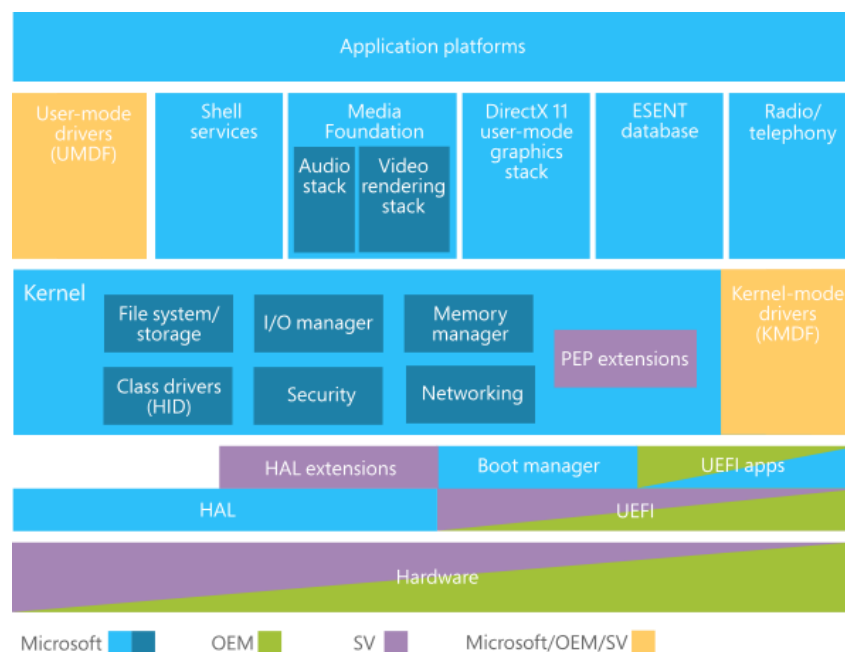


Figura 2-5. Diagrama de alto nivel de Windows Phone 8

Fuente: <http://bit.ly/1QS3f6a>

C) Versiones

Microsoft se esfuerza por cada día por brindar un software de calidad y fácil de usar por el usuario, dada esta razón Windows Phone ha tenido diferentes versiones resumidas a continuación.

| Versión | Descripción |
|-------------------|---|
| Windows Phone 7.0 | Versión inicial dada a conocer como Windows Phone 7 Series en el 2010. Se presentaba en veinticinco idiomas y poseía características funcionales limitadas. |

| | |
|---|--|
| Windows Phone 7.1 | Es la primera actualización de Windows Phone, entre sus mejoras tenía la capacidad de copiar/pegar, menor tiempo de arranque, mejor rendimiento y sincronización con Facebook. |
| Windows Phone 7.5 "Mango" | Presentada en el Mobile World Congress del 2011, este nuevo sistema operativo resolvía las deficiencias del anterior incluyendo una versión móvil de Internet Explorer 9, mejoras en el GPS e integración con Twitter, SkyDrive, Office, LinkedIn. |
| Actualización "Tango" | Lanzada en el 2012, es una actualización menor que corregía algunos errores y reducía requisitos de hardware. |
| Windows Phone 7.8 | Fue la última gran actualización de WP 7, anunciada junto con la versión 8 del sistema operativo. Esta sirve soporte para dispositivos de algunos fabricantes como LG y sus usuarios pertinentes. Entre sus mejoras fueron la pantalla de inicio, colores de tema adicionales y una actualización en la pantalla de bloqueo. |
| Windows Phone 8.0 | Es la segunda generación de Windows Phone. La versión RTM fue lanzada en septiembre del 2012 y fue liberado para consumidores en octubre del mismo año. El cambio más importante fue que cualquier app podía manejar el USB mass Storage. |
| Actualización Pórtico | Puesta en marcha en diciembre del 2012, trae mejoras en la pantalla de inicio e Internet Explorer; además correcciones de errores en mensajería, conectividad Bluetooth, entre otras. |
| Actualización General Distribution Release 2 (GDR2) | Fue publicada en julio de 2013. Dentro de esta actualización entraron mejoras menores del sistema operativo. |
| Actualización General Distribution Release 3 (GDR3) | Liberada en octubre de 2013, se incluyeron mejoras en accesibilidad, el modo "Conductor" permite contestar llamadas y mensajes, bloqueo de la rotación de pantalla, asignación de tonos de llamada y la opción de cerrar aplicaciones desde el carrusel. |
| Windows Phone 8.1 | Anunciado el 2 de abril de 2014, lanzado el 14 del mismo mes para desarrolladores y disponible para el público a lo largo de mayo y junio de 2014. Todos los modelos actuales de dispositivos lo usan y actualizan gratuitamente. Entre las características relevantes están el asistente de voz "Cortana", un centro de notificaciones, sensores de Wifi, datos y batería, entre otros. |

Tabla 2-3. Versiones del sistema operativo Windows Phone

2.1.4 Symbian

Symbian es un sistema operativo de código abierto que funciona en algunos dispositivos móviles pertenecientes a Nokia. Inicialmente fue un producto de la alianza de varias empresas como Nokia, Sony Mobile Communications, Psion, Samsung, Siemens, Arima, Benq, Fujitsu, Lenovo, LG, Motorola, Mitsubishi Electric, Panasonic, Sharp, entre otras. Cabe recalcar que Symbian tiene sus orígenes en EPOC32 (sistema operativo desarrollado por Psion Teklogix).



Figura 2-6. Captura de pantalla de Symbian OS

Fuente: <http://bit.ly/1EbJYmU>

Este sistema operativo fue creado para terminales móviles con el objetivo de competir con Palm OS o Windows Mobile. Posteriormente se pensó en la competencia con Android, iOS, Windows Phone, entre otros.

A) Reseña Histórica

La historia de este sistema operativo descrita por (Arroyo, 2011) es la siguiente:

Symbian OS nació como resultado de una alianza entre varias empresas multinacionales fabricantes de dispositivos móviles. Es descendiente de una larga saga de software que se remonta a finales de los ochenta con EPOC, un sistema operativo gráfico para dispositivos portátiles.

Tras varios avatares, en 2008 Symbian se constituyó en fundación con el objetivo de crear una plataforma abierta de software, de manera que ningún fabricante de telefonía móvil se viera obligado a encarecer su producto por el pago de licencias. Este modelo fue sustituido por otro de distribución propietaria en noviembre de 2010, tras hacerse Nokia cargo de Symbian ante la falta de apoyo de sus socios.

Nokia anunció la fecha de caducidad de Symbian en 2016, pero mientras tanto ha lanzado dos nuevas versiones, Symbian Anna y Symbian Belle, una actualización de la anterior. El teléfono Nokia 808 PureView, lanzado al mercado en el año 2012, es el último terminal Symbian.

B) Características

El grupo de desarrolladores (Nokia Developers , 2012) menciona que Symbian OS presenta las siguientes peculiaridades clave:

- **Performance** - Symbian OS está diseñado para hacer demandas mínimas en las baterías y consumir poca memoria.

- **Multitarea** - telefonía y mensajería universal, son componentes fundamentales. Todas las aplicaciones se han diseñado para funcionar perfectamente en paralelo.
- **Normas** - el uso de tecnologías basadas en estándares acordados es un principio básico del sistema operativo Symbian, asegurando que las aplicaciones sean robustas, portátiles, e interoperables.
- Gestión de memoria optimizada para entornos de software embebido.
- **Requisitos de memoria en tiempo de ejecución se reducidos al mínimo** - pequeños ejecutables y código basado en ROM que corre en su lugar.
- Los mecanismos de seguridad ofrecen comunicaciones y almacenamiento de datos seguros.
- Soporte de aplicaciones para el entorno internacional con una función de juegos de caracteres Unicode.
- Una API rica y variada que permite el acceso a componentes reutilizables en aplicaciones de desarrolladores.

C) Estructura Tecnológica

La estructura tecnológica genérica de Symbian OS es presentada por (Nokia Developers , 2012) de la siguiente forma:

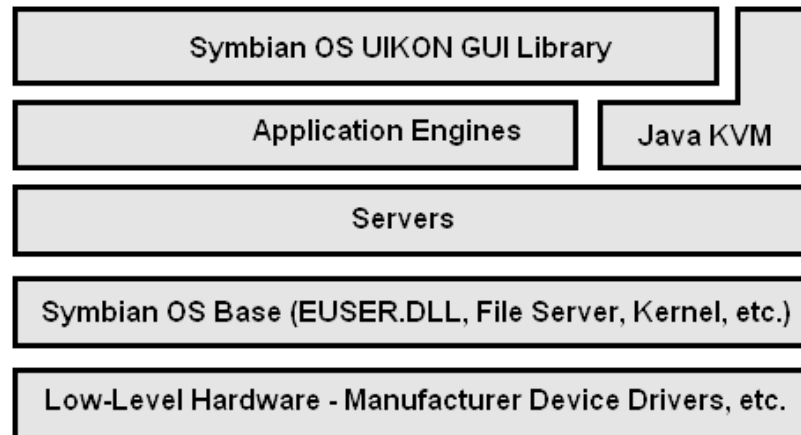


Figura 2-7. Estructura Tecnológica de Symbian OS

Fuente: <http://bit.ly/1d71qUK>

Los controladores del núcleo del sistema, archivos de servidores, administración de memoria y de dispositivos se encuentran en la capa base del sistema operativo. Symbian usa enfoque microkernel. El kernel maneja los recursos del sistema tales como memoria y es responsable por intervalo de tiempo las aplicaciones y tareas del sistema. Los controladores de dispositivos proporcionan el control y la interfaz de los elementos específicos de hardware-el teclado, la pantalla, puerto de infrarrojos y así sucesivamente.

Las capas superiores del sistema proporcionan los servicios de comunicación y extensa de computación, tales como TCP / IP, IMAP4, SMS y gestión de bases de datos. Componentes Symbian OS proporciona la gestión de datos, comunicaciones, gráficos, multimedia, seguridad, gestión de información personal (PIM) los motores de aplicaciones, mensajería motor, Bluetooth, motores de navegación y soporte para la sincronización de datos y la internacionalización.

2.1.5 Firefox O.S

“Firefox OS es el nuevo sistema operativo para móviles desarrollado por Mozilla Corporation, basado en el motor de renderizado Gecko Linux y potente de Firefox”. (Mozilla Developers, 2014)



Figura 2-8. Firefox OS en dispositivo móvil

Fuente: <http://bit.ly/1EO9wHu>

Este sistema operativo a diferencia de Android, es que es de código abierto, por lo cual es distribuido y desarrollado libremente. Esta creado completamente usando HTML5 y otros estándares web abiertos. Además mediante Javascript y otras API'S WEB se comunica con el hardware de cualquier dispositivo móvil.

“Firefox OS es el único sistema operativo con Búsqueda adaptativa de aplicaciones. Mientras todos los demás teléfonos son estáticos y muestran siempre las mismas aplicaciones, Firefox OS está en continuo cambio para ofrecer exactamente las aplicaciones que se necesitan en cada momento”.

(Mozilla Foundation, 2014)

A) Reseña Histórica

Firefox OS nace en relación con el futuro de los móviles. En el 2010, Billy May de Mozilla Labs hace una publicación de un video promocionando un móvil llamado SeaBird que poseía Android como sistema operativo; pero lamentablemente no fue difundido por Mozilla Foundation.

El mismo año se lanzó la propuesta de creación de una tienda de aplicaciones similar a la de Google, es así como surge “Open Web Apps”, donde existe un directorio de aplicaciones y aplicaciones auto-publicadas (pueden ejecutarse en el navegador).

El proyecto “*Boot to Gecko (B2G)*” inicia en julio de 2011 por Andreas Gal, director de investigación de Mozilla Corporation. El objetivo de este fue la construcción de un sistema web abierto, completo e independiente; para así lograr que los desarrolladores web puedan construir aplicaciones confiables de una manera fácil.

En el año 2012 Mozilla obtuvo el apoyo de Telefónica para el desarrollo de este proyecto. En julio de este año se reemplazó el nombre del proyecto por Firefox OS, en octubre se estrenó Firefox Marketplace, y en septiembre los analistas pronosticaron que este nuevo sistema operativo representaría el 1% en el mercado de teléfonos inteligentes para su primer año de disponibilidad comercial.

En febrero de 2013 Mozilla, dentro de la rueda de prensa antes del Mobile World Congress en Barcelona, anuncia los planes para el lanzamiento

mundial de Firefox OS junto con la presencia de dispositivos con este sistema operativo en países como Brasil, Colombia, Hungría, México, Montenegro, Polonia, Serbia, España y Venezuela.

En el año 2014 Mozilla y Telefónica anuncian nuevos terminales desarrollados por Alcatel, Huawei y ZTE que poseen este sistema operativo.

B) Características

Dentro de (Mozilla Foundation, 2014) se citan las siguientes características relevantes:

- **Almacenamiento.-** Utiliza SQLite, que es una base de datos liviana y potente.
- **Aplicaciones web.-** Todas las aplicaciones web pueden ser ejecutadas sin necesidad de ingresar al navegador.
- **Búsqueda adaptativa de aplicaciones.-** Obtén información sobre cualquier tema cuando la necesitas, Firefox OS te la lleva directamente a tu teléfono.
- **Firefox Marketplace.-** Tu fuente de información para las mejores aplicaciones de Firefox OS.
- **Navegador Firefox.-** La manera más rápida, inteligente y segura de obtener lo mejor la Web.
- **Mayor integración con redes sociales.-** Acceso completo a cada actualización de Twitter y Facebook desde cualquier lugar donde estés.

- **Cámara.-** Posee filtros integrados para fotos creativas y divertidas.
- **Radio FM.-** Toda la música que quieres sobre la marcha, en un único dispositivo.
- **Mapas.-** Incluye direcciones e información de tráfico y tránsito local.
- **Privacidad y seguridad.-** Mozilla fue la primera en implementar la característica No quiero ser rastreado. Firefox también presenta opciones de seguridad personalizables para gestionar contraseñas, historial, datos privados, cookies, opciones de sitio y complementos.

C) Arquitectura

El sistema operativo Firefox OS posee tres componentes importantes descritos a continuación:

1. *Gonk*: Es el nivel más bajo del sistema operativo Firefox OS, que consiste en un kernel Linux (basado en y una capa de abstracción de hardware (HAL). Este sistema hereda gran parte del proyecto de código abierto Android (AOSP) llevando consigo algunos drives, bibliotecas y componentes para las funciones vitales del Smartphone.
2. *Gecko*: Este es el entorno de ejecución de aplicaciones Firefox OS; es decir, la capa que proporciona todo el apoyo para los estándares web abiertos (HTML, CSS y JavaScript). Esta capa se asegura de que todas las APIs funcionan bien en cualquier sistema

operativo compatible con Gecko. “Esto significa que Gecko incluye, entre otras cosas, una pila de red, pila de gráficos, motor de diseño, una máquina virtual de JavaScript, y capas de portabilidad”. (Mozilla Developers, 2014)

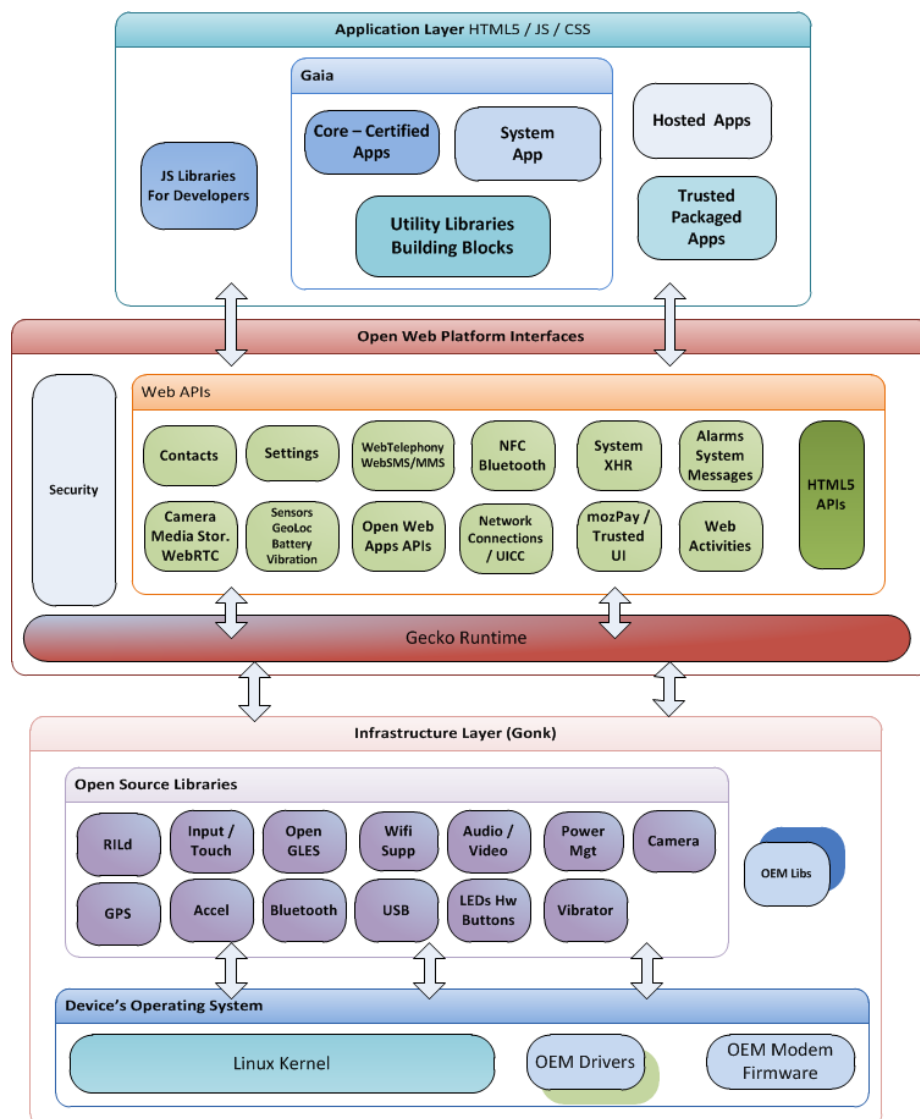


Figura 2-9. Arquitectura de Firefox OS

Fuente: <http://mzl.la/1EO9ZcE>

3. *Gaia*: Es la interfaz gráfica de usuario de Firefox OS. Todo lo que es mostrado en la pantalla una vez que Firefox OS se pone en

marcha es parte de Gaia; en esto se incluye: la pantalla de bloqueo, la pantalla de inicio, y todas las aplicaciones estándar que usted espera en un teléfono inteligente moderno. Gaia implementa en su totalidad el uso de HTML, CSS y JavaScript.

Procedimiento de arranque Firefox OS

El procedimiento de arranque de Firefox OS es descrito por (Mozilla Developers, 2014) de la siguiente manera:

Cuando un dispositivo Firefox OS se enciende por primera vez, la ejecución comienza en el gestor de arranque primario. Desde allí, el proceso de carga de los principales beneficios del sistema operativo en la forma típica; una sucesión de gestores de arranque de nivel superior cargan en el bootstrap loader (gestor de arranque). Al final del proceso, la ejecución se entrega al kernel de Linux.

Hay algunos puntos dignos de mención sobre el proceso de arranque:

- El gestor de arranque suele mostrar la primera pantalla de bienvenida vista por el usuario durante el inicio del dispositivo; esto es por lo general un logotipo.
- El gestor de arranque implementa el parpadeo de una imagen de carga para el dispositivo. Diferentes dispositivos utilizan diferentes protocolos; la mayoría de los teléfonos utilizan el protocolo fastboot, pero el Samsung Galaxy S II utiliza el protocolo odin.

- Al final del proceso de bootstrapping, la imagen está normalmente cargada y en ejecución. Como sucede esto es altamente específico del dispositivo y puede ser diferente.

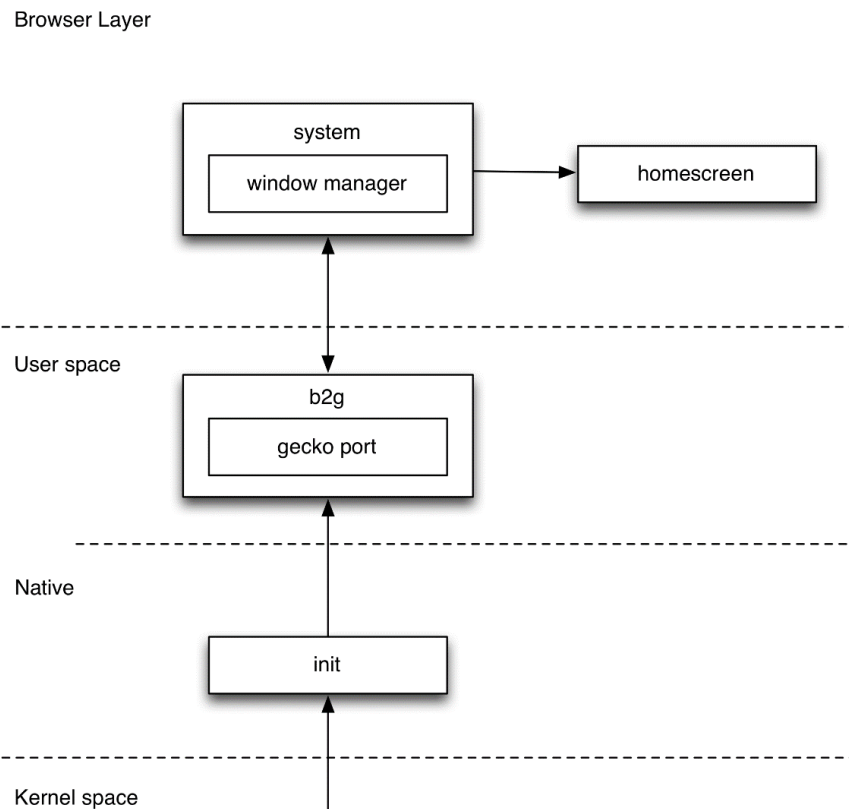


Figura 2-10. Firefox OS bootstrapping process

Fuente: <http://bit.ly/1EQ0J8Q>

2.1.6 Ubuntu Touch

Ubuntu Touch nace como parte de la rebelión de la serie Alternativa Mobile OS, es desarrollado por Canonical Ltd y está basado en Linux. Fue presentado en 2 de enero de 2013 al público mediante un anuncio en el sitio web oficial de Ubuntu, mostrado como un sistema operativo que puede ser usado en tabletas y teléfonos inteligentes.



Figura 2-11. Sistema operativo Ubuntu Touch

Fuente: <http://bit.ly/1FkySBp>

2.1.7 BlackBerry OS

Según (Barrera & Van Oorschot, 2011) este sistema operativo móvil posee la siguiente descripción:

BlackBerry OS ha sido desarrollado por Research in Motion (RIM). El sistema operativo se ejecuta en un gran número de modelos BlackBerry, y ha sido históricamente muy orientado hacia los clientes empresariales mediante la inclusión de características como correo electrónico y soporte de trabajo colaborativo (Microsoft Exchange, Lotus, Novell GroupWise y BES support).

BlackBerry OS soporta aplicaciones de terceros escritas en Java. El sistema operativo utiliza sandboxing para aislar aplicaciones en tiempo de ejecución, logrado a través del Java Virtual Machine (JVM).

Algunos desarrolladores Java crearon aplicaciones para BlackBerry que eran distribuidas en sitios fuera de la aprobación de RIM; por lo cual, se

introdujo BlackBerry AppWorld (abril de 2009), donde los usuarios de los modelos de BlackBerry más nuevos pueden acceder a un repositorio de aplicaciones de RIM aprobadas. A pesar de que RIM debe aprobar presentó cada solicitud de inclusión en AppWorld, los desarrolladores son libres para alojar sus aplicaciones en otros servidores.



Figura 2-12. BlackBerry Z30

Fuente: <http://blck.by/1QS5eHy>

En sus inicios gran cantidad de teléfonos móviles que utilizaban este sistema operativo poseían teclado QWERTY, pero actualmente hay teléfonos táctiles como BlackBerry Z10 y Z30.

La última versión de este sistema operativo es BlackBerry OS 10, que fue lanzada en enero del 2013 y ha tenido soporte hasta la actualidad.

2.2 APLICACIONES MÓVILES

“El 10 de julio de 2008 Apple inauguraba su tienda de aplicaciones para iPhone e iPod Touch con más de 800. En tan solo cuatro días ya había alcanzado los 10 millones de descargas”. (Arroyo, 2011)

El éxito mencionado anteriormente crea un modelo de negocio que cambia las reglas del desarrollo de dispositivos móviles y la industria del contenido; además crea un modelo a seguir.

2.2.1 Definición

“El término aplicaciones procede del inglés *apps*, una contracción de *application* con un cierto aire informal que se traduce también al castellano como «aplis», en alusión a la forma anglosajona y con sus mismas connotaciones”. (Arroyo, 2011).



Figura 2-13. Aplicación móvil de video2brain

Fuente: Propia

Una aplicación móvil es aquel software de aplicación diseñado exclusivamente para ejecutarse dentro de dispositivos móviles como un teléfono inteligente o tableta. Estos aplicativos proporcionan a los usuarios servicios similares a los expuestos dentro de las computadoras o PC's.

2.2.2 Utilidades de las aplicaciones

La cifra de descarga mundial de aplicaciones móviles cada vez va incrementando gracias al uso de los teléfonos inteligentes y las tabletas. Existen varias aplicaciones dentro de cada uno de los marketplaces, que brindan cierto beneficio a los usuarios.

Las utilidades más comunes de las aplicaciones móviles según (Arroyo, 2011), son:

- Los **juegos** son la categoría más popular en las tiendas de aplicaciones y siempre se encuentran varios entre los rankings de los más populares. Todo un fenómeno es Angry Birds. Creado en diciembre de 2009 por la empresa Rovio para iPhone, se ha convertido en el juego para móviles más descargado de la historia, con más de 300 millones de descargas en agosto de 2011, y ahora se encuentra también disponible para otras plataformas.
- **Redes sociales y comunicaciones.** Las grandes redes sociales han estado siempre entre los primeros habitantes de las app stores . Facebook, Twitter, Tuenti o YouTube pusieron en práctica el principio de independencia de dispositivo, uno de los básicos de la

Web 2.0. En lo que se refiere a comunicación, WhatsApp está suponiendo un nuevo canal para el envío de mensajes de texto o SMS, pues se vale de la conectividad WiFi o 3G para ahorrar así el coste del mensaje.

- Las **aplicaciones de retoque de fotografía** combinadas con redes sociales en las que compartirlas han alcanzado asimismo un gran éxito. Un buen ejemplo es Instagram (figura 5), que da a las fotografías y vídeos tomados ese aire que tienen las fotografías instantáneas. Similares son LemeLeme, Pose o Piictu. Esta última tiene el don de crear hilos de conversación con fotos. Otras, como Camera Art FX Free, convierten en dibujos de distintos tipos, según el filtro que se les aplique, nuestras imágenes. Aprovechando la cámara, también hay aplicaciones capaces de obtener imágenes panorámicas, como Photosync, y de escanear documentos.
- **Productividad y utilidades.** Calendarios, agendas, listas de cosas por hacer, software de escritura de notas o textos y otras utilidades incluidas en todas las listas de los imprescindibles son grandes éxitos de descargas. Evernote es una herramienta para tomar notas sobre la marcha, no solo con textos, sino también con imágenes, Dropbox, un servicio de almacenamiento en la nube, y Dragon Dictation, que transcribe las notas de voz que le dejamos y las envía en un mensaje, un correo electrónico o a las redes sociales, son solo algunas de ellas.

- Las **compras** están presentes en las app stores bien para mostrar el catálogo del vendedor, como en la aplicación de Ikea, o para realizar compras en línea directamente. Tiendas como Zara y outlets como Groupon o Privalia son algunas de ellas.
- **Educación.** Diccionarios, libros de texto, juegos para practicar la escritura de nuevos alfabetos o la pronunciación, libros para niños, etc. Los dispositivos móviles pueden contribuir en los procesos de aprendizaje con entornos gráficos e interactivos, multimedia, portables y disponibles en cualquier lugar.
- En las categorías de **salud y fitness** encontramos aplicaciones que nos enseñan a entrenarnos y nos ayudan a mantener una dieta, a recordar el calendario de tratamientos médicos o contienen información sobre problemas de salud concretos.
- **Prensa, radio y televisión** se pueden leer, escuchar y ver a través de aplicaciones. Además de las iniciativas individuales de los medios, se han puesto en marcha otras conjuntas, como Kiosko y más, 27 que aglutina diarios y revistas de varios grupos editoriales con ejemplares a 0,79 € y suscripciones por 9,99 €. Con Pandora y TuneIn Radio (esta última con dos versiones, una gratuita y otra de pago) podemos sintonizar canales de radio; pero también algunas emisoras, como Radio Nacional de España, tienen su propia aplicación. Siguiendo ese mismo patrón, también existen aplicaciones de cadenas de televisión, como La Sexta en directo o RTVE Clan, con la programación y algunos programas ya emitidos.

- **Libros y cómics** electrónicos se pueden descargar a través de aplicaciones como iBooks, la tienda de libros de Apple, o Comixology, pero también hay aplicaciones que son un libro. Otras, como Stanza, permiten descargar libros de pago y gratuitos (de fuentes libres), pero también transportar contenidos desde nuestro ordenador. Librerías como Amazon o Barnes & Noble ofrecen aplicaciones para descargar y leer las obras de sus tiendas. En definitiva, la lectura ha sido asumida por los dispositivos móviles como una función más, independientemente de si el tamaño de la pantalla o su retroiluminación nos resulten más o menos cómodos para leer.

2.2.3 Tipos de soluciones

Para el desarrollo de una aplicación móvil se debe considerar el tipo de solución más conveniente de acuerdo a la necesidad planteada. Existen 2 grandes grupos de soluciones en el mundo móvil que son: standalone y aplicaciones basadas en servicios web.

A) Standalone

Es aquella aplicación que funciona de manera independiente; es decir, se ejecuta en equipos específicos sin conexión a un servidor o a internet. Este tipo de soluciones son construidas para un solo sistema operativo y deben generar un paquete ejecutable compatible con el mismo.

Las soluciones standalone poseen las siguientes ventajas:

- Gran velocidad en ejecución.

- Alto rendimiento, ya que se adaptan solo a un dispositivo o sistema operativo determinado.
- Ejecución offline.

Dentro de sus desventajas tenemos:

- Incompatibilidad con otros sistemas operativos o dispositivos.
- Instalación manual de la aplicación.
- Capacidades de almacenamiento limitadas por el dispositivo móvil.
- No acceden a fuentes de información remotas.
- Peso de la aplicación.

Algunos ejemplos de estas aplicaciones pueden ser: alarmas, contactos de teléfono, juegos fuera de línea, entre otras.

B) Basadas en servicios web

Es aquella aplicación que funciona como un software cliente, esto significa que se conecta con un servicio web para tener acceso a datos o algún otro recurso necesario para que la App funcione a su totalidad. Los servicios están montados dentro de un servidor o servidores conectados a internet. También estas soluciones pueden sincronizarse con un servidor para enviar información hacia este.

Las ventajas de este tipo de solución son:

- Son ligeramente livianas, no ocupan mucho espacio dentro del almacenamiento del dispositivo,
- Compatibilidad con distintos dispositivos.

- Permiten obtener grandes volúmenes de información almacenada en servidores.
- Funcionan de manera online u offline.

Dentro de sus desventajas tenemos:

- Desarrollo complejo.
- En la mayoría de casos usan conexión permanente a internet.
- La velocidad de su ejecución dependerá de la conexión a internet y el tráfico de red del servidor.

Algunos ejemplos de estas aplicaciones son: correo electrónico, juegos en línea, redes sociales, periódicos en línea, entre otras.

2.2.4 Tipos de aplicaciones según su desarrollo

A nivel de programación, las aplicaciones móviles según se clasifican en: nativas, híbridas, y web.

A) Aplicaciones nativas

Una aplicación nativa es aquella desarrollada específicamente para un determinado sistema operativo móvil como: Android, iOS o Windows Phone. Los programadores las construyen mediante el “Software Development Kit” o SDK liberado para cada SO.

La descarga e instalación de este tipo de aplicaciones se realiza dentro de los respectivos marketplaces, donde se realiza la promoción y marketing de las mismas.



Figura 2-14. Aprovechamiento del sistema de notificaciones.

Fuente: (Cuello & Vittone, 2013)

“Una característica generalmente menospreciada de las apps nativas, es que pueden hacer uso de las notificaciones del sistema operativo para mostrar avisos importantes al usuario, aun cuando no se esté usando la aplicación, como los mensajes de Whatsapp, por ejemplo”. (Cuello & Vittone, 2013)

La ventaja con respecto a los otros tipos es la facilidad de acceso a los componentes del hardware del dispositivo como: la cámara, GPS, acelerómetro, giroscopio, dispositivo de almacenamiento, entre otros. Por lo general no requieren internet para funcionar, ofreciendo una experiencia más fluida e integración al teléfono.

A nivel de diseño, esta clase de aplicaciones tiene una interfaz basada en las guías de cada sistema operativo, logrando mayor coherencia y consistencia con el resto de aplicaciones y con el propio SO. Esto favorece

la usabilidad y beneficia directamente al usuario que encuentra interfaces familiares. (Cuello & Vittone, 2013)

B) Aplicación Web

Una **webapp** es aquella desarrollada con HTML, JavaScript y CSS; que son herramientas muy conocidas por los programadores. Las aplicaciones pueden ser usadas en diferentes sistemas operativos sin inconvenientes ya que no emplean un SDK.

Las **webapps** no tienen la necesidad de instalarse en un dispositivo, ya que su visualización es dentro de un navegador. Por la razón anterior no son distribuidas dentro de una tienda de aplicaciones, promocionándose y comercializándose de manera independiente.



Figura 2-15. Aplicación nativa y web de Facebook.

Fuente: (Cuello & Vittone, 2013)

“Las aplicaciones web suelen tener una interfaz más genérica e independiente de la apariencia del sistema operativo, por lo que la experiencia de identificación del usuario con los elementos de navegación e

interacción, suele ser menor que en el caso de las nativas”. (Cuello & Vittone, 2013)

C) Híbridas

Este tipo de aplicaciones es la combinación de los otros dos tipos citados anteriormente. El desarrollo es parecido al de una aplicación web, pero al terminarla es compilada o empaquetada con una plataforma para transformar el código en una aplicación nativa. Esto permite la reutilización de código para obtener aplicaciones para diferentes sistemas operativos móviles.



Figura 2-16. Aplicación híbrida de Netflix con visualización similar en iOS y Android.

Fuente: (Cuello & Vittone, 2013)

“Las aplicaciones híbridas, también tienen un diseño visual que no se identifica en gran medida con el del sistema operativo. Sin embargo, hay formas de usar controles y botones nativos de cada plataforma para apearse más a la estética propia de cada una”. (Cuello & Vittone, 2013)

Las herramientas para el desarrollo de este tipo de aplicaciones son: Apache Cordova, Adobe PhoneGap, Appcelerator Titanium, entre otras.

2.2.5 Distribución

Existen varios canales de distribución de las aplicaciones móviles tales como: páginas de recomendación de aplicaciones, preinstalaciones, sitios web, motores de búsqueda, redes de publicidad, entre otros. Pero el principal canal es la tienda de aplicaciones brindada para cada sistema operativo móvil.

Las tiendas más populares son:

- **Google Play.-** Plataforma de distribución de software en línea creada por Google Inc. Para dispositivos Android.
- **App Store.-** Servicio de distribución de aplicaciones perteneciente a Apple Inc. para el uso de las mismas dentro de iPhone, iPod Touch y iPad.
- **Windows Phone Store.-** Plataforma de distribución de Microsoft para los dispositivos que poseen Windows Phone.
- **BlackBerry World.-** Servicio de descarga de aplicaciones para los dispositivos de BlackBerry.
- **Amazon Appstore.-** Aplicación móvil de distribución de software para dispositivos Android.

2.2.6 Aplicaciones de pago y gratuitas

Las aplicaciones móviles son un producto relativamente nuevo dentro del mercado que responden a diferentes modelos de negocio. Existen casos donde las aplicaciones tienen como objetivo ser una forma de obtención de ganancias y otros donde estas son un canal de comunicación con los usuarios para extender una marca.

A) Apps gratuitas

El beneficio de las aplicaciones gratuitas es el alcance de usuarios potenciales a los que puede llegar, ya que estos pueden descargarla y probarla sin ningún obstáculo. Al ser tan accesibles para los usuarios, disminuye de cierta forma la expectativa del producto; es decir, que los usuarios no esperan que sea impresionante. La colocación gratuita en el mercado puede servir para conocer el comportamiento de uso de la misma y mejorar versiones futuras.

El inconveniente de una aplicación gratuita es que debido a la competencia en el mercado no se encontrará el ranking de las diferentes tiendas, ya que requiere un gran número de descargas para estar en los primeros lugares.

La más óptima utilidad de este tipo de aplicación sería de servir como brecha para promocionar versiones de pago de la misma u otras aplicaciones del desarrollador.

B) Apps pagadas

Como su nombre lo indica hay que invertir cierta cantidad de dinero para obtener este software. A diferencia de las aplicaciones gratuitas su difusión y descarga es más complicada, ya que los usuarios no se arriesgan a comprar algo desconocido.

Para que las aplicaciones de pago tengan gran éxito deben ofrecer funcionalidades diferentes que las gratuitas y también estar ubicadas en tiendas correctas para su mejor difusión. Además, algo importante que suelen tener los usuarios antes de comprar una aplicación es la valoración y comentarios dados por otros usuarios.

Es importante saber que cada una de las tiendas cobra una comisión de 30% sobre el precio de venta de la App. Esto quiere decir que el desarrollador se queda solo con el 70% de lo que el usuario pague por su descarga.

C) Freemium

La palabra *freemium* viene de la mezcla de las palabras inglesas “*free*” y “*premium*”. Este modelo consiste en la descargar de una aplicación de forma gratuita, permitiéndole un uso básico o limitado al usuario, y con posibilidad de recibir funciones avanzadas mediante un pago previo.

“Un claro ejemplo de este tipo de distribución, son aquellos juegos que dejan avanzar ciertos niveles, pero para llegar hasta el final hay que pagar por la versión completa”. (Cuello & Vittone, 2013)

2.3 SERVICIOS WEB

El (World Wide Web Consortium, 2014) define a los servicios web como “*un sistema de software diseñado para apoyar la interacción interoperable máquina a máquina sobre una red. Cuenta con una interfaz descrita en un formato procesable por máquina (específicamente WSDL). Otros sistemas interactúan con el servicio Web de una manera prescrita por su descripción utilizando mensajes SOAP, por lo general transmiten por medio de HTTP con una serialización XML en conjunción con otras normas relacionadas con la Web*”.

Para mayor comprensión se puede decir que un servicio web es un conjunto de estándares de software como SOAP, WDSL, UDDI, y RDF; para el intercambio de datos entre aplicaciones. Los estándares pueden ser desarrollados en diferentes lenguajes de programación, logrando ser implementados sobre distintas redes computacionales.

Además la W3C junto con OASIS (Organization for the Advancement of Structured Information Standards), son responsables de integrar los estándares que garanticen y mejoren la interoperabilidad de los servicios web.

2.3.1 Representational State Transfer (REST)

La explicación de los servicios REST, según (Fensel, Facca, Simperl, & Toma, 2011) es la siguiente:

REpresentational State Transfer (REST), es un estilo de diseño clave que abarca una arquitectura cliente-servidor sin estado. REST se introdujo originalmente como un estilo arquitectónico para la construcción de sistemas hipermedia distribuidos a gran escala. Este estilo de arquitectura refleja una entidad abstracta cuyos principios se han utilizado para explicar la excelente escalabilidad del protocolo HTTP 1.0 y también han limitado el diseño de su siguiente versión, HTTP 1.1. El término REST muy a menudo se utiliza junto con HTTP. Aunque REST se utiliza sobre todo en relación con la Web, REST no es realmente acerca de la web. REST es más bien una síntesis de la forma en la Web ya trabaja, un conjunto de principios que se pueden aplicar para la construcción de grandes sistemas distribuidos escalables.

Conceptualmente, REST concibe recursos como objetos de primera clase identificados por URIs. Estos recursos representan los conceptos que fundamentan la comunicación entre un cliente y un servidor. Según este paradigma, los servicios se ven así como los recursos disponibles en la Web y son manipulados a través de sus representaciones. Los clientes pueden solicitar una representación específica de todo el concepto de las representaciones que el servidor pone a disposición. Más precisamente, en caso de que los clientes deseen utilizar una de las representaciones del servidor, esta se realizará mediante la transferencia de contenido de la aplicación usando un pequeño conjunto de métodos remotos. Estos métodos describen la acción a realizar en los recursos. Los métodos utilizados son los métodos HTTP bien conocido para crear, leer, actualizar y eliminar recursos:

GET, POST, DELETE y PUT. Si los recursos se pueden considerar como los conceptos de comunicación, los métodos que hemos mencionado anteriormente son los verbos que el desarrollador puede utilizar para describir las acciones necesarias para realizarse incluyendo la creación, lectura, actualización y eliminación (CRUD).

2.3.2 SOAP y REST

SOAP (Simple Object Access Protocol) es el protocolo estándar que posibilita el intercambio de datos y mensajes XML, usado en la tecnología de los servicios web. El protocolo SOAP tiene una estructura muy compleja, pensando en dar soluciones a cualquier necesidad dentro del ámbito de comunicaciones (seguridad, transaccionalidad, mensajería, entre otras). Durante el paso del tiempo este protocolo ha ido evolucionando en medida de las especificaciones WS-* y actualmente posee muchas capacidades que en la mayoría de casos no son usadas.

Por su parte REST es simple y no da soluciones para todo, siendo muy potente y fácil en su desarrollo. REST es utilizado a menudo en aplicaciones móviles, redes sociales, sitios web y procesos de negocio automatizados. Las principales ventajas de REST sobre SOAP son las siguientes:

- Los servicios web RESTful han sido acoplados por la mayoría de herramientas, incluidas las que son gratis y de bajo costo. REST está siendo usado en la interacción de sistemas, incluyendo el uso de los

servicios web RESTful, que son la forma en proveedores de la nube externalizan sus servicios en la nube.

- Los servicios SOAP son mucho más difíciles de escalar que los servicios REST. Por lo tanto, REST se elige a menudo como la arquitectura para los servicios que se exponen a través de Internet.
- La curva de aprendizaje parece estar reducida, ya que los desarrolladores son capaces de hacer uso de REST de una forma más rápida de lo que puede realizar con SOAP. Esto ahorra tiempo, lo que ahorra dinero.
- REST utiliza un formato de mensaje de menor peso que SOAP. SOAP utiliza XML para todos los mensajes, lo que hace el tamaño del mensaje mucho más grande, y por lo tanto menos eficiente. Esto significa REST proporciona un mejor rendimiento, así como reduce los costos en el tiempo.
- Los servicios web SOAP están orientados a RPC, es decir, a invocar métodos sobre un servicio remoto; mientras que los servicios REST están orientado a recursos.

2.4 JSON

Dentro de (JSON ORG, 2014) se lo define de la siguiente manera:

JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos. Leerlo y escribirlo es simple para humanos, mientras que para las máquinas es simple interpretarlo y

generarlo. Está basado en un subconjunto del Lenguaje de Programación JavaScript, Standard ECMA-262 3rd Edition - Diciembre 1999.

JSON es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

2.4.1 Sintaxis

JSON se encuentra constituido por dos estructuras esenciales que son:

- Datos presentados en una colección de pares nombre/valor.
- Arreglos o listas ordenadas de valores.

Las estructuras descritas anteriormente son universales, y soportadas por los diferentes lenguajes de programación existentes. Es decir, que este formato de intercambio de datos es independiente del lenguaje de programación utilizado.

Cada uno de los valores usados dentro de una estructura JSON puede ser: un número (integer o float), una cadena (string), un booleano (true o false), un vector (array), un objeto (object) y nulo (null).

En JSON, los valores anteriores toman la siguiente forma:

- **Objeto**, un conjunto desordenado de pares nombre/valor.

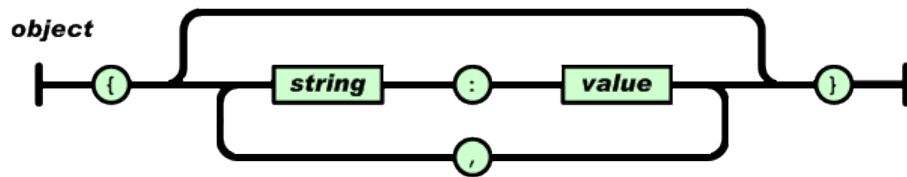


Figura 2-17. Sintaxis de objetos

Fuente: <http://www.json.org/object.gif>

Por ejemplo:

```
{"nombre":"Juan", "apellido":"Rea"}
```

- **Arreglo o vector**, una colección de valores.

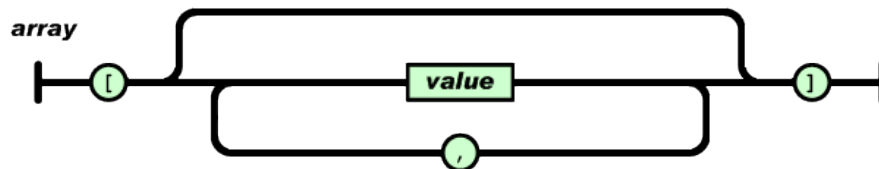


Figura 2-18. Sintaxis de arreglos

Fuente: <http://www.json.org/array.gif>

Por ejemplo:

```
"empleados":[
  {"nombre":"Juan", "apellido":"Rea"},
  {"nombre":"Luis", "apellido":"Ramirez"},
  {"nombre":"Pedro", "apellido":"Arroyo"}
]
```

- **Valor**, puede ser una cadena entre comillas dobles, o cualquiera de los valores que pueden usarse en un JSON. Las estructuras pueden anidarse.

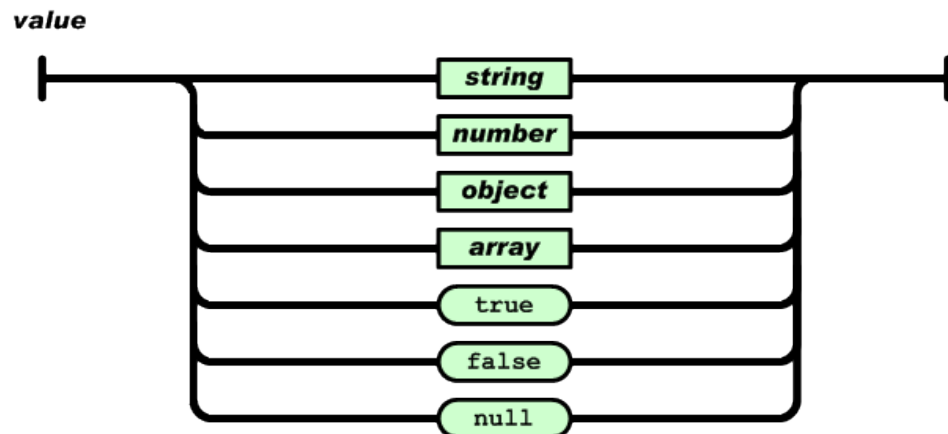


Figura 2-19. Sintaxis de valor

Fuente: <http://www.json.org/value.gif>

- **Cadena**, una colección de caracteres Unicode, que están entre comillas dobles, y usan barras divisorias invertidas como escape.

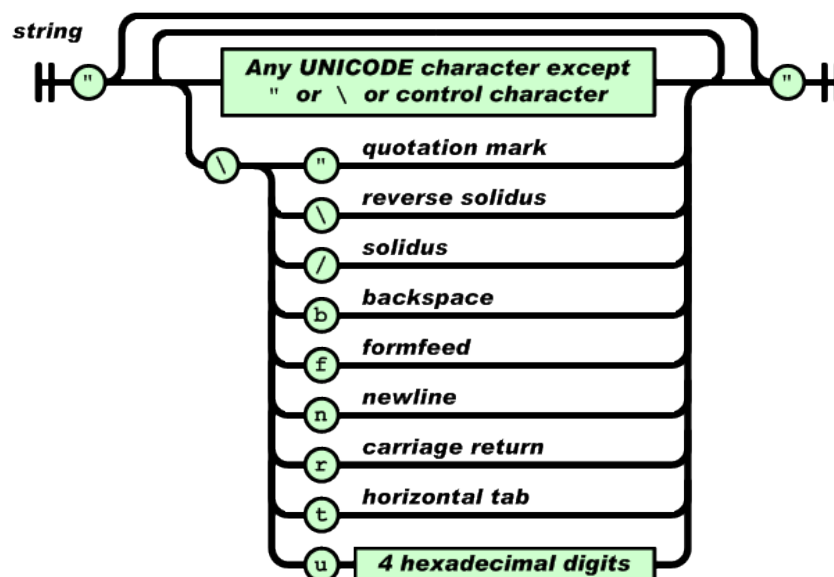


Figura 2-20. Sintaxis de cadena de caracteres

Fuente: <http://www.json.org/string.gif>

- **Número**, cualquier número que no use los formatos octales y hexadecimales.

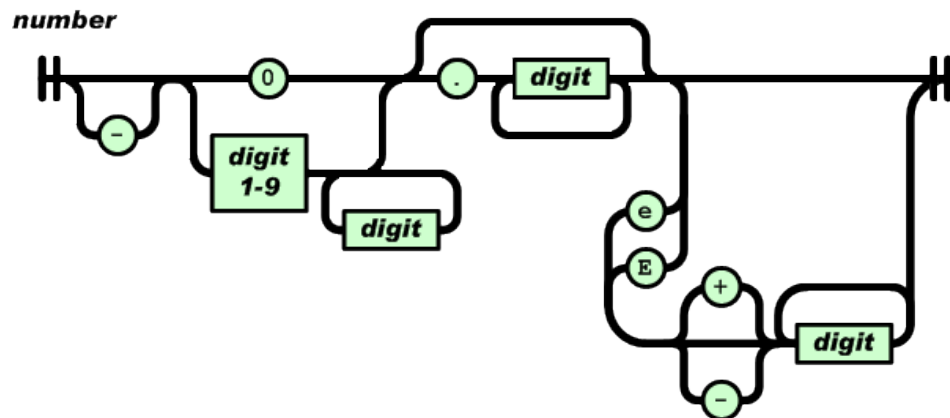


Figura 2-21. Sintaxis de arreglos

Fuente: <http://www.json.org/number.gif>

2.4.2 Diferencias con XML

JSON promete los mismos beneficios que XML, pero sin sus desventajas. A continuación se presenta una comparativa entre JSON y XML, de acuerdo con los aspectos esenciales de XML:

Tabla 2-4. Comparativa entre XML y JSON

| | |
|--------------------------|---|
| Simplicidad | JSON es más simple que XML, ya que usa estructuras de datos utilizadas también en los lenguajes de programación logrando una gramática más pequeña. |
| Extensibilidad | JSON no necesita ser extensible, ya que no es un lenguaje de marcado. Dentro de este no se necesita definir nuevas etiquetas o atributos para la representación de datos. |
| Interoperabilidad | Ambos poseen el mismo potencial de interoperabilidad. |
| Apertura | JSON al igual que XML poseen la misma apertura. |

2.5 FRAMEWORK JQUERY MOBILE

jQuery Mobile es un framework móvil basado en HTML5, diseñado para hacer responsivos a sitios web y aplicaciones que sean accesibles desde dispositivos móviles. Está construido sobre la estructura de jQuery y jQuery

UI ofreciendo características como navegación Ajax, eventos táctiles y widgets. Además es compatible con otros frameworks móviles y plataformas.



Figura 2-22. JQuery Mobile logo

Fuente: <http://bit.ly/1cGFkYn>

Para comprender mejor jQuery Mobile vamos a definir que no lo es:

- **jQuery Mobile no es una alternativa a jQuery para navegadores móviles.**- Para usarlo se debe incluir la librería de jQuery.
- **jQuery Mobile no es un framework para los amantes de JavaScript.**- Para el uso de este no es necesario usar código JavaScript, excepto en ciertos temas avanzados.
- **jQuery Mobile no es la solución para todas las aplicaciones móviles, sitios web o juegos.**- A pesar de su gran funcionalidad no puede cubrir todo tipo de soluciones.

2.5.1 ¿Por qué usar jQuery Mobile?

La mayoría de personas usan dispositivos móviles como tabletas o teléfonos inteligentes, dentro de estos se muestran sitios web o aplicaciones móviles que probablemente están desarrolladas con este framework u otro similar.

jQuery Mobile lleva el slogan "escribir menos, hacer más", permitiendo que los desarrolladores diseñen de una manera automática páginas web o Apps

con un atractivo peculiar, fáciles de usar, y que funcionará de la misma manera en todos los dispositivos móviles.

Hay razones que hacen que jQuery Mobile sea uno de los frameworks más usados a nivel mundial:

- **Responsivo.-** Permite adaptar la apariencia a las dimensiones del dispositivo usado.
- **Flexible.-** Puede ser personalizado; es decir, cambiaremos la apariencia del tema y los widgets.
- **Simple.-** Mediante el uso de los estándares HTML, posee una estructura fácil de comprender y usar.
- **Soportado en varios dispositivos.-** Funciona en todos los más populares teléfonos inteligentes y tabletas.
- **Rentable.-** Basado en la increíble biblioteca jQuery. jQuery es una biblioteca de JavaScript rápida, eficiente y confiable para el desarrollo web.

2.5.2 Funcionamiento

Mediante el uso de HTML5 y CSS3 jQuery Mobile mejora la estructura de HTML básico y brinda una experiencia móvil a través de las plataformas soportadas. jQuery Mobile se basa en las especificaciones HTML5 dadas por la W3C para la personalización de data-attributes.

Una vez cargado o iniciado jQuery Mobile, se aplicarán automáticamente los atributos HTML, los estilos CSS y los eventos. Esto hace que los

desarrolladores se concentren en la estructura del documento y la creación de interfaces de usuario.

2.5.3 Características

Las principales características son:

- Multiplataforma.
- Interfaz de usuario optimizada para dispositivos táctiles.
- Diseño personalizable y de aspecto configurable.
- Uso de sólo HTML5.
- Uso de AJAX para contenido dinámico
- Basado en jQuery.
- Peso ligero (12Kb comprimido).
- Mejora continua.
- Brinda soporte para aplicar accesibilidad web.

2.5.4 Contenido de la librería

jQuery Mobile se compone de cuatro archivos principales:

- Un archivo JavaScript cuya funcionalidad es realizar tareas como: creación de widgets, aplicación de eventos y ejecución del API. Este archivo debe ser cargado luego de librería de jQuery.
- Una hoja CSS que especifica el diseño, animaciones y transiciones de los elementos de jQuery Mobile.
- Un pequeño conjunto de gráficos para la presentación. Estos son íconos simples que están estandarizados para la navegación.

Este paquete mencionado anteriormente puede ser descargado desde la página del proyecto jQuery Mobile: <http://jquerymobile.com/>.

2.5.5 Soporte de plataformas

Dentro de (jQuery Foundation, 2014) se especifica el soporte para la versión más reciente de jQuery Mobile (1.4), usando un sistema gradual de 3 niveles: A (completa), B (completo menos Ajax), C (HTML básico).

A) Grado A

Brinda una experiencia completa con transiciones de página animadas basadas en Ajax.

- **Apple iOS 4-7.0** – Probado en los originales iPad (4.3 / 5.0), iPad 2 (4.3 / 5.1 / 6.1), iPad 3 (5.1 / 6.0), iPad Mini (6.1), iPad Retina (7.0), iPhone 3GS (4.3), iPhone 4 (4.3 / 5.1), iPhone 4S (5.1 / 6.0), iPhone 5 (6.0), y iPhone 5S (7.0)
- **Android 4.4 (KitKat)** – Probado en un Nexus 5.
- **Android 4.1-4.3 (Jelly Bean)** – Probado en un Galaxy Nexus y Galaxy 7.
- **Android 4.0 (ICS)** – Probado en un Galaxy Nexus.
- **Android 3.2 (Honeycomb)** – Probado en the Samsung Galaxy Tab 10.1 y Motorola XOOM.
- **Android 2.1-2.3** – Probado en HTC Incredible (2.2), Original Droid (2.2), HTC Aria (2.1), Google Nexus S (2.3). Funcional en 1.5 & 1.6 pero el rendimiento decae, Probado en Google G1 (1.5).

- **Windows Phone 7.5-8** – Probado en HTC Surround (7.5), HTC Trophy (7.5), LG-E900 (7.5), Nokia 800 (7.8), HTC Maza (7.8), Nokia Lumia 520 (8), Nokia Lumia 920 (8), HTC 8x (8).
- **Blackberry 6-10** – Probado en BlackBerry Torch 9800 (6) y Style 9670 (6), BlackBerry® Torch 9810 (7), BlackBerry Z10 (10).
- **Blackberry Playbook (1.0-2.0)** – Probado en BlackBerry PlayBook.
- **Palm WebOS (1.4-3.0)** – Probado en Palm Pixi (1.4), Pre (1.4), Pre 2 (2.0), HP TouchPad (3.0).
- **Firefox Mobile 18** – Probado en dispositivos Android 2.3 and 4.1.
- **Chrome for Android 18** – Probado en dispositivos Android 4.0 and 4.1.
- **Skyfire 4.1** – Probado en dispositivos Android 2.3.
- **Opera Mobile 11.5-12** – Probado en dispositivos Android 2.3.
- **Meego 1.2** – Probado en Nokia 950 and N9.
- **Samsung Bada 2.0** – Probado en Samsung Wave 3, Dolphin browser.
- **UC Browser** – Probado en dispositivos Android 2.3.
- **Kindle 3, Fire, and Fire HD** – Probado en el built-in WebKit browser de cada una.
- **Nook Color 1.4.1** – Probado en Nook Color, no en Nook Tablet.
- **Chrome Desktop 16-24** – Probado en OS X 10.7 y Windows 7.
- **Safari Desktop 5-6** – Probado en OS X 10.8

- **Firefox Desktop 10-18** – Probado en OS X 10.7 y Windows 7.
- **Internet Explorer 8-10** – Probado en Windows XP, Vista y 7, Windows Surface RT.
- **Opera Desktop 10-12** – Probado en OS X 10.7 y Windows 7.

B) Grado B

Posee la experiencia completa, excepto las funciones de navegación Ajax.

- **Opera Mini 7** – Probado en iOS 6.1 y Android 4.1.
- **Nokia Symbian³** – Probado en Nokia N8 (Symbian³), C7 (Symbian³), también funciona en N97 (Symbian¹).

C) Grado C

Muestra una funcionalidad HTML básica.

- **Internet Explorer 7 e inferior** – Probado en Windows XP
- **Apple iOS 3.x e inferior** – Probado en iPhone (3.1), iPhone 3 (3.2)
- **Blackberry 4-5** – Probado en Curve 8330 (4), Storm 2 9550 (5), y Bold 9770 (5)
- **Windows Mobile** – Probado en HTC Leo (WinMo 5.2)

2.5.6 Documentación

La documentación oficial de este framework se encuentra en <http://api.jquerymobile.com/>, donde existe información sobre los elementos CSS, eventos, íconos, widgets, entre otros.

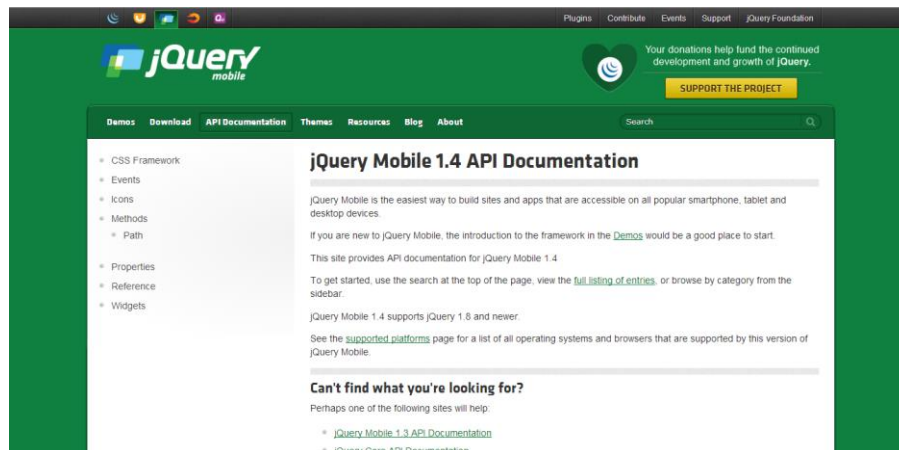


Figura 2-23. Sitio de documentación oficial de jQuery Mobile

Fuente: Propia

El sitio oficial para el aprendizaje de jQuery, jQuery UI y jQuery Mobile encontrado en <http://learn.jquery.com/>, ofrece una guía para resolver inconvenientes, fomentar conocimientos y ayudar en el proceso de desarrollo con estas herramientas. Además existe un tutorial completo que ofrece el sitio de aprendizaje de la W3C (W3Schools), donde aprenderá a usar jQuery Mobile para sus proyectos, dentro de la dirección <http://www.w3schools.com/jquerymobile/default.asp>.

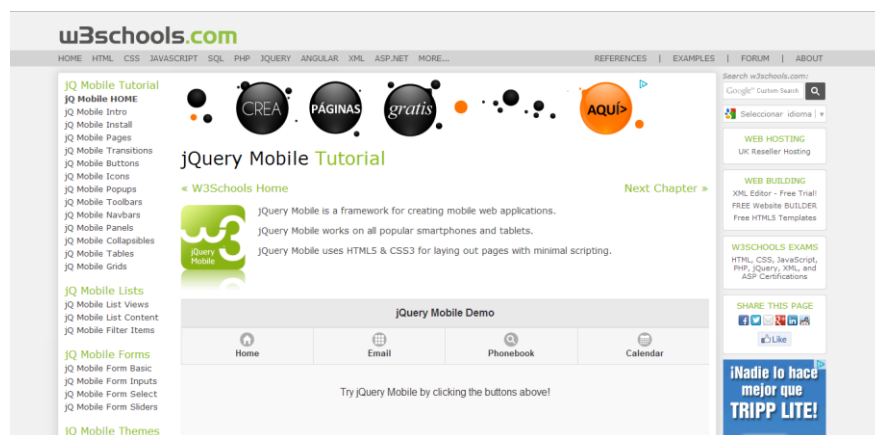


Figura 2-24. Sitio del tutorial de jQuery Mobile de la W3C

Fuente: Propia

2.5.7 Estructura básica

Antes de empezar se debe conocer que este framework se maneja mediante páginas, que serán cada una de pantallas dentro de la interfaz del aplicativo.

Para crear aplicaciones con jQuery Mobile se debe partir de una estructura HTML5 vacía, y añadir los recursos CSS y JavaScript correspondientes.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.css">
  <script src="http://code.jquery.com/jquery-1.10.2.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.js"></script>
</head>
<body>
</body>
</html>
```

Figura 2-25. Estructura inicial de jQuery Mobile

Fuente: Propia

Una vez establecida la estructura base se añade al cuerpo del documento HTML las páginas que se crea conveniente; la cabecera, cuerpo y pie de las mismas; y sus respectivos enlaces hacia otras páginas.

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.css">
  <script src="http://code.jquery.com/jquery-1.10.2.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.4.2/jquery.mobile-1.4.2.min.js"></script>
</head>
<body>
  <div data-role="page" id="pageuno">
    <div data-role="header">
      <h1>Página Principal</h1>
    </div>
    <div data-role="main" class="ui-content">
      <p>Bienvenidos...</p>
      <a href="#pagesdos">Ir a la página 2</a>
    </div>
    <div data-role="footer">
      <h1>Pie Uno</h1>
    </div>
  </div>

  <div data-role="page" id="pagesdos">
    <div data-role="header">
      <h1>Página 2</h1>
    </div>
    <div data-role="main" class="ui-content">
      <p>Esta es la segunda página </p>
      <a href="#pageuno">Regresar</a>
    </div>
    <div data-role="footer">
      <h1>Pie Dos</h1>
    </div>
  </div>
</body>
</html>
```

Figura 2-26. Estructura básica multi-página de jQuery Mobile

Fuente: Propia

2.6 PLATAFORMA APACHE CORDOVA

El desarrollador de software (Wargo, 2013), explica lo siguiente sobre Apache Cordova:

Apache Cordova (<http://cordova.apache.org>) es un framework gratuito de código abierto, usado para la construcción de aplicaciones multiplataforma nativas con HTML5. Los creadores de Apache Cordova querían una manera más sencilla de crear aplicaciones móviles multiplataforma, y decidieron crearlo como una combinación de tecnologías nativas y aplicaciones web. Este tipo de aplicación móvil se denomina una aplicación híbrida.

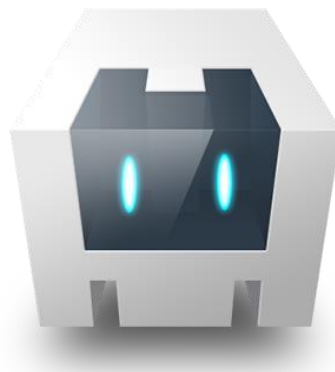


Figura 2-27. Logo de Apache Cordova

Fuente: <http://bit.ly/1PN6NnS>

La ventaja inicial de Apache Cordova es usar la tecnología web (HTML5, CSS3 y JavaScript) en el desarrollo de aplicaciones móviles que se ejecuten dentro navegador del dispositivo como software nativo. Cordova implementa un conjunto de APIs que extiende el uso de las capacidades nativas de los

dispositivos (tales como la cámara, acelerómetro, la aplicación contactos, entre otras.) para la creación de una aplicación.

2.6.1 Funcionamiento

El paquete de aplicación Cordova (aplicación web), es puesto dentro de un contenedor; luego este es probado y depurado, llevando a cabo la transformación en un software nativo para distintos sistemas operativos móviles.

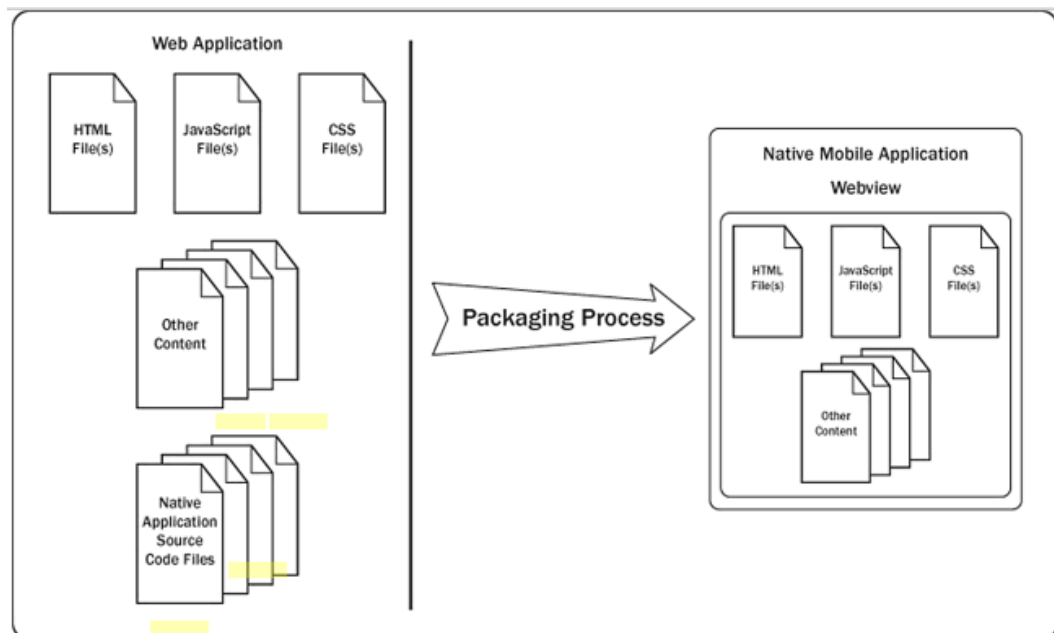


Figura 2-28. Proceso de empaquetamiento de Cordova

Fuente: (Wargo, 2013)

Dentro de la aplicación nativa, se muestra una interfaz de usuario con una sola pantalla que consume todo el espacio del dispositivo. Cuando la aplicación inicia se carga la página web de inicio (index.html) en la WebView, y finalmente llega al controlador de vista web para permitir la interacción con el usuario mediante el navegador.

A medida que el usuario interactúa con la aplicación, se irá ejecutando todos los códigos JavaScript para que la aplicación cargue los archivos y recursos de la misma. El funcionamiento de esta aplicación es similar al de una desarrollada con código nativo.

2.6.2 Arquitectura

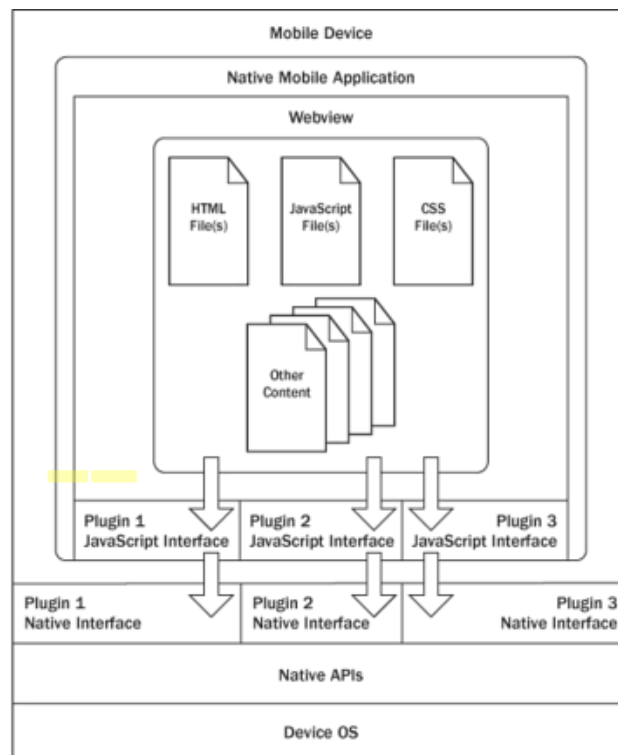


Figura 2-29. Arquitectura de una aplicación Apache Cordova

Fuente: (Wargo, 2013)

La arquitectura de una aplicación realizada con Apache Cordova consta de dos partes principales:

- El JavaScript de la lógica de negocios, que impulsa la interfaz de usuario y su funcionalidad.

- El JavaScript que accede y controla el dispositivo (teléfono inteligente o tableta).

Para entender un poco más sobre la arquitectura de la plataforma consideraremos como ejemplo la aplicación móvil de Facebook.

Las partes principales de la aplicación serían la página de acceso y la descarga de las galerías de fotos (código JavaScript de la lógica de negocio). Ahora si se desea agregar un módulo donde se pueda tomar a foto y subirla a nuestro perfil Facebook, se usa la API cámara (código JavaScript de control del dispositivo) para obtener acceso a la cámara del teléfono, tomar una foto, y obtener el archivo de imagen. El último paso es una llamada AJAX al servidor de Facebook, con el fin de cargar la imagen.

2.6.3 Phonegap y Apache Cordova

La relación entre Phonegap y Apache Cordova es según (Natili, 2013), es la siguiente:

PhoneGap fue desarrollado originalmente por Nitobi, una empresa adquirida por Adobe en 2011, después de que fue adquirido Nitobi donó la base de código PhoneGap a la Apache Software Foundation (ASF) bajo el nombre de proyecto Cordova; que es el nombre de la calle en Vancouver, donde oficinas de Nitobi fueron localizados y donde la compañía creó la primera versión de la plataforma.

Una de las mayores ventajas de mover la base de código a la ASF es que las grandes organizaciones puedan contribuir fácilmente al proyecto;

Además, el proyecto está ahora bajo un gobierno abierto y transparente (su comunidad).

PhoneGap es una distribución libre y abierta con licencia de Apache Cordova. Cordova pasa a ser el motor sobre el que PhoneGap y sus servicios relacionados (debug, emulan y construir servicios) se construyen.

Adobe sigue desempeñando un papel importante en el proyecto, con una inversión en su desarrollo en curso, y la compañía decidió mantener el nombre PhoneGap para describir su propia distribución del proyecto de Cordova. Otros colaboradores del proyecto Apache Cordova incluyen Google, RIM, Microsoft, IBM, Nokia, Intel y Hewlett-Packard.

2.6.4 Componentes Básicos

Dentro de la documentación oficial de la plataforma, la organización (Apache Software Foundation, 2014) destaca los siguientes componentes:

Las aplicaciones Apache Cordova se basan en un archivo común ***config.xml***, que proporciona información sobre la aplicación y especifica los parámetros que afectan la forma en que funciona; por ejemplo, si responde a los cambios de orientación. Este archivo se adhiere a la especificación W3C Web Packaged App.

La aplicación en sí se implementa como una página web ***index.html***, llamada por defecto, que contiene referencias a archivos CSS, archivos JavaScript, imágenes, multimedia u otros recursos que son necesarios para que se ejecute.

La aplicación se ejecuta como un WebView dentro de la envoltura de aplicación nativa, que se distribuye en las tiendas de aplicaciones. El WebView habilitado Cordova puede proporcionar la aplicación de la totalidad de su interfaz de usuario.

Una variedad de plugins están disponible para que Cordova pueda comunicarse con componentes nativos. Esto permite invocar código nativo de JavaScript. Desde la versión 3.0, los plugins proporcionan enlaces a las API del dispositivo estándar. Plugins de terceros proporcionan enlaces adicionales a funciones no necesariamente disponibles en todas las plataformas. Estos plugins de terceros se encuentran disponibles en el registro de plugins. Además se puede desarrollar propios plugins, que sean necesarios para utilizar características nativas.

2.6.5 Plugins

Como se explicó anteriormente, Apache Cordova posee un sinnúmero de plugins JavaScript que permiten la comunicación con componentes de hardware del dispositivo.

Las funcionales principales que cubren estos plugins son:

- **Acelerómetro:** Sirve para monitorear el movimiento, orientación y captar agitaciones del dispositivo.
- **Camera:** Sirve para que los usuarios puedan tomar fotos dentro de la aplicación o acceder a fotos almacenadas.
- **Captura:** Captura audio y video.

- **Compass:** Obtiene la dirección en la que el dispositivo está apuntando.
- **Conexión:** Sirve para conocer si existe una conexión a la red.
- **Dispositivo:** Accede a los metadatos del dispositivo.
- **Eventos:** Permite soportar eventos relacionados con el dispositivo como la batería.
- **Ficheros:** Permite el acceso al sistema de ficheros del dispositivo.
- **Geolocalización:** Proporciona información acerca de la ubicación del dispositivo.
- **Globalización:** Obtiene información para realizar operaciones específicas de configuración regional y zona horaria.
- **InAppBrowser:** Es un navegador web que aparece en la aplicación al llamar ***window.open***.
- **Media:** Ofrece la posibilidad de grabar y reproducir archivos de audio.
- **Notification:** Permite notificaciones visuales, audibles y táctiles.
- **SplashScreen:** Muestra y oculta la SplashScreen de la aplicación.
- **Storage:** Proporciona acceso a las opciones de almacenamiento del dispositivo.

2.6.6 Soporte de la plataforma

Apache Cordova compatible con los siguientes sistemas operativos móviles como: Android (Google), Bada (Samsung), BlackBerry 10 (BlackBerry), iOS

(Apple), Firefox OS, Tizen (Linux Foundation), Windows Phone 7(Microsoft), Windows Phone 8 (Microsoft), y Windows 8 (Microsoft).

Además presenta soporte de las diferentes API's, presentado en la figura:

| | amazon-fireos | android | blackberry10 | Firefox OS | ios | Ubuntu | wp8 (Windows Phone 8) | win8 (Windows 8) | tizen |
|--------------------------|-----------------------|-----------------------|-----------------|-----------------------|-----------------|----------|----------------------------|----------------------------|-------|
| cordova CLI | ✓ Mac, Windows, Linux | ✓ Mac, Windows, Linux | ✓ Mac, Windows | ✓ Mac, Windows, Linux | ✓ Mac | ✓ Ubuntu | ✓ Windows | ✓ | ✗ |
| Embedded WebView | ✓ (see details) | ✓ (see details) | ✗ | ✗ | ✓ (see details) | ✓ | ✗ | ✗ | ✗ |
| Plug-in Interface | ✓ (see details) | ✓ (see details) | ✓ (see details) | ✗ | ✓ (see details) | ✓ | ✓ (see details) | ✓ | ✗ |
| Platform APIs | | | | | | | | | |
| Accelerometer* | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Camera* | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Capture* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✗ | ✗ |
| Compass* | ✓ | ✓ | ✓ | ✗ | ✓ (3GS+) | ✓ | ✓ | ✓ | ✓ |
| Connection* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Contacts* | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | ✗ |
| Device* | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Events | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ |
| File* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✗ |
| Geolocation* | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Globalization* | ✓ | ✓ | ✗ | ✗ | ✓ | ✓ | ✓ | ✗ | ✗ |
| InAppBrowser* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | uses iframe | ✗ |
| Media* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Notification* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Splashscreen* | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✗ |
| Storage | ✓ | ✓ | ✓ | ✗ | ✓ | ✓ | ✓ localStorage & indexedDB | ✓ localStorage & indexedDB | ✓ |

Figura 2-30. Soporte de Apache Cordova

Fuente: (Apache Software Foundation, 2014)

2.7 OPENSIFT



Figura 2-31. OpenShift logo

Fuente: <http://bit.ly/1PN74af>

“OpenShift es la plataforma como servicio (PaaS) de Red Hat, que permite a los desarrolladores crear, gestionar y desplegar rápidamente aplicaciones escalables en un entorno de nube.” (Red Hat Inc., 2014)

2.7.1 OpenShift Runtime Environments

OpenShift Online es compatible con una variedad de lenguajes de programación y runtimes. Entre los principales están:

- **Java.-** Posee soporte para el desarrollo de aplicaciones que utilicen JBoss AS, JBoss, JBoss EAP, Apache Tomcat, y Spring.
- **PHP.-** OpenShift hace uso de diferentes frameworks PHP como son: CodeIgniter, CakePHP y Yii. Además posee una amplia selección de PHP quickstarts, generando aplicaciones en Wordpress, Drupal y Magento.
- **Node.js.-** Una plataforma construida en JavaScript para la construcción de aplicaciones escalables en tiempo real.
- **Python.-** Permite implementar y ejecutar fácilmente aplicaciones Python usando sus marcos y bases de datos preferidas.
- **Ruby.-** Permite la creación de aplicaciones robustas con Ruby.
- **Vert.x.-** Una plataforma ligera y con alto rendimiento para el desarrollo de aplicaciones JVM modernas, móviles, web, y empresariales.

2.7.2 Bases de datos de OpenShift

OpenShift brinda los siguientes tipos de bases de datos dentro de su catálogo:

- **MySQL.-** Posee este famoso sistema de gestión de bases de datos SQL relacional, multi-hilo y multi-usuario. Además implementa phpMyAdmin para la administración de las bases de datos.
- **MongoDB.-** Una base de datos NoSQL escalable por 10gen. Permite instalar el cartridge RockMongo para la administración de la instancia en la web.
- **PostgreSQL.-** Presenta este sistema de gestión de bases de datos objeto-relacionales avanzado y libre.

2.7.3 Versiones de OpenShift

Este PaaS posee tres diferentes tipos de versiones: OpenShift Origin, OpenShift Online, y OpenShift Enterprise.

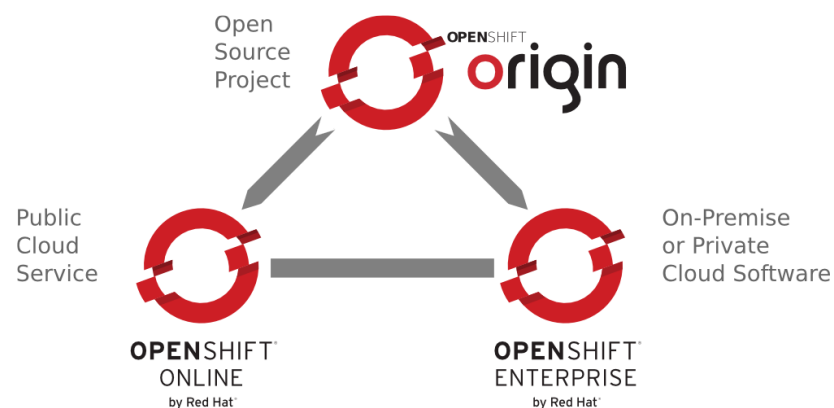


Figura 2-32. Relación entre las tres versiones de OpenShift

Fuente: (Pousty & Miller, 2014)

- **Openshift Origin.-** Es el proyecto gratuito y de código abierto de OpenShift, que engloba las otras dos versiones. Se encuentra en GitHub bajo una licencia Apache 2. Todos los cambios de código que se realizan por Red Hat y otros desarrolladores externos, son puestos

dentro del repositorio. Solo basta descargarlo e instalarlo en tu servidor o nube para utilizarlo.

- **Openshift Online.-** Al crear una cuenta dentro del sitio <https://www.openshift.com>, Red Hat se encarga de organizar toda la lógica PaaS dentro de Amazon Web Services (AWS); es decir, la actualización del sistema operativo y gestión de redes, está cubierta por el equipo de operaciones OpenShift. Logrando que los desarrolladores solo se concentre en la codificación.
- **OpenShift Enterprise.-** Es el PaaS privado de OpenShift, orientado hacia los clientes que requieren escalabilidad; es decir, aumentar la eficiencia operativa y ampliar la utilización del hardware. Este paquete le permite tomar el servicio de PAAS y ejecutarlo en cualquier lugar deseado (servidores propios, Rackspace o AWS).

2.7.4 Planes de Openshift Online

Openshift Online presenta tres tipos de planes con diferentes especificaciones: Free, Bronze y Silver. Las diferencias entre los planes serán expuestas en la siguiente tabla:

| | FREE | BRONZE | SILVER |
|--|----------|---------|-------------|
| Precio Base | Libre | Libre | \$ 20 / mes |
| Funciones centrales | | | |
| Administración del sistema por Red Hat | X | X | X |
| Aplicación autoescalable | X | X | X |
| Máximo de Gears | 3 | 16 | 16+ |
| Reposo de aplicación | 24 horas | Nunca | Nunca |
| Gears básicos | | | |
| Región Hosting | EE.UU. | EE.UU. | EE.UU. |
| Incluido gratis | 3 gears | 3 gears | 3 gears |

| | pequeños | pequeños | pequeños |
|--------------------------------|--------------|-------------------------|-------------------------|
| Pequeño (4+) | ~ | \$ 0.02 / hora | \$ 0.02 / hora |
| Gears de producción | | | |
| Región Hosting | No aplicable | Estados Unidos y Europa | Estados Unidos y Europa |
| Small highcpu | ~ | \$ 0,025 / hora | \$ 0,025 / hora |
| Medio | ~ | \$ 0.05 / hora | \$ 0.05 / hora |
| Grande | ~ | \$ 0.10 / hora | \$ 0.10 / hora |
| Almacenamiento | | | |
| Base (por gear) | 1GB | 1GB | 6GB |
| Adicional | ~ | \$ 1.00 / GB / mes | \$ 1.00 / GB / mes |
| Dominios y SSL | | | |
| Subdominios en rhcloud.com | X | X | X |
| SSL compartido en rhcloud.com | X | X | X |
| Dominios personalizados | X | X | X |
| Certificados SSL privadas | ~ | X | X |
| Colaboración | | | |
| Controles compartidos | X | X | X |
| Gestión de equipos | ~ | X | X |
| Soporte de la Comunidad | | | |
| Centro de ayuda | X | X | X |
| Documentación y guías | X | X | X |
| Comunidad de desarrolladores | X | X | |
| Soporte Premium | | | |
| Preguntas de facturación | ~ | X | X |
| Acceso al Portal del Cliente | ~ | ~ | X |
| Basado en Ticket | ~ | ~ | X |
| Teléfono | ~ | ~ | X |

Tabla 2-5. OpenShift Online Pricing

Fuente: <https://www.openshift.com/products/pricing>

2.8 METODOLOGÍA XP

“Extreme Programming (XP), es una disciplina de desarrollo de software basada en los valores: simplicidad, comunicación retroalimentación, coraje y respeto. Su acción consiste en llevar todo el equipo unido mediante prácticas

sencillas, sincronizando las prácticas a su situación particular”. (Jeffries, 2014)

XP al ser parte de las metodologías ágiles está diseñada para desarrollar software rápidamente, llevándolo al cliente dentro de tiempo que lo necesita. Además hace que los desarrolladores de software respondan al cambio de requerimientos de los clientes durante todo el proceso de desarrollo.

Dentro de XP tanto gerentes, como clientes, como diseñadores y desarrolladores; es decir, cada contribuyente al proyecto es una parte del “Equipo completo” y hace que este logre un software de calidad.

2.8.1 Agile Manifesto

(Blankenship, Bussa, & Millett, 2011), relatan sobre el Agile Manifesto:

En la década de 1990 había varias personas que estaban hablando acerca de cambiar la forma en que escribimos software. Estas conversaciones llegaron a un punto crítico en 2001, cuando una serie de luminarias de desarrollo de software, incluyendo la talla de Martin Fowler, Kent Beck, Bob Martin, Ken Schwaber, Jess Sutherland, y Dave Thomas se reunieron en una casa de campo en la estación de esquí de Snowbird en el Montañas Wasatch de Utah. Lo que salió de este encuentro se hizo conocido como el Manifiesto Ágil.



Figura 2-33. Agile Manifesto

Fuente: (Blankenship, Bussa, & Millett, 2011)

Además del manifiesto, doce principios de agilidad se crearon para ampliar la declaración del manifiesto.

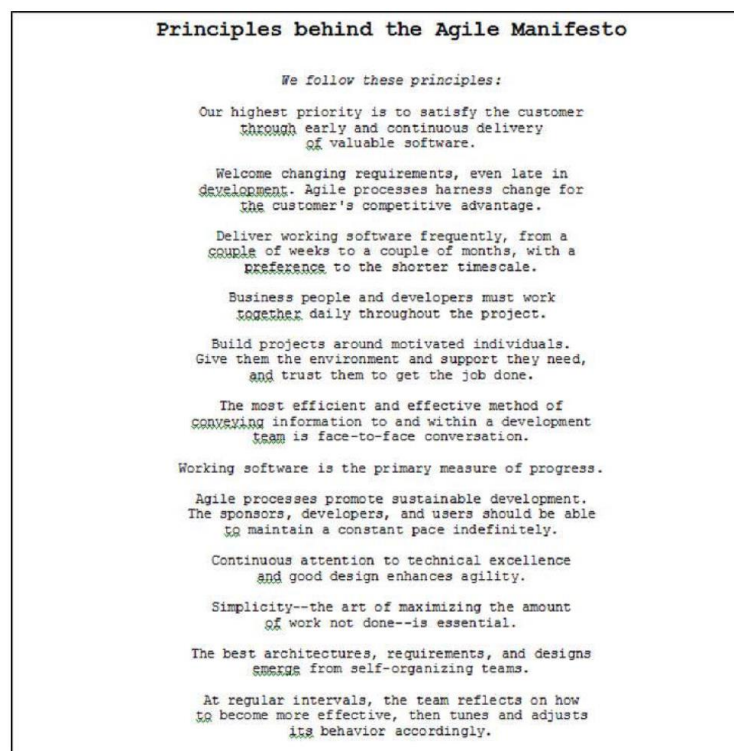


Figura 2-34. Los 12 principios creados con el manifiesto

Fuente: (Blankenship, Bussa, & Millett, 2011)

Los principios mencionados con anterioridad son los siguientes:

- Satisfacer al cliente a través de entregas continuas y tempranas es la mayor prioridad.
- Los cambios a los requerimientos son bienvenidos, aún en fases tardías del desarrollo.
- Entregar frecuentemente software que funciona, desde un par de semanas a un par de meses, prefiriendo los periodos más cortos.
- Desarrolladores, gerentes y clientes deben trabajar juntos diariamente, a lo largo del proyecto.
- Construir proyectos alrededor de personas motivadas, dándoles el entorno y soporte que necesitan, y confiando en que realizarán el trabajo.
- El método más eficiente y efectivo de transmitir información entre un equipo de desarrolladores es la conversación frontal (cara a cara).
- Tener software que funciona es la medida primaria del progreso.
- El proceso ágil promueve el desarrollo sostenible. Los sponsors, desarrolladores y usuarios deben ser capaces de mantener un ritmo de
- trabajo constante en forma permanente a lo largo del proyecto.
- La atención continua a la excelencia técnica y el buen diseño mejoran la agilidad.
- La simplicidad es esencial.

- Las mejores arquitecturas, requerimientos y diseños surgen de los equipos auto-organizados.
- A intervalos regulares, el equipo debe reflexionar sobre cómo ser más efectivos, y ajustar su comportamiento de acuerdo a ello.

2.8.2 Bases de la programación extrema

Según (Bahit, 2012) la programación extrema posee los siguientes valores:

- **Comunicación.-** En XP, todo es trabajado en equipo: desde el relevamiento y análisis hasta el código fuente desarrollado. Todo se conversa cara a cara, procurando hallar soluciones en conjunto a los problemas que puedan surgir.
- **Simplicidad.-** Se pretende desarrollar solo lo necesario y no perder tiempo en detalles que no sean requeridos en el momento. En este aspecto, se asemeja a otra metodología ágil, denominada Kanban, en la cual, un proceso “anterior” solo produce lo que el proceso posterior demanda.
- **Retroalimentación.-** El objetivo de XP es entregar lo necesario al cliente, en el menor tiempo posible. A cambio, demanda al cliente, un feedback continuo -retroalimentación-, a fin de conocer sus requerimientos e implementar los cambios tan pronto como sea posible.
- **Respeto.-** El equipo respeta la idoneidad del cliente como tal (sólo éste, es quien conoce el valor para el negocio) y el cliente, a la vez, respeta la idoneidad del equipo (confiando en ellos

profesionalmente para definir y decidir el “cómo” se llevará a cabo el desarrollo de lo requerido).

- **Coraje.-** Se dice que en XP un equipo debe tener el valor para decir la verdad sobre el avance del proyecto y las estimaciones del mismo, planificando el éxito en vez de buscar excusas sobre los errores.

2.8.3 Prácticas

Según (Blankenship, Bussa, & Millett, 2011), la metodología XP posee las siguientes prácticas:

1. **Planificando el juego:** El juego de la planificación captura las características del sistema como las historias de usuario, y define las versiones del proyecto. Durante la planificación de la iteración, historias de usuarios seleccionados para la iteración se descomponen en tareas de desarrollo.
2. **Metáfora del sistema:** En la comunicación con la empresa, ayuda a tener un lenguaje adecuado para que los sistemas complejos puedan ser explicados fácilmente.
3. **Diseño simple:** el diseño del sistema de alto nivel se produce al inicio y durante una iteración.

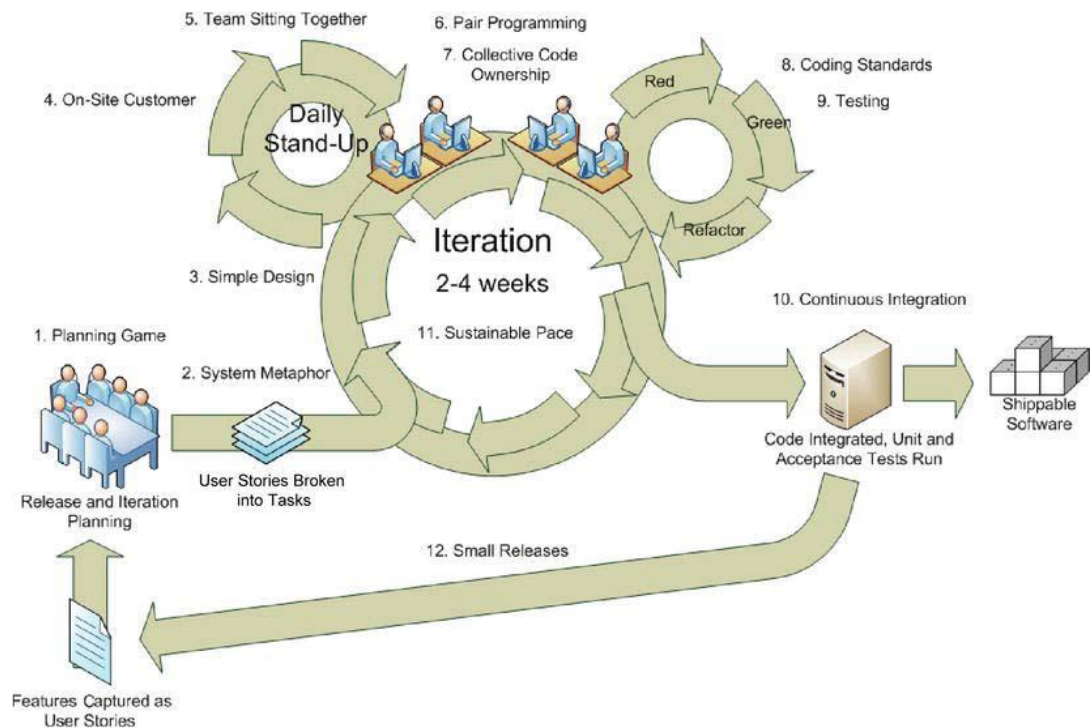


Figura 2-35. Prácticas de XP

Fuente: (Blankenship, Bussa, & Millett, 2011)

4. **Sitio de Atención al Cliente:** Durante una iteración, es ideal que el cliente esté en el lugar para ayudar a responder rápidamente los detalles de una historia de usuario. Aunque las reuniones físicas son ideales, las reuniones virtuales son mejor que ninguna reunión en absoluto. El punto importante es tener un cliente disponible que puede proporcionar rápidamente las respuestas como los desarrolladores exploran los detalles de una historia de usuario.
5. **Equipo sentado en conjunto:** Es importante para todo el equipo estar comprometido con el proyecto y mantenerse junto el uno del otro. Esto mejora la comunicación y transmite una sensación de apoyo.

- 6. Programación en parejas:** Es una práctica de desarrollo para que pares de desarrolladores puedan trabajar conjuntamente en un problema. Un par compartirá una máquina de desarrollo, y mientras unen los códigos, el otro le ayudará con las decisiones de diseño y mirar hacia adelante sobre la siguiente función. Con la programación en parejas, las revisiones de código se convierten en "tiempo real", y es poco probable que se pase por alto errores en la programación.
- 7. Trabajo colectivo del código:** A través de estar sentados juntos y la programación en parejas, todo el equipo comparte una propiedad colectiva de la base del código. A todos los desarrolladores se les permite fijar y trabajar en cualquier parte del código.
- 8. Estándares de Codificación:** Para ayudar con la propiedad colectiva, se utilizan las mejores prácticas para mantener un diseño simple y garantizar que se crea todo el código de una manera consistente. Esto asegura que el código fuente se puede entender rápidamente al trabajar en cualquier parte del sistema y con cualquier otro desarrollador.
- 9. Pruebas:** Para garantizar la calidad de entrega al cliente, se hace hincapié en las pruebas a través del proceso de XP. Primero, se identifican los criterios de aceptación de las historias de usuario. Estos criterios de aceptación se utilizan para escribir las pruebas unitarias y comenzar el desarrollo en el estilo TDD. Las pruebas de

aceptación del usuario también se deriva de los criterios de aceptación y serán automatizadas tanto como sea posible.

10. Integración continua: Para asegurarse de que todo el código que el equipo está desarrollando en conjunto realmente funciona, el equipo se integra a menudo y con tiempo. Así se logra corregir errores con más facilidad.

11. Ritmo sostenible: Debido que dentro de los ciclos de lanzamientos cortos y la naturaleza iterativa de XP, puede suceder algún inconveniente. Esto significa que los problemas encontrados después de la revisión y las pruebas pueden ser incorporados en la siguiente iteración. Este modelo de retroalimentación rápida asegura que los desarrolladores pueden trabajar a un ritmo sostenible. De este modo, se ahorra largas semanas en comparación con las metodologías de desarrollo más tradicionales que han etapas claramente definidas para el ciclo de vida del desarrollo, de votar y prueba hasta el final. El ritmo sostenible también ayuda a la gestión para planificar de manera más eficiente para las vacaciones del personal, ausencias imprevistas, y la producción ocasional.

12. Pequeños Releases: XP tiene que ver con la satisfacción del cliente y la entrega de valor de negocio a través de software de calidad. En lo que se refiere al negocio, más a menudo esto puede suceder, mejor. XP promueve lanzamientos frecuentes, que pueden ser relativamente pequeños, pero ponen en relieve las

características priorizados por el negocio. Esta transparencia de versiones frecuentes da ánimos a los clientes, mostrándoles que el equipo está agregando valor al proyecto a lo largo del camino.

2.8.4 Roles

Dentro de XP se encuentran los siguientes roles:

- **Programador.-** Es el responsable de transformar los requerimientos de los clientes en código perteneciente al sistema. Los programadores estiman los tiempos de desarrollo de cada historia de usuario y las tareas de las mismas.
- **Cliente.-** El cliente representa al usuario final; es decir, una empresa, un proyecto o un usuario cualquiera. A partir de él nacen los requerimientos del sistema, que están escritos en forma de historias de usuario.

Mediante el juego de la planificación, el cliente elige el orden en que las historias serán realizadas. También define las pruebas de aceptación para demostrar que el sistema es fiable y hace lo que requiere.
- **Encargado de pruebas (Tester).-** Tiene la responsabilidad de ayudar al cliente a definir e implementar las pruebas de aceptación de las historias de usuario. También es responsable de ejecutar las pruebas con frecuencia y la publicación de los resultados para todo el equipo pueda verlos.

- **Encargado de seguimiento (Tracker).**- Esta persona está encargada de dar seguimiento al plan de liberación (historias de usuario), al plan de iteración (tareas) y a las pruebas de aceptación. Un buen tracker tiene la capacidad de recoger la información sin perturbar el proceso de manera significativa.
- **Entrenador (Coach).**- Dentro de XP es el guía y mentor del equipo (cliente, desarrolladores, organización). Un entrenador trae una perspectiva externa para ayudar al equipo a ver el trabajo con más claridad. Las decisiones tomadas por los entrenadores siempre deben provenir de los valores y prácticas de XP; siendo estas un requisito previo. El entrenador debe poseer habilidades de comunicación y ser eficaz para influir en las acciones de los equipos.
- **Consultor.**- Es aquel que posee conocimientos de algún tema en específico, y brinda al equipo soporte sobre este.
- **Gestor (Big boss).**- Es el jefe del proyecto encargado de: establecer una buena relación entre usuarios y desarrolladores, cubrir las necesidades del equipo y asegurar el alcance del proyecto.

2.8.5 Ciclo de vida

La metodología XP posee las siguientes fases:

1. **Exploración:** En esta fase definimos el alcance del proyecto. El cliente describe al proyecto mediante la redacción de las “historias

de usuarios". Dadas las historias el grupo de programadores estiman el tiempo de desarrollo de las mismas. El tiempo de duración de esta fase es generalmente de dos semanas.

2. **Planificación:** Es una fase corta del proceso de XP donde el cliente, el jefe del proyecto y los desarrolladores se ponen de acuerdo sobre el tiempo y evolución del proyecto. Esto se realiza mediante algunas reuniones donde se entiende el contexto del negocio del software.
3. **Diseño.-** El diseño debe ser basado en el principio MS (mantenerlo sencillo), para esto se crea un prototipo base que será evaluado y rediseñado en cada iteración conforme existan sugerencias del cliente.

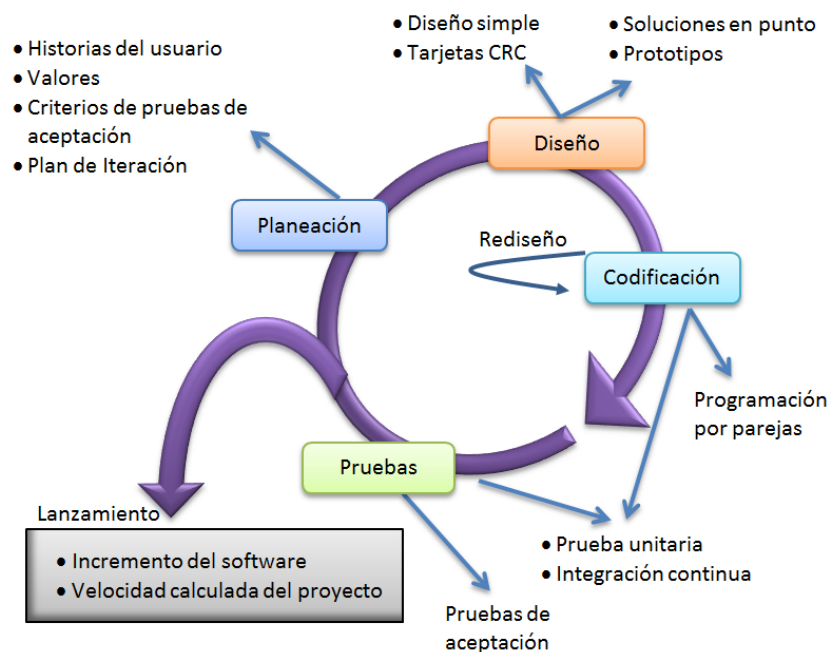


Figura 2-36. Ciclo de vida de la metodología XP

Fuente: <http://bit.ly/1Hnh6Qn>

4. Codificación y Pruebas.- Ya establecidas las historias de usuario y el diseño preliminar, el equipo pone en marcha desarrollo, sin descuidar la realización de pruebas durante el proceso; las cuales permiten capacitarse mejor para resolver la problemática de cada historia.

Existen varios tipos de pruebas dentro del enfoque de XP. Las pruebas unitarias que describen el diseño de las clases y los métodos, y las pruebas de aceptación que especifican las características y funcionalidades del sistema dadas por el cliente.

5. Puesta en producción.- Se trata de la entrega final y completa del software al cliente. En esta fase no se realizan más desarrollos funcionales, pero pueden ser necesarias tareas de ajuste.

CAPÍTULO III

3 TURISMO EN IBARRA

Dentro de este capítulo trataremos sobre la oferta turística que posee la ciudad de Ibarra, y también se dará a conocer información característica del Departamento de turismo del GAD-I.

3.1 ANTECEDENTES

3.1.1 Ibarra



Figura 3-1. Arte de la ciudad de Ibarra 2014 - 2019

Fuente: <http://www.ibarra.gob.ec/>

San Miguel de Ibarra, también conocida como “Ciudad Blanca”, es la capital de la provincia de Imbabura. Se encuentra ubicada a 115 km al noroeste de Quito, capital del Ecuador. Su altura es de 2.220 msnm, y posee una superficie de 1126 km².

Ibarra se encuentra distribuida en 5 parroquias urbanas: San Francisco, Sagrario, Caranqui, Alpachaca y Priorato; y 8 parroquias rurales: San Antonio, La Esperanza, Angochagua, Ambuqui, Salinas, La Carolina, Lita y Cochapamba.

Su entorno natural se compone de montañas, ríos y cascadas con una flora y fauna diversa. La “Ciudad Blanca” posee un hermoso centro histórico con plazas, parques y edificaciones únicas dentro del país. A esto se le suman los servicios de calidad y la infraestructura hotelera, convirtiéndola en un destino ideal para visitar.

A) Clima

Ibarra se caracteriza por la variedad de microclimas que posee, los cuales van desde el frío del páramo en la zona de Angochagua, pasando por cálido húmedo de la zona de Lita y Carolina, hasta el cálido seco del Valle del Chota.

Posee una temperatura promedio de 15.9 grados centígrados, con una variación 0.3 grados. La temperatura mínima varía ente los 7 y 11 grados centígrados. Al tener una variación en las precipitaciones debido a las diferentes alturas, presenta épocas de sequía entre junio y septiembre, con dos épocas adicionales en abril y noviembre.

B) Datos geomorfológicos

Alrededor de la ciudad de Ibarra se encuentran elevaciones montañosas importantes como: el Imbabura, el Cubilche y el Cunro.



Figura 3-2. Principales elevaciones montañosas

Fuente: <http://bit.ly/1Pq7mcn>

C) Datos demográficos

El VII censo nacional de población realizado en el 2010, determina que el total de habitantes dentro del cantón Ibarra es de 181.175 personas, de los cuales 87.786 corresponden a hombres y 93.389 a mujeres. (INEC, 2010)

D) Símbolos

Ibarra posee dos símbolos característicos que son:



Figura 3-3. Símbolos de Ibarra

Fuente: <http://bit.ly/1Pq7nwV>

- **Escudo.-** Se encuentra dividido en tres partes horizontales iguales. En la parte superior se encuentran dos alas y una corona de dos puntas; la parte intermedia posee un recuadro de tres

corazones, y otro con la bandera de Castilla La Vieja; la parte inferior tiene la corona del Marqués de Selva.

- **Bandera.-** Consta de dos franjas horizontales del mismo tamaño, la superior es de color rojo y la inferior de color blanco.

E) Fechas Cívicas

La ciudad de Ibarra celebra las siguientes fechas cívicas:

- Fiestas de El Retorno: 28 de Abril.
- Batalla de Ibarra: 17 de Julio.
- Fundación de Ibarra: 28 de Septiembre.

3.1.2 Historia de Ibarra

La historia de la ciudad de Ibarra, según la (Dirección de Turismo GAD-I, 2012) es la siguiente:

La Villa de San Miguel de Ibarra fue fundada el 28 de Septiembre de 1606, por el Capitán Cristóbal de Troya, bajo la orden del entonces Presidente de la Real Audiencia de Quito, Miguel de Ibarra y bajo mandato del Rey Felipe de España.

Los motivos para la construcción de la ciudad de Ibarra fueron: la necesidad de tener una ciudad de paso en el camino Quito – Pasto y una ciudad con acceso rápido al mar. Es así como el asentamiento y la Villa de San Miguel de Ibarra fue construida en el Valle de Carangue, en los terrenos donados por Juana Atabalipa, nieta del Inca Atahualpa, el último de los Incas y Señor del Tahuantinsuyo.

La Junta Soberana de Quito, el 16 de noviembre de 1811, dio a la Villa de San Miguel de Ibarra el título de ciudad. Posteriormente el 11 de noviembre de 1829, el libertador, Simón Bolívar, nombró a la ciudad capital de la provincia de Imbabura.



Figura 3-4. Retorno a Ibarra

Fuente: <http://www.ibarra.gob.ec/>

En la madrugada del 16 de agosto de 1868, la erupción del volcán Imbabura y el consiguiente terremoto devastaron a la ciudad y la provincia. Ibarra quedó prácticamente destruida y fallecieron cerca de 20.000 personas. Los sobrevivientes de este suceso se trasladaron a los llanos de Santa María de la Esperanza, donde se reubicaron y vivieron por cuatro años.

Gabriel García Moreno, fue comisionado por el Presidente de la República para reconstruir la ciudad. Finalmente el 28 de abril de 1872, los ibarreños regresaron a la reconstruida ciudad. Esta fecha se considera la segunda más importante después de la Fundación de Ibarra.

3.2 TURISMO EN EL CANTÓN

3.2.1 Concepto de turismo

La definición del término turismo dada por la Organización Mundial del Turismo (OMT) es el siguiente: *“Conjunto de actividades de producción y consumo a las que dan lugar determinados desplazamientos de personas seguidos de al menos una noche pasada fuera del domicilio habitual siendo el motivo del viaje el recreo, los negocios, la salud, o la participación en una reunión profesional, deportiva o religiosa”*. (Jiménez Bulla & Jiménez Barbosa, 2013)

Para mayor comprensión diremos que el turismo es el desplazamiento de las personas de un lugar a otro, dentro de un tiempo determinado, dado por el interés de conocer sitios naturales, ciudades, países, culturas, gastronomía, entre otros.

3.2.2 Desarrollo turístico

El cantón Ibarra al estar ubicado en un sitio estratégico del norte del país, está rodeado de una singular riqueza natural, se caracteriza por su diversidad de atractivos, los cuales están ligados a la cultura local y se complementan con los servicios y actividades de aventura y artesanales.

A raíz de la dolarización, Ibarra sufrió un decaimiento en la actividad turística por su debilitamiento del principal mercado (turistas del país Colombia), y la falta de diversificación de su oferta turística. Pero al no haber flujo turístico, sus habitantes cambiaron sus expectativas y adoptaron actividades comerciales para sobrevivir.

Finalmente los habitantes de Ibarra sienten la necesidad de recuperar su liderazgo como destino turístico aprovechando su potencial natural y cultural que engloba todos los hermosos escenarios naturales que permiten la práctica de un gran número de deportes de aventura y actividades turísticas.

3.2.3 Turismo de Aventura

El turismo de aventura consiste en la realización de actividades físicas, que incluyen cierto riesgo real o percibido, relacionadas con la práctica de diversos deportes o actividades desarrolladas en medios naturales o áreas remotas. Para la ejecución de estas actividades es necesario un equipamiento adecuado, servicio de guía, cumplir con normas de seguridad y aplicar medidas de protección ambiental.

Este tipo de turismo está relacionado con los deportes de aventura, los cuales cubren actividades como: pesca, caza, rafting, canopy, montañismo, esquí, puenting, entre otras.



Figura 3-5. Práctica de rafting

Fuente: <http://bit.ly/1B3Q495>

El cantón Ibarra ofrece la práctica de deportes al aire libre, entre los cuales se destacan:

- Senderismo
- Hiking
- Treking
- Cabalgatas
- Bicicleta de montaña
- Parapente
- Rafting
- Canyoning
- Kayaking
- Puenting

3.2.4 Turismo comunitario

El turismo comunitario es aquel que involucra a los grupos étnicos de una región y permite al turista experimentar la vida dentro de una comunidad específica (indígena, afro descendiente o mestiza).

La característica principal de este tipo de turismo es que los benefactores netos son los miembros de las comunidades, logrando que las mismas fortalezcan su desarrollo. Además se encuentra ligado a la conservación del medio ambiente, patrimonios naturales y culturales, y costumbres ancestrales.



Figura 3-6. Comunidad indígena de Zuleta

Fuente: <http://bit.ly/1QS7uib>

El cantón Ibarra brinda a sus visitantes este tipo de turismo dentro de sus parroquias rurales, con el fin de revalorizar las costumbres ancestrales (sabiduría y medicina) y expresiones culturales (música, danza, rituales, mitos, leyendas, entre otras) de las diferentes comunidades.

El turismo comunitario se hace presente en las siguientes parroquias:

- Parroquia de Ambuquí
- Parroquia de Angochagua
- Parroquia de Lita
- Parroquia de Salinas
- Parroquia La Carolina
- Parroquia La Esperanza
- Parroquia San Antonio de Ibarra

3.2.5 Servicios Turísticos

Ibarra cuenta con un catastro turístico que incorpora a todos los establecimientos del cantón que se encuentran registrados como turísticos.



Figura 3-7. Vista nocturna hostería "La Estelita"

Fuente: <http://bit.ly/1Ile5ZE>

En el catastro de establecimientos turísticos constan diferentes grupos de servicios que se ofrecen a turistas nacionales y extranjeros, entre ellos están:

- Agencias de Viajes
- Operadoras Turísticas
- Transporte Turístico
- Hoteles
- Hostales
- Hosterías
- Restaurantes
- Cafeterías
- Bares

- Discotecas
- Fuentes de Soda

Los establecimientos turísticos cumplen con normas establecidas por el Ministerio de Turismo del Ecuador y además reciben un control permanente a través de inspecciones que verifican las instalaciones y el servicio que se entrega. Todo esto con la finalidad de ofrecer a los visitantes un servicio de calidad y garantizado.

3.3 DEPARTAMENTO DE TURISMO GAD-I

La Dirección de desarrollo y planificación turística municipal de Ibarra tiene la responsabilidad de impulsar el desarrollo turístico integral como una de las actividades económicas prioritarias del cantón para posicionarlo como un destino turístico de calidad.

3.3.1 Misión y Visión

3.3.2 Misión

Liderar las acciones tendientes a conseguir un desarrollo armónico y sostenido de las actividades turísticas en el cantón, a través de actividades en concertación con la empresa privada, los gobiernos locales, sector académico y comunitario, tomando en cuenta las ventajas competitivas, desarrollando acciones que potencialicen y faciliten la actividad turística del cantón, para hacer del turismo una de las principales actividades económicas que se proyecte como un producto diferenciado sustentable y de calidad.

3.3.3 Visión

Ofrecer al turista un producto y destino de calidad, respondiendo a las expectativas creadas por la oferta, apoyadas en los atractivos, servicios y facilidades turísticas.

3.3.4 Objetivos departamentales

A) Objetivo General

Impulsar el desarrollo turístico integral como una de las actividades económicas prioritarias del cantón para posicionarlo como un destino turístico de calidad.

B) Objetivos Específicos

- ✓ Promover el desarrollo integral turístico de las parroquias y los atractivos del cantón Ibarra.
- ✓ Mejorar de la gestión de la Dirección de Planificación y Desarrollo Turístico.
- ✓ Dinamizar el mercadeo, promoción y apoyo a la comercialización turística del cantón Ibarra.
- ✓ Mejorar la calidad de los servicios turísticos y fomentar el desarrollo turístico integral en el centro de ciudad de Ibarra
- ✓ Mejorar los servicios y beneficios brindados a través de la Dirección de Planificación y Desarrollo Turístico a los establecimientos turísticos del cantón.

3.3.5 Organigrama

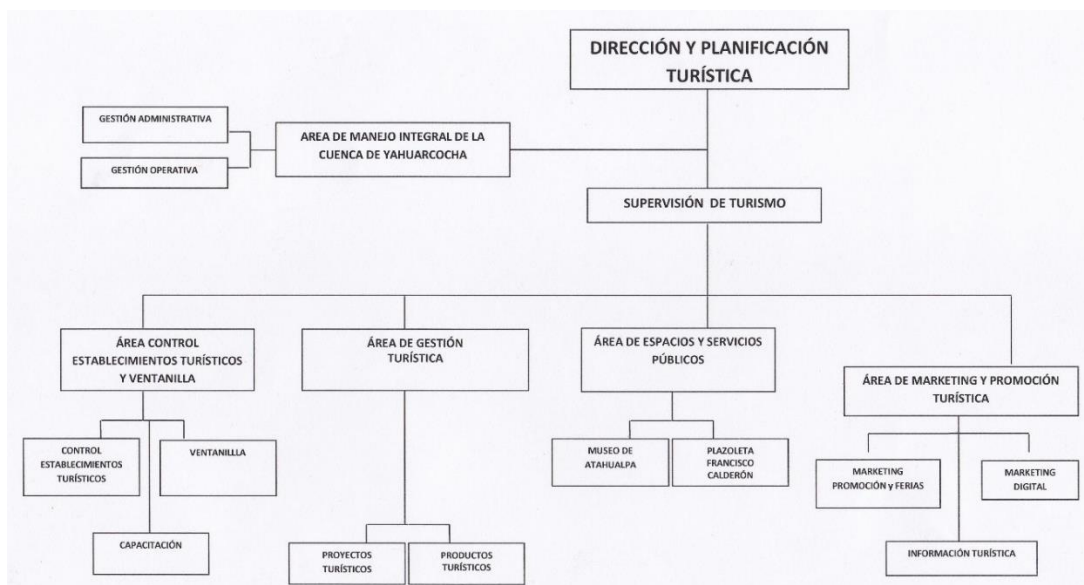


Figura 3-8. Organigrama del Departamento de Turismo GAD-I

Fuente: Departamento de Turismo GAD-I.

3.3.6 Procesos

A) Actualización y control de catastro turístico

Permite conservar el documento de catastro de establecimientos turísticos actualizado, logrando un mejor control de las actividades turísticas de los establecimientos.

Roles y responsabilidades

Tabla 3-1. Roles y responsabilidades para el proceso de actualización y control de catastro turístico

| Rol | Responsabilidad |
|--|---|
| Director de Turismo | El Director del departamento realiza la revisión de los informes entregados de las fichas de cada inspección. |
| Asistente de Mejoramiento Turístico | <ul style="list-style-type: none"> Realiza inspecciones de control de calidad. Emite informes técnicos. Actualiza el catastro turístico del cantón Ibarra. |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

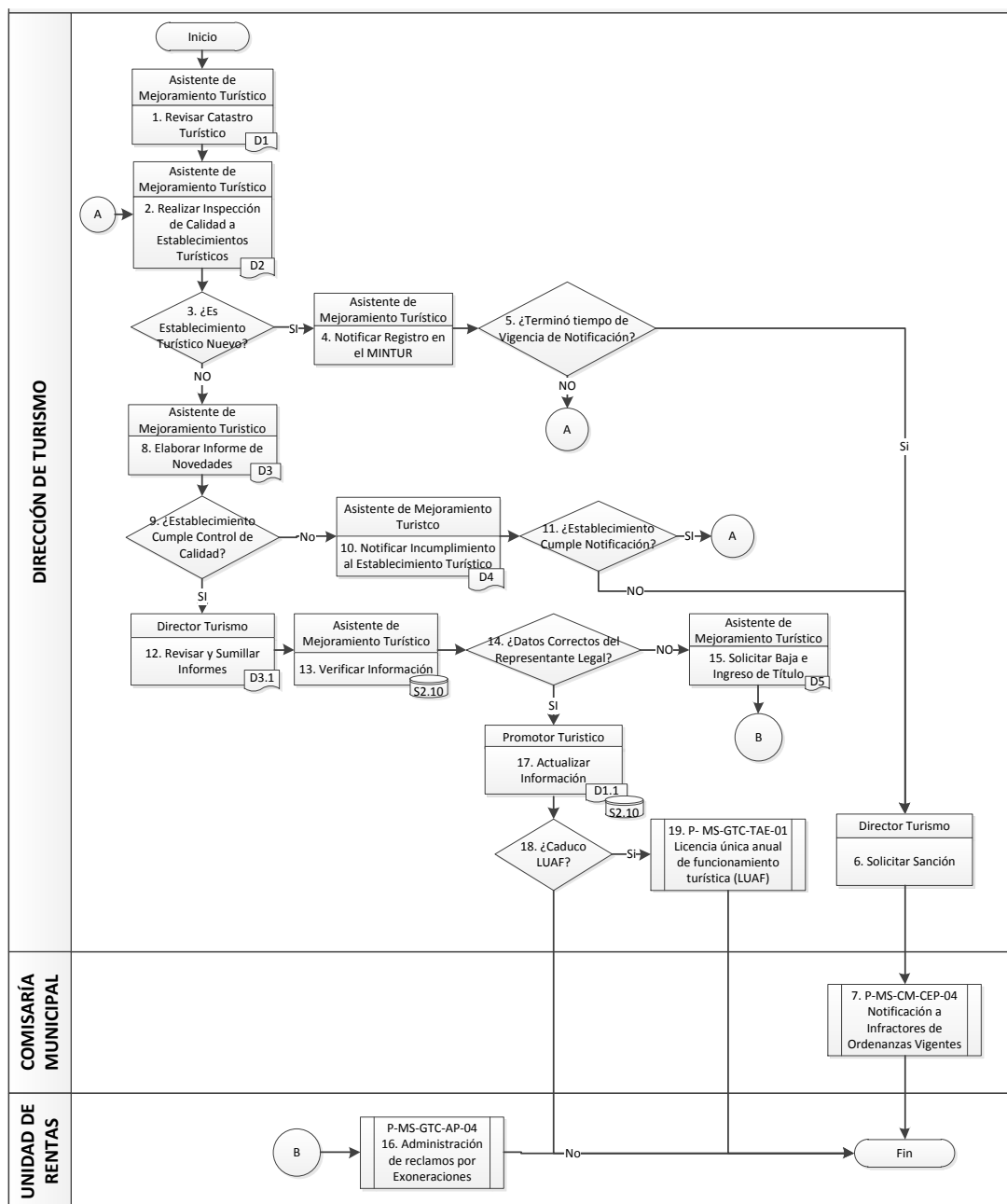


Figura 3-9. Diagrama de Flujo: Actualización y control de catastro turístico

Fuente: Departamento de Turismo GAD-I.

B) Entrega de información a los ciudadanos

Permite la entrega de información turística a los visitantes nacionales o extranjeros, satisfaciendo sus necesidades informativas sobre algún sitio o atractivo.

Roles y responsabilidades

Tabla 3-2. Roles y responsabilidades para el proceso de entrega de información a los ciudadanos

| Rol | Responsabilidad |
|----------------------------------|--|
| Asistente de Promoción Turística | Procesa y mantiene actualizados los expedientes y archivos |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

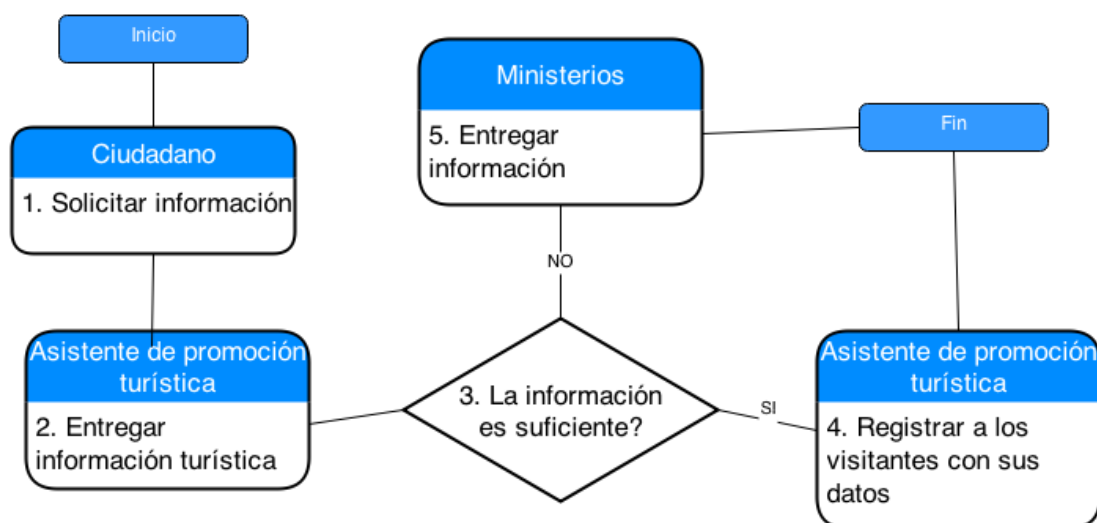


Figura 3-10. Diagrama de Flujo: Entrega de información a los ciudadanos

Fuente: Departamento de Turismo GAD-I.

C) Señalética turística externa e interna

Consiste en la implementación de los elementos de señalética turística para facilitar la orientación y acceso a los diferentes atractivos existentes en

Ibarra. Estos elementos son basados en los parámetros del Manual de Señalética Turística del Ministerio de Turismo.

Roles y responsabilidades

Tabla 3-3. Roles y responsabilidades para el proceso de señalética turística externa e interna

| Rol | Responsabilidad |
|---------------------------|--|
| Director de Turismo | Planifica las actividades y necesidades integrales para la implementación de señalización turística a nivel cantonal. |
| Técnica/o Dir. De Turismo | Realiza el levantamiento técnico, seguimiento y procedimiento para la implementación de la señalización turística en el cantón Ibarra. |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

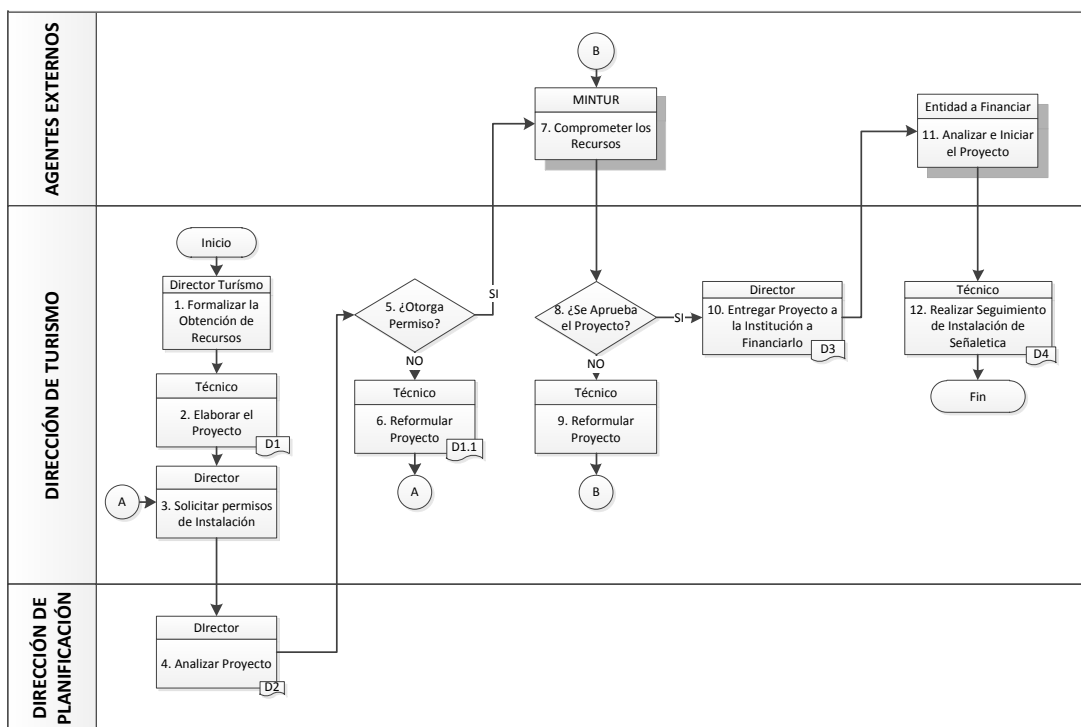


Figura 3-11. Diagrama de Flujo: Señalética turística externa e interna

Fuente: Departamento de Turismo GAD-I.

D) Gestión de promoción y marketing turístico

Consiste en lograr el posicionamiento de Ibarra como un destino turístico, mejorando la economía del mismo.

Roles y responsabilidades

Tabla 3-4. Roles y responsabilidades para el proceso de gestión de promoción y marketing turístico

| Rol | Responsabilidad |
|---------------------|--|
| Director de Turismo | Da directrices para promocionar el patrimonio arquitectónico del cantón. Da lineamientos para preservar y difundir tradiciones, folclore y gastronomía del cantón. |
| Promotor de Turismo | Revisa e informa sobre el estado de los atractivos turísticos del Cantón y apoya en la planificación de desarrollo turístico. Difunde políticas, normas y procedimientos para el desarrollo y promoción turística. |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

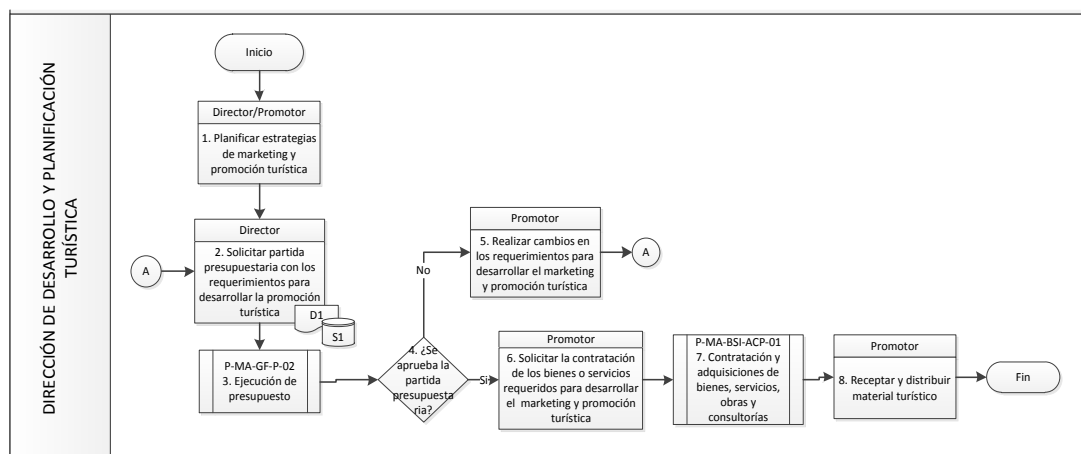


Figura 3-12. Diagrama de Flujo: Gestión de promoción y marketing turístico

Fuente: Departamento de Turismo GAD-I.

E) Capacitación turística

Consiste en el fortalecimiento de los conocimientos de los miembros sector turístico (trabajadores/administradores), logrando la mejora de los servicios que brindan a los visitantes nacionales o extranjeros.

Roles y responsabilidades

Tabla 3-5. Roles y responsabilidades para el proceso de capacitación turística

| Rol | Responsabilidad |
|-------------------------------------|---|
| Director de Turismo | El Director del departamento realiza la autogestión con Instituciones de Estado para conseguir los fondos. |
| Asistente de Mejoramiento Turístico | <ul style="list-style-type: none"> • Coordinar la infraestructura académica para la ejecución de los cursos de capacitación. • Socialización y convocatoria a los establecimientos turísticos para la participación de los cursos de capacitación. • Supervisión del cumplimiento de las capacitaciones. |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

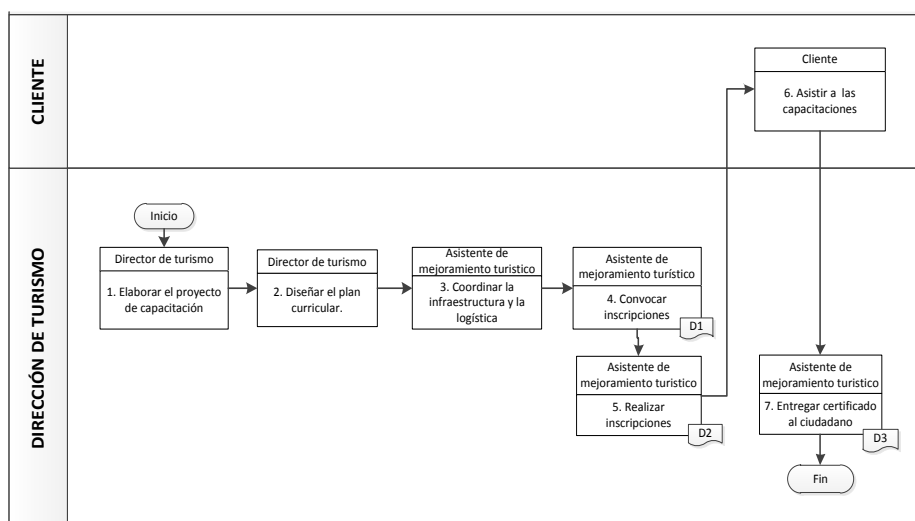


Figura 3-13. Diagrama de Flujo: Capacitación turística

Fuente: Departamento de Turismo GAD-I.

F) Administración de la guianza en museo Atahualpa

Consiste en facilitar a los visitantes el servicio de guianza dentro del museo Atahualpa.

Roles y responsabilidades

Tabla 3-6. Roles y responsabilidades para el proceso de administración de la guianza en museo Atahualpa

| Rol | Responsabilidad |
|---------------------|--|
| Director de Turismo | El Director departamental planifica, coordina, evalúa los requerimientos y procesos de los usuarios. |
| Promotor turístico | Coordinar trabajos de mantenimiento y funcionamiento museo Atahualpa |
| Asistente de Museo | Facilita el servicio de guianza a los visitantes que llegan al museo. |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

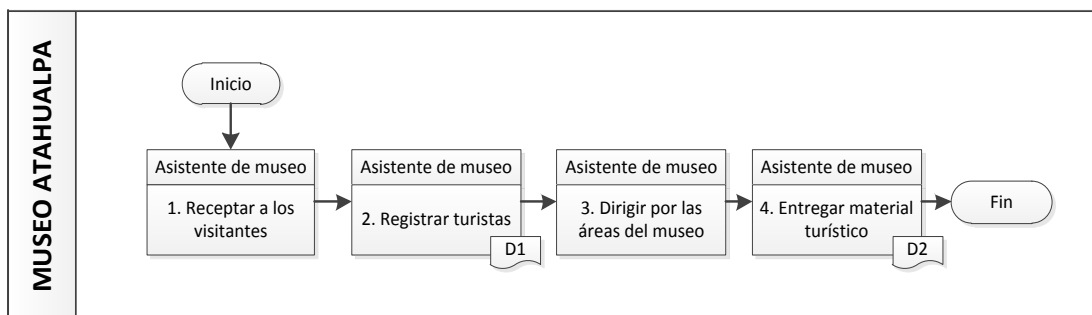


Figura 3-14. Diagrama de Flujo: Administración de la guianza en museo Atahualpa

Fuente: Departamento de Turismo GAD-I.

G) Ferias Turísticas

Permite la difusión de la oferta turística, artesanal y cultural ibarreña al turista a través de eventos como las ferias.

Roles y responsabilidades

Tabla 3-7. Roles y responsabilidades para el proceso de ferias turísticas

| Rol | Responsabilidad |
|---------------------------------------|--|
| Director de Turismo | El Director de Turismo dirige y planifica la actividad turística integral en el cantón Ibarra. |
| Promotor de Turismo Área Marketing | Planifica las estrategias para la ejecución y realización de la feria |

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

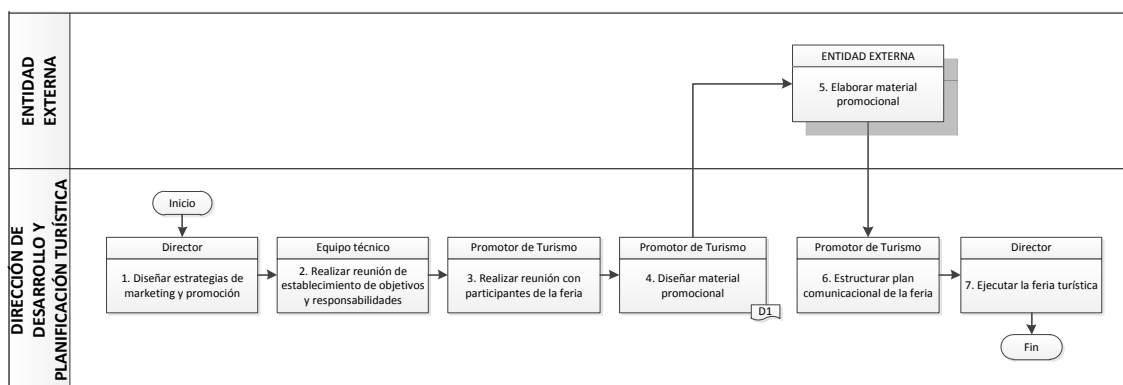


Figura 3-15. Diagrama de Flujo: Ferias turísticas

Fuente: Departamento de Turismo GAD-I.

H) Entrega de licencia anual de funcionamiento turístico

Permite la legalización del funcionamiento de los establecimientos turísticos a través de la entrega de la Licencia única anual de funcionamiento (LUAF).

Roles y responsabilidades

| Rol | Responsabilidad |
|---------------------------|---|
| Director de Turismo | Revisa documentos habilitantes y firma la LUAF previa entrega definitiva. |
| Promotor de turismo | <ul style="list-style-type: none"> Actualiza el catastro turístico del cantón Ibarra Verifica los documentos habilitantes para obtención de LUAF Registra y control de los establecimientos dentro del sistema |
| Asistente de Mejoramiento | <ul style="list-style-type: none"> Actualización del catastro de turístico |

Turístico

- Inspección a establecimientos turísticos
- Actualización de ficha catastral

Tabla 3-8. Roles y responsabilidades para el proceso de entrega de licencia anual de funcionamiento turístico

Fuente: Departamento de Turismo GAD-I.

Diagrama de Flujo

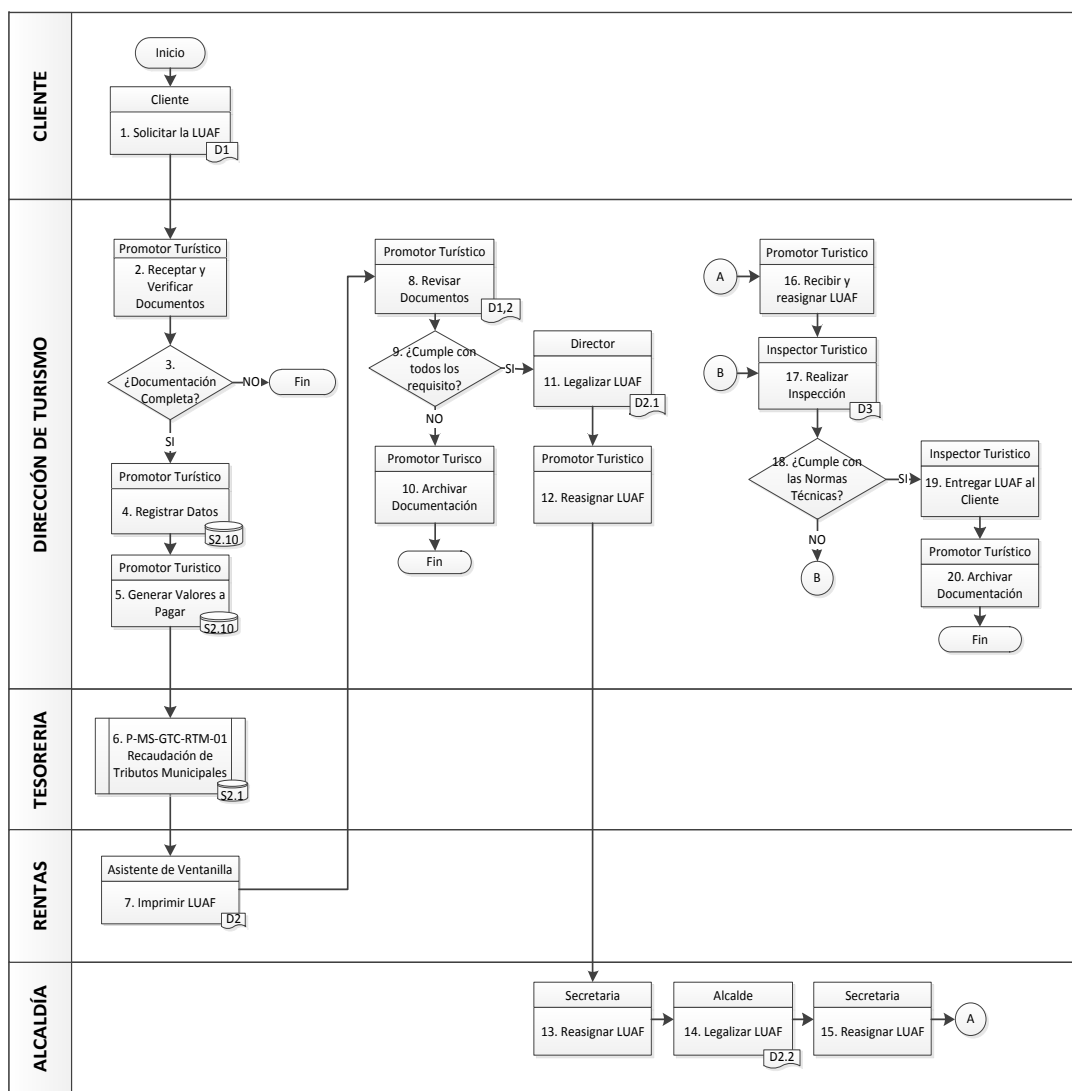


Figura 3-16. Diagrama de Flujo: Entrega de licencia anual de funcionamiento turístico

Fuente: Departamento de Turismo GAD-I.

CAPÍTULO IV

4 METODOLOGÍA DE DESARROLLO

Dentro de este capítulo se pone en práctica la metodología de desarrollo de software XP, dando a cabo sus fases con la determinada documentación de las mismas. Mediante una metodología de desarrollo crearemos un ambiente con las mejores prácticas para resolver un problema relacionado con un determinado software, garantizando su calidad y satisfaciendo las necesidades del cliente.

4.1 FASE DE EXPLORACIÓN

El aplicativo a desarrollarse tiene como propósito ofrecer al turista una guía de los atractivos que presenta la ciudad de Ibarra, y logran agilizar el proceso de entrega de información turística.

Al poseer las historias de usuario iniciales se procederá a rediseñar las mismas, siendo estas la base de los requerimientos dados por el cliente y servirán para el equipo de desarrollo XP.

4.1.1 Historias de Usuario

- Ingreso al Sistema (Sistema Administrador)

| Historia de Usuario | |
|--|------------------------------------|
| Número: 1 | Usuario: Usuario Registrado |
| Nombre historia: Ingreso al Sistema (Sistema Administrador) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 0,8 | Iteración asignada: 1 |

| |
|---|
| Programador responsable: Luis Cisneros |
| <p>Descripción: Un usuario para tener acceso al sistema debe poseer una cuenta. Al entrar a la pantalla de acceso se procederá a escribir su nombre y contraseña, logrando así acceder a la pantalla correspondiente.</p> <p>Indicaciones generales.</p> <ul style="list-style-type: none"> - Se utilizará colores sencillos para el diseño y maquetado. - Se centrará el formulario - La pantalla deberá poseer una imagen representativa del GAD-I - Se centrarán los componentes del formulario. <p>Validación y acciones.</p> <ul style="list-style-type: none"> - Los campos de texto dentro del formulario deben contener datos adecuados, en caso contrario se muestra un mensaje de alerta. - Si un usuario no existe o ingresa mal sus datos, deberá mostrar un mensaje de alerta. - Si todas las validaciones fueron aprobadas, deberá acceder la pantalla correspondiente. |
| <p>Observaciones: No existen observaciones.</p> |

Tabla 4-1. Historia Ingreso al Sistema

- Gestión de Usuarios (Sistema Administrador)

| Historia de Usuario | |
|--|------------------------------------|
| Número: 2 | Usuario: Administrador |
| Nombre historia: Gestión de Usuarios (Sistema Administrador) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 0,6 | Iteración asignada: 1 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario administrador del sistema al entrar a la pantalla de administración, tendrá la capacidad de ingresar, editar y eliminar usuarios operadores del sistema.</p> <p>Indicaciones generales.</p> <ul style="list-style-type: none"> - Se utilizará colores sencillos para el diseño y maquetado. - La pantalla de administración debe poseer una imagen representativa y un menú. - Se centrarán los componentes de los formularios. <p>Validación y acciones.</p> <ul style="list-style-type: none"> - Los campos deben contener datos adecuados, en caso contrario se | |

| |
|---|
| muestra un mensaje de alerta. -Si todas las validaciones fueron aprobadas, se debe realizar la acción correspondiente de creación o edición. -Se debe alertar si un usuario va a ser eliminado. |
| Observaciones: No existen observaciones. |

Tabla 4-2. Historia Gestión de Usuarios

- Gestión de Categorías (Sistema Administrador)

Tabla 4-3. Historia Gestión de Categorías

| Historia de Usuario | |
|---|------------------------------------|
| Número: 3 | Usuario: Operador |
| Nombre historia: Gestión de Categorías (Sistema Administrador) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Media |
| Puntos estimados: 0,6 | Iteración asignada: 1 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario operador del sistema al entrar a la pantalla correspondiente, tendrá la capacidad de ingresar, editar y eliminar categorías.</p> <p>Indicaciones generales. - Se utilizará colores sencillos para el diseño y maquetado. - El panel del operador debe poseer una imagen representativa y un menú. - El menú permitirá el acceso al manejo de categorías. - Se centrarán los componentes de los formularios.</p> <p>Validación y acciones. -Los campos deben contener datos adecuados, en caso contrario se muestra un mensaje de alerta. -Si todas las validaciones fueron aprobadas, se debe realizar la acción correspondiente de creación o edición. -Se debe alertar si una categoría va a ser eliminada.</p> | |
| Observaciones: No existen observaciones. | |

- Gestión de Información general (Sistema Administrador)

Historia de Usuario

| | |
|--|------------------------------------|
| Número: 4 | Usuario: Operador |
| Nombre historia: Gestión de Información general (Sistema Administrador) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 0,6 | Iteración asignada: 1 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario operador del sistema al entrar a la pantalla correspondiente, tendrá la capacidad de ingresar, editar y eliminar información característica de la ciudad de Ibarra como: historia, clima, entre otros.</p> <p>Indicaciones generales.</p> <ul style="list-style-type: none"> - Se utilizará colores sencillos para el diseño y maquetado. - El panel del operador debe poseer un enlace desde el menú. - Se centrarán los componentes de los formularios. <p>Validación y acciones.</p> <ul style="list-style-type: none"> -Los campos deben contener datos adecuados, en caso contrario se muestra un mensaje de alerta. -Si todas las validaciones fueron aprobadas, se debe realizar la acción correspondiente de creación o edición. -Se debe alertar si se va a realizar una eliminación. | |
| Observaciones: No existen observaciones. | |

Tabla 4-4 Historia Gestión de Información general

- Gestión de Atractivos (Sistema Administrador)

| Historia de Usuario | |
|--|------------------------------------|
| Número: 5 | Usuario: Operador |
| Nombre historia: Gestión de Atractivos (Sistema Administrador) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Medio |
| Puntos estimados: 0,8 | Iteración asignada: 1 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario operador del sistema al entrar a la pantalla correspondiente, tendrá la capacidad de ingresar, editar y eliminar atractivos pertenecientes a una categoría.</p> <p>Indicaciones generales.</p> <ul style="list-style-type: none"> - Se utilizará colores sencillos para el diseño y maquetado. - El panel del operador debe poseer un enlace desde el menú. - Se centrarán los componentes de los formularios. | |

| |
|---|
| <p>Validación y acciones.</p> <p>-Los campos deben contener datos adecuados, en caso contrario se muestra un mensaje de alerta.</p> <p>-Si todas las validaciones fueron aprobadas, se debe realizar la acción correspondiente de creación o edición.</p> <p>-Se debe alertar si se va a realizar una eliminación.</p> |
| <p>Observaciones:</p> <p>No existen observaciones.</p> |

Tabla 4-5 Historia Gestión de Atractivos

- Acceso a datos (Sistema Administrador y Servicio REST)

Tabla 4-6 Historia Acceso a Datos

| Historia de Usuario | |
|--|-----------------------------------|
| Número: 6 | Usuario: Cliente |
| Nombre historia: Acceso a datos (Sistema Administrador y Servicio REST) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Alto |
| Puntos estimados: 0,8 | Iteración asignada: 1 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción:</p> <p>Se pretende la configuración del ambiente en la nube, con el fin de montar el servicio REST que serán de utilidad para la aplicación móvil. Además dentro del mismo se pretende añadir el sistema administrador.</p> | |
| <p>Observaciones:</p> <p>No existen observaciones.</p> | |

- Ingreso a la aplicación (App móvil)

| Historia de Usuario | |
|---|-----------------------------------|
| Número: 7 | Usuario: Cliente |
| Nombre historia: Cliente Móvil (App móvil) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Alto |
| Puntos estimados: 0,6 | Iteración asignada: 2 |
| Programador responsable: Luis Cisneros | |

| |
|---|
| <p>Descripción: Un usuario que haya instalado la aplicación móvil en su dispositivo Android, al entrar a la misma aparecerá una animación y proseguirá a la elección de idioma.</p> <p>Indicaciones generales. - Se utilizará colores sencillos para el diseño y maquetado. - La animación será sencilla y llamativa. - Se usara una imagen de la ciudad en la selección de idioma.</p> <p>Validación y acciones. -Se debe verificar la conexión a internet. -Se selecciona el idioma para acceder a la pantalla principal de la aplicación.</p> |
| <p>Observaciones: No existen observaciones.</p> |

Tabla 4-7 Historia Ingreso a la aplicación

- Visualización de categorías (App móvil)

Tabla 4-8 Historia Visualización de categorías

| Historia de Usuario | |
|---|-----------------------------------|
| Número: 8 | Usuario: Cliente |
| Nombre historia: Visualización de categorías (App móvil) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Alto |
| Puntos estimados: 0,6 | Iteración asignada: 2 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario que haya instalado la aplicación móvil en su dispositivo Android, al encontrarse en la pantalla principal deberá acceder a un menú donde se desplegarán las categorías existentes.</p> <p>Indicaciones generales. - Se utilizará colores sencillos para el diseño y maquetado. - Cada categoría en el menú poseerá un ícono representativo por cada una.</p> <p>Acciones. -Al ingresar a una categoría se presentarán la lista de sitios pertenecientes a la misma.</p> | |
| Observaciones: | |

| |
|---------------------------|
| No existen observaciones. |
|---------------------------|

- Visualización de Información de la ciudad (App móvil)

Tabla 4-9 Historia Visualización de Información de la ciudad

| Historia de Usuario | |
|--|-----------------------------------|
| Número: 9 | Usuario: Cliente |
| Nombre historia: Visualización de Información de la ciudad (App móvil) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Alto |
| Puntos estimados: 0,6 | Iteración asignada: 2 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario que haya instalado la aplicación móvil en su dispositivo Android, al encontrarse en la pantalla principal observará enlaces a la información general de la ciudad de Ibarra.</p> <p>Indicaciones generales. - Se utilizará colores sencillos para el diseño y maquetado. - Cada elemento poseerá un ícono representativo por cada una.</p> <p>Acciones. -Al ingresar a un elemento se presentarán la información del mismo estilizada.</p> | |
| Observaciones: No existen observaciones. | |

- Visualización de Atractivos (App móvil)

| Historia de Usuario | |
|---|-----------------------------------|
| Número: 10 | Usuario: Cliente |
| Nombre historia: Visualización de Atractivos (App móvil) | |
| Prioridad en negocio: Alta | Riesgo en desarrollo: Alto |
| Puntos estimados: 0,8 | Iteración asignada: 2 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Un usuario que haya instalado la aplicación móvil en su dispositivo Android, al encontrarse en la pantalla de atractivos por categoría podrá</p> | |

acceder a la información de cada uno mediante un enlace.

Indicaciones generales.

- Se utilizará colores sencillos para el diseño y maquetado.
- El listado poseerá un ícono.

Acciones.

-Al ingresar a un atractivo se presentarán la información del mismo estilizada.

Observaciones:

No existen observaciones.

Tabla 4-10 Historia Visualización de Atractivos

4.2 FASE DE PLANIFICACIÓN

4.2.1 Plan de entregas

Al ordenar las historias de usuario, que son los requerimientos del aplicativo, se logró definir un plan de entregas que será cumplido durante dos iteraciones.

Cabe recalcar que el tiempo de desarrollo será medido de la siguiente manera:

- 1 día = 8 horas
- 1 semana = 5 días
- 1 mes = 20 días

Tabla 4-11 Plan de Entregas

| Módulos | Nro. | Historias de usuario | Fechas estimadas | Esfuerzo en desarrollo | | | Iteraciones | | Entregas | |
|---------------------------------|------|--------------------------------------|-------------------------|------------------------|------|-------|-------------|---|----------|---|
| | | | | Semanas | Días | Horas | 1 | 2 | 1 | 2 |
| Control y acceso al sistema | 1 | Ingreso al Sistema | 16/12/2014 a 19/12/2014 | 0,8 | 4 | 32 | X | | X | |
| | 2 | Gestión de usuarios | 22/12/2014 a 24/12/2014 | 0,6 | 3 | 24 | X | | X | |
| Manejo de información turística | 3 | Gestión de Categorías | 25/12/2014 a 29/12/2014 | 0,6 | 3 | 24 | X | | X | |
| | 4 | Gestión de Información general | 30/12/2014 a 01/01/2015 | 0,6 | 3 | 24 | X | | X | |
| | 5 | Gestión de Atractivos | 02/01/2015 a 07/01/2015 | 0,8 | 4 | 32 | X | | X | |
| Cliente móvil | 6 | Acceso a datos | 08/01/2015 a 13/01/2015 | 0,8 | 4 | 32 | X | | X | |
| | 7 | Ingreso a la aplicación | 14/01/2015 a 16/01/2015 | 0,6 | 3 | 24 | | X | | X |
| | 8 | Visualización de categorías | 19/01/2015 a 21/01/2015 | 0,6 | 3 | 24 | | X | | X |
| | 9 | Visualización de información general | 22/01/2015 a 26/01/2015 | 0,6 | 3 | 24 | | X | | X |
| | 10 | Visualización de atractivos | 27/01/2015 a 30/01/2015 | 0,8 | 4 | 32 | | X | | X |

4.2.2 Módulos del aplicativo

1) Control y acceso al sistema.

El sistema web administrador maneja dos tipos de usuario: administrador y operador. El usuario administrador gestiona el acceso al sistema creando usuarios operadores. Los usuarios operadores dispondrán de las demás funcionalidades del sistema. Cualquier usuario deberá acceder a la aplicación utilizando un nick y contraseña correspondiente, logrando el acceso a la pantalla principal según su rol.

2) Manejo de información turística.

El usuario operador al ingresar al sistema podrá registrar la información perteneciente a la ciudad, las categorías de atractivos, y los atractivos correspondientes.

3) Cliente móvil.

El cliente móvil poseerá las siguientes características:

- Disponibilidad para las tiendas de Android.
- Diseño responsivo tanto para teléfonos móviles como para tabletas
- Mostrará un listado de los atractivos turísticos pertenecientes a Ibarra con información relevante, imagen representativa y un mapa de su ubicación.
- Contendrá información relevante de la ciudad.

4.2.3 Planificación: Iteración I

A) Cronograma

Tabla 4-12 Cronograma Iteración I

| Nro. | Historia de usuario | Fecha estimada | Duración | |
|--------------|--------------------------------|----------------------------|------------|------------|
| | | | Semanas | Horas |
| 1 | Ingreso al Sistema | 16/12/2014 a 19/12/2014 | 0,8 | 32 |
| 2 | Gestión de usuarios | 22/12/2014 a 24/12/2014 | 0,6 | 24 |
| 3 | Gestión de Categorías | 25/12/2014 a 29/12/2014 | 0,6 | 24 |
| 4 | Gestión de Información general | 30/12/2014 a 01/01/2015 | 0,6 | 24 |
| 5 | Gestión de Atractivos | 02/01/2015 a 07/01/2015 | 0,8 | 32 |
| 6 | Acceso a datos | 08/01/2015 a 13/01/2015 | 0,8 | 32 |
| TOTAL | | | 4,2 | 168 |

B) Tareas Historia 1: Ingreso al Sistema

Tabla 4-13 Tareas Historia 1

| Número | Nombre | Tiempo estimado |
|--------|--|-----------------|
| 1 | Configuración del entorno de programación | 3 horas |
| 2 | Creación del script de la base de datos | 3 horas |
| 3 | Diseño de la interfaz de login y ventanas de usuario | 8 horas |
| 4 | Mapeo de entidades | 1 hora |
| 5 | Creación de clases JAVA | 14 horas |
| 6 | Implementación de funcionalidad | 3 horas |
| | | 32 horas |

✓ Configuración del entorno de programación

| Tarea | |
|-----------------|--------------------|
| Número tarea: 1 | Número historia: 1 |

| | |
|---|---------------------------------|
| Nombre tarea: Configuración del entorno de programación. | |
| Tipo de tarea : General | Tiempo estimado: 3 horas |
| Fecha inicio: 16/12/14 | Fecha fin: 16/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: Se procede a la creación del ambiente de trabajo para el desarrollo del sistema administrador. | |
| Componentes a implementarse: <ul style="list-style-type: none"> • Bases de datos: PostgreSQL • Servidor de aplicaciones: JBOSS • IDE de desarrollo: Eclipse | |

Tabla 4-14 Tarea: Configuración del entorno de programación

✓ **Creación del script de la base de datos**

Tabla 4-15 Tarea: Creación del script de la base de datos

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 1 |
| Nombre tarea: Creación del script de la base de datos | |
| Tipo de tarea : Administración de base de datos | Tiempo estimado: 3 horas |
| Fecha inicio: 16/12/14 | Fecha fin: 16/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación de varios scripts SQL pertenecientes a la base de datos. <ul style="list-style-type: none"> • createBD.sql: Permite la creación de la base de datos "appturismo_db". • usuario.sql: Crea la tabla usuario, con sus respectivos atributos y claves. • tipousuario.sql: Crea la tabla tipousuario, respectivos atributos y claves. • sec_usr.sql: Crea las secuencias pertenecientes a las tablas antes mencionadas. | |

✓ **Diseño de la interfaz de login y ventanas de usuario**

Tabla 4-16 Tarea: Diseño de la interfaz de login y ventanas de usuario

| Tarea | |
|--|---------------------------------|
| Número tarea: 3 | Número historia: 1 |
| Nombre tarea: Diseño de la interfaz de login y ventanas de usuario | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 8 horas |
| Fecha inicio: 16/12/14 | Fecha fin: 17/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación y diseño de los archivos <i>.xhtml</i> pertenecientes al login y a cada ventana principal de administración y operación | |

✓ **Mapeo de entidades**

Tabla 4-17 Tarea: Mapeo de entidades

| Tarea | |
|--|--------------------------------|
| Número tarea: 4 | Número historia: 1 |
| Nombre tarea: Mapeo de entidades | |
| Tipo de tarea : Programación | Tiempo estimado: 1 hora |
| Fecha inicio: 17/12/14 | Fecha fin: 17/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe el proceso de mapeo de las entidades Usuario y Tipousuario mediante JPA, aplicando el paradigma de la programación orientada a objetos. | |

✓ **Creación de clases JAVA**

Tabla 4-18 Tarea Creación de clases JAVA

| Tarea | |
|---|----------------------------------|
| Número tarea: 5 | Número historia: 1 |
| Nombre tarea: Creación de clases JAVA | |
| Tipo de tarea : Programación | Tiempo estimado: 14 horas |
| Fecha inicio: 18/12/14 | Fecha fin: 19/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación de las clases y métodos que permitan llevar un control de datos y el manejo de los mismos en el lado del cliente. Para esto se creará las siguientes clases: <ul style="list-style-type: none"> • ManagerTour.java: Clase controlador con los métodos correspondientes lógica de negocio y utilidades. • LogeoBean.java: Clase utilizada para conectar con los formularios JSF de logeo. | |

✓ **Implementación de funcionalidad**

Tabla 4-19 Tarea Implementación de funcionalidad

| Tarea | |
|--|---------------------------------|
| Número tarea: 6 | Número historia: 1 |
| Nombre tarea: Implementación de funcionalidad | |
| Tipo de tarea : Programación | Tiempo estimado: 3 horas |
| Fecha inicio: 19/12/14 | Fecha fin: 19/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de las clases Bean dentro de la vista (archivos <i>.xhtml</i>), además se añadirá validaciones y mensajes informativos. | |

C) Tareas Historia 2: Gestión de Usuarios

Tabla 4-20 Tareas Historia 2

| Número | Nombre | Tiempo estimado |
|--------|--|-----------------|
| 1 | Diseño de la interfaz de gestión de usuarios | 8 horas |
| 2 | Creación de clases JAVA | 8 horas |
| 3 | Implementación de funcionalidad | 8 horas |
| | | 24 horas |

- ✓ Diseño de la interfaz de gestión de usuarios

Tabla 4-21 Tarea Historia Diseño de la interfaz de gestión de usuarios

| Tarea | |
|--|---------------------------------|
| Número tarea: 1 | Número historia: 2 |
| Nombre tarea: Diseño de la interfaz de gestión de usuarios | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 8 horas |
| Fecha inicio: 22/12/14 | Fecha fin: 22/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación y diseño de los archivos <i>.xhtml</i> pertenecientes a la gestión de usuarios. | |

- ✓ **Creación de clases JAVA**

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 2 |
| Nombre tarea: Creación de clases JAVA | |
| Tipo de tarea : Programación | Tiempo estimado: 8 horas |
| Fecha inicio: 23/12/14 | Fecha fin: 23/12/14 |
| Programador responsable: Luis Cisneros | |

| |
|---|
| <p>Descripción: La tarea describe la creación de las clases y métodos que permitan llevar un control de datos y el manejo de los mismos en el lado del cliente. Para esto se creará las siguientes clases:</p> <ul style="list-style-type: none"> • ManagerTour.java: Creación de los métodos correspondientes lógica de negocio de la gestión de usuarios dentro de la clase controlador. • TipousuarioBean.java: Clase utilizada para conectar con los formularios JSF para el uso de roles. • UsuarioBean.java: Clase utilizada para conectar con los formularios JSF para el manejo de usuarios. |
|---|

Tabla 4-22 Tarea Creación de clases JAVA

✓ **Implementación de funcionalidad**

Tabla 4-23 Tarea Implementación de funcionalidad

| Tarea | |
|---|---------------------------------|
| Número tarea: 3 | Número historia: 2 |
| Nombre tarea: Implementación de funcionalidad | |
| Tipo de tarea : Programación | Tiempo estimado: 8 horas |
| Fecha inicio: 24/12/14 | Fecha fin: 24/12/14 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: La tarea describe la utilización de las clases Bean dentro de la vista (archivos <i>.xhtml</i>), además se añadirá validaciones y mensajes informativos.</p> | |

D) Tareas Historia 3: Gestión de Categorías

| Número | Nombre | Tiempo estimado |
|--------|---|-----------------|
| 1 | Creación del script de la base de datos | 2 horas |
| 2 | Diseño de la interfaz de manejo de categorías | 8 horas |
| 3 | Mapeo de entidades | 1 hora |

| | | |
|---|---------------------------------|-----------------|
| 4 | Creación de clases JAVA | 10 horas |
| 5 | Implementación de funcionalidad | 3 horas |
| | | 24 horas |

Tabla 4-24 Tareas Historia 3

✓ **Creación del script de la base de datos**

Tabla 4-25 Tarea: Creación del script de la base de datos

| Tarea | |
|--|---------------------------------|
| Número tarea: 1 | Número historia: 3 |
| Nombre tarea: Creación del script de la base de datos | |
| Tipo de tarea : Administración de base de datos | Tiempo estimado: 2 horas |
| Fecha inicio: 25/12/14 | Fecha fin: 25/12/14 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: La tarea describe la creación de varios scripts SQL pertenecientes a la base de datos.</p> <ul style="list-style-type: none"> • idioma.sql: Crea la tabla idioma, con sus respectivos atributos y claves. • categoria.sql: Crea la tabla categoria, respectivos atributos y claves. • sec_idioma_categoria.sql: Crea las secuencias pertenecientes a las tablas antes mencionadas. | |

✓ **Diseño de la interfaz de gestión de categorías**

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 3 |
| Nombre tarea: Diseño de la interfaz de gestión de categorías | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 8 horas |
| Fecha inicio: 25/12/14 | Fecha fin: 26/12/14 |
| Programador responsable: Luis Cisneros | |

Descripción:

La tarea describe la creación y diseño de los archivos *.xhtml* pertenecientes a gestión de usuarios.

Tabla 4-26 Tarea: Diseño de la interfaz de gestión de categorías

✓ **Mapeo de entidades**

Tabla 4-27 Tarea: Mapeo de entidades

| Tarea | |
|---|--------------------------------|
| Número tarea: 3 | Número historia: 3 |
| Nombre tarea: Mapeo de entidades | |
| Tipo de tarea : Programación | Tiempo estimado: 1 hora |
| Fecha inicio: 26/12/14 | Fecha fin: 26/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe el proceso de mapeo de las entidades Idioma y Categoría mediante JPA, aplicando el paradigma de la programación orientada a objetos. | |

✓ **Creación de clases JAVA**

| Tarea | |
|---|----------------------------------|
| Número tarea: 4 | Número historia: 3 |
| Nombre tarea: Creación de clases JAVA | |
| Tipo de tarea : Programación | Tiempo estimado: 10 horas |
| Fecha inicio: 27/12/14 | Fecha fin: 29/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación de las clases y métodos que permitan llevar un control de datos y el manejo de los mismos en el lado del cliente. Para esto se creará las siguientes clases: | |

- **ManagerTour.java:** Creación de los métodos correspondientes lógica de negocio de la gestión de categorías dentro de la clase controlador.
- **IdiomaBean.java:** Clase utilizada para conectar con los formularios JSF para el uso de idiomas.
- **CategoriaBean.java:** Clase utilizada para conectar con los formularios JSF para el manejo de categorías

Tabla 4-28 Tarea Creación de clases JAVA

✓ **Implementación de funcionalidad**

Tabla 4-29 Tarea Implementación de funcionalidad

| Tarea | |
|--|---------------------------------|
| Número tarea: 5 | Número historia: 3 |
| Nombre tarea: Implementación de funcionalidad | |
| Tipo de tarea : Programación | Tiempo estimado: 3 horas |
| Fecha inicio: 29/12/14 | Fecha fin: 29/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de las clases Bean dentro de la vista (archivos <i>.xhtml</i>), además se añadirá validaciones y mensajes informativos. | |

E) Tareas Historia 4: Gestión de información general

Tabla 4-30 Tareas Historia 4

| Número | Nombre | Tiempo estimado |
|--------|---|-----------------|
| 1 | Creación del script de la base de datos | 2 horas |
| 2 | Diseño de la interfaz de gestión de información general | 8 horas |
| 3 | Mapeo de entidades | 1 hora |
| 4 | Creación de clases JAVA | 10 horas |
| 5 | Implementación de funcionalidad | 3 horas |
| | | 24 horas |

✓ **Creación del script de la base de datos**

Tabla 4-31 Tarea: Creación del script de la base de datos

| Tarea | |
|---|---------------------------------|
| Número tarea: 1 | Número historia: 4 |
| Nombre tarea: Creación del script de la base de datos | |
| Tipo de tarea : Administración de base de datos | Tiempo estimado: 2 horas |
| Fecha inicio: 30/12/14 | Fecha fin: 30/12/14 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: La tarea describe la creación de varios scripts SQL pertenecientes a la base de datos.</p> <ul style="list-style-type: none"> • informacion.sql: Crea la tabla información, con sus respectivos atributos y claves. • sec_informacion.sql: Crea las secuencias pertenecientes a la tabla antes mencionada. | |

✓ **Diseño de la interfaz de gestión de información general**

Tabla 4-32 Tarea: Diseño de la interfaz de gestión de información general

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 4 |
| Nombre tarea: Diseño de la interfaz de gestión de información general | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 8 horas |
| Fecha inicio: 30/12/14 | Fecha fin: 31/12/14 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: La tarea describe la creación y diseño de los archivos <i>.xhtml</i> pertenecientes a gestión de información general de la ciudad de Ibarra.</p> | |

✓ **Mapeo de entidades**

Tabla 4-33 Tarea: Mapeo de entidades

| Tarea | |
|---|--------------------------------|
| Número tarea: 3 | Número historia: 4 |
| Nombre tarea: Mapeo de entidades | |
| Tipo de tarea : Programación | Tiempo estimado: 1 hora |
| Fecha inicio: 31/12/14 | Fecha fin: 31/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe el proceso de mapeo de la entidad Informacion mediante JPA, aplicando el paradigma de la programación orientada a objetos. | |

✓ **Creación de clases JAVA**

Tabla 4-34 Tarea Creación de clases JAVA

| Tarea | |
|---|----------------------------------|
| Número tarea: 4 | Número historia: 4 |
| Nombre tarea: Creación de clases JAVA | |
| Tipo de tarea : Programación | Tiempo estimado: 10 horas |
| Fecha inicio: 31/12/14 | Fecha fin: 01/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación de las clases y métodos que permitan llevar un control de datos y el manejo de los mismos en el lado del cliente. Para esto se creará las siguientes clases: <ul style="list-style-type: none"> • ManagerTour.java: Creación de los métodos correspondientes lógica de negocio de la gestión de la información general de la ciudad de Ibarra dentro de la clase controlador. • InformacionBean.java: Clase utilizada para conectar con los formularios JSF para el manejo de la información general de la ciudad de Ibarra. | |

✓ **Implementación de funcionalidad**

Tabla 4-35 Tarea Implementación de funcionalidad

| Tarea | |
|---|---------------------------------|
| Número tarea: 5 | Número historia: 4 |
| Nombre tarea: Implementación de funcionalidad | |
| Tipo de tarea : Programación | Tiempo estimado: 3 horas |
| Fecha inicio: 01/01/15 | Fecha fin: 01/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de las clases Bean dentro de la vista (archivos <i>.html</i>), además se añadirá validaciones y mensajes informativos. | |

F) Tareas Historia 5: Gestión de Atractivos

Tabla 4-36 Tareas Historia 5

| Número | Nombre | Tiempo estimado |
|--------|--|-----------------|
| 1 | Creación del script de la base de datos | 2 horas |
| 2 | Diseño de la interfaz de gestión de atractivos | 8 horas |
| 3 | Mapeo de entidades | 1 hora |
| 4 | Creación de clases JAVA | 16 horas |
| 5 | Implementación de funcionalidad | 5 horas |
| | | 32 horas |

✓ **Creación del script de la base de datos**

| Tarea | |
|--|---------------------------------|
| Número tarea: 1 | Número historia: 5 |
| Nombre tarea: Creación del script de la base de datos | |
| Tipo de tarea : Administración de base de datos | Tiempo estimado: 2 horas |

| | |
|---|----------------------------|
| Fecha inicio: 02/01/15 | Fecha fin: 02/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación de varios scripts SQL pertenecientes a la base de datos. <ul style="list-style-type: none"> • sitio.sql: Crea la tabla sitio, con sus respectivos atributos y claves. • sec_sitio.sql: Crea las secuencias pertenecientes a la tabla antes mencionada. | |

Tabla 4-37 Tarea: Creación del script de la base de datos

✓ **Diseño de la interfaz de gestión de atractivos**

Tabla 4-38 Tarea: Diseño de la interfaz de gestión de atractivos

| Tarea | |
|--|---------------------------------|
| Número tarea: 2 | Número historia: 4 |
| Nombre tarea: Diseño de la interfaz de gestión de información general | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 8 horas |
| Fecha inicio: 02/12/14 | Fecha fin: 05/01/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación y diseño de los archivos <i>.html</i> pertenecientes a gestión de información general de la ciudad de Ibarra. | |

✓ **Mapeo de entidades**

| Tarea | |
|---|--------------------------------|
| Número tarea: 3 | Número historia: 4 |
| Nombre tarea: Mapeo de entidades | |
| Tipo de tarea : Programación | Tiempo estimado: 1 hora |

| | |
|---|----------------------------|
| Fecha inicio: 05/01/14 | Fecha fin: 05/01/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe el proceso de mapeo de la entidad Sitio mediante JPA, aplicando el paradigma de la programación orientada a objetos. | |

Tabla 4-39 Tarea: Mapeo de entidades

✓ **Creación de clases JAVA**

| Tarea | |
|---|----------------------------------|
| Número tarea: 4 | Número historia: 4 |
| Nombre tarea: Creación de clases JAVA | |
| Tipo de tarea : Programación | Tiempo estimado: 16 horas |
| Fecha inicio: 05/01/14 | Fecha fin: 07/01/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la creación de las clases y métodos que permitan llevar un control de datos y el manejo de los mismos en el lado del cliente. Para esto se creará las siguientes clases: <ul style="list-style-type: none"> • ManagerTour.java: Creación de los métodos correspondientes lógica de negocio de la gestión de atractivos dentro de la clase controlador. • SitioBean.java: Clase utilizada para conectar con los formularios JSF para el manejo de atractivos. | |

Tabla 4-40 Tarea Creación de clases JAVA

✓ **Implementación de funcionalidad**

| Tarea | |
|--|---------------------------------|
| Número tarea: 5 | Número historia: 4 |
| Nombre tarea: Implementación de funcionalidad | |
| Tipo de tarea : Programación | Tiempo estimado: 5 horas |
| Fecha inicio: 07/01/15 | Fecha fin: 07/01/15 |
| Programador responsable: Luis Cisneros | |

Descripción:

La tarea describe la utilización de las clases Bean dentro de la vista (archivos *.xhtml*), además se añadirá validaciones y mensajes informativos.

Tabla 4-41 Tarea Implementación de funcionalidad

G) Tareas Historia 6: Acceso a Datos

Tabla 4-42 Tareas Historia 6

| Número | Nombre | Tiempo estimado |
|--------|--------------------------------------|-----------------|
| 1 | Configuración del entorno en la nube | 10 horas |
| 2 | Estructurado del objeto JSON | 4 horas |
| 3 | Creación del servicio REST | 18 horas |
| | | 32 horas |

✓ **Configuración del entorno en la nube**

Tabla 4-43 Tarea Configuración del entorno en la nube

| Tarea | |
|--|----------------------------------|
| Número tarea: 1 | Número historia: 6 |
| Nombre tarea: Configuración del entorno en la nube. | |
| Tipo de tarea : General | Tiempo estimado: 10 horas |
| Fecha inicio: 08/01/15 | Fecha fin: 09/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: Se procede a la creación del ambiente de trabajo dentro del PAS (plataforma como servicio) de openshift.com, donde ubicaremos la base de datos, el sistema administrador y el servicio web. | |
| Componentes a implementarse dentro de la plataforma: <ul style="list-style-type: none"> • Bases de datos: PostgreSQL • Servidor de aplicaciones: JBOSS | |
| Otros componentes: <ul style="list-style-type: none"> • IDE de desarrollo: Eclipse | |

✓ Estructurado del objeto JSON

Tabla 4-44 Tarea Estructurado del objeto JSON

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 6 |
| Nombre tarea: Estructurado del objeto JSON | |
| Tipo de tarea : Programación | Tiempo estimado: 4 horas |
| Fecha inicio: 09/01/15 | Fecha fin: 09/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: Mediante el formato ligero JSON guardaremos toda la información perteneciente a los diferentes atractivos turísticos que se encuentran en la base de datos dentro de un solo objeto. Para esto debemos analizar detenidamente la estructura del mismo para su consumo. | |

✓ Creación del servicio REST

Tabla 4-45 Creación del servicio REST

| Tarea | |
|---|----------------------------------|
| Número tarea: 3 | Número historia: 6 |
| Nombre tarea: Creación del servicio REST | |
| Tipo de tarea : Programación | Tiempo estimado: 18 horas |
| Fecha inicio: 28/11/14 | Fecha fin: 02/12/14 |
| Programador responsable: Luis Cisneros | |
| Descripción: La presente tarea pretende la creación del servicio web, el cual consultará la base de datos obteniendo la información de los sitios turísticos para la devolver dicha consulta mediante un objeto JSON. Este objeto JSON será consumido en la aplicación móvil. | |

4.2.4 Planificación: Iteración II

A) Cronograma

Tabla 4-46 Cronograma Iteración II

| Nro. | Historia de usuario | Fecha estimada | Duración | |
|--------------|--------------------------------------|-------------------------|------------|------------|
| | | | Semanas | Horas |
| 7 | Ingreso a la aplicación | 14/01/2015 a 16/01/2015 | 0,6 | 24 |
| 8 | Visualización de categorías | 19/01/2015 a 21/01/2015 | 0,6 | 24 |
| 9 | Visualización de información general | 22/01/2015 a 26/01/2015 | 0,6 | 24 |
| 10 | Visualización de atractivos | 27/01/2015 a 30/01/2015 | 0,8 | 32 |
| TOTAL | | | 2,6 | 104 |

B) Tareas Historia 7: Ingreso a la aplicación

Tabla 4-47 Tareas Historia 7

| Número | Nombre | Tiempo estimado |
|--------|---|-----------------|
| 1 | Configuración del entorno de programación | 5 horas |
| 2 | Diseño de la interfaz de inicio y selección de idioma | 8 horas |
| 3 | Implementación de funcionalidad en la App | 13 horas |
| | | 24 horas |

✓ Configuración del entorno de programación

| Tarea | |
|---|---------------------------------|
| Número tarea: 1 | Número historia: 7 |
| Nombre tarea: Configuración del entorno de programación. | |
| Tipo de tarea : General | Tiempo estimado: 5 horas |
| Fecha inicio: 14/01/15 | Fecha fin: 14/01/15 |
| Programador responsable: Luis Cisneros | |

| |
|---|
| <p>Descripción: Se procede a la creación del ambiente de trabajo para el desarrollo de la aplicación móvil.</p> <p>Componentes a implementarse:</p> <ul style="list-style-type: none"> • Servidor de aplicaciones Node JS • Herramienta de programación Apache ANT • SDK de Android • IDE de desarrollo Eclipse • Plugin de desarrollo ADT Android para Eclipse • Editor de texto Sublime Text • Cliente de la Plataforma Apache Cordova |
|---|

Tabla 4-48 Tarea Configuración del entorno de programación

- ✓ Diseño de la interfaz de inicio y selección de idioma

Tabla 4-49 Tarea Diseño de la interfaz de inicio y selección de idioma.

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 7 |
| Nombre tarea: Diseño de la interfaz de inicio y selección de idioma. | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 8 horas |
| Fecha inicio: 14/01/15 | Fecha fin: 15/01/15 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Esta tarea describe el diseño de la interfaz de inicio y selección de idioma dentro de la aplicación móvil. Cabe recalcar que la base del diseño es JQuery Mobile. Para la modificación de estilos se creará un archivo theme.css.</p> <p>La interfaz de inicio constará de lo siguiente:</p> <ul style="list-style-type: none"> • Escudo de la ciudad • Animación de carga <p>La interfaz de selección de idioma constará de lo siguiente:</p> <ul style="list-style-type: none"> • Imagen representativa de la ciudad. • Botón de selección de idioma español. • Botón de selección de idioma inglés. • Los botones deben estar en 2 idiomas. | |

- ✓ Implementación de funcionalidad en la App

Tabla 4-50 Tarea Implementación de funcionalidad en la App

| Tarea | |
|--|----------------------------------|
| Número tarea: 3 | Número historia: 7 |
| Nombre tarea: Implementación de funcionalidad en la App | |
| Tipo de tarea : Programación | Tiempo estimado: 13 horas |
| Fecha inicio: 15/01/15 | Fecha fin: 16/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de los métodos de Apache Cordova y jQuery Mobile para la lectura del servicio REST y verificación de conexión Internet para ingresar a la pantalla de selección de idioma. | |

C) Tareas Historia 8: Visualización de Categorías

Tabla 4-51 Tareas Historia 8

| Número | Nombre | Tiempo estimado |
|--------|---|-----------------|
| 1 | Diseño del menú de categorías y pantalla de lista de sitios | 12 horas |
| 2 | Creación de métodos JavaScript | 4 horas |
| 3 | Implementación de funcionalidad en la App | 8 horas |
| | | 24 horas |

- ✓ Diseño del menú de categorías y pantalla de lista de sitios

| Tarea | |
|---|----------------------------------|
| Número tarea: 1 | Número historia: 8 |
| Nombre tarea: Diseño del menú de categorías y pantalla de lista de sitios. | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 12 horas |

| | |
|--|----------------------------|
| Fecha inicio: 19/01/15 | Fecha fin: 20/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: | |
| <p>Esta tarea describe el diseño del menú de categorías y pantalla de lista de sitios dentro de la aplicación móvil. Cabe recalcar que la base del diseño es JQuery Mobile. Para la modificación de estilos se creará un archivo theme.css.</p> <p>El menú constará de lo siguiente:</p> <ul style="list-style-type: none"> • Título del menú. • Botones con las diferentes categorías. • Cada categoría posee su propio ícono. • Al hacer clic en el título se cerrará el menú. • Los botones debe abrir la lista de atractivos por categoría. <p>La interfaz de atractivos por categoría constará de lo siguiente:</p> <ul style="list-style-type: none"> • Lista estilizada de nombres de atractivos. • Ícono representativo que indique visualización. | |

Tabla 4-52 Tarea Diseño del menú de categorías y pantalla de lista de sitios

- ✓ Creación de métodos JavaScript

Tabla 4-53 Tarea Creación de métodos JavaScript

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 8 |
| Nombre tarea: Creación de métodos JavaScript | |
| Tipo de tarea : Programación | Tiempo estimado: 4 horas |
| Fecha inicio: 20/01/15 | Fecha fin: 20/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: | |
| <p>La tarea describe la utilización de JQuery Mobile para la lectura del objeto JSON y la creación de los métodos que permitan cargar las categorías y los sitios según la categoría.</p> | |

- ✓ Implementación de funcionalidad en la App

Tabla 4-54 Tarea Implementación de funcionalidad en la App

| Tarea | |
|---|---------------------------------|
| Número tarea: 3 | Número historia: 8 |
| Nombre tarea: Implementación de funcionalidad en la App | |
| Tipo de tarea : Programación | Tiempo estimado: 8 horas |
| Fecha inicio: 21/01/15 | Fecha fin: 21/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de los métodos anteriormente creados para combinarlos con las interfaces ya establecidas, logrando así una carga y muestra dinámica de los datos. | |

D) Tareas Historia 9: Visualización de información general

Tabla 4-55 Tareas Historia 9

| Número | Nombre | Tiempo estimado |
|--------|---|-----------------|
| 1 | Diseño de pantalla home y visualización de la información general | 12 horas |
| 2 | Creación de métodos JavaScript | 4 horas |
| 3 | Implementación de funcionalidad en la App | 8 horas |
| | | 24 horas |

- ✓ Diseño de pantalla home y visualización de la información general

Tabla 4-56 Tarea Diseño de pantalla de home y visualización de la información general

| Tarea | |
|---|----------------------------------|
| Número tarea: 1 | Número historia: 9 |
| Nombre tarea: Diseño de la pantalla de home y visualización de la información general. | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 12 horas |
| Fecha inicio: 22/01/15 | Fecha fin: 23/01/15 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Esta tarea describe el diseño del menú de la pantalla de inicio de la aplicación y la pantalla de cada información relevante de la aplicación móvil. Cabe recalcar que la base del diseño es JQuery Mobile. Para la modificación de estilos se creará un archivo theme.css.</p> <p>La pantalla de inicio constará de lo siguiente:</p> <ul style="list-style-type: none"> • Nombre de la ciudad en la cabecera. • Imagen de la ciudad. • Presentación de clasificación de la información general en una lista. • Cada ítem debe tener un icono. • Poseer un botón que acceda al menú. • Poseer un botón que acceda a la página de selección de idioma. <p>La pantalla de información constará de lo siguiente:</p> <ul style="list-style-type: none"> • Título de la información. • Imagen representativa. • Texto estilizado de la información. • Botón para el regreso a la página principal | |

- ✓ Creación de métodos JavaScript

| Tarea | |
|---|---------------------------|
| Número tarea: 2 | Número historia: 9 |
| Nombre tarea: Creación de métodos JavaScript | |

| | |
|---|---------------------------------|
| Tipo de tarea : Programación | Tiempo estimado: 4 horas |
| Fecha inicio: 23/01/15 | Fecha fin: 23/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de jQuery Mobile para la lectura del objeto JSON y la creación de los métodos que permitan cargar la información característica de la ciudad de Ibarra. | |

Tabla 4-57 Tarea Creación de métodos JavaScript

- ✓ Implementación de funcionalidad en la App

Tabla 4-58 Tarea Implementación de funcionalidad en la App

| Tarea | |
|---|---------------------------------|
| Número tarea: 3 | Número historia: 9 |
| Nombre tarea: Implementación de funcionalidad en la App | |
| Tipo de tarea : Programación | Tiempo estimado: 8 horas |
| Fecha inicio: 26/01/15 | Fecha fin: 26/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de los métodos anteriormente creados para combinarlos con las interfaces ya establecidas, logrando así una carga y muestra dinámica de los datos. | |

E) Tareas Historia 10: Visualización de Atractivos

Tabla 4-59 Tareas Historia 10

| Número | Nombre | Tiempo estimado |
|--------|--|-----------------|
| 1 | Diseño de la visualización de la información de los atractivos | 10 horas |
| 2 | Creación de métodos JavaScript | 4 horas |
| 3 | Implementación de Google Maps | 10 horas |
| 4 | Implementación de funcionalidad en la App | 8 horas |
| | | 32 horas |

- ✓ Diseño del menú de categorías y pantalla de lista de sitios

Tabla 4-60 Tarea Diseño de la visualización de la información de los atractivos

| Tarea | |
|---|----------------------------------|
| Número tarea: 1 | Número historia: 10 |
| Nombre tarea: Diseño de la visualización de la información de los atractivos. | |
| Tipo de tarea : Diseño y maquetación | Tiempo estimado: 10 horas |
| Fecha inicio: 27/01/15 | Fecha fin: 28/01/15 |
| Programador responsable: Luis Cisneros | |
| <p>Descripción: Esta tarea describe el diseño de la visualización de la información de los atractivos dentro de la aplicación móvil. Cabe recalcar que la base del diseño es JQuery Mobile. Para la modificación de estilos se creará un archivo theme.css.</p> <p>La página constará de los siguiente:</p> <ul style="list-style-type: none"> • Nombre del atractivo en la cabecera. • Imagen del atractivo. • Texto de descripción del mismo. • Enlace a visualización de ubicación. • Poseer un botón que regrese a la página principal. | |

- ✓ Creación de métodos JavaScript

Tabla 4-61 Tarea Creación de métodos JavaScript

| Tarea | |
|---|---------------------------------|
| Número tarea: 2 | Número historia: 10 |
| Nombre tarea: Creación de métodos JavaScript | |
| Tipo de tarea : Programación | Tiempo estimado: 4 horas |
| Fecha inicio: 28/01/15 | Fecha fin: 28/01/15 |
| Programador responsable: Luis Cisneros | |

Descripción:

La tarea describe la utilización de jQuery Mobile para la lectura del objeto JSON y la creación de los métodos que permitan cargar la información de los atractivos.

- ✓ Implementación de Google Maps

Tabla 4-62 Tarea Implementación de Google Maps

| Tarea | |
|--|----------------------------------|
| Número tarea: 3 | Número historia: 8 |
| Nombre tarea: Implementación de Google Maps | |
| Tipo de tarea : Programación | Tiempo estimado: 10 horas |
| Fecha inicio: 28/01/15 | Fecha fin: 29/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización del Api de Google Maps para mostrar la ubicación de los atractivos, esto se realizará usando método de Apache Córdoba mediante su plugin de mapas. | |

- ✓ Implementación de funcionalidad en la App

Tabla 4-63 Tarea Implementación de funcionalidad en la App

| Tarea | |
|---|---------------------------------|
| Número tarea: 4 | Número historia: 8 |
| Nombre tarea: Implementación de funcionalidad en la App | |
| Tipo de tarea : Programación | Tiempo estimado: 8 horas |
| Fecha inicio: 30/01/15 | Fecha fin: 30/01/15 |
| Programador responsable: Luis Cisneros | |
| Descripción: La tarea describe la utilización de los métodos anteriormente creados para combinarlos con las interfaces ya establecidas, logrando así una carga y muestra dinámica de los datos. | |

4.3 FASE DE DISEÑO

Dentro de esta fase estableceremos el diseño del aplicativo, su patrón de arquitectura y diagrama relacional; los cuales parten de los requerimientos plasmados en las distintas historias de usuario.

4.3.1 Arquitectura del sistema administrador

Para el desarrollo del sistema de administración de la aplicación móvil se ha establecido seguir el patrón de arquitectura MVC (Modelo Vista Controlador), ya que este trata de manera separada los datos, la lógica del negocio, y las interfaces de usuario; logrando así un software de calidad.

Para detallar un poco más sobre el patrón de arquitectura MVC explicaremos cada uno de sus componentes (capas): (Eslava Muñoz, 2013)

- **Modelo.-** Es aquella capa donde trabajan todos los datos correspondientes a la aplicación. Además gestiona todo el acceso y modificación de la información.
- **Controlador.-** Posee todo el código necesario para responder a las acciones solicitadas de la aplicación.
- **Vista.-** Presenta las distintas interfaces de usuario que serán utilizadas en la aplicación. Estas usualmente son presentadas en código HTML, XHTML, PHP, entre otros.

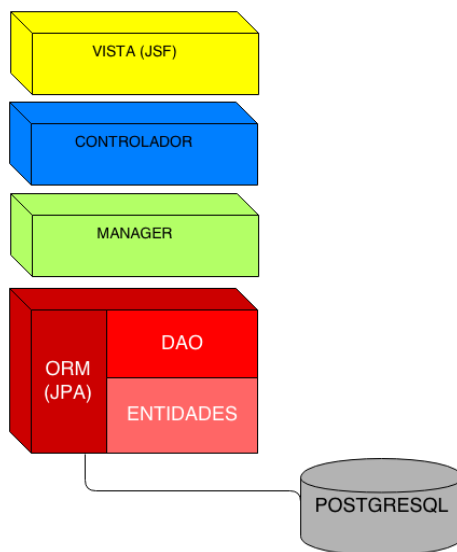


Figura 4-1. Arquitectura del Sistema Administrador

Fuente: Propia

4.3.2 Diagrama Entidad-Relación

Mediante el diagrama entidad-relación se pretende mostrar el diseño del esquema de base de datos que se usara en el aplicativo.

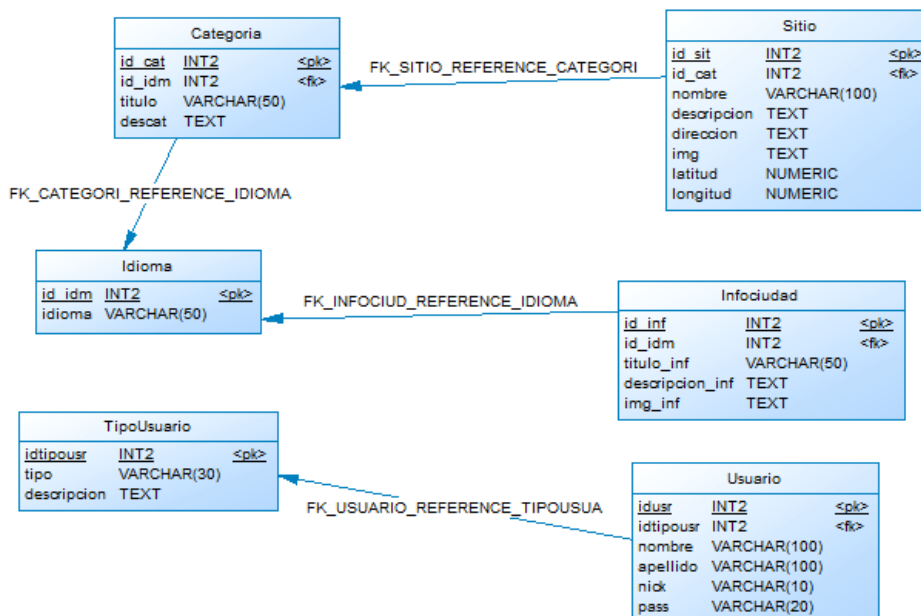


Figura 4-2 Diagrama Entidad-Relación

Fuente: Propia

4.3.3 Funcionalidad del aplicativo

El aplicativo que consta de un sistema web administrador como de una aplicación móvil, ambos construidos bajo herramientas Open Source de distribución libre. Cabe recalcar que el sistema web generará toda la información turística que será consumida por la aplicación móvil mediante un servicio REST. Para hacer uso del sistema administrador se debe estar conectado a internet y poseer un usuario con su respectiva contraseña.

El uso de la aplicación móvil es solo para dispositivos con sistema operativo Android y se la obtiene gratuitamente en Google Play.

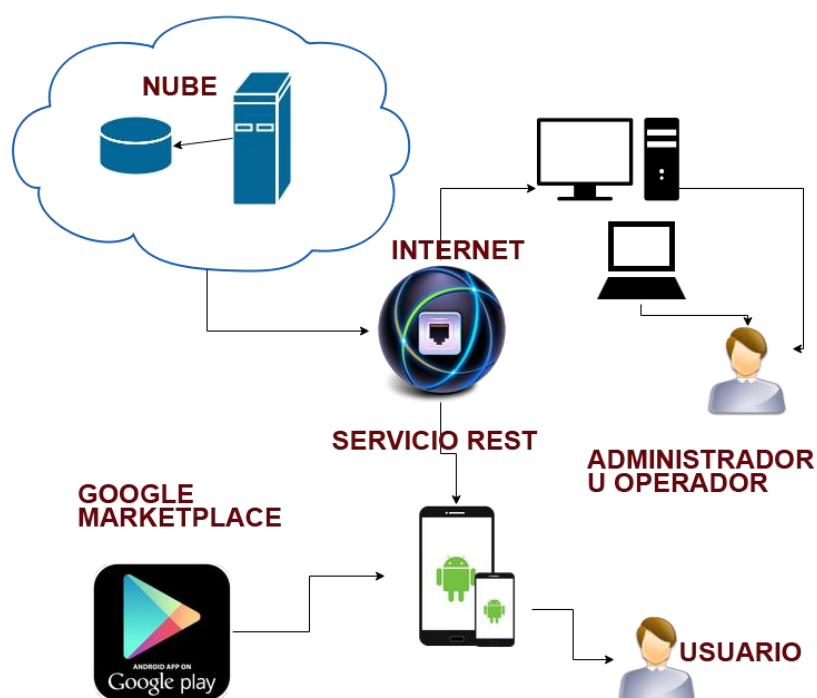


Figura 4-3 Funcionalidad del aplicativo

Fuente: Propia

4.3.4 Diseño de la aplicación móvil

La aplicación móvil pretende ser atractiva para el usuario y servir como fuente de información sobre los atractivos turísticos que posee la ciudad de Ibarra. Para mayor facilidad de uso por los turistas, esta se encuentra en el idioma español e inglés.

El diseño de la aplicación es simple y consta de:

- Animación de inicio.
- Selección de idioma.
- Menú con categorías de atractivos.
- Enlaces a información general de la ciudad.
- Presentación de atractivos por categoría.
- Información específica de cada atractivo junto a su ubicación en la ciudad.



Figura 4-4 Diseño de la App móvil

Fuente: Propia

4.4 FASE DE PRUEBAS

4.4.1 Pruebas Iteración I

- Ingreso correcto de datos

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Acceder al Sistema | Opción de prueba: Ingreso |
| Nro. de caso de prueba: 01 | Nro. de historia de usuario: 01 |
| Nombre de caso de prueba: Ingreso correcto de datos | |
| Descripción: Se desea acceder al sistema para gestionar la aplicación móvil o los usuarios pertenecientes al sistema. | |
| Condiciones de ejecución: Acceder a la URL del sistema. | |
| Datos de entrada: <ul style="list-style-type: none"> • Nick: nombre único perteneciente al usuario. • Contraseña: conjunto de caracteres clave para el acceso del usuario. | |
| Pasos de ejecución: Se ingresan correctamente los datos y se accede al panel principal de administración según el rol del usuario. | |
| Resultado esperado: <ul style="list-style-type: none"> ✓ El nick de usuario y la contraseña son verificados correctamente. ✓ El usuario según su rol accede al panel de administración. | |
| Evaluación: El usuario accede al sistema correctamente. | |

Tabla 4-64 Prueba: Ingreso correcto de datos



Figura 4-5. Prueba ingreso correcto de datos

Fuente: Propia

- Ingreso incorrecto de datos

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Acceder al Sistema | Opción de prueba: Ingreso |
| Nro. de caso de prueba: 02 | Nro. de historia de usuario: 01 |
| Nombre de caso de prueba: Ingreso incorrecto de datos | |
| Descripción: Se desea acceder al sistema para gestionar la aplicación móvil o los usuarios pertenecientes al sistema, pero se ingresa los datos de forma incorrecta. | |
| Condiciones de ejecución: Acceder a la URL del sistema. | |
| Datos de entrada: <ul style="list-style-type: none"> • Nick: nombre único perteneciente al usuario. • Contraseña: conjunto de caracteres clave para el acceso del usuario. | |
| Pasos de ejecución: <ol style="list-style-type: none"> 1. Nick incorrecto o usuario inexistente.- Al introducir este campo de forma errónea, el sistema lanzará un mensaje para informar al usuario sobre este problema ya que este usuario no es encontrado. 2. Contraseña incorrecta.- Si un usuario no ingresa su contraseña de forma correcta, el sistema lanzará un mensaje para informar al usuario el ingreso correcto de dicho campo. | |
| Resultado esperado: Cuando se ingresan los datos de forma errónea, el sistema notifica al usuario sobre el ingreso de datos erróneos. | |
| Evaluación: Mensajes de notificación para el usuario. | |

Tabla 4-65 Prueba: Ingreso incorrecto de datos



Figura 4-6. Prueba ingreso incorrecto de datos

Fuente: Propia

- Registro correcto de usuario

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Registrar usuario | Opción de prueba: Usuarios |
| Nro. de caso de prueba: 03 | Nro. de historia de usuario: 02 |
| Nombre de caso de prueba: Registro correcto de usuario | |
| Descripción: Se realiza el registro de la información perteneciente a un usuario (nombre, apellido, nick, tipo de usuario y contraseña). | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol ADMINISTRADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> • Nombre: nombre del usuario a ingresar. • Apellido: apellido del usuario a ingresar. • Nick: alias del usuario a ingresar. • Tipo de usuario: rol del usuario a ingresar. • Contraseña: código secreto temporal del usuario para el acceso al sistema. | |
| Pasos de ejecución: El usuario ADMINISTRADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Usuarios</i> e ingresa al formulario de gestión de usuarios. Para la creación de un usuario se ingresa los datos personales, se selecciona el tipo de usuario y se asigna una contraseña temporal. Al presionar el botón <i>Guardar</i> el usuario es agregado y aparece un mensaje de notificación. | |
| Resultado esperado: | |

| |
|---|
| <ul style="list-style-type: none"> ✓ Los datos son verificados correctamente. ✓ Se presenta un mensaje de éxito en la creación del usuario. |
| <p>Evaluación: Se notifica el ingreso correcto de usuario.</p> |

Tabla 4-66 Prueba: Registro correcto de usuario

The screenshot shows a web interface for user registration. At the top left, there are links for 'Ayuda' and 'Regresar'. The main heading is 'Ingreso de Usuarios'. Below this, there are input fields for 'Nombre', 'Apellido', 'Nick de usuario', and 'Contraseña'. A dropdown menu for 'Tipo de Usuario' is set to 'Operador'. A blue 'Ingresar' button is at the bottom left. On the right side, a blue notification box displays the message: 'Bienvenido Luis', 'Usuario ingresado correctamente', and 'Usuario ingresado correctamente'.

Figura 4-7. Prueba registro correcto de usuario

Fuente: Propia

- Registro incorrecto de usuario

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Registrar usuario | Opción de prueba: Usuarios |
| Nro. de caso de prueba: 04 | Nro. de historia de usuario: 02 |
| Nombre de caso de prueba: Registro incorrecto de usuario | |
| Descripción: Se realiza el registro de la información perteneciente a un usuario (nombre, apellido, nick, tipo de usuario y contraseña); se introduce de forma errónea uno o varios datos. | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol ADMINISTRADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> Código usuario.- identificador de la tabla Usuario. Nombre: nombre del usuario a ingresar. Apellido: apellido del usuario a ingresar. Nick: alias del usuario a ingresar. Tipo de usuario: rol del usuario a ingresar. Contraseña: código secreto temporal del usuario para el acceso al sistema. | |
| Pasos de ejecución: El usuario ADMINISTRADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Usuarios</i> e ingresa al formulario de gestión de usuarios. Para la creación de un usuario se ingresa los datos personales, se selecciona el | |

tipo de usuario y se asigna una contraseña temporal.
Si el administrador no llena los datos correspondientes, al presionar el botón *Guardar* aparecen los respectivos mensajes de notificación.

Resultado esperado:

- ✓ Se presentan los debidos mensajes de notificación al momento de intentar crear el usuario.
- ✓ No se ingresa el usuario.

Evaluación:

Se notifica las condiciones erróneas del proceso de registro de usuario.

Tabla 4-67 Prueba: Registro incorrecto de usuario

Figura 4-8. Prueba registro incorrecto de usuario

Fuente: Propia

- Modificación de usuario

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Editar usuario | Opción de prueba: Usuarios |
| Nro. de caso de prueba: 05 | Nro. de historia de usuario: 02 |
| Nombre de caso de prueba: Modificación de usuario | |
| Descripción: Se desea cambiar la contraseña de un usuario. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol ADMINISTRADOR. ▪ Debe existir el usuario para proceder a la modificación. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código usuario.- identificador de la tabla Usuario. • Nombre: nombre del usuario a ingresar. • Apellido: apellido del usuario a ingresar. • Nick: alias del usuario a ingresar. | |

| |
|---|
| <ul style="list-style-type: none"> • Tipo de usuario: rol del usuario a ingresar. • Contraseña: código secreto temporal del usuario para el acceso al sistema. |
| <p>Pasos de ejecución: El usuario ADMINISTRADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Usuarios</i> e ingresa al formulario de gestión de usuarios. Dentro de esta pantalla encuentra una lista con los usuarios pertenecientes al sistema, selecciona el usuario a modificar y accede al formulario de edición. Se procede al cambio de contraseña y se guardan los datos.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Los datos son verificados correctamente. ✓ Se presenta un mensaje de error, si los datos no son correctos. ✓ Se modifica la contraseña del usuario. |
| <p>Evaluación: Modificación correcta de contraseña.</p> |

Tabla 4-68 Prueba: Modificación de usuario

The screenshot shows a web form titled "Ingreso de Usuarios". It contains the following fields and controls:

- Nombres:** Text input field containing "Juan".
- Apellidos:** Text input field containing "Orbe".
- Nick de usuario:** Text input field containing "jorbe".
- Contraseña:** Text input field containing "MD5pass".
- Tipo de Usuario:** Dropdown menu with "Operador" selected.
- Buttons:** Two blue buttons labeled "Modificar" and "Cancelar".

Figura 4-9. Prueba modificación de usuario

Fuente: Propia

- Desactivar de usuario

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Desactivar usuario | Opción de prueba: Usuarios |
| Nro. de caso de prueba: 06 | Nro. de historia de usuario: 02 |
| Nombre de caso de prueba: Desactivación de usuario | |
| Descripción: Se debe eliminar un usuario existente. | |

| |
|---|
| <p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol ADMINISTRADOR. ▪ Debe existir el usuario para proceder a la desactivación. |
| <p>Datos de entrada:</p> <ul style="list-style-type: none"> • Código usuario.- identificador de la tabla Usuario. |
| <p>Pasos de ejecución:</p> <p>El usuario ADMINISTRADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Usuarios</i> e ingresa al formulario de gestión de usuarios. Dentro de esta pantalla encuentra una lista con los usuarios pertenecientes al sistema, selecciona el usuario a desactivar; el sistema notifica sobre la desactivación del usuario. Si se procede a desactivar se muestra un mensaje de éxito en la eliminación.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se presenta un cuadro de dialogo para decidir desactivar o no el usuario. ✓ Al seleccionar SI se desactiva el usuario con éxito. ✓ Al seleccionar NO, no se realiza ninguna acción. |
| <p>Evaluación:</p> <ul style="list-style-type: none"> • Proceso de desactivación correcto. • Notificación para desactivar un usuario. • Mensaje exitoso sobre el proceso de desactivación. |

Tabla 4-69 Prueba: Desactivación de usuario



Figura 4-10. Prueba desactivación de usuario

Fuente: Propia

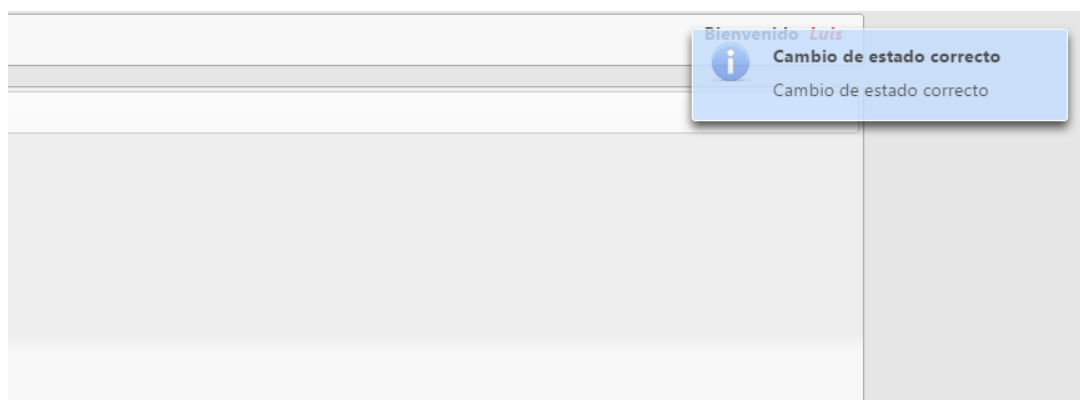


Figura 4-11. Desactivación de usuario exitosa

Fuente: Propia

- Ingreso correcto de categoría

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Ingresar categoría | Opción de prueba: Categorías |
| Nro. de caso de prueba: 07 | Nro. de historia de usuario: 03 |
| Nombre de caso de prueba: Ingreso correcto de categoría | |
| Descripción: Se realiza el registro de la información perteneciente a una categoría (título, descripción e idioma). | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Categoría. • Título: nombre de la categoría a ingresar, debe ser una palabra representativa. • Descripción: texto explicativo de la categoría. • Idioma: idioma de la categoría a ingresar. | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Categorías</i> e ingresa al formulario de gestión de categorías. Para la creación de una categoría se debe ingresar el título, su descripción y seleccionar el idioma. Al presionar el botón <i>Guardar</i> la categoría es agregada y aparece un mensaje de notificación. | |
| Resultado esperado: <ul style="list-style-type: none"> ✓ Los datos son verificados correctamente. ✓ Se presenta un mensaje de éxito en la creación de categoría. | |

Evaluación:

Se notifica el ingreso correcto de una categoría.

Tabla 4-70 Prueba: Ingreso correcto de categoría

The screenshot shows a web application interface. At the top left, there are links for 'Ayuda' and 'Regresar'. In the top right corner, a blue notification box displays the user name 'Usuario: Veronica' and the message 'Se ha ingresado correctamente' (Successfully logged in) twice. The main content area is titled 'Registro de Información de Categorías' and contains a form with three fields: 'Titulo *' (a text input), 'Descripción *' (a larger text area), and 'Idioma' (a dropdown menu currently showing 'Seleccionar...'). Below the form are two buttons: 'Nuevo' and 'Cancelar'.

Figura 4-12. Prueba ingreso correcto de categoría

Fuente: Propia

- Ingreso incorrecto de categoría

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Ingresar categoría | Opción de prueba: Categorías |
| Nro. de caso de prueba: 08 | Nro. de historia de usuario: 03 |
| Nombre de caso de prueba: Ingreso incorrecto de categoría | |
| Descripción: Se realiza el registro de la información perteneciente a una categoría (título, descripción e idioma), se procede al ingreso de datos erróneos. | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Categoría. • Título: nombre de la categoría a ingresar, debe ser una palabra representativa. • Descripción: texto explicativo de la categoría. • Idioma: idioma de la categoría a ingresar. | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Categorías</i> e ingresa al formulario de gestión de categorías. Para la creación de una categoría se debe ingresar el título, su descripción y seleccionar el idioma. El usuario realiza de forma errónea el ingreso de datos y presiona el botón <i>Guardar</i> , en ese instante el sistema mostrará el debido mensaje de notificación y la categoría no se ingresará. | |

| |
|---|
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se presentan los debidos mensajes de notificación al momento de intentar crear la categoría. ✓ Los mensajes informan sobre las condiciones de escritura del título, la inexistencia de una descripción y no haber seleccionado un idioma. ✓ No se ingresa la categoría. |
| <p>Evaluación: Se notifica las condiciones erróneas del proceso de registro de categoría.</p> |

Tabla 4-71 Prueba: Ingreso incorrecto de categoría

Figura 4-13. Prueba ingreso incorrecto de categoría

Fuente: Propia

- Modificación de una categoría

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Modificar categoría | Opción de prueba: Categorías |
| Nro. de caso de prueba: 09 | Nro. de historia de usuario: 03 |
| Nombre de caso de prueba: Modificación de una categoría | |
| Descripción: Se desea modificar los datos de una categoría. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. ▪ Debe existir la categoría a modificar. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Categoría. • Título: nombre de la categoría a ingresar, debe ser una palabra representativa. • Descripción: texto explicativo de la categoría. • Idioma: idioma de la categoría a ingresar. | |
| Pasos de ejecución: | |

| |
|---|
| <p>El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Categorías</i> e ingresa al formulario de gestión de categorías. Dentro de esta ventana se encuentra un listado de las categorías existentes.</p> <p>Para la modificación de una categoría se debe cambiar algún dato que sea conveniente; entre estos están: título, descripción e idioma. Al presionar el botón <i>Modificar</i> la categoría es modificada.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se verifican los datos de modificación. ✓ Se presentan los debidos mensajes de notificación al momento de intentar modificar la categoría. ✓ Se modifica la categoría. |
| <p>Evaluación:</p> <p>Los datos de una categoría son modificados satisfactoriamente.</p> |

Tabla 4-72 Prueba: Modificación de categoría

Figura 4-14. Prueba modificación de categoría

Fuente: Propia

- Eliminación de una categoría

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Eliminar categoría | Opción de prueba: Categorías |
| Nro. de caso de prueba: 10 | Nro. de historia de usuario: 03 |
| Nombre de caso de prueba: Eliminación de una categoría | |
| Descripción: Se desea eliminar una categoría. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |

| |
|---|
| <ul style="list-style-type: none"> ▪ Debe existir la categoría a eliminar. |
| Datos de entrada: <ul style="list-style-type: none"> ▪ Código: identificador de la tabla Categoría. |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Categorías</i> e ingresa al formulario de gestión de categorías. Dentro de esta ventana se encuentra un listado de las categorías existentes. Para la eliminación de una categoría elegirla del listado, al pulsar sobre el botón <i>Eliminar</i> aparecerá un cuadro de dialogo avisando al usuario si seguir y eliminar tanto la categoría como los atractivos pertenecientes a ella, como cancelar la acción. Al presionar el botón SI la categoría es eliminada. |
| Resultado esperado: <ul style="list-style-type: none"> ✓ Se presentan el debido cuadro de dialogo al momento de intentar eliminar la categoría. ✓ Si se decide eliminar la categoría, esta es borrada con sus respectivos atractivos. ✓ Si se decide no proceder, no sucede ningún cambio. |
| Evaluación: Los datos de una categoría, como los de sus atractivos son eliminados satisfactoriamente. |

Tabla 4-73 Prueba: Eliminación de categoría

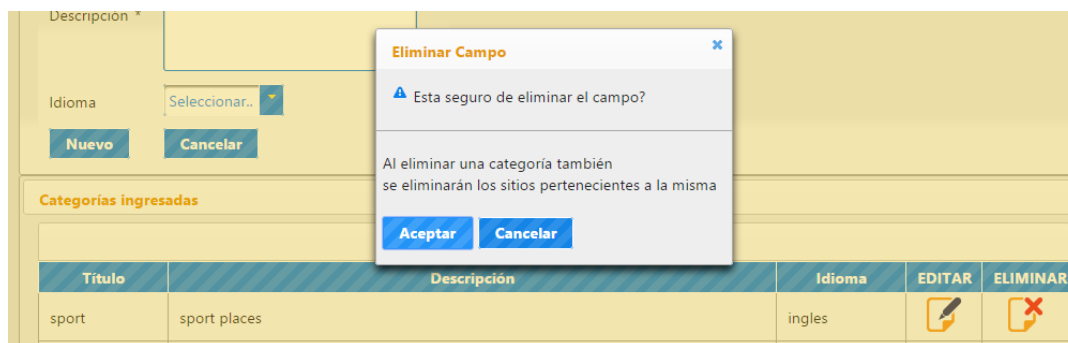


Figura 4-15. Prueba eliminación de categoría

Fuente: Propia

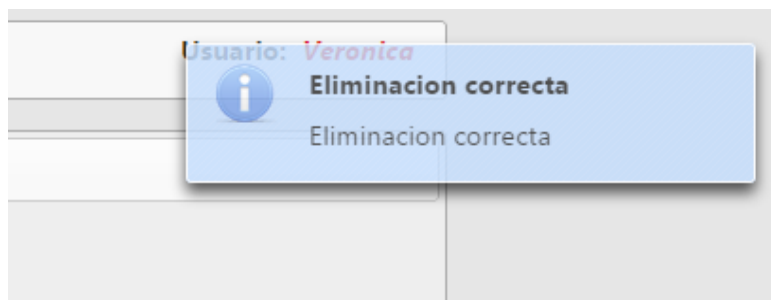


Figura 4-16. Mensaje de eliminación para categoría

Fuente: Propia

- Ingreso correcto de información general

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Registrar información general | Opción de prueba: Información General |
| Nro. de caso de prueba: 11 | Nro. de historia de usuario: 04 |
| Nombre de caso de prueba: Ingreso correcto de información general | |
| Descripción: Se realiza el registro de la información general perteneciente a la ciudad de Ibarra (título, descripción, imagen e idioma). | |
| Condiciones de ejecución: <ul style="list-style-type: none"> • El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Infocidad. • Título: nombre de característico de la información a ingresar, debe ser una palabra representativa. • Descripción: texto explicativo para dicha información. • Imagen: foto representativa para dicha información (tamaño ideal 1024x512 pixeles) • Idioma: idioma de la categoría a ingresar | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Información General</i> e ingresa al formulario de gestión de información de la ciudad. Para la creación de una nueva información se debe ingresar el título, su descripción, seleccionar una imagen y el idioma. Al presionar el botón Guardar la entidad es agregada y aparece un mensaje de notificación. | |
| Resultado esperado: <ul style="list-style-type: none"> ✓ Los datos son verificados correctamente. ✓ Se presenta un mensaje de éxito en la creación dentro de la base de datos. | |

Evaluación:

Se notifica el ingreso correcto de la información.

Tabla 4-74 Prueba: Ingreso correcto de información general

Figura 4-17. Prueba ingreso correcto de información general

Fuente: Propia

- Ingreso incorrecto de información general

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Registrar información general | Opción de prueba: Información General |
| Nro. de caso de prueba: 12 | Nro. de historia de usuario: 04 |
| Nombre de caso de prueba: Ingreso incorrecto de información general | |
| Descripción: Se realiza el registro de la información general perteneciente a la ciudad de Ibarra (título, descripción, imagen e idioma), se ingresan los datos de forma incorrecta. | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Infocuidad. • Título: nombre de característico de la información a ingresar, debe ser una palabra representativa. • Descripción: texto explicativo para dicha información. • Imagen: foto representativa para dicha información (tamaño ideal 1024x512 pixeles) • Idioma: idioma de la categoría a ingresar | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Información General</i> e ingresa al formulario de gestión de información de la ciudad. Para la creación de una nueva información se | |

| |
|---|
| <p>debe ingresar el título, su descripción, seleccionar una imagen y el idioma. El usuario realiza de forma errónea el ingreso de datos y presiona el botón Guardar, en ese instante el sistema mostrará el debido mensaje de notificación y la información de la ciudad no se ingresará.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se presentan los debidos mensajes de notificación al momento de intentar crear la entidad. ✓ Los mensajes informan sobre las condiciones de escritura del título, la inexistencia de una descripción y no haber seleccionado un idioma. ✓ No se ingresa la entidad. |
| <p>Evaluación: Se notifica las condiciones erróneas del proceso de registro de información.</p> |

Tabla 4-75 Prueba: Ingreso incorrecto de información general



Figura 4-18. Prueba ingreso incorrecto de información general

Fuente: Propia

- Modificación de información general

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Editar información general | Opción de prueba: Información General |
| Nro. de caso de prueba: 13 | Nro. de historia de usuario: 04 |
| Nombre de caso de prueba: Modificación de información general | |
| Descripción: Se desea modificar la información previamente ingresada. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. ▪ La información debe estar previamente creada. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Infociedad. | |

| |
|---|
| <ul style="list-style-type: none"> • Título: nombre de característico de la información a ingresar, debe ser una palabra representativa. • Descripción: texto explicativo para dicha información. • Imagen: foto representativa para dicha información (tamaño ideal 1024x512 pixeles) • Idioma: idioma de la categoría a ingresar |
| <p>Pasos de ejecución:</p> <p>El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Información General</i> e ingresa al formulario de gestión de información de la ciudad. Dentro de esta ventana se encuentra un listado de los datos ya existentes.</p> <p>Para la modificación se debe cambiar algún dato que sea conveniente; entre estos están: título, descripción, imagen e idioma. Al presionar el botón <i>Modificar</i> la información es modificada.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se verifican los datos de modificación. ✓ Se presentan los debidos mensajes de notificación al momento de intentar modificar la información. ✓ Se modifica la información de la ciudad. |
| <p>Evaluación:</p> <p>Los datos son modificados satisfactoriamente.</p> |

Tabla 4-76 Prueba: Modificación de información general

The screenshot shows a web form titled "Registro de Información de Categorías". It includes the following fields and controls:

- Título *:** A text input field containing the word "restaurants".
- Descripción *:** A larger text area containing the text "Restaurants of Ibarra".
- Idioma:** A dropdown menu currently displaying "ingles".
- Buttons:** Two blue buttons labeled "Modificar" and "Cancelar" are positioned at the bottom left of the form.

Figura 4-19. Prueba modificación de información general

Fuente: Propia

- Eliminación de información general

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Eliminar Información General | Opción de prueba: Información General |
| Nro. de caso de prueba: 14 | Nro. de historia de usuario: 04 |
| Nombre de caso de prueba: Eliminación de información general | |
| Descripción: Se desea eliminar la información previamente ingresada. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. ▪ La información debe estar previamente creada. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Infocidad. | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Información General</i> e ingresa al formulario de gestión de información de la ciudad. Dentro de esta ventana se encuentra un listado de los datos ya existentes. Para la eliminación se debe elegirla del listado, al pulsar sobre el botón <i>Eliminar</i> aparecerá un cuadro de dialogo avisando al usuario si seguir y eliminar la información, como cancelar la acción. Al presionar el botón SI la información general de la ciudad es eliminada. | |
| Resultado esperado: <ul style="list-style-type: none"> ✓ Se presentan el debido cuadro de dialogo al momento de intentar eliminar la información. ✓ Si se decide eliminar la entidad, esta es borrada. ✓ Si se decide no proceder, no sucede ningún cambio. | |
| Evaluación: Los datos de información general son eliminados satisfactoriamente. | |

Tabla 4-77 Prueba: Eliminación de información general

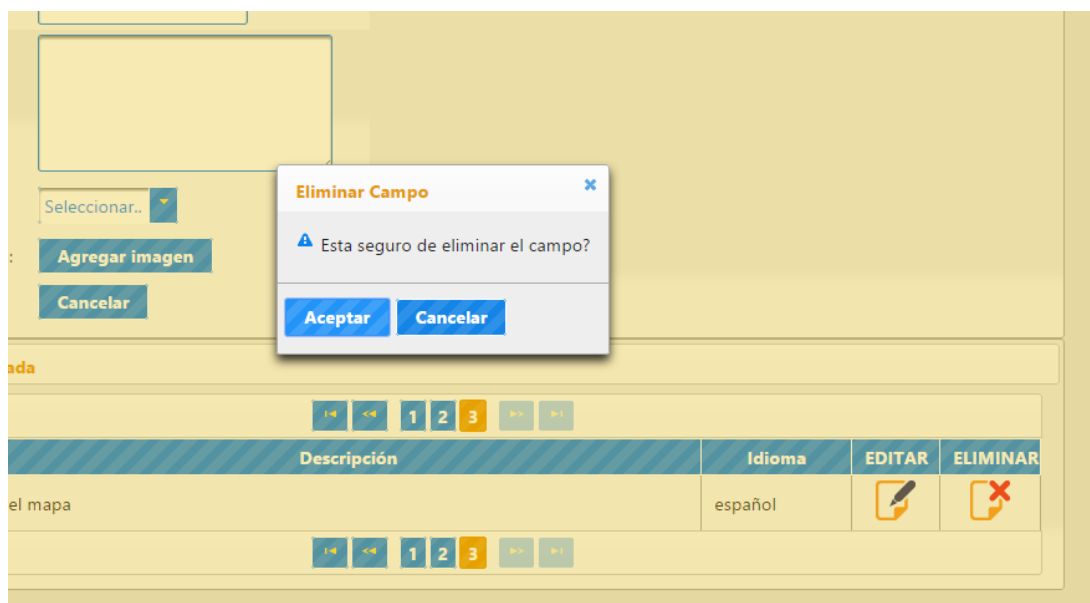


Figura 4-20. Prueba eliminación de información general

Fuente: Propia

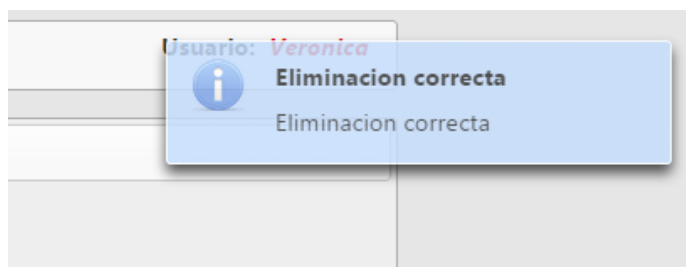


Figura 4-21. Mensaje de eliminación para información general

Fuente: Propia

- Ingreso correcto de atractivo

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Registrar atractivos | Opción de prueba: Atractivos |
| Nro. de caso de prueba: 15 | Nro. de historia de usuario: 05 |
| Nombre de caso de prueba: Ingreso correcto de atractivo | |
| Descripción: Se realiza el registro de un atractivo perteneciente a una categoría (nombre, descripción, dirección, imagen, longitud, latitud y categoría). | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |

| |
|--|
| <p>Datos de entrada:</p> <ul style="list-style-type: none"> • Código: identificador de la tabla Sitio. • Nombre: nombre de característico del atractivo. • Descripción: texto informativo del atractivo. • Dirección: ubicación del atractivo. • Imagen: foto representativa (tamaño ideal 1024x512 pixeles). • Longitud y latitud: ubicación geográfica del atractivo. • Categoría: categoría perteneciente del atractivo. |
| <p>Pasos de ejecución:</p> <p>El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Atractivos</i> e ingresa al formulario de gestión de atractivos de la ciudad. Para la creación de un nuevo atractivo debe seleccionar una categoría de la lista, si no ha seleccionado el sistema notificara la ausencia de selección.</p> <p>Una vez seleccionada la categoría, pulsa el botón <i>Nuevo</i> y accede al formulario de ingreso de atractivo turístico para el llenado de información (nombre, descripción, dirección, imagen); además mediante el movimiento de un marcador en el mapa de Google asignara la longitud y latitud del atractivo.</p> <p>Al presionar el botón <i>Guardar</i> el atractivo es agregado y aparece un mensaje de notificación.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Los datos son verificados correctamente. ✓ Se presenta un mensaje de éxito en la creación dentro de la base de datos. |
| <p>Evaluación:</p> <p>Se notifica el ingreso correcto de la información de un atractivo.</p> |

Tabla 4-78 Prueba: Ingreso correcto de atractivo

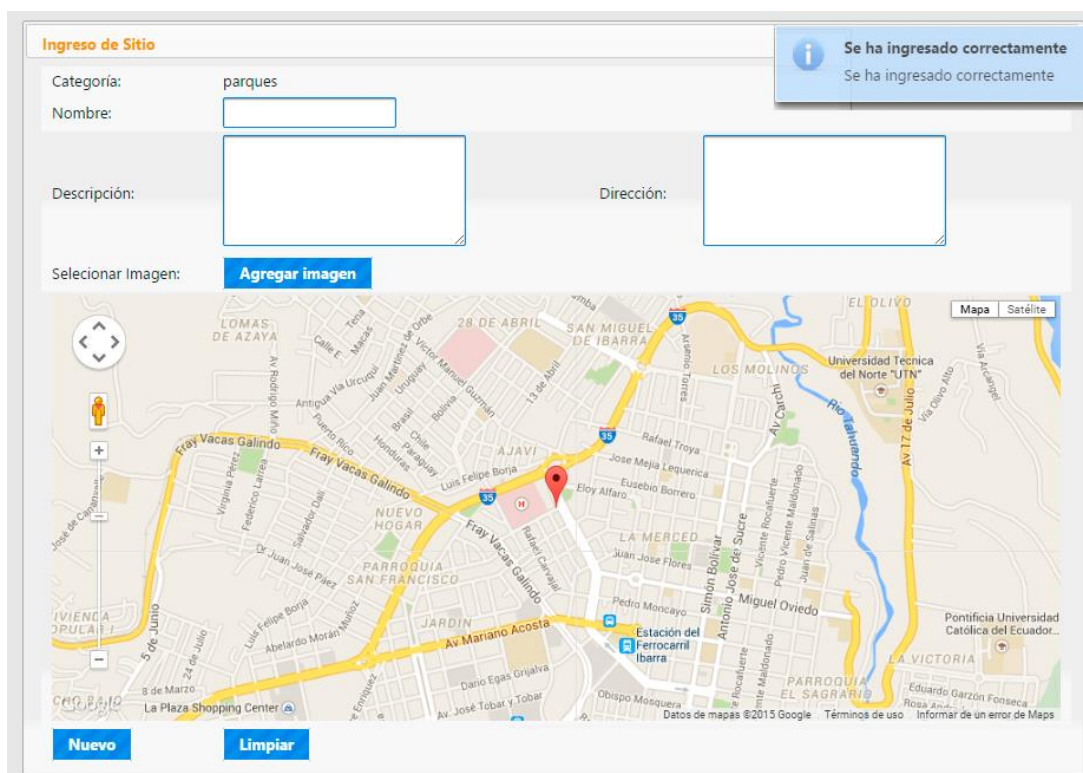


Figura 4-22. Prueba ingreso correcto de atractivo

Fuente: Propia

- Ingreso incorrecto de atractivo

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Registrar atractivos | Opción de prueba: Atractivos |
| Nro. de caso de prueba: 16 | Nro. de historia de usuario: 05 |
| Nombre de caso de prueba: Ingreso incorrecto de atractivo | |
| Descripción: Se realiza el registro de un atractivo perteneciente a una categoría (nombre, descripción, dirección, imagen, longitud, latitud y categoría), se ingresan los datos de forma errónea. | |
| Condiciones de ejecución: El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Sitio. • Nombre: nombre de característico del atractivo. • Descripción: texto informativo del atractivo. • Dirección: ubicación del atractivo. | |

| |
|--|
| <ul style="list-style-type: none"> • Imagen: foto representativa (tamaño ideal 1024x512 pixeles). • Longitud y latitud: ubicación geográfica del atractivo. • Categoría: categoría perteneciente del atractivo. |
| <p>Pasos de ejecución:</p> <p>El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Atractivos</i> e ingresa al formulario de gestión de atractivos de la ciudad. Para la creación de un nuevo atractivo debe seleccionar una categoría de la lista, si no ha seleccionado el sistema notificara la ausencia de selección.</p> <p>Una vez seleccionada la categoría, pulsa el botón <i>Nuevo</i> y accede al formulario de ingreso de atractivo turístico para el llenado de información (nombre, descripción, dirección, imagen); además mediante el movimiento de un marcador en el mapa de Google asignara la longitud y latitud del atractivo.</p> <p>El usuario realiza de forma errónea el ingreso de datos y presiona el botón Guardar, en ese instante el sistema mostrará el debido mensaje de notificación y la información del atractivo no se ingresará</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se presentan los debidos mensajes de notificación al momento de intentar crear la entidad. ✓ Los mensajes informan sobre las condiciones de escritura del nombre, la inexistencia de una descripción, y la inexistencia de una dirección. ✓ No se ingresa la entidad. |
| <p>Evaluación:</p> <p>Se notifica las condiciones erróneas del proceso de registro del atractivo.</p> |

Tabla 4-79 Prueba: Ingreso incorrecto de atractivo

The screenshot shows a web form titled "Ingreso de Sitio". The form has the following fields and values:

- Categoría: parques
- Nombre: Parque del Águila
- Descripción: (empty text box)
- Dirección: (empty text box)

Below the form is a map with a red location pin. A button labeled "Agregar imagen" is visible. On the right side of the form, two error messages are displayed in blue boxes:

- formsitio:descripcion: Error de validación: se necesita un valor.
- formsitio:direccion: Error de validación: se necesita un valor.

Figura 4-23. Prueba ingreso incorrecto de atractivo

Fuente: Propia

- Modificación de atractivo

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Editar atractivos | Opción de prueba: Atractivos |
| Nro. de caso de prueba: 17 | Nro. de historia de usuario: 05 |
| Nombre de caso de prueba: Modificación de atractivo | |
| Descripción: Se desea modificar la información de un atractivo previamente ingresado. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. ▪ La información del atractivo debe estar previamente creada. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Sitio. • Nombre: nombre de característico del atractivo. • Descripción: texto informativo del atractivo. • Dirección: ubicación del atractivo. • Imagen: foto representativa (tamaño ideal 1024x512 pixeles). • Longitud y latitud: ubicación geográfica del atractivo. • Categoría: categoría perteneciente del atractivo. | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> selecciona <i>Atractivos</i> e ingresa al formulario de gestión de atractivos de la ciudad. Dentro de esta ventana se encuentra un listado de los atractivos ya existentes, estos pueden ser filtrados por categoría. Para la modificación se debe pulsar el botón <i>Modificar</i> del listado y proceder a acceder al formulario de modificación; donde se debe cambiar algún dato que sea conveniente, entre estos están: nombre, descripción, dirección, imagen y la ubicación dentro del mapa. Al presionar el botón <i>Modificar</i> la información es modificada. | |
| Resultado esperado: <ul style="list-style-type: none"> ✓ Se verifican los datos de modificación. ✓ Se presentan los debidos mensajes de notificación al momento de intentar modificar la información. ✓ Se modifica la información del atractivo. | |
| Evaluación: Los datos son modificados satisfactoriamente. | |

Tabla 4-80 Prueba: Modificación de atractivo

Ingreso de Sitio

Categoría: parques

Nombre:

Descripción:

Dirección:

Seleccionar Imagen:




Figura 4-24. Prueba modificación de atractivo

Fuente: Propia

- Eliminación de atractivo

| Prueba de aceptación | |
|---|--|
| Caso de prueba: Eliminar atractivos | Opción de prueba: Atractivos |
| Nro. de caso de prueba: 18 | Nro. de historia de usuario: 05 |
| Nombre de caso de prueba: Eliminación de atractivo | |
| Descripción: Se desea eliminar la información de un atractivo previamente ingresado. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. ▪ La información del atractivo debe estar previamente creada. | |
| Datos de entrada: <ul style="list-style-type: none"> • Código: identificador de la tabla Sitio. • Categoría: categoría perteneciente del atractivo. | |
| Pasos de ejecución: El usuario OPERADOR accede al sistema, dentro del menú <i>Gestión</i> | |

selecciona *Atractivos* e ingresa al formulario de gestión de atractivos de la ciudad. Dentro de esta ventana se encuentra un listado de los atractivos ya existentes, estos pueden ser filtrados por categoría.

Para la eliminación se debe elegirla del listado, al pulsar sobre el botón Eliminar aparecerá un cuadro de dialogo avisando al usuario si seguir y eliminar la información, como cancelar la acción. Al presionar el botón SI la información del atractivo es eliminada.

Resultado esperado:

- ✓ Se presentan el debido cuadro de dialogo al momento de intentar eliminar la información.
- ✓ Si se decide eliminar la entidad, esta es borrada.
- ✓ Si se decide no proceder, no sucede ningún cambio.

Evaluación:

Los datos del atractivo son eliminados satisfactoriamente.

Tabla 4-81 Prueba: Eliminación de atractivo

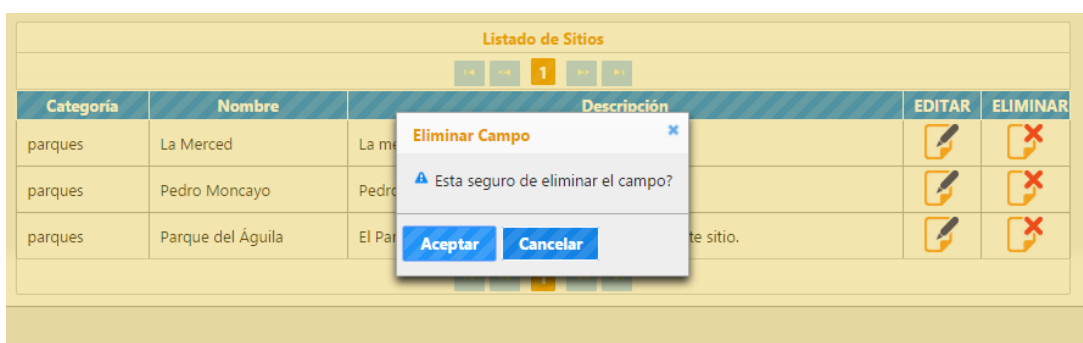


Figura 4-25. Prueba eliminación de atractivo

Fuente: Propia

- Perfil de usuario

| Prueba de aceptación | |
|---|---------------------------------------|
| Caso de prueba: Editar perfil de usuario | Opción de prueba: Usuarios |
| Nro. de caso de prueba: 19 | Nro. de historia de usuario: - |
| Nombre de caso de prueba: Perfil de usuario | |
| Descripción: Un usuario puede cambiar su contraseña. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ▪ El usuario debe ser parte de los usuarios del sistema con rol OPERADOR. ▪ El usuario debe acceder al sistema. | |
| Datos de entrada: | |

| |
|--|
| <ul style="list-style-type: none"> • Contraseña actual: código secreto actual para acceder al sistema. • Contraseña nueva: código secreto nuevo para acceder al sistema. • Contraseña nueva confirmación: repetición del código secreto nuevo. |
| <p>Pasos de ejecución:</p> <p>El usuario OPERADOR dentro del menú <i>Ver</i> selecciona <i>Perfil</i> e ingresa al formulario de gestión de perfil. Dentro de esta ventana se muestra los datos de perfil como nombre, apellido y nick. Además posee una sección de cambio de contraseña.</p> <p>Dentro de esta sección el usuario debe introducir los siguientes campos: contraseña actual, contraseña nueva y verificación de contraseña nueva. Si alguno de estos campos es erróneo, cuando se pulse el botón <i>Cambiar</i> el sistema mostrará una notificación.</p> <p>Si todos los datos son correctos se cambiará la contraseña satisfactoriamente.</p> |
| <p>Resultado esperado:</p> <ul style="list-style-type: none"> ✓ Se presentan los datos de la información del usuario. ✓ Se puede realizar el cambio de contraseña. ✓ Para el cambio se verifican los datos ingresados, mostrando mensajes de notificación en caso de errores. |
| <p>Evaluación:</p> <p>Cambio de contraseña satisfactorio.</p> |

Tabla 4-82 Prueba: Perfil de usuario

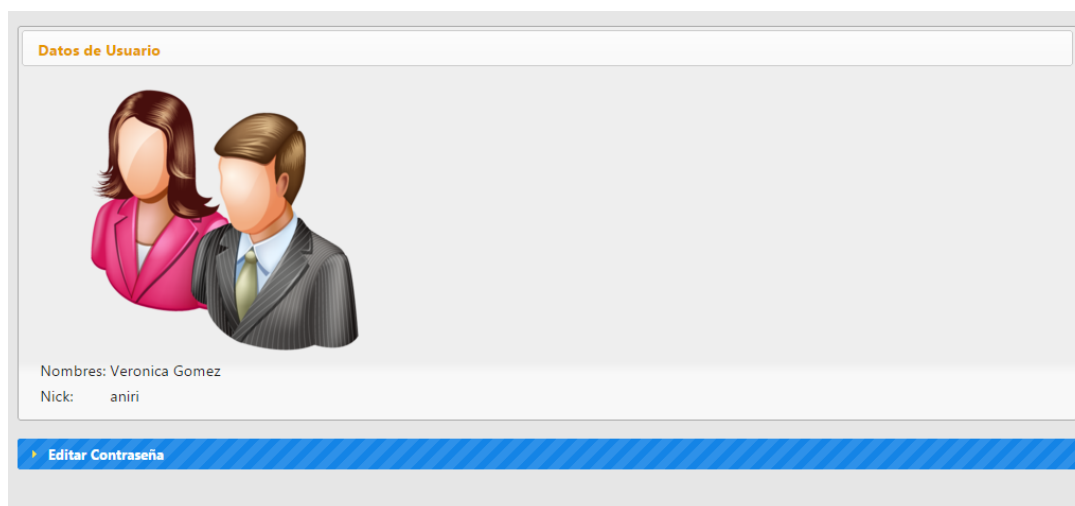


Figura 4-26. Prueba perfil de usuario

Fuente: Propia

- Funcionamiento del servicio REST

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Verificar servicio REST | Opción de prueba: Datos |
| Nro. de caso de prueba: 20 | Nro. de historia de usuario: 06 |
| Nombre de caso de prueba: Funcionamiento del servicio REST | |
| Descripción: Se desea verificar el correcto funcionamiento del servicio web. | |
| Condiciones de ejecución: No existen condiciones de ejecución. | |
| Datos de entrada: No existen datos de entrada. | |
| Pasos de ejecución: Para verificar el funcionamiento del servicio, dentro de un navegador web ingresamos la URL: http://tourapp-ibarratourapp.rhcloud.com/ServiceData . Automáticamente se mostrarán los datos en formato JSON. (Este es el ambiente de pruebas). | |
| Resultado esperado: Visualización de objeto JSON. | |
| Evaluación: Se muestra en el navegador el objeto JSON. | |

Tabla 4-83 Prueba: Funcionamiento del servicio REST

4.4.2 Pruebas Iteración II

- Carga y verificación de internet

| Prueba de aceptación | |
|--|---|
| Caso de prueba: Verificación de la conexión de internet | Opción de prueba: Estado de conexión |
| Nro. de caso de prueba: 21 | Nro. de historia de usuario: 07 |
| Nombre de caso de prueba: Carga y verificación de internet | |
| Descripción: El usuario que ha instalado la aplicación en su dispositivo ingresa a ella. | |
| Condiciones de ejecución: Tener instalada la aplicación en nuestro dispositivo Android. | |
| Datos de entrada: No existen datos de entrada. | |
| Pasos de ejecución: | |

Al abrir la aplicación se mostrará una animación de inicio, mientras esta dura se verificará la conexión a internet, si hay conexión se hace la petición de datos al servidor y se procede a la pantalla de selección de idioma. Caso contrario esta muestra un mensaje sobre el estado de conexión a internet.

Resultado esperado:

La aplicación inicia con normalidad, verifica la conexión a internet, consume el servicio o muestra una notificación de error.

Evaluación:

Verificación de internet y consumo de datos correctos.

Tabla 4-84 Prueba: Carga y verificación de internet

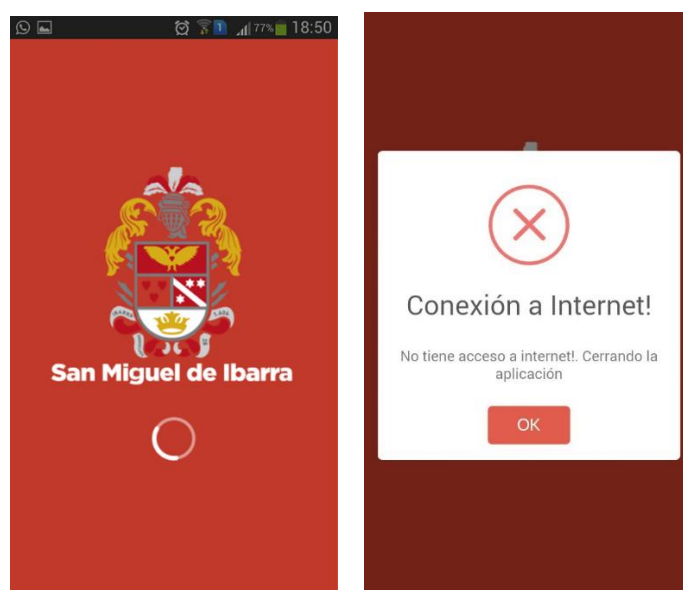


Figura 4-27. Prueba carga y verificación de internet

Fuente: Propia

- Visualización de categorías

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Menú de categorías | Opción de prueba: Datos |
| Nro. de caso de prueba: 22 | Nro. de historia de usuario: 08 |
| Nombre de caso de prueba: Visualización de categorías | |
| Descripción: Dentro de la aplicación se despliega un menú que contiene el listado de categorías de atractivos. | |

| |
|--|
| <p>Condiciones de ejecución:</p> <ul style="list-style-type: none"> ○ Tener instalada la aplicación. ○ Ingresar a la aplicación y seleccionar el idioma. |
| <p>Datos de entrada:</p> <p>No existen datos de entrada.</p> |
| <p>Pasos de ejecución:</p> <p>Iniciada la aplicación y seleccionado el idioma (inglés o español), accederemos a la pantalla principal de la misma donde podremos desplegar al lado izquierdo un menú lateral; el cual posee la lista de categorías de los atractivos pertenecientes a Ibarra.</p> |
| <p>Resultado esperado:</p> <p>Se muestra la lista de atractivos con sus respectivos íconos en ambos idiomas.</p> |
| <p>Evaluación:</p> <p>Correcto despliegue del menú de categorías de atractivos.</p> |

Tabla 4-85 Prueba: Visualización de categorías

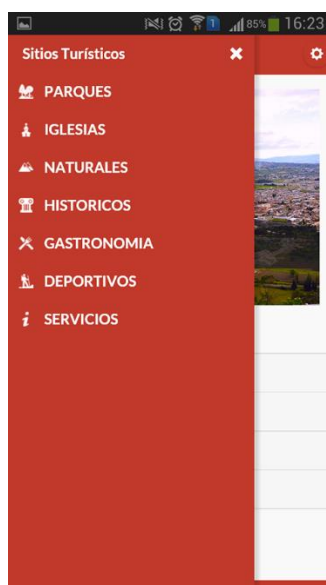


Figura 4-28. Prueba visualización de categorías

Fuente: Propia

- Visualización de Información General

| Prueba de aceptación | |
|--|--|
| Caso de prueba: Ver información de la ciudad. | Opción de prueba: Datos |
| Nro. de caso de prueba: 23 | Nro. de historia de usuario: 09 |
| Nombre de caso de prueba: Visualización de Información General | |

| |
|---|
| Descripción: En la pantalla principal se muestran links para acceder a información sobre la ciudad, por ejemplo: clima, festividades, entre otras. |
| Condiciones de ejecución: <ul style="list-style-type: none"> ○ Tener instalada la aplicación. ○ Ingresar a la aplicación y seleccionar el idioma. |
| Datos de entrada: No existen datos de entrada. |
| Pasos de ejecución: Dentro de la pantalla principal se muestran enlaces para el acceso a la información de la ciudad, al seleccionar uno de ellos visualizaremos una imagen representativa y el su contenido. |
| Resultado esperado: Visualización de la información en ambos idiomas. |
| Evaluación: Correcto despliegue de información. |

Tabla 4-86 Prueba: Visualización de Información General

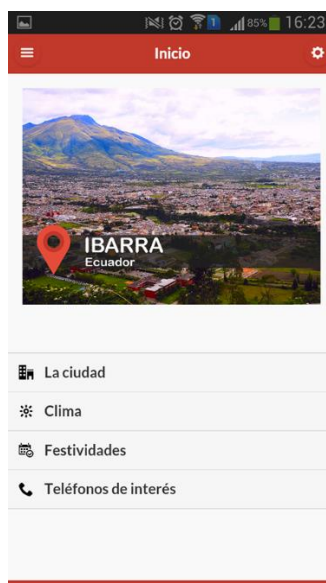


Figura 4-29. Prueba visualización de Información General

Fuente: Propia

- Visualización de Atractivos

| Prueba de aceptación | |
|---|--------------------------------|
| Caso de prueba: Ver información de un atractivo. | Opción de prueba: Datos |

| | |
|--|--|
| Nro. de caso de prueba: 24 | Nro. de historia de usuario: 10 |
| Nombre de caso de prueba: Visualización de Atractivos | |
| Descripción: Al seleccionar un atractivo podemos obtener información sobre este. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ○ Tener instalada la aplicación. ○ Ingresar a la aplicación y seleccionar el idioma. ○ Seleccionar una categoría de atractivos. | |
| Datos de entrada: No existen datos de entrada. | |
| Pasos de ejecución: Al seleccionar una categoría, se despliega una lista de atractivos pertenecientes a la misma; cuando elegimos uno accedemos a la información de este (imagen, descripción y dirección). | |
| Resultado esperado: Visualización de la información de atractivos en ambos idiomas. | |
| Evaluación: Correcto despliegue de información. | |

Tabla 4-87 Prueba: Visualización de Atractivos

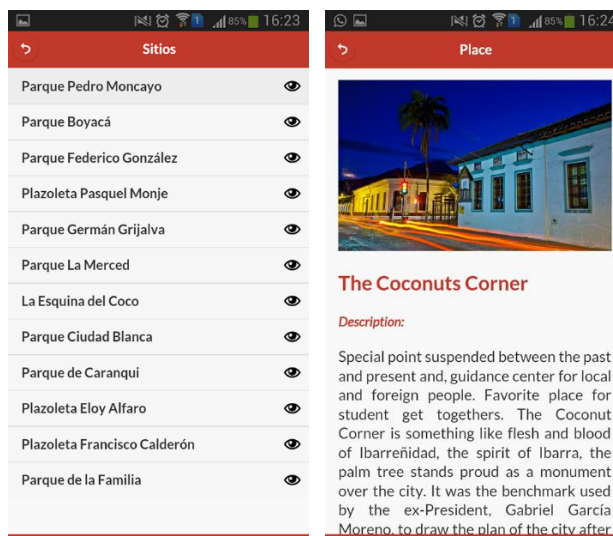


Figura 4-30. Prueba visualización de Atractivos

Fuente: Propia

- Visualización de ubicación de atractivo

| Prueba de aceptación | |
|--|------------------------------------|
| Caso de prueba: Ubicar al atractivo en el mapa. | Opción de prueba: Ubicación |

| | |
|--|--|
| Nro. de caso de prueba: 25 | Nro. de historia de usuario: 10 |
| Nombre de caso de prueba: Visualización de ubicación de atractivo | |
| Descripción: Dentro de la información de un atractivo se encuentra un enlace para poder ubicarlo dentro del mapa. | |
| Condiciones de ejecución: <ul style="list-style-type: none"> ○ Tener instalada la aplicación. ○ Ingresar a la aplicación y seleccionar el idioma. ○ Seleccionar una categoría de atractivos. ○ Seleccionar un atractivo. | |
| Datos de entrada: No existen datos de entrada. | |
| Pasos de ejecución: Dentro de la visualización de la información del atractivo, se encuentra un botón que nos indica si deseamos ver la ubicación de este en el mapa. Al seleccionar el botón se despliega el mapa de Google y nos muestra nuestra ubicación (ícono persona), y la ubicación del atractivo (ícono marcador). | |
| Resultado esperado: Visualización del atractivo en el mapa. | |
| Evaluación: Correcto despliegue de ubicación en el mapa. | |

Tabla 4-88 Prueba: Visualización de ubicación de atractivo

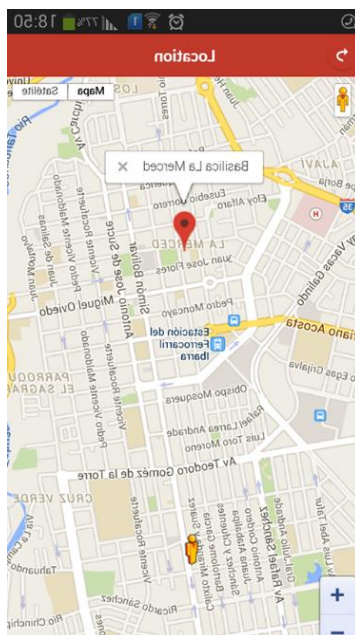


Figura 4-31. Prueba visualización de ubicación de atractivo

Fuente: Propia

CAPÍTULO V

5 CONCLUSIONES, RECOMENDACIONES Y ANÁLISIS DE IMPACTO

Dentro del presente capítulo se redactan las conclusiones y recomendaciones del aplicativo desarrollado. También se realiza el análisis de impacto dentro de varias áreas.

5.1 Conclusiones

- ✓ El uso de software libre muestra grandes ventajas como: evitar o reducir el costo de licencias, usar y redistribuir este las veces necesarias, acceder al código fuente, reducir el tiempo de desarrollo por la existencia amplia de herramientas, entre otras; permitiendo a los programadores construir aplicaciones de alta calidad y con la misma funcionalidad que el software privativo.
- ✓ Las aplicaciones móviles brindan beneficios como: ocio y diversión, comunicación con amigos y familiares, aprendizaje, contacto empresarial, compras en línea, utilidades como edición de fotografías, entre otros; logrando que los usuarios tengan un amplio mundo de posibilidades dentro de un solo dispositivo al alcance de su mano.
- ✓ Mediante el uso de servicios REST podemos generar soluciones de manera simple, centradas en la escalabilidad y rendimiento.

- ✓ Mediante el formato de objetos JSON podemos lograr un intercambio de datos, de una manera simple e independiente del lenguaje de programación.
- ✓ JQuery Mobile permite el diseño de aplicaciones para dispositivos móviles de forma fácil y asegurando la compatibilidad con todos los teléfonos inteligentes y tabletas.
- ✓ Mediante Apache Cordova podemos desarrollar aplicaciones nativas usando tecnologías web (html, css y javascript), y reutilizar el mismo código para emplearlo en distintos sistemas operativos móviles.
- ✓ OpenShift brinda un espacio en la nube donde podemos desplegar y gestionar distintas aplicaciones en diferentes lenguajes de programación, olvidándonos de la administración del hardware.
- ✓ El uso de metodologías ágiles como Extreme Programming (XP) permiten la creación de software de manera rápida, manteniendo su calidad y centrándose en las necesidades del cliente.
- ✓ El desarrollo de una aplicación móvil turística facilita la difusión de la oferta turística del cantón Ibarra, haciendo uso de las nuevas tecnologías.

5.2 Recomendaciones

- ✓ Se puede hacer uso de otras plataformas como PhoneGap, Appcelerator Titanium, CocoonJS, entre otras; para el desarrollo de aplicaciones híbridas.

- ✓ Para desarrollar aplicaciones híbridas necesitamos tener un amplio conocimiento sobre tecnologías web, especialmente HTML, CSS y JAVASCRIPT.
- ✓ Al diseñar aplicaciones móviles debemos centrarnos en que sean agradables para el usuario y que este pueda acceder a la información máximo en 5 pasos.
- ✓ Usar el plan FREE de OpenShift solo para desplegar aplicaciones pequeñas, ya que restringe funcionalidades. Se recomienda utilizar el plan BRONZE para pagar por el consumo justo de la aplicación y que esta vaya creciendo de acuerdo a sus necesidades.
- ✓ Desarrollar aplicaciones híbridas cuando estas poseen funcionalidades simples, caso contrario realizar aplicaciones nativas.
- ✓ Usar metodologías ágiles para desarrollar aplicaciones móviles, con el fin de liberar carga en documentación y centrarse en las necesidades del cliente.
- ✓ Mejorar contenidos del aplicativo, especialmente imágenes para atraer la atención de los turistas y obtener más visitantes nacionales o extranjeros dentro de la ciudad.
- ✓ Reutilizar código web generado para la realización de una aplicación móvil para el sistema operativo iOS, con el fin de lograr mayor cobertura de dispositivos.
- ✓ Realización de respaldos de la base de datos para no perder la información del aplicativo.

| | | | | | | | | |
|-----------------------------------|--|--|--|--|---|---|----|-----------|
| desarrollo | | | | | | | | |
| Utilización del aplicativo | | | | | | | | |
| Tiempo de trabajo | | | | | | | x | 3 |
| Presentación de resultados | | | | | | | x | 3 |
| Facilidad de manejo | | | | | | x | | 2 |
| TOTAL | | | | | 1 | 4 | 12 | 17 |

Tabla 5-2. Impacto productivo

Fuente: Propia

Total de impacto productivo: 17/7

Valor total de impacto productivo: 2.4

Nivel de impacto productivo: Medio positivo.

5.3.2 Impacto económico.

| Indicador | -3 | -2 | -1 | 0 | 1 | 2 | 3 | Total |
|-----------------------------------|----|----|----|---|---|---|---|----------|
| Desarrollo del aplicativo | | | | | | | | |
| Costo de comercialización | | | | x | | | | 0 |
| Costo de desarrollo | | | | x | | | | 0 |
| Costo de equipos | | | | x | | | | 0 |
| Utilización del aplicativo | | | | | | | | |
| Reducción de uso de papel | | | | | | x | | 2 |
| Reducción de soporte técnico | | | | | | | x | 3 |
| TOTAL | | | | 0 | | 2 | 3 | 5 |

Tabla 5-3. Impacto económico

Fuente: Propia

Total de impacto económico: 5/5

Valor total de impacto económico: 1

Nivel de impacto económico: Bajo positivo.

5.3.3 Impacto de innovación.

| Indicador | -3 | -2 | -1 | 0 | 1 | 2 | 3 | Total |
|-----------------------------------|----|----|----|---|---|---|----|-----------|
| Desarrollo del aplicativo | | | | | | | | |
| Aplicar herramientas nuevas | | | | | | x | | 1 |
| Usar de estándares | | | | | | | x | 3 |
| Usar técnicas de programación | | | | | | | x | 3 |
| Reutilizar de código | | | | | | | x | 3 |
| Utilización del aplicativo | | | | | | | | |
| Utilizar en dispositivos Android | | | | | | | x | 3 |
| Utilizar en otros dispositivos | | | | x | | | | 0 |
| TOTAL | | | | 0 | | 2 | 12 | 14 |

Tabla 5-4. Impacto de innovación

Fuente: Propia

Total de impacto de innovación: 14/6

Valor total de impacto de innovación: 2.33

Nivel de impacto de innovación: Medio positivo.

5.3.4 Impacto humano.

| Indicador | -3 | -2 | -1 | 0 | 1 | 2 | 3 | Total |
|-----------------------------------|----|----|----|---|---|---|---|-----------|
| Desarrollo del aplicativo | | | | | | | | |
| Aceptación de la herramienta | | | | | | x | | 2 |
| Equipo de trabajo | | | | | | | x | 3 |
| Utilización del aplicativo | | | | | | | | |
| Aceptación del aplicativo | | | | | | | x | 3 |
| Administración del aplicativo | | | | | | | x | 3 |
| Uso del aplicativo | | | | | | x | | 2 |
| TOTAL | | | | | | 4 | 9 | 13 |

Tabla 5-5 Impacto humano

Fuente: Propia

Total de impacto humano: 13/5

Valor total de impacto humano: 2.6

Nivel de impacto humano: Alto positivo.

5.3.5 Impacto social

| Indicador | -3 | -2 | -1 | 0 | 1 | 2 | 3 | Total |
|--|----|----|----|---|---|---|---|-----------|
| Utilización del aplicativo | | | | | | | | |
| Generación de empleo | | | | | | x | | 2 |
| Intercambio cultural | | | | | | | x | 3 |
| Valorar la cultura local | | | | | | | x | 3 |
| Inversión en atractivos naturales y de entretenimiento | | | | | | | x | 3 |
| TOTAL | | | | | | 2 | 9 | 11 |

Tabla 5-6 Impacto Social

Fuente: Propia

Total de impacto social: 11/4

Valor total de impacto social: 2.75

Nivel de impacto social: Alto positivo.

5.3.6 Impacto turístico

| Indicador | -3 | -2 | -1 | 0 | 1 | 2 | 3 | Total |
|--------------------------------------|----|----|----|---|---|---|---|----------|
| Utilización del aplicativo | | | | | | | | |
| Incremento de demanda turística | | | | | | x | | 2 |
| Potenciar turismo ibarreño | | | | | | x | | 2 |
| Aumento en la calidad de servicios | | | | | | | x | 3 |
| Permanencia de visitantes en la zona | | | | | | x | | 2 |
| TOTAL | | | | | | 6 | 3 | 9 |

Tabla 5-7 Impacto Turístico

Fuente: Propia

Total de impacto turístico: 9/4

Valor total de impacto turístico: 2.25

Nivel de impacto turístico: Medio positivo.

5.3.7 Impacto ambiental

| Indicador | -3 | -2 | -1 | 0 | 1 | 2 | 3 | Total |
|---|----|----|----|---|---|---|---|-----------|
| Utilización del aplicativo | | | | | | | | |
| Mayor conciencia ambiental | | | | | | x | | 2 |
| Interés en áreas naturales | | | | | | | x | 3 |
| Apoyo a programas de conservación | | | | | | | x | 3 |
| Impulsar la creación de nuevos atractivos naturales | | | | | | x | | 2 |
| TOTAL | | | | | | 4 | 6 | 10 |

Tabla 5-8 Impacto Ambiental

Fuente: Propia

Total de impacto ambiental: 10/4

Valor total de impacto ambiental: 2.5

Nivel de impacto ambiental: Alto positivo.

BIBLIOGRAFÍA

- Apache Software Foundation. (20 de Setptiembre de 2014). *Apache Cordova (Sitio Oficial)*. Recuperado de <http://cordova.apache.org>
- Apple Inc. (2014). *Apple*. Recuperado de <http://www.apple.com>
- Arroyo, N. (2011). *Información en el móvil*. España: Editorial UOC.
- Bahit, E. (2012). *Scrum y eXtreme Programming para Programadores*. Buenos Aires, Argentina: SafeCreative.
- Bai, G. (2011). *jQuery Mobile First Look*. Birmingham: Packt Publishing Ltd.
- Barrera, D. (3 de Noviembre de 2010). Secure Software Installation on Smartphones. Carleton University. Recuperado de <http://www.informatica-hoy.com.ar>
- BlackBerry Ltd. (2014). *BlackBerry*. Recuperado de <http://es.blackberry.com/>
- Blankenship, J., Bussa, M., & Millett, S. (2011). *Pro Agile .NET Development with Scrum*. Apress.
- Canonical Ltd. Ubuntu. (2014). *Ubuntu*. Recuperado de <http://www.ubuntu.com/phone>
- Comunidad Oficial del estándar UDDI. (2014). *Online community for the UDDI OASIS Standard*. Recuperado de <http://uddi.xml.org/>
- Cuello, J., & Vittone, J. (2013). *Diseñando apps para móviles*. Barcelona.
- Developers - Apple Inc. (2014). *Developers Apple*. Recuperado de Introduction to iOS: <https://developer.apple.com/>

- Dimas Luján, J. (17 de Enero de 2014). *Desarrollo Web*. Recuperado de <http://www.desarrolloweb.com/manuales/android-basico.html>
- Dirección de Turismo de Ibarra. (4 de Octubre de 2014). *Portal oficial de Turismo de Ibarra*. Recuperado de <http://touribarra.gob.ec>
- Dirección de Turismo GAD-I. (2012). Historia de Ibarra. *Ibarra: Revista y Guía turística*, 4.
- Eslava Muñoz, V. J. (2013). *El nuevo PHP: Conceptos avanzados*. España: Bubok Publishing S.L.
- Fensel, D., Facca, F. M., Simperl, E., & Toma, I. (2011). *Semantic Web Services*. Berlin: Springer Science & Business Media.
- GAD-Ibarra. (11 de 11 de 2014). *Sitio Oficial de Ibarra*. Recuperado de <http://www.ibarra.gob.ec/>
- Gironés, J. T. (2013). *El gran libro de Android*. México D.F.: MARCOMBO S.A.
- IBM® Knowledge Center. (2013). *Centro de documentación de productos IBM*. Recuperado de <http://www-01.ibm.com/support/knowledgecenter/?lang=es>
- Jeffries, R. E. (2014). *Xprogramming : An Agile Software Development Resource*. Recuperado de <http://xprogramming.com/>
- Jiménez Bulla, L. H., & Jiménez Barbosa, W. G. (2013). *Turismo: tendencias globales y planificación estratégica*. Colombia: Ecoe Ediciones.
- jQuery Foundation. (2014). *jQuery Mobile*. Recuperado de <http://jquerymobile.com/>

- JSON ORG. (2014). *The JSON Data Interchange Standard*. Recuperado de <http://json.org/>
- Microsoft. (2014). *Windows Phone*. Recuperado de <http://www.windowsphone.com/>
- Microsoft Developers. (2014). *Windows Phone Dev Center*. Recuperado de Windows Phone architecture overview: <https://dev.windowsphone.com>
- Mozilla Developers. (31 de Agosto de 2014). *Mozilla Developer Network*. Recuperado de https://developer.mozilla.org/es/Firefox_OS
- Mozilla Foundation. (2014). *Mozilla*. Recuperado de <https://www.mozilla.org>
- Natili, G. (2013). *PhoneGap 3 Beginner's Guide*. Birmingham: Packt Publishing Ltd.
- Nokia Developers . (26 de Julio de 2012). *Nokia Developer Beta*. Recuperado de Wiki Symbian OS: http://developer.nokia.com/community/wiki/Symbian_OS
- Pousty, S., & Miller, K. (2014). *Getting Started with OpenShift*. O'Reilly Media, Inc.
- Red Hat Inc. (2014). *OpenShift Developers*. Recuperado de <https://developers.openshift.com/>
- Rojas Zamora, C. E. (5 de Junio de 2013). *Codejobs: Aprende a programar*. Recuperado de <http://www.codejobs.biz/es/blog/2013/06/05/programacion-extrema-xp>
- Wargo, J. M. (2013). *Apache Cordova 3 Programming*. Addison-Wesley Professional.

- Wikipedia. (19 de Agosto de 2014). *Symbian OS*. Recuperado de <http://en.wikipedia.org/wiki/Symbian>
- World Wide Web Consortium. (2014). *W3C*. Recuperado de <http://www.w3.org/>



ANEXOS





MANUAL TÉCNICO

Tabla de contenido

| | | |
|-------|--|----|
| 1 | PROPÓSITO..... | 1 |
| 2 | OPENSIFT..... | 1 |
| 2.1 | Introducción..... | 1 |
| 2.2 | Creación de cuenta | 1 |
| 2.3 | Registro de dominio | 4 |
| 2.4 | Creación de una aplicación | 5 |
| 2.5 | Instalación del cliente de OpenShift | 7 |
| 2.5.1 | Ruby..... | 7 |
| 2.5.2 | Gema RHC..... | 10 |
| 2.5.3 | Git | 11 |
| 2.5.4 | Conexión hacia OpenShift..... | 15 |
| 3 | SISTEMA ADMINISTRADOR | 18 |
| 3.1 | Lenguaje..... | 18 |
| 3.2 | Arquitectura..... | 18 |
| 3.3 | Diagrama Entidad-Relación..... | 19 |
| 3.4 | Proyecto WEB | 21 |
| 3.4.1 | Estructura..... | 21 |
| 3.4.2 | Paquetes y clases | 22 |
| 3.4.3 | Vistas | 24 |
| 4 | APLICACIÓN MÓVIL | 24 |
| 4.1 | Tipo de Aplicación | 24 |
| 4.2 | Plataforma de desarrollo | 25 |
| 4.3 | Diseño | 25 |

| | |
|-----------------------------------|----|
| 4.4 Proyecto CORDOVA..... | 26 |
| 4.4.1 Estructura de Archivos..... | 26 |
| 4.4.2 Archivos utilizados | 27 |

1 PROPÓSITO

Este documento pretende brindar información sobre la estructura de la aplicación móvil para turismo de Ibarra y las partes que la conforman, con el fin que se dé a entender cómo fue construida.

2 OPENSIFT

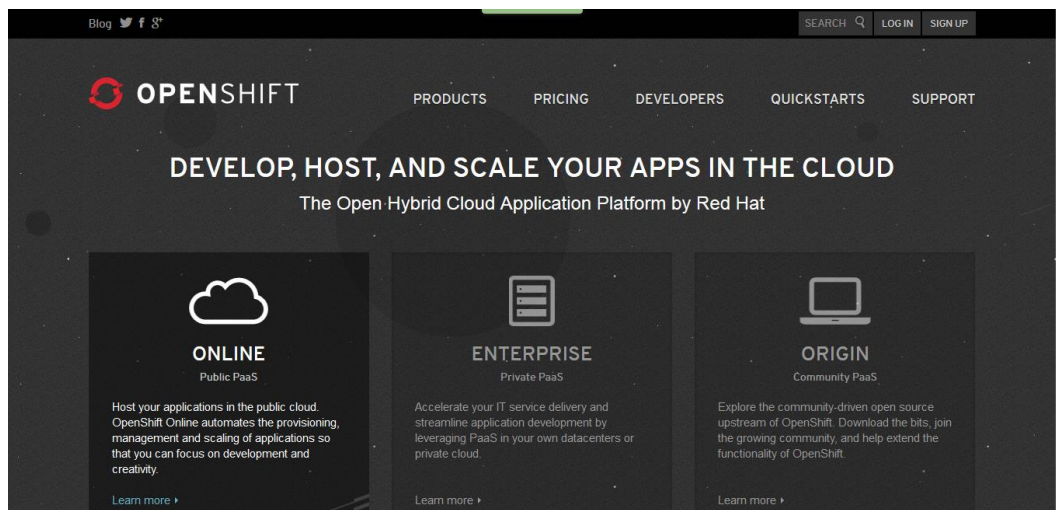
2.1 Introducción

OpenShift es la plataforma como servicio (PaaS) de Red Hat, que permite a los desarrolladores crear, gestionar y desplegar rápidamente aplicaciones escalables en un entorno de nube.

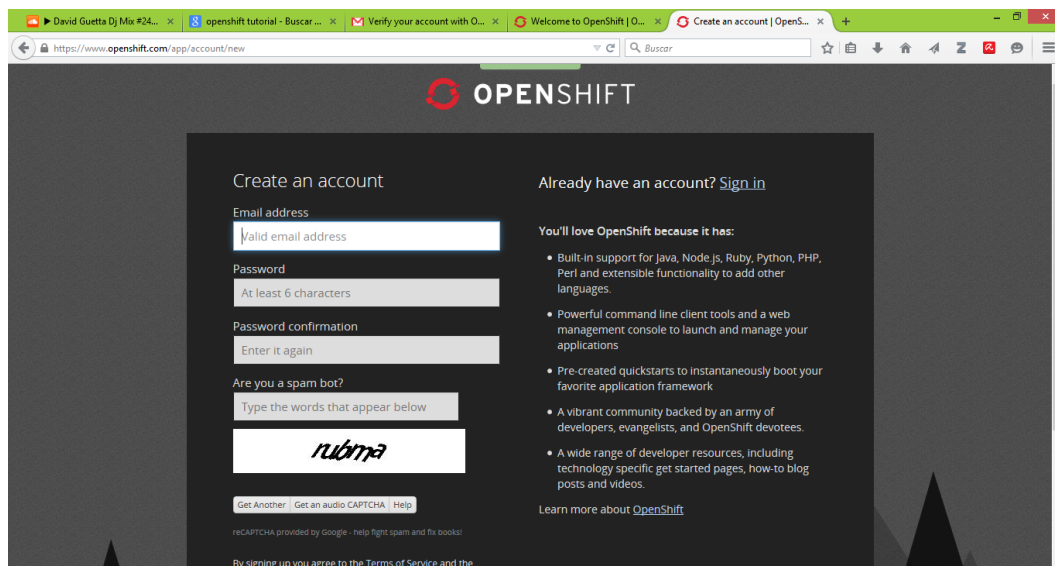
Aquí es donde se ha puesto en marcha el sistema administrador de la aplicación móvil de turismo de Ibarra, es por esta razón la vital importancia de su conocimiento.

2.2 Creación de cuenta

Para crear una cuenta nos dirigimos a <https://www.openshift.com/>, y pulsamos el botón **Sign Up**, que se encuentra en la parte superior derecha.



Luego se mostrara un formulario de registro donde ingresaremos los datos de nuestra cuenta, y pulsaremos el botón **Sign Up**.



The screenshot shows the OpenShift account creation page. The URL is <https://www.openshift.com/app/account/new>. The page features the OpenShift logo at the top. Below the logo, there are two main sections: "Create an account" and "Already have an account? [Sign in](#)".

The "Create an account" section includes the following fields:

- Email address: A text input field with a placeholder "Valid email address".
- Password: A text input field with a placeholder "At least 6 characters".
- Password confirmation: A text input field with a placeholder "Enter it again".
- Are you a spam bot?: A text input field with a placeholder "Type the words that appear below". Below this field is a CAPTCHA image showing the word "rubma".

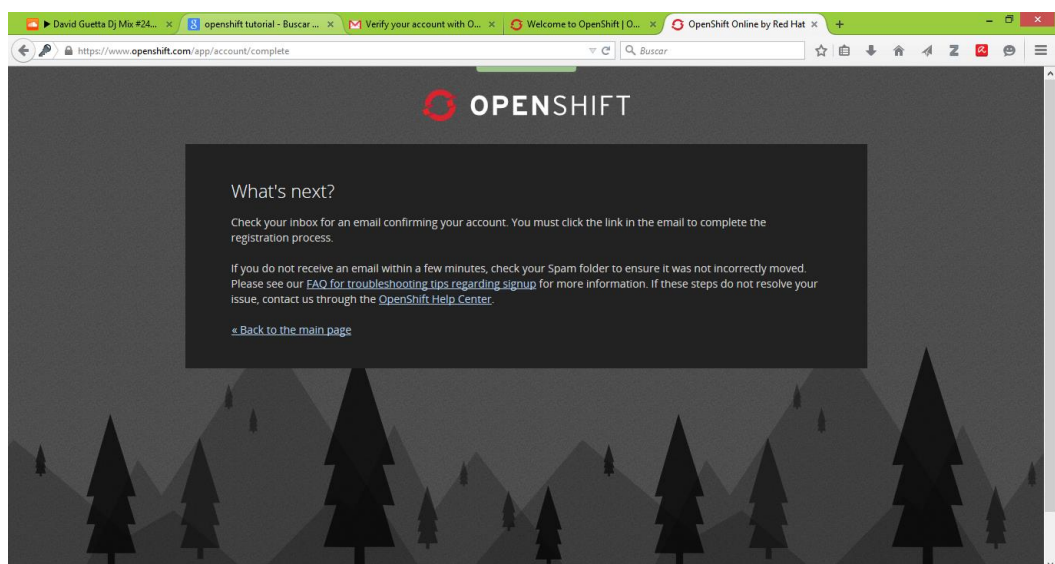
Below the CAPTCHA field, there are links for "Get Another", "Get an audio CAPTCHA", and "Help". A small note indicates "reCAPTCHA provided by Google - help fight spam and fix books!". At the bottom of the form, it says "By signing up you agree to the [Terms of Service](#) and the [Privacy Policy](#)".

The "Already have an account? [Sign in](#)" section includes a list of features under the heading "You'll love OpenShift because it has:"

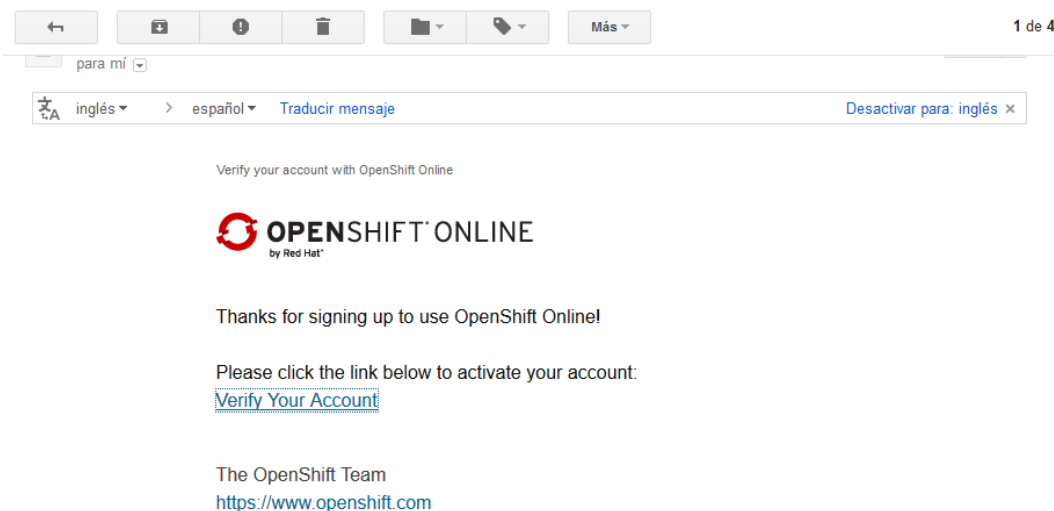
- Built-in support for Java, Node.js, Ruby, Python, PHP, Perl and extensible functionality to add other languages.
- Powerful command line client tools and a web management console to launch and manage your applications.
- Pre-created quickstarts to instantaneously boot your favorite application framework.
- A vibrant community backed by an army of developers, evangelists, and OpenShift devotees.
- A wide range of developer resources, including technology specific get started pages, how-to blog posts and videos.

At the bottom of this section, there is a link "Learn more about [OpenShift](#)".

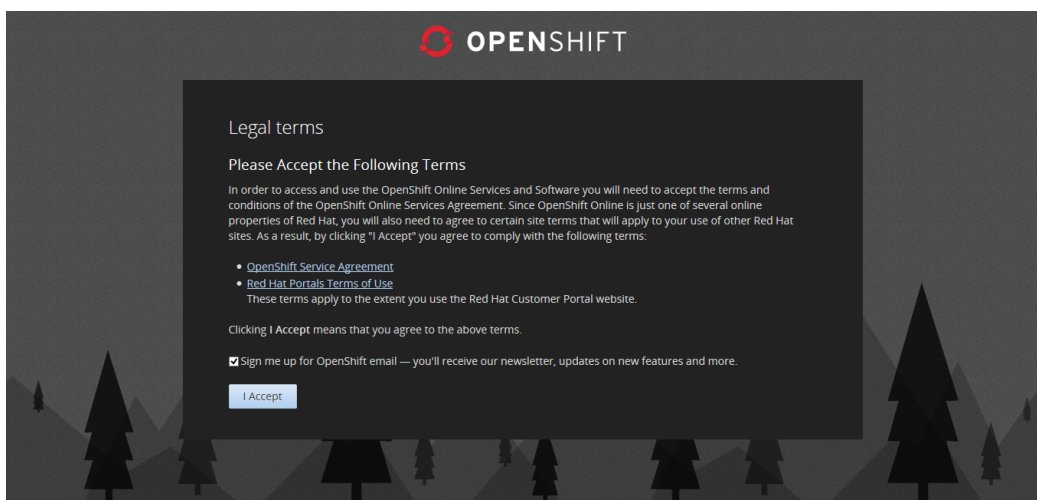
A continuación nos aparecerán las indicaciones para activar nuestra cuenta ya creada, en resumen no indican que debemos realizar este proceso mediante nuestro correo electrónico.



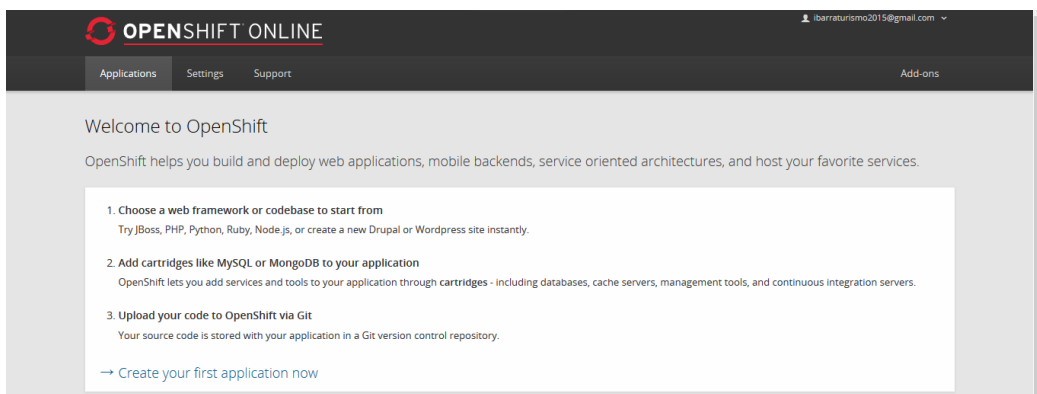
Al abrir nuestro mensaje dentro del correo electrónico, nos aparece un enlace de activación de cuenta.



Al dar clic en este nos aparece la aceptación de términos legales de uso de OpenShift.

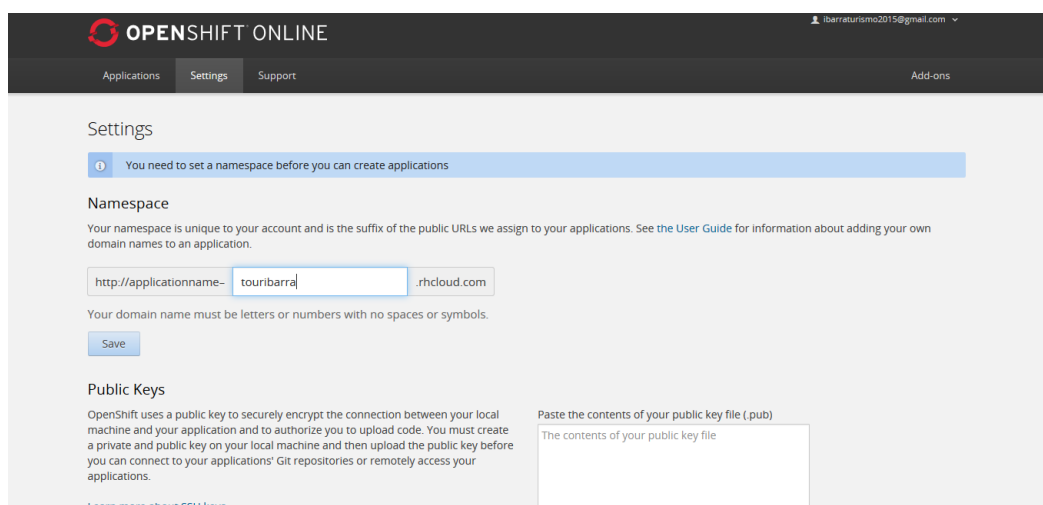


Finalmente se muestra nuestro escritorio de trabajo virtual de OpenShift.



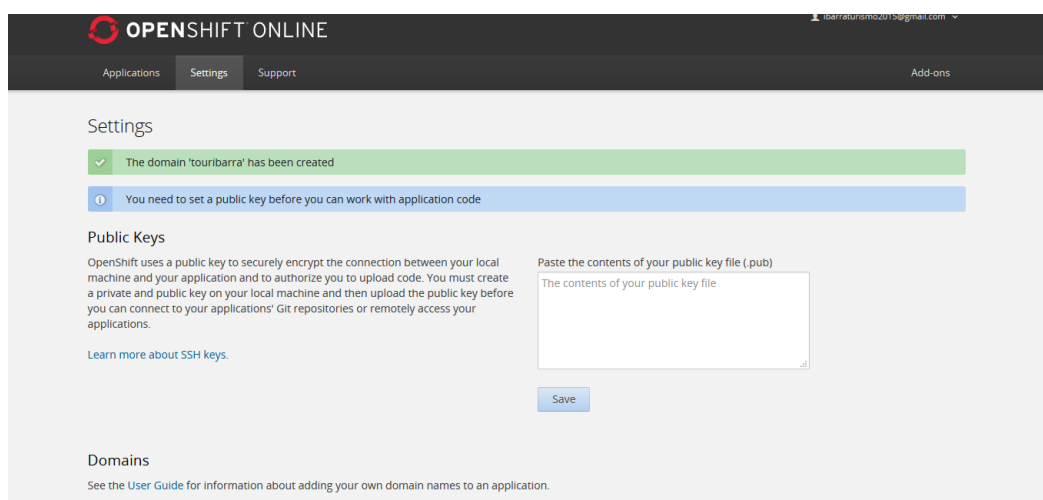
2.3 Registro de dominio

El primer paso dentro de nuestra cuenta es la creación de un dominio, para esto nos dirigimos a la pestaña de **Settings** y en la sección **Namespace** nos aparece el cuadro de texto donde escribiremos el nombre del dominio, el cual será verificado para su disponibilidad.



The screenshot shows the OpenShift Online 'Settings' page. At the top, there is a navigation bar with 'Applications', 'Settings', and 'Support' tabs. Below the navigation bar, a blue message bar states: 'You need to set a namespace before you can create applications'. The 'Namespace' section explains that the namespace is unique to the account and is the suffix of the public URLs. A text input field contains 'http://applicationname- touribarra .rhcloud.com'. Below the input field, a note says: 'Your domain name must be letters or numbers with no spaces or symbols.' A 'Save' button is visible. The 'Public Keys' section is partially visible, with a text area for pasting the contents of a public key file (.pub).

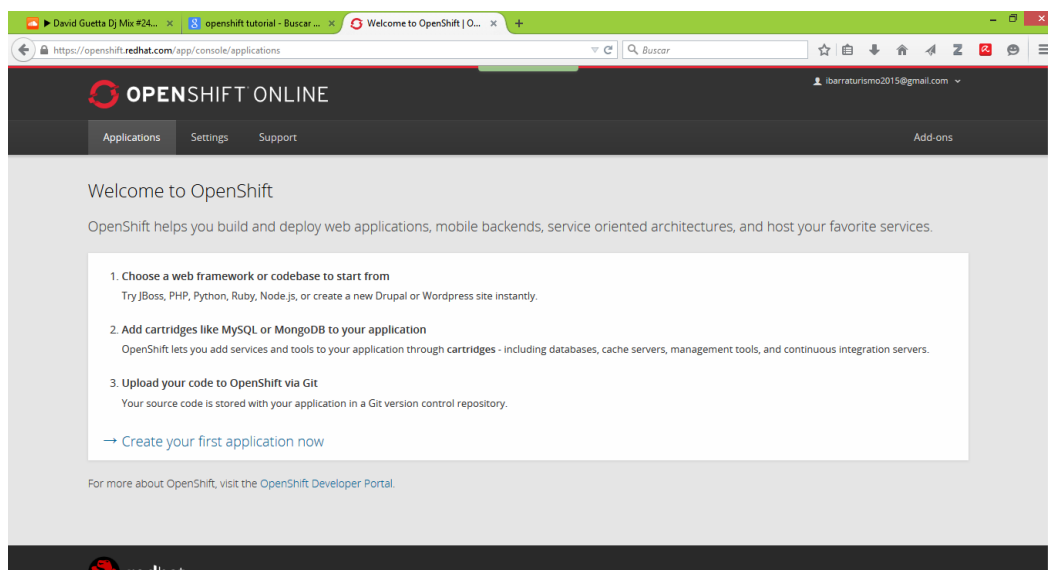
Al presionar el botón **Save** finalizaremos el proceso de creación de nuestro dominio y nos aparecerá un mensaje de éxito.



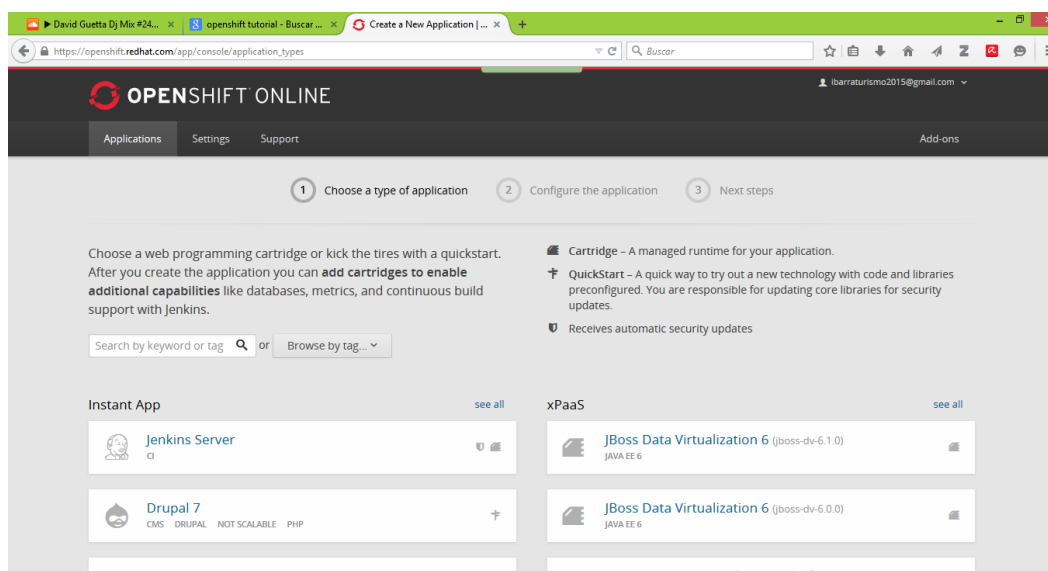
The screenshot shows the OpenShift Online 'Settings' page after the domain creation process. A green message bar at the top states: 'The domain 'touribarra' has been created'. Below this, a blue message bar states: 'You need to set a public key before you can work with application code'. The 'Public Keys' section is visible, with a text area for pasting the contents of a public key file (.pub) and a 'Save' button. The 'Domains' section is also visible, with a note: 'See the User Guide for information about adding your own domain names to an application.'

2.4 Creación de una aplicación

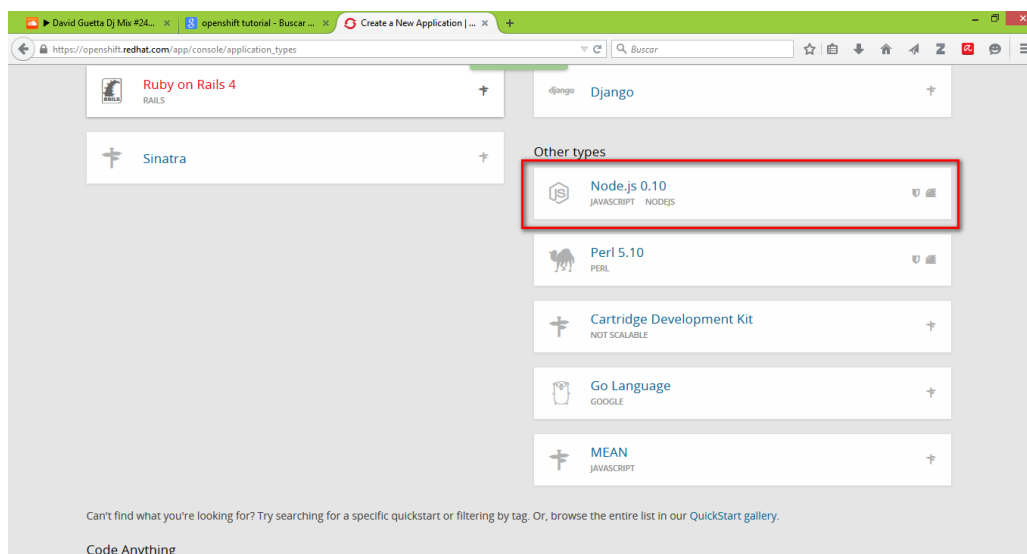
Dentro de nuestra cuenta tenemos la posibilidad de crear nuestras aplicaciones para modificarlas posteriormente. Para crear una aplicación nos dirigimos a la pestaña **Applications** y pulsamos **Create application**.



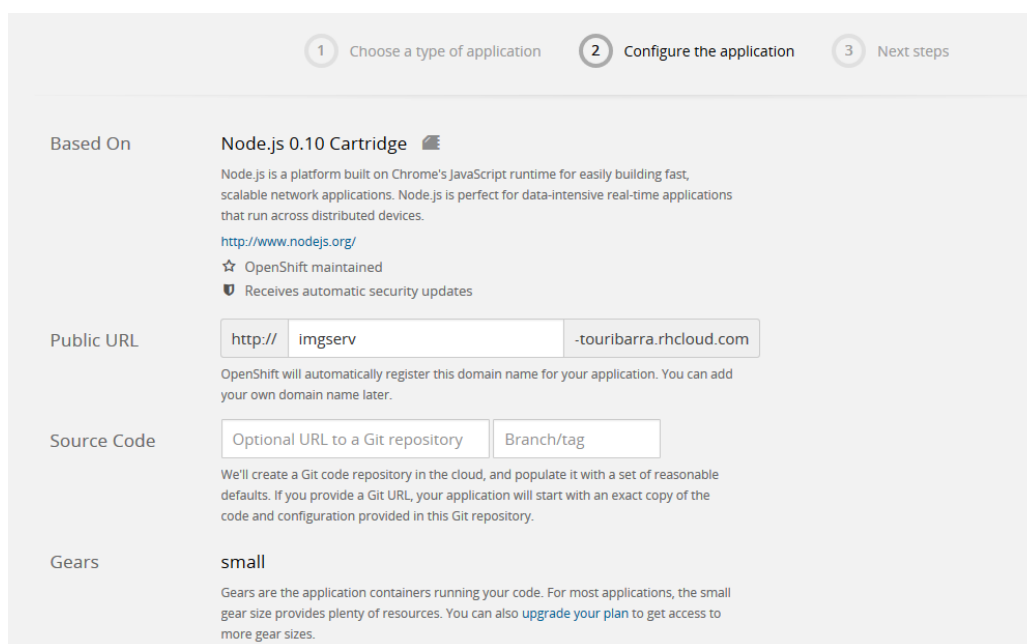
OpenShift ofrece varias posibilidades a los desarrolladores de aplicaciones como son JAVA, PHP, PHYTON, RUBY, entre otras. Basta con seleccionar la que necesitamos y seguir los pasos de forma intuitiva.



En este caso navegamos por las opciones y seleccionamos Node.js.



Ahora configuraremos nuestra aplicación y procederemos a crearla.



Una vez creada nos aparecerán opciones de ayuda y manejo de la misma.

1 Choose a type of application 2 Configure the application 3 Next steps

Your application has been created. [Continue to the application overview page.](#)

Making code changes

Install the Git client for your operating system, and from your command line run

```
git clone ssh://559f79634382ec4fab00041@imgserv-touribarra.rhcloud.com/~/.git
/imgserv.git/
cd imgserv/
```

This will create a folder with the source code of your application. After making a change, add, commit, and push your changes.

```
git add .
git commit -m 'My changes'
git push
```

When you push changes the OpenShift server will report back its status on deploying your code. The server will run any of your configured [deploy hooks](#) and then restart the application.

Manage your app

The console is convenient, but if you need deeper control try our other client tools

Command-Line

All of the capabilities of OpenShift are exposed through our command line tool, [rhc](#). [Follow these steps to install the client on Linux, Mac OS X, or Windows.](#)

After installing the RHC read more on [how to manage your application from the command line in our User Guide.](#)

JBoss Developer Studio

The JBoss Developer Studio is a full featured IDE with OpenShift integration built in. It gives you the ability to create, edit and deploy applications without having to leave the IDE. Links to download, install and use the JBoss Developer Studio for Linux, Mac OS X, or Windows can be found on the [JBoss Developer Studio tools page.](#)

Podemos verificar la creación de la aplicación escribiendo la dirección en nuestro navegador.

David Guetta's stream on S... openshift tutorial - Buscar... Next Steps | OpenShift Onli... Welcome to OpenShift

imgserv-touribarra.rhcloud.com

Welcome to your Node.js application on OpenShift

Deploying code changes

OpenShift uses the [Git version control system](#) for your source code, and grants you access to it via the Secure Shell (SSH) protocol. In order to upload and download code to your application you need to give us your [public SSH key](#). You can upload it within the web console or install the [RHC command line tool](#) and run `rhc setup` to generate and upload your key automatically.

Working in your local Git repository

If you created your application from the command line and uploaded your SSH key, `rhc` will automatically download a copy of that source code repository (Git calls this "cloning") to your local system.

If you created the application from the web console, you'll need to manually clone the repository to your local system. Copy the application's source code Git URL and then run:

```
$ git clone <git_url> <directory_to_create>
# Within your project directory
# Commit your changes and push to OpenShift
$ git commit -a -m 'Some commit message'
$ git push
```

Managing your application

Web Console

You can use the OpenShift web console to enable additional capabilities via cartridges, add collaborator access authorizations, designate custom domain aliases, and manage domain memberships.

Command Line Tools

Installing the [OpenShift RHC client tools](#) allows you complete control of your cloud environment. Read more on how to manage your application from the command line in our [User Guide](#).

Development Resources

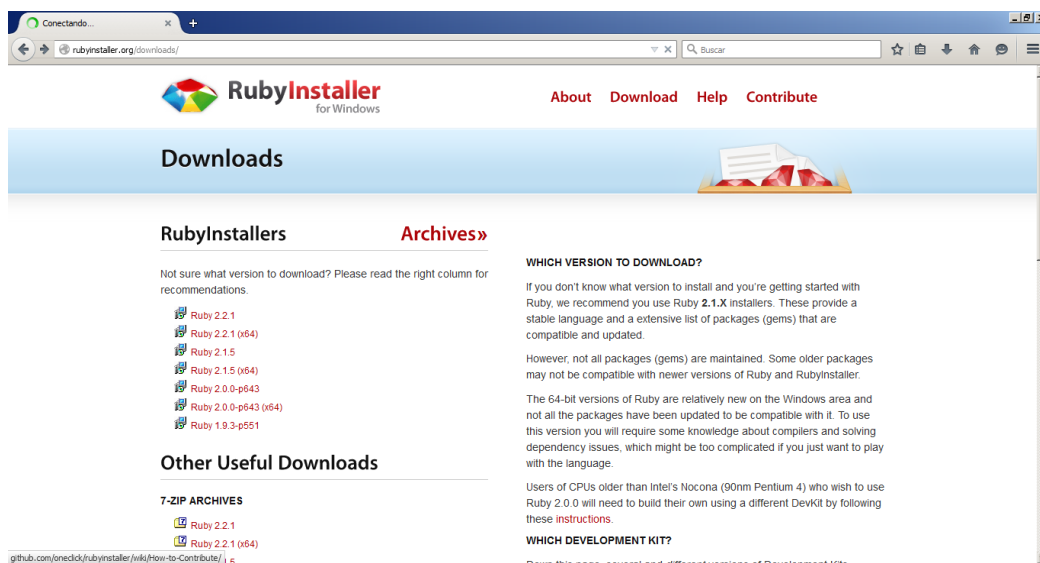
- [Getting Started with Node.js on OpenShift](#)
- [Developer Center](#)
- [User Guide](#)
- [Help Center](#)
- [Stack Overflow questions for OpenShift](#)
- [Git documentation](#)

2.5 Instalación del cliente de OpenShift

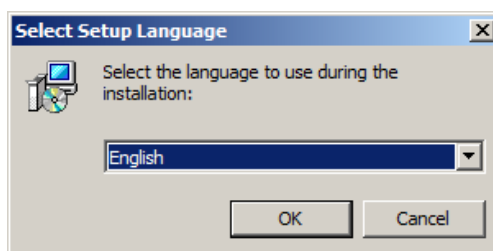
Para poder manejar OpenShift debemos instalar dependencias importantes, como son: Ruby, Git y RHC Gem.

2.5.1 Ruby

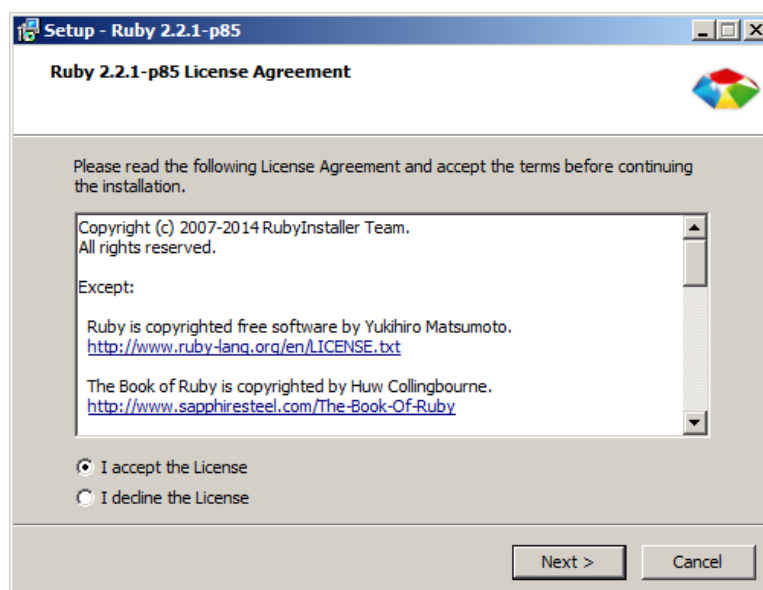
Para instalar Ruby en el sistema operativo Windows vamos a el sitio <http://rubyinstaller.org/downloads>, y elegimos la versión del instalador.



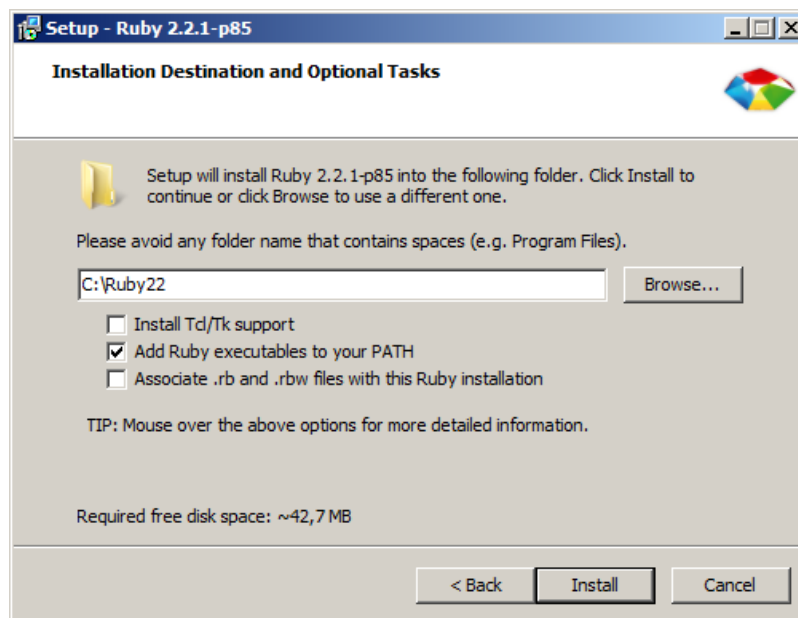
Al ejecutar el instalador nos mostrará un mensaje de selección de idioma para la instalación, lo dejaremos en English por defecto.



Luego aceptaremos las condiciones de la licencia del lenguaje de programación.



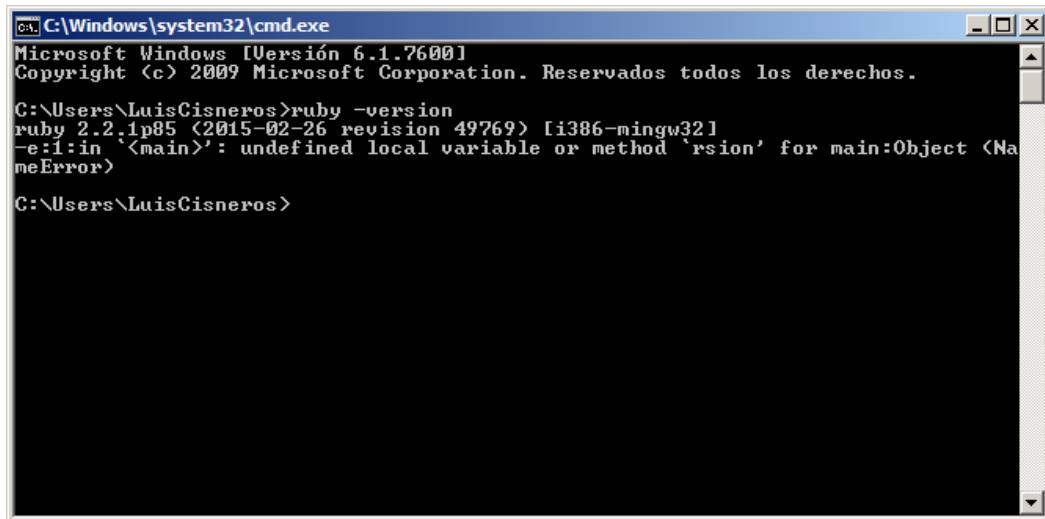
Escogeremos la carpeta de instalación y seleccionaremos la segunda casilla, permitiendo añadir las variables al PATH.



Finalmente al pulsar el botón Install se inicia el proceso de instalación y cuando este termine, se mostrará la ventana de finalización exitosa.



Si deseamos comprobar la instalación, nos dirigimos al terminal de Windows y ejecutamos el comando **ruby -version**.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

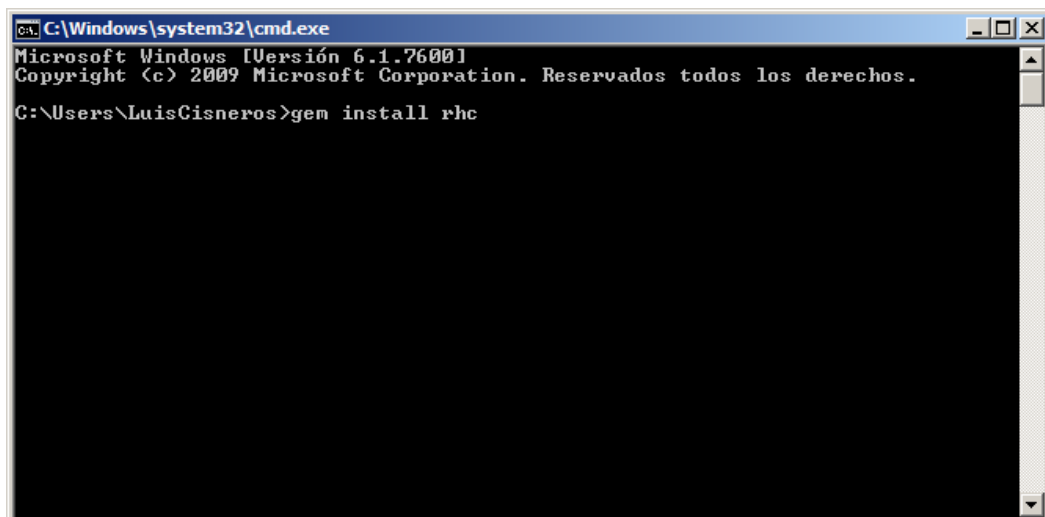
C:\Users\LuisCisneros>ruby -version
ruby 2.2.1p85 (2015-02-26 revision 49769) [i386-mingw32]
-e:1:in <main>: undefined local variable or method <rsion> for main:Object <NameError>

C:\Users\LuisCisneros>
```

2.5.2 Gema RHC

La gema RHC, es la herramienta por así decirlo para la conexión hacia los servicios de OpenShift y el manejo de cada uno de nuestras aplicaciones mediante SSH.

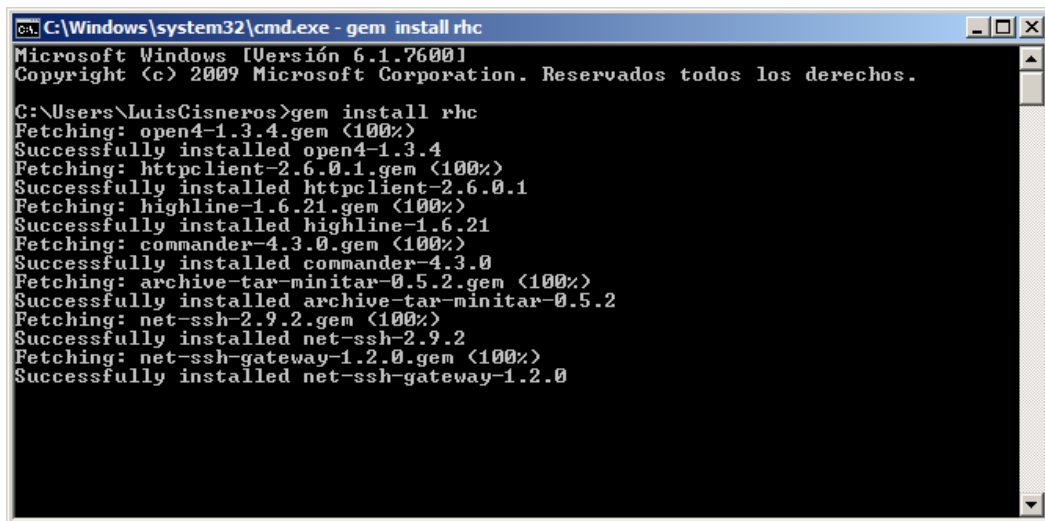
Para instalar la gema, ingresamos al terminal y ejecutamos el comando **gem install rhc**.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\LuisCisneros>gem install rhc
```

Rápidamente el proceso de instalación inicia y se instalan las otras gemas dependientes.



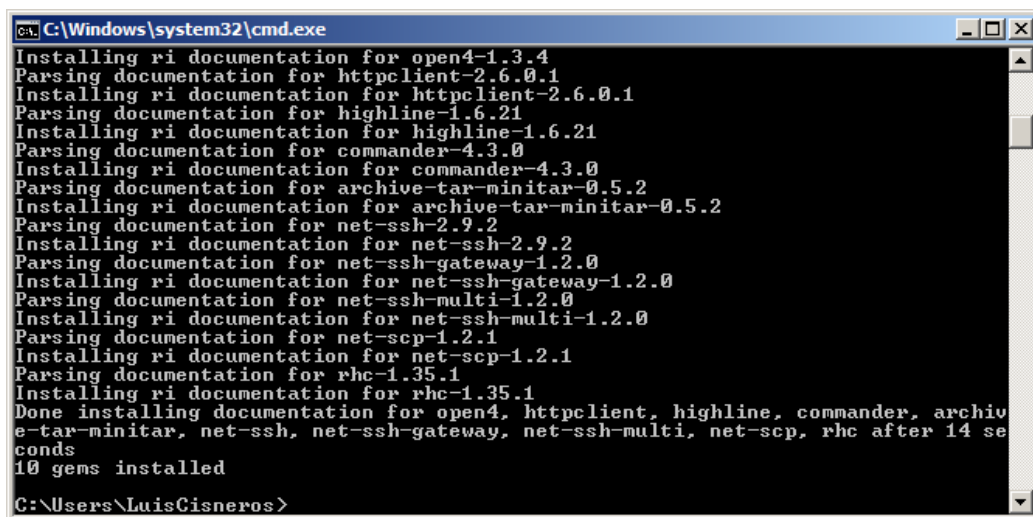
```

C:\Windows\system32\cmd.exe - gem install rhc
Microsoft Windows [Versión 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.

C:\Users\LuisCisneros>gem install rhc
Fetching: open4-1.3.4.gem (100%)
Successfully installed open4-1.3.4
Fetching: httpclient-2.6.0.1.gem (100%)
Successfully installed httpclient-2.6.0.1
Fetching: highline-1.6.21.gem (100%)
Successfully installed highline-1.6.21
Fetching: commander-4.3.0.gem (100%)
Successfully installed commander-4.3.0
Fetching: archive-tar-minitar-0.5.2.gem (100%)
Successfully installed archive-tar-minitar-0.5.2
Fetching: net-ssh-2.9.2.gem (100%)
Successfully installed net-ssh-2.9.2
Fetching: net-ssh-gateway-1.2.0.gem (100%)
Successfully installed net-ssh-gateway-1.2.0

```

Una vez finalizado el proceso nos aparecerá un mensaje de éxito.



```

C:\Windows\system32\cmd.exe
Installing ri documentation for open4-1.3.4
Parsing documentation for httpclient-2.6.0.1
Installing ri documentation for httpclient-2.6.0.1
Parsing documentation for highline-1.6.21
Installing ri documentation for highline-1.6.21
Parsing documentation for commander-4.3.0
Installing ri documentation for commander-4.3.0
Parsing documentation for archive-tar-minitar-0.5.2
Installing ri documentation for archive-tar-minitar-0.5.2
Parsing documentation for net-ssh-2.9.2
Installing ri documentation for net-ssh-2.9.2
Parsing documentation for net-ssh-gateway-1.2.0
Installing ri documentation for net-ssh-gateway-1.2.0
Parsing documentation for net-ssh-multi-1.2.0
Installing ri documentation for net-ssh-multi-1.2.0
Parsing documentation for net-scp-1.2.1
Installing ri documentation for net-scp-1.2.1
Parsing documentation for rhc-1.35.1
Installing ri documentation for rhc-1.35.1
Done installing documentation for open4, httpclient, highline, commander, archiv
e-tar-minitar, net-ssh, net-ssh-gateway, net-ssh-multi, net-scp, rhc after 14 se
conds
10 gems installed

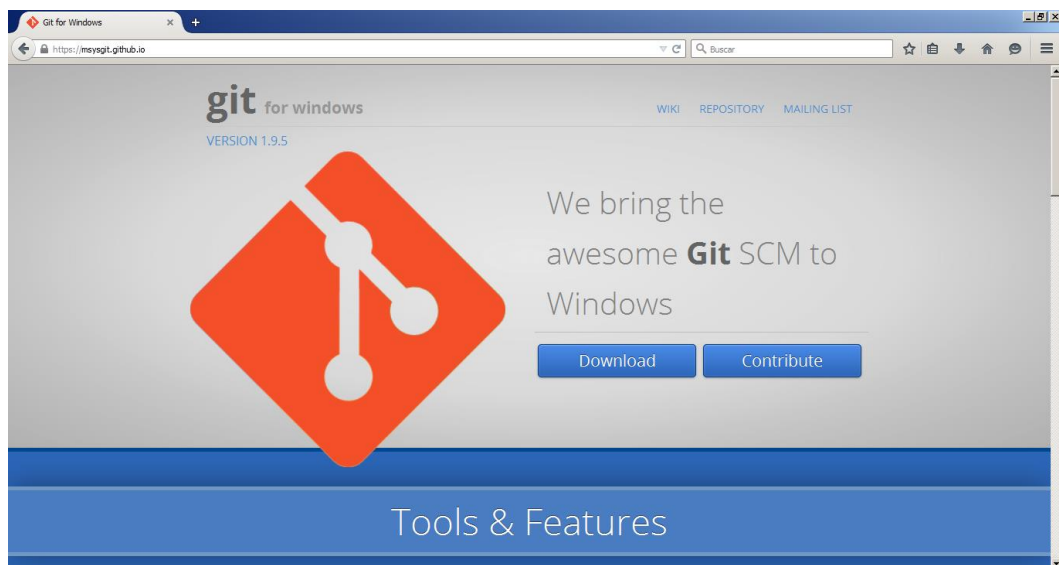
C:\Users\LuisCisneros>

```

2.5.3 Git

Git es un software de control de versiones, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando estas tienen un gran número de archivos de código fuente. Además nos servirá para clonar nuestros archivos de las aplicaciones de la plataforma OpenShift.

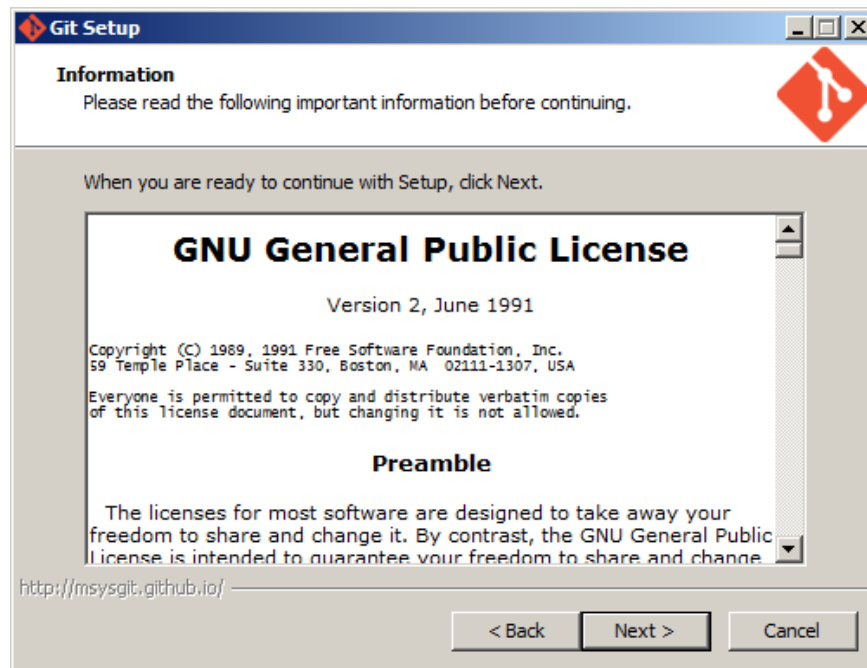
La instalación del software es muy sencilla, primero nos dirigimos a la siguiente dirección web <https://msysgit.github.io/> y pulsaremos el botón descargar.



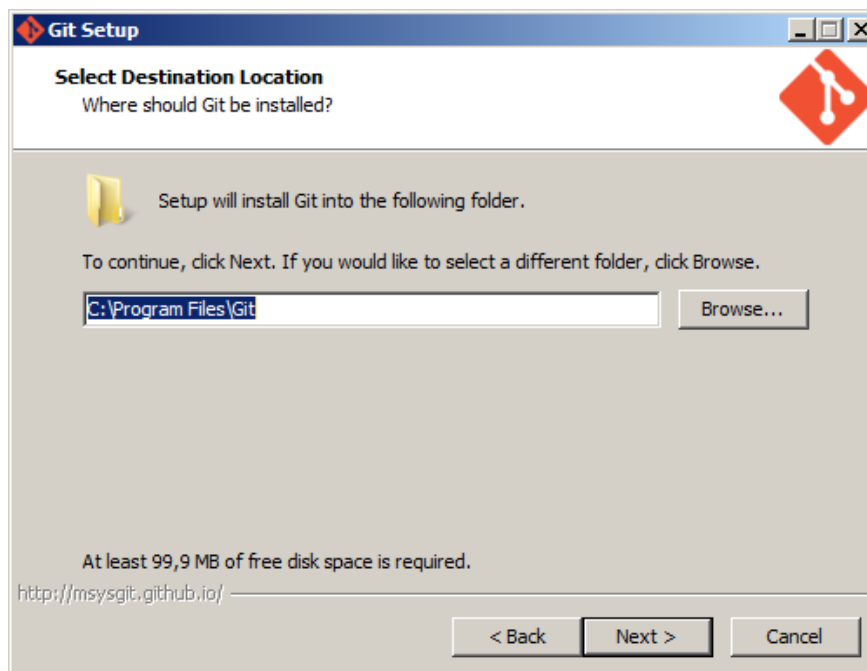
Una vez descargado el instalador, lo ejecutamos iniciando así el proceso de instalación dentro del computador.



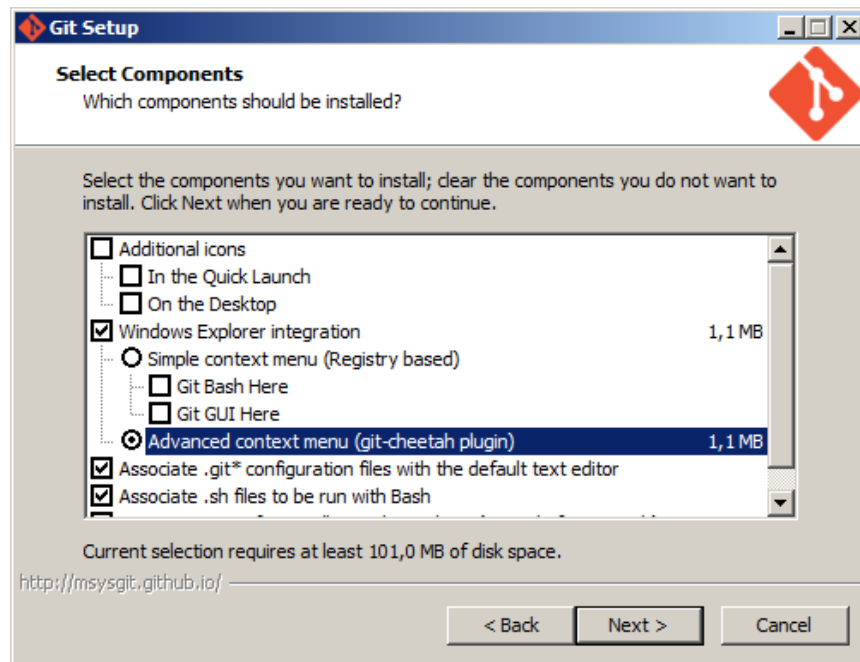
Aceptaremos los términos y condiciones de la licencia.



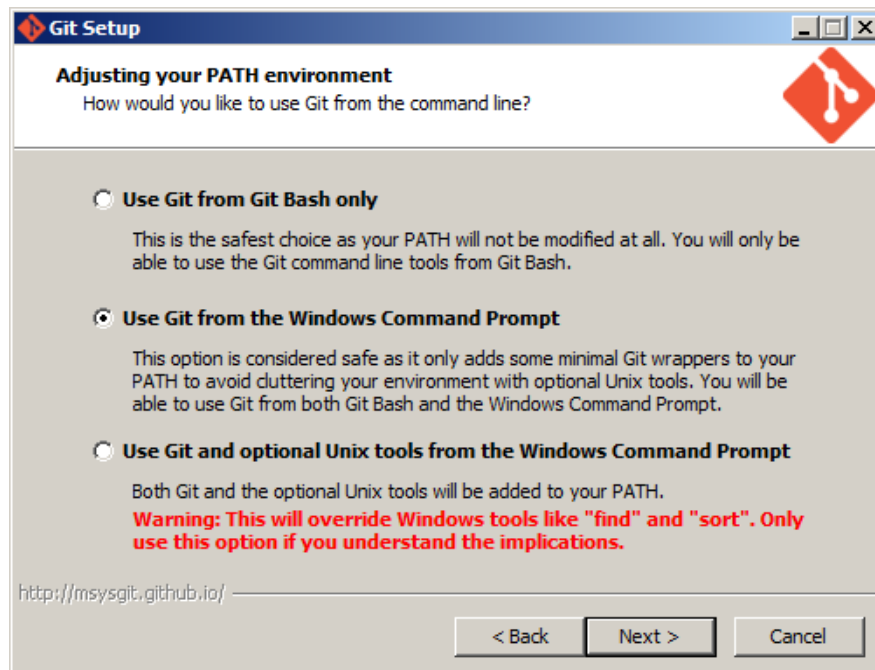
Elegimos el directorio de instalación.



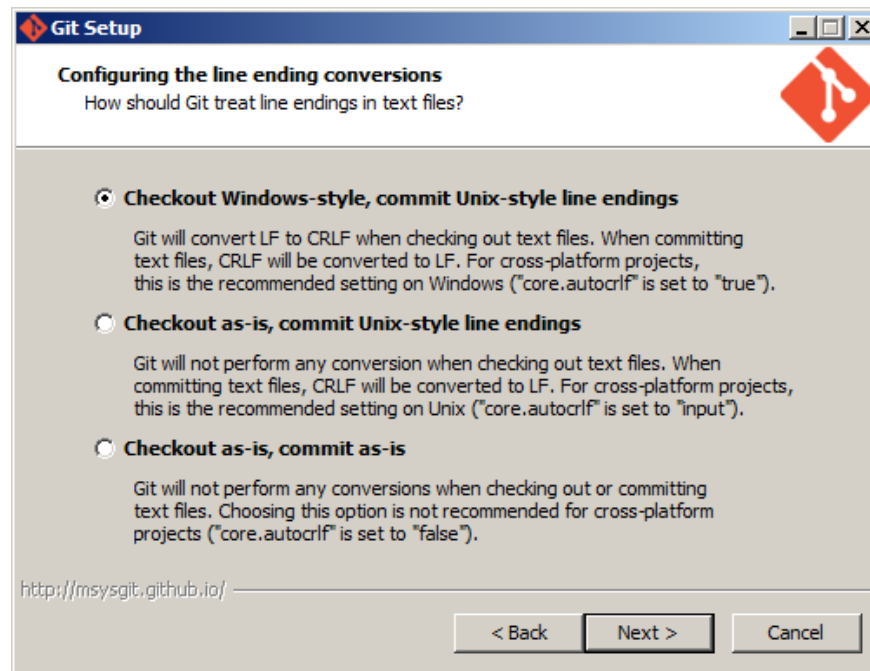
Seleccionamos los componentes a instalar.



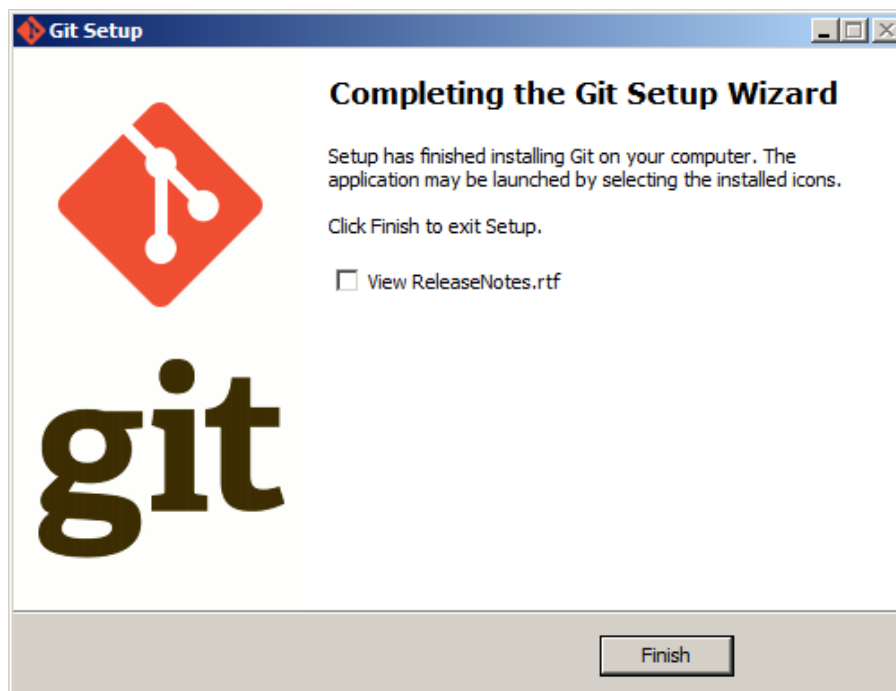
Asignaremos las variables a nuestro PATH.



Dejaremos las últimas configuraciones por defecto e inicia el proceso de instalación.



Una vez terminado saldrá una ventana con un mensaje de éxito.



2.5.4 Conexión hacia OpenShift

Para realizar nuestra primera conexión hacia la plataforma y la configuración de nuestra cuenta, nos dirigimos al terminal y ejecutamos el comando **rhc**

setup. En seguida se muestra un mensaje del cliente de OpenShift y daremos **Enter** en la selección del nombre del servidor. Si poseemos un servidor propio con la plataforma OpenShift Origin, escribiremos el nombre de nuestro host.

```
C:\>rhc setup
OpenShift Client Tools (RHC) Setup Wizard

This wizard will help you upload your SSH keys, set your application namespace,
and check that other programs like Git are properly installed.

If you have your own OpenShift server, you can specify it now. Just hit enter to
use the server for OpenShift Online: openshift.redhat.com.
Enter the server hostname: !openshift.redhat.com!
```

Luego nos pedirá darnos de alta, digitando nuestro e-mail y contraseña de la cuenta creada anteriormente.

```
C:\Windows\system32\cmd.exe - rhc setup
    from C:/Ruby22/bin/rhc:23:in `load'
    from C:/Ruby22/bin/rhc:23:in `<main>'

C:\>gem install net-ssh -v 2.9.3.beta1
Fetching: net-ssh-2.9.3.beta1.gem (100%)
Successfully installed net-ssh-2.9.3.beta1
Parsing documentation for net-ssh-2.9.3.beta1
Installing ri documentation for net-ssh-2.9.3.beta1
Done installing documentation for net-ssh after 3 seconds
1 gem installed

C:\>rhc setup
OpenShift Client Tools (RHC) Setup Wizard

This wizard will help you upload your SSH keys, set your application namespace,
and check that other programs like Git are properly installed.

If you have your own OpenShift server, you can specify it now. Just hit enter to
use the server for OpenShift Online: openshift.redhat.com.
Enter the server hostname: !openshift.redhat.com!

You can add more servers later using 'rhc server'.

Login to openshift.redhat.com: ibarraturismo2015@gmail.com
Password: *****
```

Crearemos las claves de conexión para evitar iniciar sesión a cada instante escribiendo **yes** y pulsando **Enter**.

```

C:\Windows\system32\cmd.exe - rhc setup
Done installing documentation for net-ssh after 3 seconds
1 gem installed

C:\>rhc setup
OpenShift Client Tools (RHC) Setup Wizard

This wizard will help you upload your SSH keys, set your application namespace,
and check that other programs like Git are properly installed.

If you have your own OpenShift server, you can specify it now. Just hit enter to
use the server for OpenShift Online: openshift.redhat.com.
Enter the server hostname: lopenshift.redhat.com!

You can add more servers later using 'rhc server'.

Login to openshift.redhat.com: ibarraturismo2015@gmail.com
Password: *****

OpenShift can create and store a token on disk which allows to you to access the
server without using your password. The key is stored in your home directory and
should be kept secret. You can delete the key at any time by running 'rhc
logout'.
Generate a token now? (yes|no) yes

```

También crearemos nuestra clave SSH pública que nos permitirá manejar el entorno de OpenShift, para esto escribiremos **yes** y pulsaremos **Enter**.

```

C:\Windows\system32\cmd.exe - rhc setup
use the server for OpenShift Online: openshift.redhat.com.
Enter the server hostname: lopenshift.redhat.com!

You can add more servers later using 'rhc server'.

Login to openshift.redhat.com: ibarraturismo2015@gmail.com
Password: *****

OpenShift can create and store a token on disk which allows to you to access the
server without using your password. The key is stored in your home directory and
should be kept secret. You can delete the key at any time by running 'rhc
logout'.
Generate a token now? (yes|no) yes
Generating an authorization token for this client ... lasts about 1 month

Saving configuration to C:\Users\LuisCisneros\.openshift\express.conf ... done

No SSH keys were found. We will generate a pair of keys for you.

Created: C:/Users/LuisCisneros/.ssh/id_rsa.pub

Your public SSH key must be uploaded to the OpenShift server to access code.
Upload now? (yes|no)
yes

```

Finalmente estará configurada nuestra herramienta cliente para el manejo de la plataforma OpenShift.

```

C:\Windows\system32\cmd.exe
JBoss Unified Push Server      rhc create-app <app name>
1.0.0.Beta1                   jboss-unified-push-1
Jenkins Server                 rhc create-app <app name> jenkins-1
Node.js 0.10                  rhc create-app <app name> node.js-0.10
PHP 5.3                       rhc create-app <app name> php-5.3
PHP 5.4                       rhc create-app <app name> php-5.4
PHP 5.4 with Zend Server 6.1  rhc create-app <app name> zend-6.1
Perl 5.10                     rhc create-app <app name> perl-5.10
Python 2.6                    rhc create-app <app name> python-2.6
Python 2.7                    rhc create-app <app name> python-2.7
Python 3.3                    rhc create-app <app name> python-3.3
Ruby 1.8                      rhc create-app <app name> ruby-1.8
Ruby 1.9                      rhc create-app <app name> ruby-1.9
Ruby 2.0                      rhc create-app <app name> ruby-2.0
Tomcat 6 (JBoss EWS 1.0)      rhc create-app <app name> jbossews-1.0
Tomcat 7 (JBoss EWS 2.0)      rhc create-app <app name> jbossews-2.0
Vert.x 2.1                    rhc create-app <app name> jboss-vertx-2.1
WildFly Application Server     rhc create-app <app name> jboss-wildfly-8

You are using 0 of 3 total gears
The following gear sizes are available to you: small

Your client tools are now configured.
C:\>

```

3 SISTEMA ADMINISTRADOR

3.1 Lenguaje

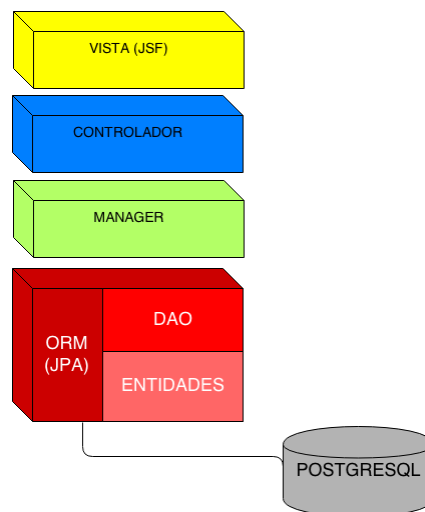
El sistema administrador GESTUR, está construido en el lenguaje de programación JAVA mediante el framework para aplicaciones web JSF (Java Server Faces), que permite crear aplicaciones J2EE basadas en el patrón MVC.

3.2 Arquitectura

Para el desarrollo del sistema de administración de la aplicación móvil se ha establecido seguir el patrón de arquitectura MVC (Modelo Vista Controlador), ya que este trata de manera separada los datos, la lógica del negocio, y las interfaces de usuario; logrando así un software de calidad.

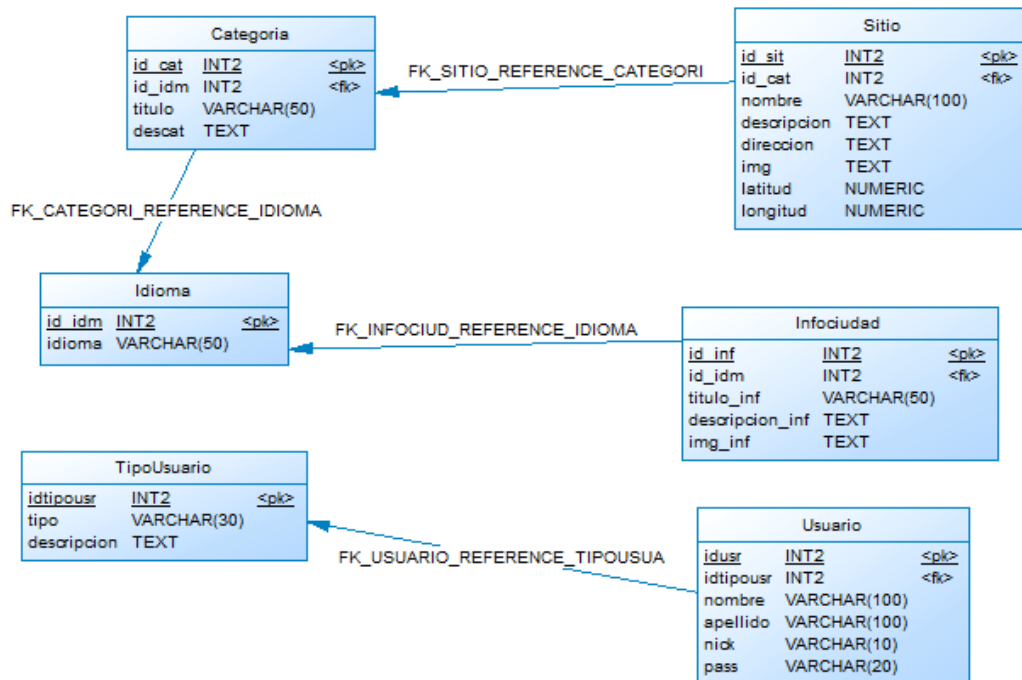
Para detallar un poco más sobre el patrón de arquitectura MVC explicaremos cada uno de sus componentes (capas):

- **Modelo.-** Es aquella capa donde trabajan todos los datos correspondientes a la aplicación. Además gestiona todo el acceso y modificación de la información.
- **Controlador.-** Posee todo el código necesario para responder a las acciones solicitadas de la aplicación.
- **Vista.-** Presenta las distintas interfaces de usuario que serán utilizadas en la aplicación. Estas usualmente son presentadas en código HTML, XHTML, PHP, entre otros.



3.3 Diagrama Entidad-Relación

Mediante el diagrama entidad-relación se pretende mostrar el diseño del esquema de base de datos que se usará en el aplicativo.



Cada tabla anteriormente presentada tiene su propósito, este es:

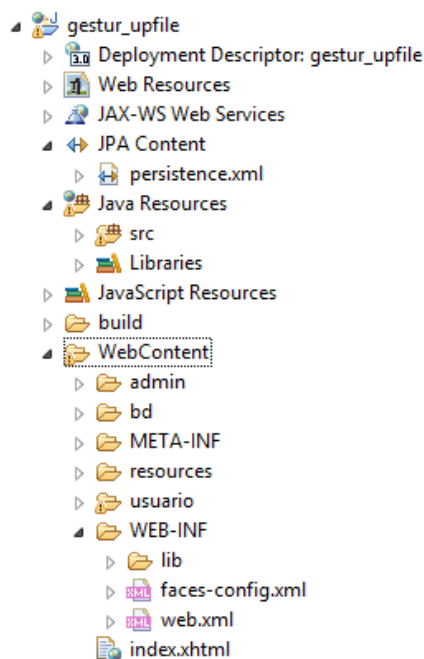
- TipoUsuario: Permite clasificar a los usuarios que intervienen dentro del sistema.
- Usuario: Representa a los datos personales de cada usuario del sistema.
- Idioma: Es el idioma que se usará para la información de la ciudad y sus atractivos; es decir, puede ser Español o Inglés.
- Infociedad: Contiene la información representativa de la ciudad de Ibarra.
- Categoria: Representa a la clasificación de los atractivos turísticos de la ciudad.

- Sitio: Contiene la información de los atractivos turísticos de la ciudad de Ibarra, estos dependen de la categoría para su clasificación.

3.4 Proyecto WEB

3.4.1 Estructura

Dentro de nuestro IDE de desarrollo, desplegamos la carpeta del proyecto y observamos su estructura en forma de árbol de directorios. Los directorios más importantes serán explicados a continuación:

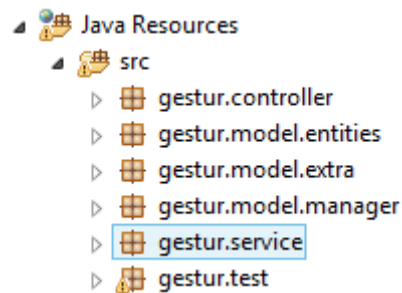


- WebContent: Contiene tanto las páginas web (vistas), como dos directorios que son necesarios para definir la configuración de la aplicación. Estos directorios son:
 - ✓ META-INF: Contiene el fichero context.xml.
 - ✓ WEB-INF: Contiene los ficheros de configuración:

- faces-config.xml: Contiene instrucciones de configuración que atañen a las clases java de respaldo de las páginas Web. Aquí se inicializan variables, se enlaza código Java con páginas Web, se indica cómo se salta de una página Web a otra, etc.
- web.xml: Recoge parámetros de configuración generales de la aplicación, tales como cuáles son las páginas de inicio y error, qué recursos adicionales usa la aplicación, temas de seguridad, etc.
- Java Resources: Contiene directorios con los recursos de programación; es decir, paquetes, clases y librerías. Estos directorios son:
 - ✓ Source Packages: Contiene nuestro código Java.
 - ✓ Libraries: Las librerías que necesita nuestro programa.
- JPA Content: Posee la unidad de persistencia de JPA que define un conjunto de todas las entidades (clases) que son gestionadas por la instancia del EntityManager en una aplicación. Este conjunto de clases de entidad representa los datos contenidos en una única base de datos y su conexión hacia la misma. Las unidades de persistencia se definen en el fichero de configuración persistence.xml.

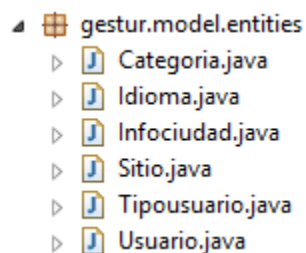
3.4.2 Paquetes y clases

El proyecto consta de 6 paquetes que contienen distintas clases JAVA que siguiendo el patrón MVC otorgan funcionalidad al aplicativo.

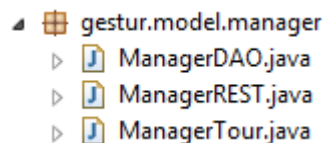


Cada uno de los paquetes será expuesto a continuación:

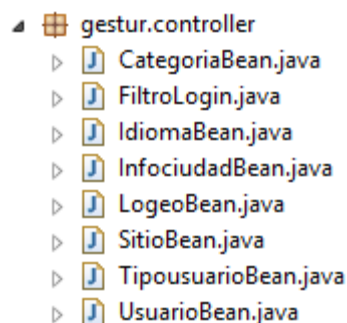
- gestur.model.entities.- Posee las clases entidades que representan a una tabla en el modelo de datos relacional y contienen los atributos de las mismas.



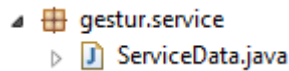
- gestur.model.manager.- Posee las clases que permiten el acceso a los objetos representados por las entidades y contiene métodos que gestionan los datos dentro de cada tabla.



- gestur.controller.- Posee las clases que interactúan con las vistas.

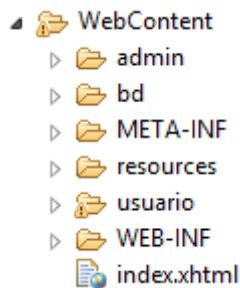


- `gestur.service`.- Contiene el Servlet que permite la creación del servicio web REST.

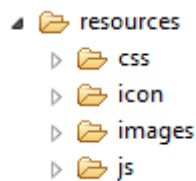


3.4.3 Vistas

Las representaciones de las distintas interfaces de usuario se encuentran en el WebContent, la vista principal es **index.html** y las otras se encuentran respectivamente en las carpetas **admin** y **usuario**.



Cabe recalcar que los recursos CSS y JavaScript, como las imágenes que usan cada una de las vistas se encuentran dentro de la carpeta **resources**.



4 APLICACIÓN MÓVIL

4.1 Tipo de Aplicación

El cliente móvil desarrollado forma parte de las aplicaciones móviles híbridas, estas son creadas con tecnologías Web: HTML5, JavaScript y se ejecutan sobre un contenedor nativo utilizando el motor de un navegador móvil para procesar su contenido de manera local (WebView).

4.2 Plataforma de desarrollo

Apache Cordova (<http://cordova.apache.org>) es un framework gratuito de código abierto, usado para la construcción de aplicaciones multiplataforma nativas con tecnologías web (HTML, CSS y JavaScript). Córdoba implementa un conjunto de APIs que extiende el uso de las capacidades nativas de los dispositivos (tales como la cámara, acelerómetro, la aplicación contactos, entre otras.) para la creación de una aplicación.

4.3 Diseño

La aplicación móvil pretende ser atractiva para el usuario y servir como fuente de información sobre los atractivos turísticos que posee la ciudad de Ibarra. Para mayor facilidad de uso por los turistas, esta se encuentra en el idioma español e inglés.

El diseño de la aplicación es simple y consta de:

- Animación de inicio.
- Selección de idioma.
- Menú con categorías de atractivos.
- Enlaces a información general de la ciudad.
- Presentación de atractivos por categoría.
- Información específica de cada atractivo junto a su ubicación en la ciudad.

4.4 Proyecto CORDOVA

4.4.1 Estructura de Archivos

Al crear un proyecto CORDOVA aparecerá un directorio con el nombre de nuestra aplicación, dentro de esta encontramos diferentes carpetas. Las más importantes son **platforms** y **www**.

| Name | Date modified | Type |
|-----------|--------------------|-------------|
| .cordova | 1/31/2014 12:41 AM | File folder |
| merges | 1/31/2014 1:51 AM | File folder |
| platforms | 1/31/2014 1:50 AM | File folder |
| plugins | 2/2/2014 6:52 PM | File folder |
| www | 2/16/2014 8:03 PM | File folder |

PLATFORMS contiene las plataformas de los distintos sistemas operativos soportados por CORDOVA, si en nuestro caso trabajamos el proyecto para el sistema operativo Android; dentro de esa carpeta encontraremos una carpeta con el nombre android.

La carpeta WWW contiene todos los archivos y directorios relacionados con la estructura de nuestra aplicación. Esta estructura de directorios y archivos es la misma que una aplicación web cliente.

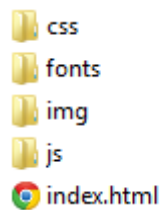
| Name | Date modified | Type |
|------------|-------------------|-------------------|
| css | 4/1/2014 11:28 PM | File folder |
| img | 4/1/2014 11:28 PM | File folder |
| js | 4/1/2014 11:28 PM | File folder |
| config.xml | 4/1/2014 11:28 PM | XML File |
| index.html | 4/1/2014 11:28 PM | Opera Web Docu... |

La única diferencia que encontraremos es un archivo “config.xml”, el cual no deberemos borrar, todo lo demás podemos borrarlo y pegar nuestra

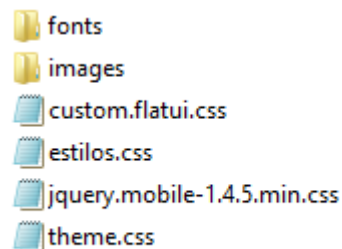
aplicación web; ya que este archivo contiene la configuración esencial de nuestro proyecto CORDOVA. Si revisamos ese archivo podemos ver que hay información del proyecto; como el nombre, con que archivo va a iniciar la aplicación (index.html), y datos de plugins si en un caso usamos.

4.4.2 Archivos utilizados

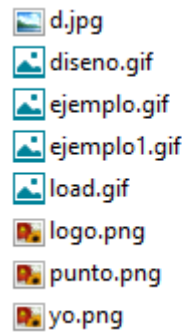
El proyecto web consta de la estructura de archivos donde existen carpetas para almacenar los archivos de extensiones .css y .js correspondientemente. También consta de una carpeta para almacenar las imágenes usadas y de nuestro archivo principal index.html.



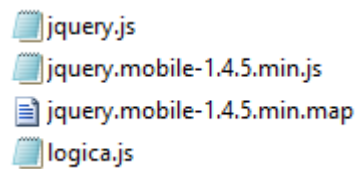
Dentro de la carpeta **css** se muestran los archivos correspondientes a los estilos del framework JQuery Mobile, nuestro theme personalizado y otros estilos personalizados. A demás consta de una carpeta para almacenar fuentes e imágenes de nuestro theme.



Como explicamos anteriormente existe una carpeta donde almacenamos las imágenes usadas dentro de nuestro diseño, las cuales no pertenecen al theme.



En la carpeta **js** se encuentran los archivos de JQuery y JQuery Mobile, que son esenciales para el uso de este framework. También está nuestro archivo de la funcionalidad para la aplicación que es **lógica.js**.





MANUAL DE USUARIO

Tabla de Contenido

| | | |
|-------|--|----|
| 1 | INTRODUCCIÓN | 1 |
| 2 | CONCEPTOS IMPORTANTES..... | 1 |
| 2.1 | Tipos de usuario..... | 1 |
| 2.2 | Acceso al sistema | 1 |
| 3 | GUÍA DE USO..... | 4 |
| 3.1 | Administración de usuarios | 4 |
| 3.1.1 | Agregar usuarios..... | 5 |
| 3.1.2 | Modificar datos de usuarios | 6 |
| 3.1.3 | Habilitar y deshabilitar usuarios | 7 |
| 3.2 | Perfil de usuario | 8 |
| 3.2.1 | Cambiar contraseña..... | 8 |
| 3.3 | Administración de información de la ciudad | 10 |
| 3.3.1 | Crear información..... | 11 |
| 3.3.2 | Modificar información | 15 |
| 3.3.3 | Eliminar información..... | 16 |
| 3.4 | Administración de categorías | 17 |
| 3.4.1 | Crear categorías | 18 |
| 3.4.2 | Modificar categorías..... | 20 |
| 3.4.3 | Eliminar categorías | 21 |
| 3.5 | Administración de atractivos..... | 22 |
| 3.5.1 | Buscar atractivos por categoría..... | 23 |
| 3.5.2 | Crear atractivo..... | 24 |
| 3.5.3 | Modificar atractivos | 29 |

| | |
|--------------------------------|----|
| 3.5.4 Eliminar atractivos..... | 31 |
| 3.6 Salir del sistema | 31 |

1 INTRODUCCIÓN

Este documento pretende explicar el funcionamiento del sistema web para la gestión de la aplicación móvil turística de Ibarra.

Esta aplicación permite al operador el manejo de la información presentada en la aplicación móvil, es decir: categorías, descripciones, direcciones, imágenes y ubicaciones de los atractivos de la ciudad de Ibarra.

2 CONCEPTOS IMPORTANTES

2.1 Tipos de usuario

El sistema web maneja dos tipos de usuario: administrador y operador. El usuario administrador gestiona el acceso al sistema creando usuarios operadores. Los usuarios operadores dispondrán de las demás funcionalidades del sistema; es decir, podrá registrar la información perteneciente a la ciudad, las categorías de atractivos, y los atractivos correspondientes.

2.2 Acceso al sistema

El ingreso a la aplicación se realizará mediante el empleo de un navegador web. Cabe destacar que esta aplicación es compatible con "Google Chrome".

Cualquier usuario deberá acceder a la aplicación utilizando un nick y contraseña correspondiente, logrando el ingreso a la página principal según el rol adquirido.

Para dirigirnos a la aplicación, dentro de nuestro navegador digitamos la siguiente URL: gestur-touribarra.rhcloud.com. Así se nos presenta el formulario de ingreso.



Formulario de Acceso

Administrador de la aplicación móvil de Turismo de la ciudad de Ibarra.

Nick de usuario

Contraseña

Ingresar

Si los datos no son correctos, el sistema notificará los resultados erróneos.



Formulario de Acceso

Administrador de la aplicación móvil de Turismo de la ciudad de Ibarra.

Contraseña incorrecta

Contraseña incorrecta

Ingresar

Si los datos son correctos, accederemos al panel correspondiente.



Página principal del usuario administrador



Página principal del usuario operador

3 GUÍA DE USO

3.1 Administración de usuarios

Esta funcionalidad del sistema pertenece solo a usuarios que se incluyen en el grupo ADMINISTRADOR. Dentro de la página principal observaremos una barra de menú, donde al ingresar en **Gestionar** se desplegará la opción **Usuarios**.



Al acceder a esta opción se mostrará una página donde podremos ingresar un nuevo usuario y visualizar los existentes. Además en el listado de usuarios existentes tendremos acceso a opciones como: **Cambio de Estado** que permite habilitar/deshabilitar usuarios, y **Modificar** que permite editar sus datos.

| Nombres | Apellidos | Nick | Estado | Tipo | Modificar | Cambio estado |
|----------|-----------|---------|--------|---------------|-----------------------------|-----------------------------|
| Usuario | Apellido | usuario | no | Operador | Seleccionar | Seleccionar |
| Luis | Cisneros | tiko | si | Administrador | Seleccionar | Seleccionar |
| Veronica | Gomez | aniri | si | Operador | Seleccionar | Seleccionar |

3.1.1 Agregar usuarios

Dentro de **Gestión de Usuarios**, en la sección superior podemos ingresar nuevos usuarios del sistema. Para esto escribimos el nombre, apellido, nick, contraseña provisional y seleccionamos el tipo de usuario.

Cuando tenemos lleno el formulario, pulsamos el botón **Ingresar** para finalizar el proceso de creación de usuario.

Si se ha ingresado algún dato erróneo, el sistema notificara mediante mensajes; caso contrario se ingresará el nuevo usuario y se mostrará un mensaje de éxito. Además aparecerá en la lista de usuarios y este se encontrará habilitado.

[? Ayuda](#) [Regresar](#)

Bienvenido Luis
 Usuario ingresado correctamente
 Usuario ingresado correctamente

Ingreso de Usuarios

Nombre
 Apellido
 Nick de usuario
 Contraseña
 Tipo de Usuario Operador

Ingresar

Usuarios Existentes...

| Nombres | Apellidos | Nick | Estado | Tipo | Modificar | Cambio estado |
|----------|-----------|---------|--------|---------------|-----------------------------|-----------------------------|
| Usuario | Apellido | usuario | no | Operador | Seleccionar | Seleccionar |
| Luis | Cisneros | tiko | si | Administrador | Seleccionar | Seleccionar |
| Veronica | Gomez | aniri | si | Operador | Seleccionar | Seleccionar |
| Juan | Orbe | jorbe | si | Operador | Seleccionar | Seleccionar |

3.1.2 Modificar datos de usuarios

Dentro de la lista de usuarios existentes se encuentra una columna **Modificar**, si deseamos cambiar los datos de un usuario existente nos ubicaremos en la fila correspondiente y pulsaremos **Seleccionar**.

Usuarios Existentes...

| Nombres | Apellidos | Nick | Estado | Tipo | Modificar | Cambio estado |
|----------|-----------|---------|--------|---------------|-----------------------------|-----------------------------|
| Usuario | Apellido | usuario | no | Operador | Seleccionar | Seleccionar |
| Luis | Cisneros | tiko | si | Administrador | Seleccionar | Seleccionar |
| Veronica | Gomez | aniri | si | Operador | Seleccionar | Seleccionar |
| Juan | Orbe | jorbe | si | Operador | Seleccionar | Seleccionar |

A continuación nos re-direccionaremos a la página de edición de usuario que contiene los campos similares a los de ingreso, con la excepción que no podremos editar el nick de usuario y que la contraseña no se muestra.

Si deseamos modificar los datos estos serán igualmente validados y al cambiar el texto de contraseña esta será modificada. Se recomienda el cambio de contraseña cuando un usuario ha olvidado la misma,

proporcionándole una clave temporal que podrá ser cambiada en el perfil de usuario.

Ingreso de Usuarios

Nombres:

Apellidos:

Nick de usuario: jorbe

Contraseña:

Tipo de Usuario:

3.1.3 Habilitar y deshabilitar usuarios

Si un usuario está habilitado podrá ingresar al sistema según su rol, caso contrario no. Para habilitar o deshabilitar un usuario nos dirigimos a la lista de usuarios existente y en la columna **Cambiar estado** pulsaremos **Seleccionar** según el usuario. En seguida se mostrará un mensaje de confirmación, mediante el cual al pulsar **Aceptar** cambiaremos el estado y al pulsar **Cancelar** no se realizarán cambios.

Nick de usuario:

Contraseña:

Tipo de Usuario:

Usuarios Existentes...

| Nombres | Apellidos | Nick | Estado | Tipo | Modificar | Cambio estado |
|----------|-----------|---------|--------|---------------|-----------------------------|-----------------------------|
| Usuario | Apellido | usuario | no | Operador | Seleccionar | Seleccionar |
| Luis | Cisneros | tiko | si | Administrador | Seleccionar | Seleccionar |
| Veronica | Gomez | aniri | si | Operador | Seleccionar | Seleccionar |
| Juan | Orbe | jorbe | si | Operador | Seleccionar | Seleccionar |

Si decidimos habilitar/deshabilitar un usuario, el sistema mostrará un mensaje de éxito y dentro de la columna **Estado** aparecerá si o no según el caso.

The screenshot shows a web application interface. At the top left, there are links for 'Ayuda' and 'Regresar'. A notification bubble in the top right corner says 'Bienvenido Luis' and 'Cambio de estado correcto'. Below this is a section titled 'Ingreso de Usuarios' with input fields for 'Nombre' (Juan), 'Apellido' (Orbe), 'Nick de usuario' (jorbe), and 'Contraseña'. A dropdown menu for 'Tipo de Usuario' is set to 'Operador', and an 'Ingresar' button is present. Below the login form is a section titled 'Usuarios Existentes...' containing a table with columns: Nombres, Apellidos, Nick, Estado, Tipo, Modificar, and Cambio estado. The table lists four users: Luis Cisneros (Administrador, Estado: si), Veronica Gomez (Operador, Estado: si), and Juan Orbe (Operador, Estado: no). Each row has 'Seleccionar' links under the 'Modificar' and 'Cambio estado' columns. Navigation buttons are visible above and below the table.

| Nombres | Apellidos | Nick | Estado | Tipo | Modificar | Cambio estado |
|----------|-----------|---------|--------|---------------|-----------------------------|-----------------------------|
| Usuario | Apellido | usuario | no | Operador | Seleccionar | Seleccionar |
| Luis | Cisneros | tiko | si | Administrador | Seleccionar | Seleccionar |
| Veronica | Gomez | aniri | si | Operador | Seleccionar | Seleccionar |
| Juan | Orbe | jorbe | no | Operador | Seleccionar | Seleccionar |

3.2 Perfil de usuario

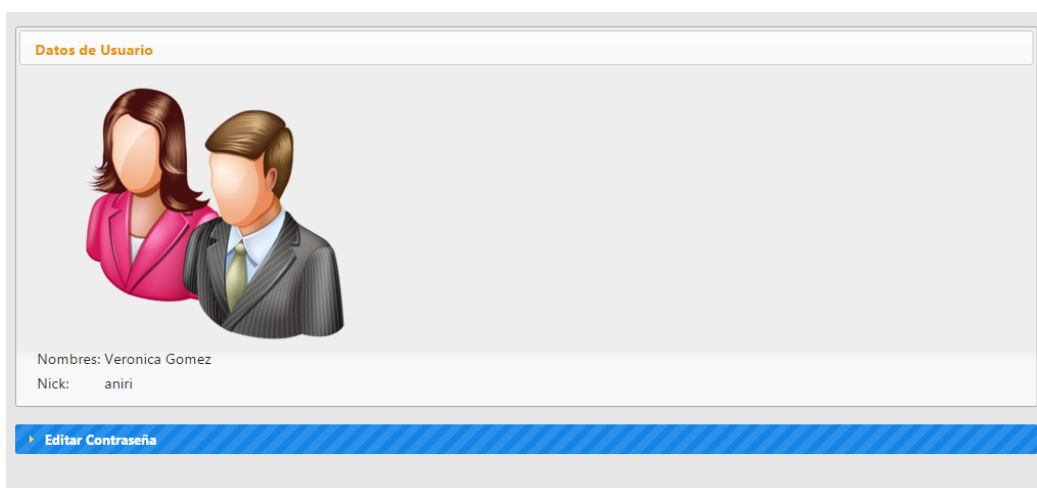
Los usuarios pertenecientes al grupo **Operador**, dentro del menú en la página principal podrán acceder a su perfil. Dentro de este encontrarán sus datos generales y tendrán la capacidad de cambiar su contraseña de acceso al sistema.

3.2.1 Cambiar contraseña

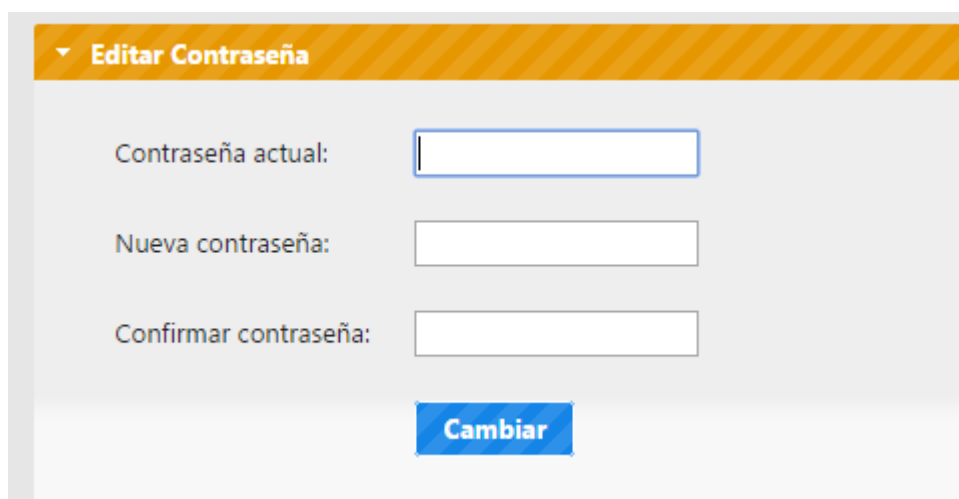
Dentro de la página principal escogeremos la opción **Perfil** del menú **Ver.**

The screenshot shows a user profile page. At the top left, there are links for 'Gestionar', 'Ver.', and 'Ayuda'. A dropdown menu is open under 'Ver.', showing the 'Perfil' option. At the top right, it says 'Bienvenido Veronica' and 'Cerrar Sesión'. The main content area features the logo 'IBARRA' in large red letters, with the tagline 'CIUDAD BLANCA A LA QUE SIEMPRE SE VUELVE' below it. A small green mountain graphic is visible in the bottom right corner.

Ingresaremos a la página de visualización del perfil, donde se encuentran nuestros datos personales y la sección para cambiar la contraseña de nuestra cuenta.



Para cambiar la contraseña desplegamos la barra **Editar Contraseña** e ingresaremos los datos de contraseña actual y nueva.

The image shows a form for changing a password. The form has a yellow header with a dropdown arrow and the text "Editar Contraseña". Below the header, there are three input fields: "Contraseña actual:", "Nueva contraseña:", and "Confirmar contraseña:". At the bottom of the form, there is a blue button with the text "Cambiar".

Finalmente pulsaremos el botón **Cambiar**, logrando un cambio de contraseña de ingreso al sistema exitoso.

3.3 Administración de información de la ciudad

Dentro del manejo de contenidos de la aplicación móvil, el usuario **Operador** puede ingresar los datos referentes a la ciudad de Ibarra. Para esto en la página principal accede al menú **Gestionar** y escoge la opción **Información Ciudad**.



Dentro de la página de información general se presenta un formulario para el ingreso de nuevos contenidos y un listado de los existentes.

Registro de Información de la Ciudad

Título: *

Descripción: *

Idioma:

Selección de imagen:

Información ingresada

| Título | Descripción | Idioma | EDITAR | ELIMINAR |
|---------|-------------|---------|--------|----------|
| ciudad | ciudad | español | | |
| city | city | ingles | | |
| clima | clima | español | | |
| weather | weather | ingles | | |

3.3.1 Crear información

En la parte superior de la página se encuentra el formulario para ingresar los datos referentes a un nuevo ítem de información de la ciudad de Ibarra.

Dentro de estos están:

- Título.- Una sola palabra característica.
- Descripción.- Texto informativo que no incluye comillas simples, ni dobles y otros caracteres especiales; en vez de usar comillas podemos introducir guiones medios (-). Además vale recalcar que dentro de la descripción si necesitamos un punto a parte usaremos el carácter arroba (@).
- Idioma.- Elección del idioma de la información sea este inglés o español.
- Imagen.- Foto característica de dicha información, tamaño ideal 1024 x 640 pixeles. Si no se elige esta se creará con la imagen por defecto.

Registro de Información de la Ciudad

Título: *

Descripción: *

Idioma:

Seleccin de imagen:

Al pulsar el botón de **Seleccionar imagen** se abrirá una ventana flotante que nos permite realizar el proceso de subida de imagen.

Selección de imagen

Seleccionar archivo No se eligió archivo Subir

el mapa

Descripción: *

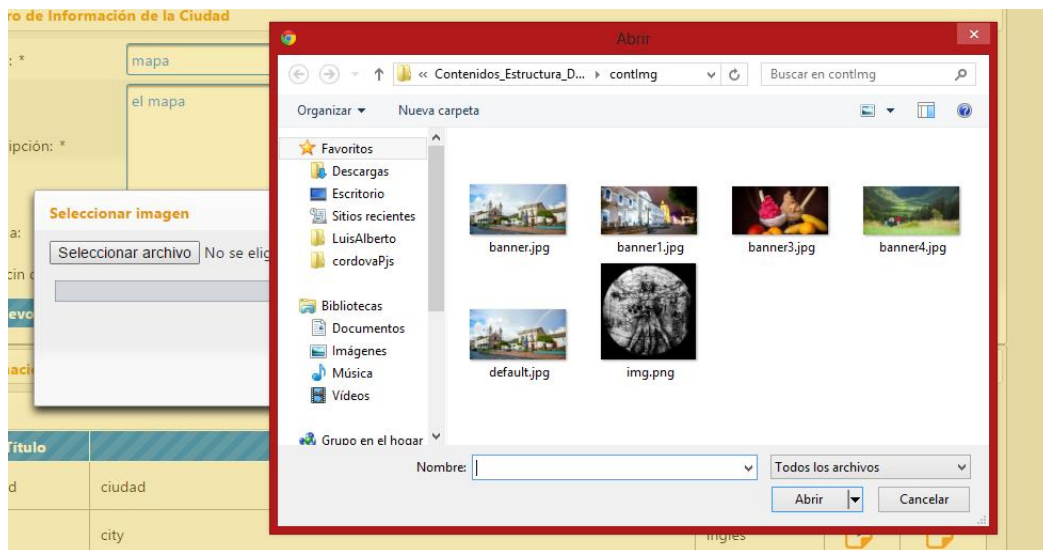
Idioma:

Seleccin de imagen:

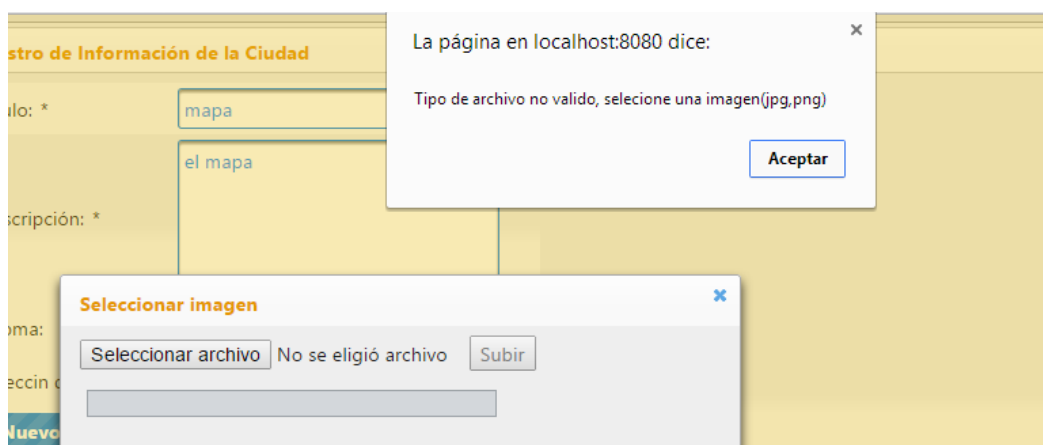
Información ingresada

| Título | Descripción | Idioma | EDITAR | ELIMINAR |
|--------|-------------|--------|---------------------------------------|---|
| | | | <input type="button" value="EDITAR"/> | <input type="button" value="ELIMINAR"/> |

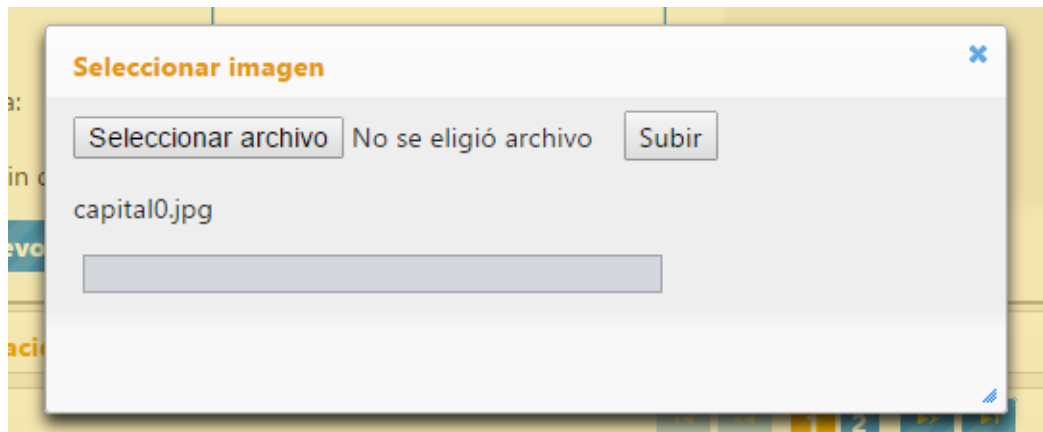
Para subir la imagen daremos clic en el botón **Seleccionar archivo**, con esto se mostrará la ventana de búsqueda de imagen, donde podremos navegar por el computador y elegir la imagen característica recordando su tamaño ideal de 1024 x 640 pixeles.



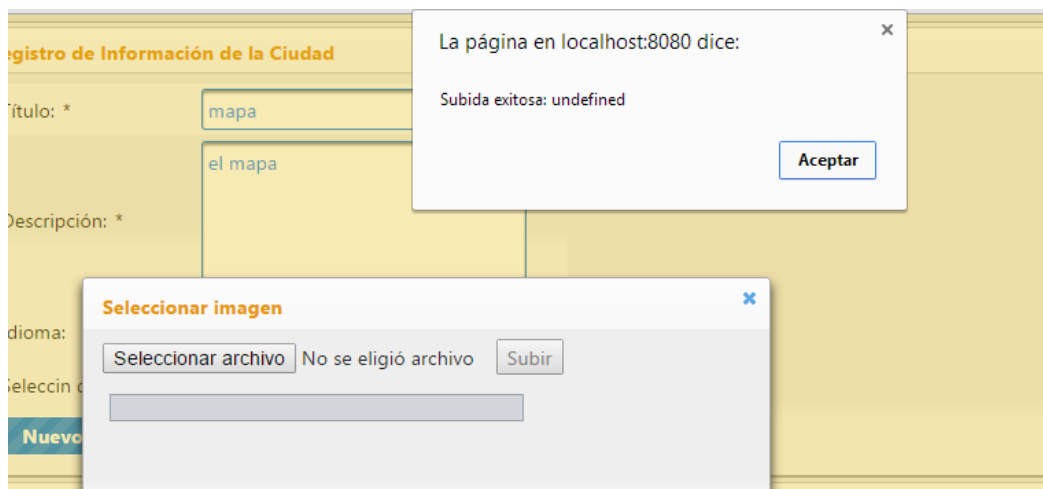
Si durante el proceso de selección de imagen, escogemos un archivo de distinta extensión u formato, se mostrará un mensaje de ayuda para informarnos de nuestra elección



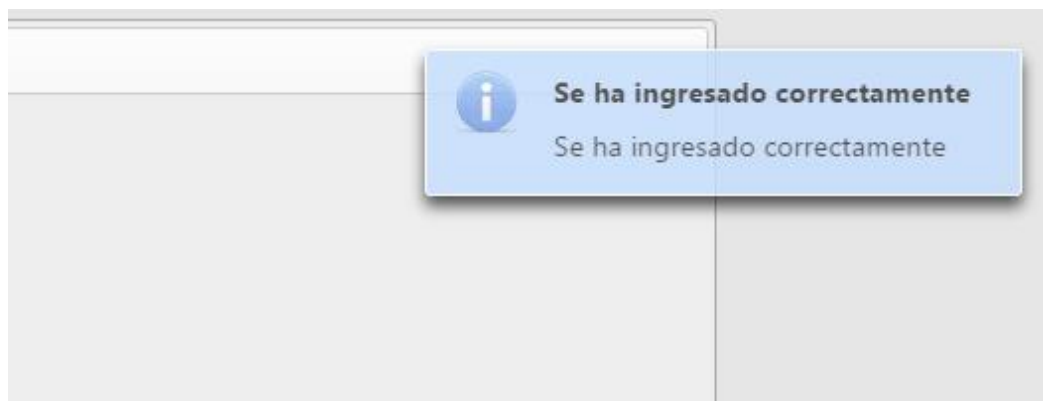
Cuando seleccionamos correctamente nuestra imagen, el nombre de esta se mostrará en la parte inferior.





Para iniciar el proceso de subida damos clic en el botón **Subir**, cabe recalcar que existe la barra de progreso que nos indicará como avanza el proceso de subida de la imagen. Una vez finalizado este, se muestra una alerta de éxito del proceso.



Finalmente guardaremos los cambios pulsando el botón **Nuevo**, a continuación aparecerá un mensaje de éxito en la creación de la información.




Además se agregará al listado de **Información ingresada**.

| Información ingresada | | | | |
|-----------------------|-------------|---------|---|---|
| Título | Descripción | Idioma | EDITAR | ELIMINAR |
| mapa | el mapa | español |  |  |

3.3.2 Modificar información

Dentro de la lista de información ingresada se encuentra una columna **Editar**, si deseamos cambiar los datos existente nos ubicaremos en la fila


correspondiente y pulsaremos la imagen  .

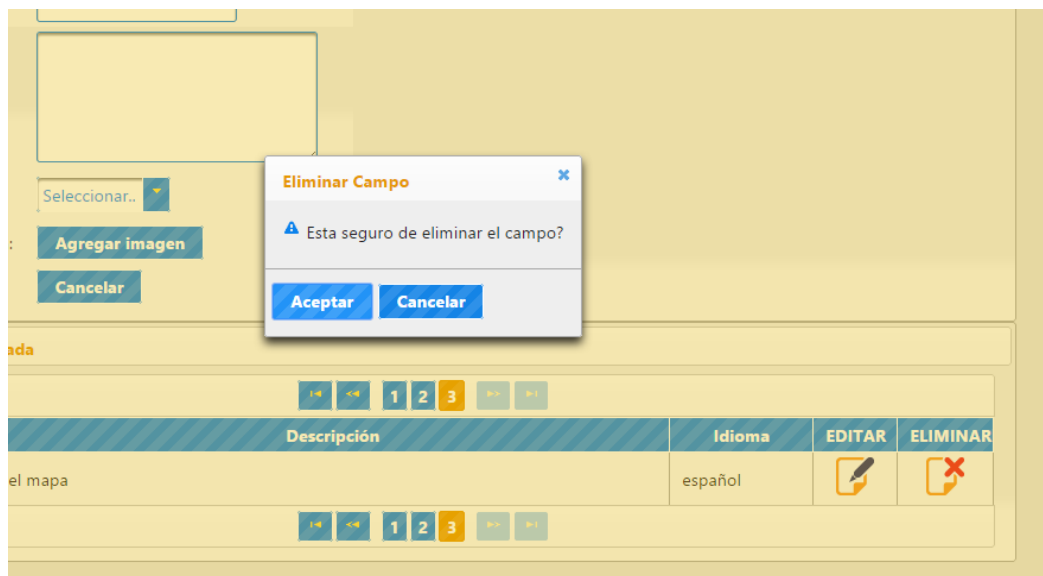
Enseguida nos re-direccionaremos a la página de modificación donde tendremos las mismas condiciones que en el formulario de ingreso. El proceso de modificación es similar al de ingreso de nueva información.

Existe un ligero cambio dentro del formulario, este se refiere al botón **Ver imagen actual**; este botón nos permite visualizar la imagen que pertenece a dicha información actualmente mediante una ventana flotante.

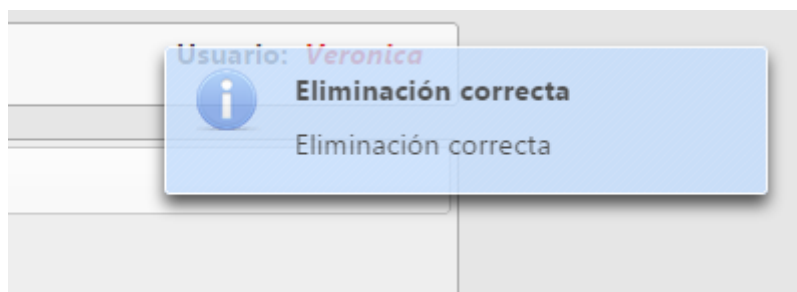


3.3.3 Eliminar información

Dentro de la lista de información ingresada se encuentra una columna **Eliminar**, si deseamos borrar los datos de aquella información existente nos ubicaremos en la fila correspondiente y pulsaremos la imagen .



Al presionar el botón nos aparecerá una ventana de aceptación de eliminación del campo; si pulsamos **Cancelar** no se realizará ninguna acción, por el contrario al pulsar **Aceptar** dicha información será eliminada y se mostrará un mensaje de éxito.



3.4 Administración de categorías

Dentro del manejo de contenidos de la aplicación móvil, el usuario **Operador** puede ingresar categorías referentes a los atractivos de Ibarra. Para esto en la página principal accede al menú **Gestionar** y escoge la opción **Categorías**.



Dentro de la página de categorías se presenta un formulario para el ingreso de nuevas categorías y un listado de las existentes.

Registro de Información de Categorías

Título *

Descripción *

Idioma

Categorías ingresadas

| Título | Descripción | Idioma | EDITAR | ELIMINAR |
|-----------|-----------------------|---------|--------|----------|
| parques | los parques de ibarra | español | | |
| parks | parks of ibarra | ingles | | |
| iglesias | las iglesias | español | | |
| naturales | los sitios naturales | español | | |

3.4.1 Crear categorías

En la parte superior de la página se encuentra el formulario para ingresar los datos referentes a una nueva categoría. Dentro de estos están:

- Título.- Una sola palabra característica.
- Descripción.- Texto informativo que no incluye comillas simples, ni dobles y otros caracteres especiales; en vez de usar comillas podemos introducir guiones medios (-).

- Idioma.- Elección del idioma de la información sea este inglés o español.

Registro de Información de Categorías

Título * restaurants ibarra

Descripción * Restaurants of Ibarra

Idioma Seleccionar..

Seleccionar..
ingles
español

Nuevo

Categorías ingresadas

Si deseamos guardar la categoría pulsaremos el botón **Nuevo**; si ingresamos datos incorrectos se notificará al usuario sobre los errores y no se ingresará la categoría, caso contrario se ingresa normalmente la categoría mostrando un mensaje de éxito.

Registro de Información de Categorías

Título * restaurants ibarra

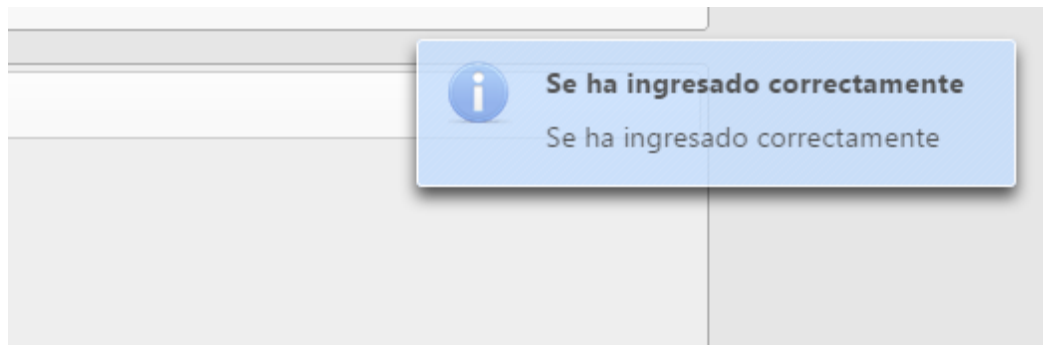
Descripción * Restaurants of Ibarra

Idioma ingles

Nuevo Cancelar

El título debe ser una palabra en minúsculas
El título debe ser una palabra en minúsculas


Además se añadirá a la lista de **categorías ingresadas**.



Categorías ingresadas

| Título | Descripción | Idioma | EDITAR | ELIMINAR |
|-------------|-----------------------|--------|---|---|
| sport | sport places | ingles |  |  |
| services | other services | ingles |  |  |
| restaurants | Restaurants of Ibarra | ingles |  |  |

3.4.2 Modificar categorías

Dentro de la lista de categorías ingresadas se encuentra una columna **Editar**, si deseamos cambiar los datos existente nos ubicaremos en la fila correspondiente y pulsaremos la imagen . Enseguida nos redireccionaremos a la página de modificación donde tendremos las mismas condiciones que en el formulario de ingreso. El proceso de modificación es similar al de ingreso de nueva información.

Registro de Información de Categorías

Título *

Descripción *

Idioma

3.4.3 Eliminar categorías

Dentro de la lista de categorías ingresadas se encuentra una columna **Eliminar**, si deseamos borrar los datos de aquella categoría existente nos

ubicaremos en la fila correspondiente y pulsaremos la imagen  .




The screenshot shows a web application interface for managing categories. It consists of two main sections: a form for adding new categories and a table of existing categories.

Registro de Información de Categorías

Form fields:

- Título *
- Descripción *
- Idioma (dropdown menu: Seleccionar..)
- Buttons: Nuevo, Cancelar

Categorías ingresadas

| Título | Descripción | Idioma | EDITAR | ELIMINAR |
|-------------|-----------------------|--------|---|---|
| sport | sport places | ingles |  |  |
| services | other services | ingles |  |  |
| restaurants | Restaurants of Ibarra | ingles |  |  |

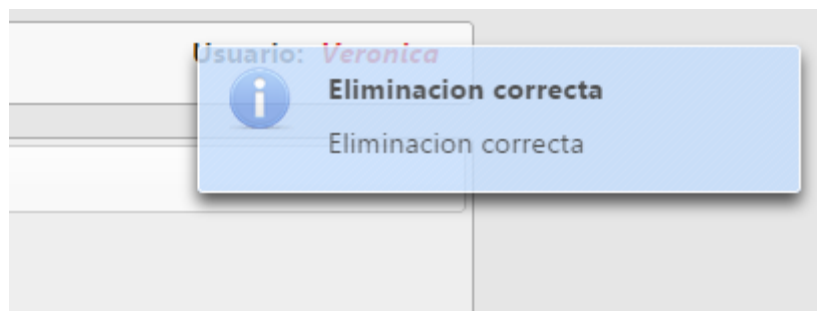
Modal dialog box: **Eliminar Campo**

Esta seguro de eliminar el campo?

Al eliminar una categoría también se eliminarán los sitios pertenecientes a la misma

Buttons: Aceptar, Cancelar

Al presionar el botón nos aparecerá una ventana de aceptación de eliminación de categoría que nos advierte sobre lo que ocurrirá con los atractivos pertenecientes a la misma; si pulsamos **Cancelar** no se realizará ninguna acción, por el contrario al pulsar **Aceptar** la categoría como los atractivos pertenecientes a ella serán eliminados y se mostrará un mensaje de éxito.



3.5 Administración de atractivos

Dentro del manejo de contenidos de la aplicación móvil, el usuario **Operador** puede ingresar atractivos referentes categorías anteriormente creadas. Para esto en la página principal accede al menú **Gestionar** y escoge la opción **Atractivos**.



A continuación se presenta una la página principal del manejo de atractivos, donde se presentan los atractivos existentes, además consta con una búsqueda de atractivos por categoría, y la creación de uno nuevo.

Sitios pertenecientes a Ibarra

Búsqueda por categoría Ingresar nuevo Sitio

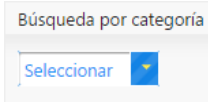
Seleccionar

Listado de Sitios

| Categoría | Nombre | Descripción | EDITAR | ELIMINAR |
|------------|-------------------------|-------------------------|--------|----------|
| parques | La Merced | La merced | | |
| parques | Pedro Moncayo | Pedro Moncayo | | |
| parks | La Merced park | La Merced park | | |
| parks | Pedro Moncayo park | Pedro Moncayo park | | |
| iglesias | La Catedral | La Catedral | | |
| iglesias | Basilica de La Dolorosa | Basilica de La Dolorosa | | |
| naturales | Laguna de Yahuarcocha | Laguna de Yahuarcocha | | |
| naturales | Loma de Guayabillas | Loma de Guayabillas | | |
| historicos | Inca Huasi | Inca Huasi | | |
| historicos | La Piedra Chapetona | La Piedra Chapetona | | |

3.5.1 Buscar atractivos por categoría

Dentro de la página principal de atractivos, podemos conocer qué sitios pertenecen a una categoría específica; para ellos escogeremos la misma en

el listado de categorías  y pulsaremos el botón de búsqueda



Cabe recalcar que si presionamos el botón antes mencionado si seleccionar una categoría nos aparecerá un mensaje de alerta solicitándonos este proceso.

Atención!
Seleccione Categoría para continuar

Sitios pertenecientes a Ibarra

Búsqueda por categoría Ingresar nuevo Sitio

Seleccionar  



Listado de Sitios

| Categoría | Nombre | Descripción | EDITAR | ELIMINAR |
|-----------|-----------|-------------|---|---|
| parques | La Merced | La merced |  |  |





Una vez seleccionada nuestra categoría y pulsado el botón de búsqueda, se mostrará en el **Listado de sitios** los atractivos pertenecientes a esta.

Sitios pertenecientes a Ibarra

Búsqueda por categoría Ingresar nuevo Sitio

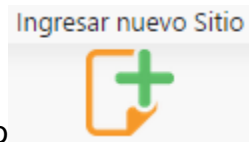
parques  

Listado de Sitios

| Categoría | Nombre | Descripción | EDITAR | ELIMINAR |
|-----------|---------------|---------------|---|---|
| parques | La Merced | La merced |  |  |
| parques | Pedro Moncayo | Pedro Moncayo |  |  |

3.5.2 Crear atractivo


Dentro de la página principal de atractivos, una vez ya seleccionada nuestra categoría procedemos a crear un nuevo sitio perteneciente a la misma; para



ello pulsamos el botón nuevo

Esto nos re-direccionará a la página de ingreso de atractivos, donde se encuentra el formulario para ingresar los datos referentes a este. Dentro de ellos están:

- Nombre.- Nombre del atractivo, este puede contener caracteres en mayúsculas, espacios y tildes.
- Descripción.- Texto informativo que no incluye comillas simples, ni dobles y otros caracteres especiales; en vez de usar comillas podemos introducir guiones medios (-). Además vale recalcar que dentro de la descripción si necesitamos un punto a parte usaremos el carácter arroba (@).
- Dirección.- Texto que informa la ubicación del atractivo, este puede contener mayúsculas, tildes, números y guiones medios (-).
- Idioma.- Elección del idioma de la información sea este inglés o español.
- Imagen.- Foto característica de dicha información, tamaño ideal 1024 x 640 pixeles. Si no se elige esta se creará con la imagen por defecto.

- Mapa.- Posición del atractivo dentro del mapa de Ibarra, esta se realiza mediante el arrastre del puntero o marcador  y la navegación en el mapa de Google.

Ingreso de Sitio

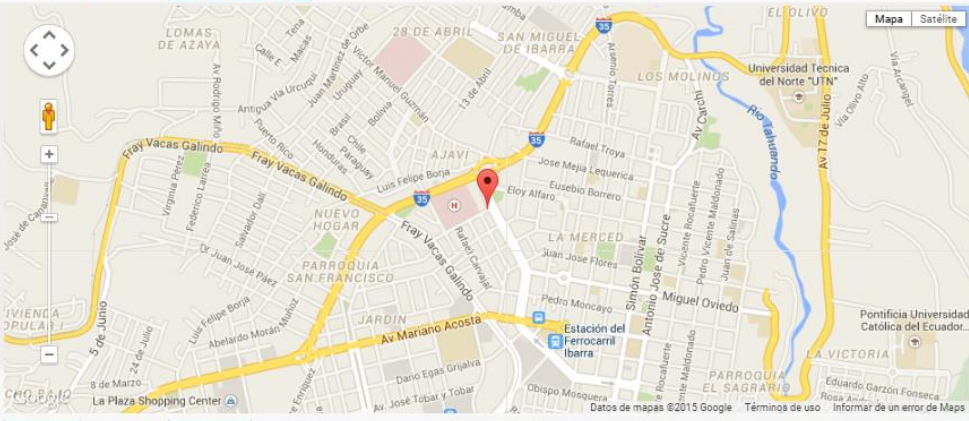
Categoría: parques

Nombre:

Descripción:

Dirección:

Seleccionar Imagen:



Si dejamos campos vacíos o no llenamos correctamente alguno, el sistema nos notificará sobre estos eventos mediante mensajes.

Ingreso de Sitio


Categoría: parques

Nombre:

Descripción:

Dirección:

Seleccionar Imagen:



✘ formsitio:descripcion: Error de validación: se necesita un valor.

✘ formsitio:direccion: Error de validación: se necesita un valor.

Como explicamos anteriormente un ejemplo de escritura de los datos característicos de un atractivo sería el siguiente.

Ingreso de Sitio

Categoría: parques

Nombre: Parque del Águila

Descripción: El Parque del águila es genial.@Debes visitar este sitio.

Dirección: Calle Sucre esquina.

Seleccionar Imagen: **Agregar imagen**

Nuevo **Limpiar**

Al pulsar el botón de **Seleccionar imagen** se abrirá una ventana flotante que nos permite realizar el proceso de subida de imagen.

Ingreso de Sitio

Categoría: parques

Nombre: Parque del Águila

Descripción: El Parque del águila es genial.@Debes visitar este sitio.

Dirección: Calle Sucre esquina.

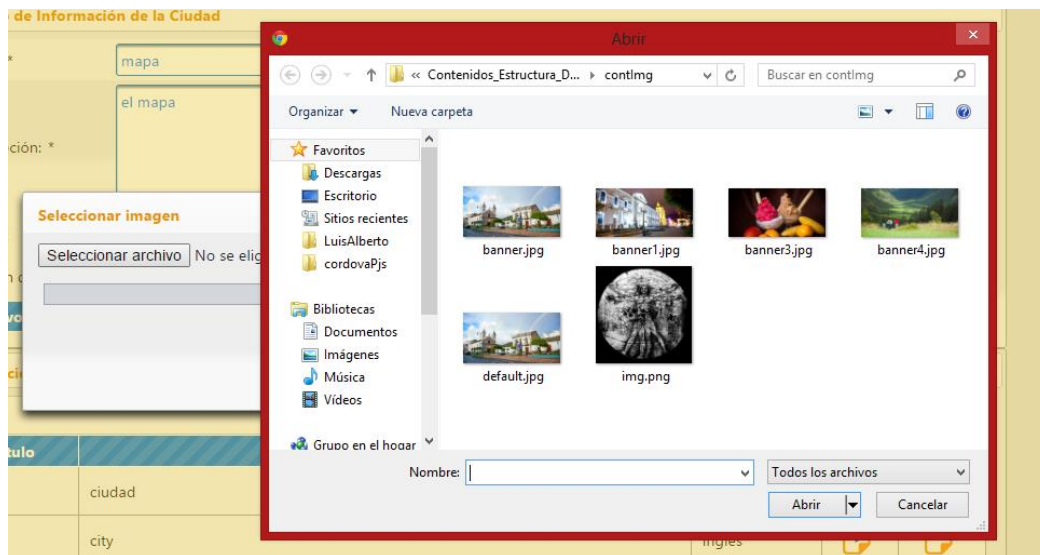
Seleccionar Imagen: **Agregar imagen**

Seleccionar imagen

Seleccionar archivo No se eli...ó archivo **Subir**

Para subir la imagen daremos clic en el botón **Seleccionar archivo**, con esto se mostrará la ventana de búsqueda de imagen, donde podremos

navegar por el computador y elegir la imagen característica recordando su tamaño ideal de 1024 x 640 pixeles.

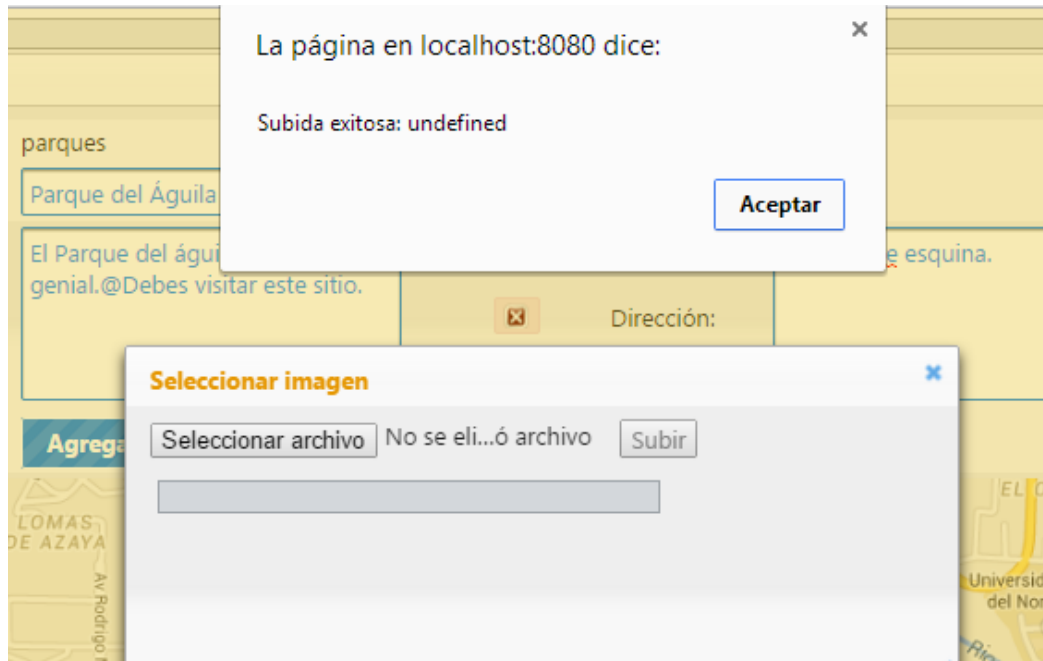


Si durante el proceso de selección de imagen, escogemos un archivo de distinta extensión u formato, se mostrará un mensaje de ayuda para informarnos de nuestra elección. Cuando seleccionamos correctamente nuestra imagen, el nombre de esta se mostrará en la parte inferior.

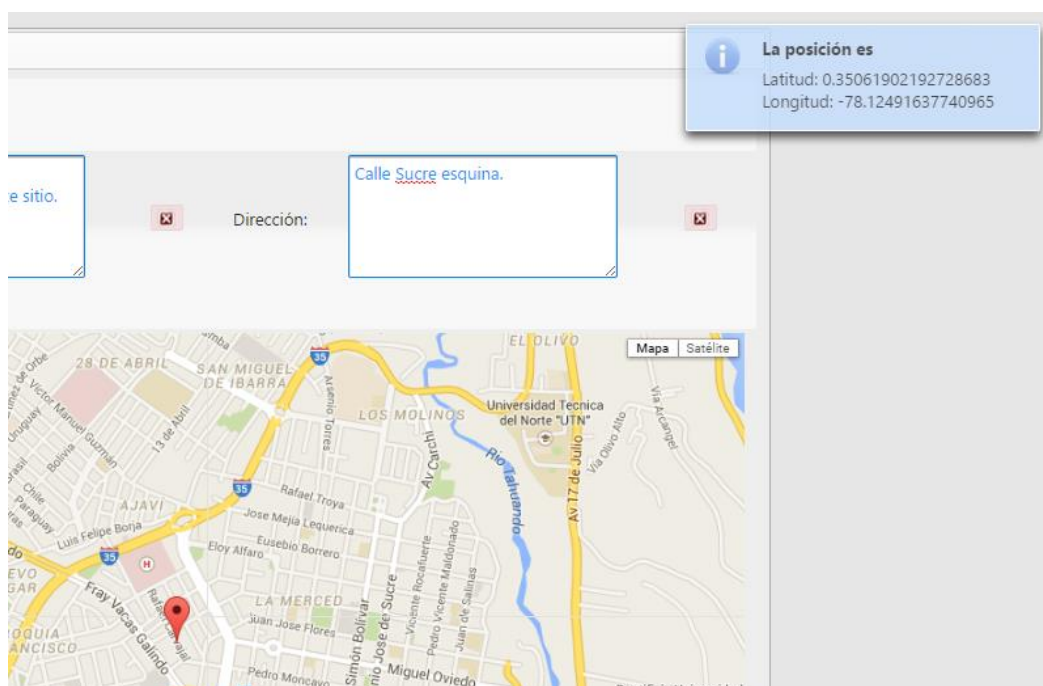


Para iniciar el proceso de subida damos clic en el botón **Subir**, cabe recalcar que existe la barra de progreso que nos indicará como avanza el proceso de

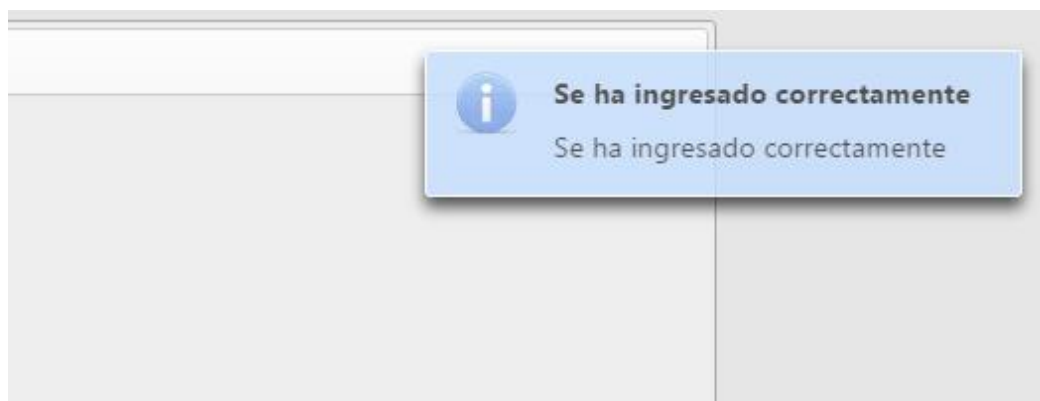
subida de la imagen. Una vez finalizado este, se muestra una alerta de éxito del proceso.



Al arrastrar el marcador dentro del mapa nos mostrará un mensaje de la latitud y longitud del atractivo, verificando que el proceso de ubicación dentro del mapa sea correcto





Finalmente guardaremos los cambios pulsando el botón **Nuevo**, a continuación aparecerá un mensaje de éxito en la creación de la información.









Además se regresamos a la página principal de atractivos, el nuevo sitio se agregará al listado de **Atractivos**.

Sitios pertenecientes a Ibarra


Búsqueda por categoría Ingresar nuevo Sitio

parques  

Listado de Sitios

| Categoría | Nombre | Descripción | EDITAR | ELIMINAR |
|-----------|-------------------|---|---|---|
| parques | La Merced | La merced |  |  |
| parques | Pedro Moncayo | Pedro Moncayo |  |  |
| parques | Parque del Águila | El Parque del águila es genial.@Debes visitar este sitio. |  |  |

3.5.3 Modificar atractivos

Dentro de la lista de atractivos se encuentra una columna **Editar**, si deseamos cambiar los datos existente nos ubicaremos en la fila correspondiente y pulsaremos la imagen . Enseguida nos redireccionaremos a la página de modificación donde tendremos las mismas

condiciones que en el formulario de ingreso. El proceso de modificación es similar al de ingreso de nueva información.

Ingreso de Sitio


Categoría: parques

Nombre: Parque del Águila

Descripción: El Parque del águila es genial.@Debes visitar este sitio.

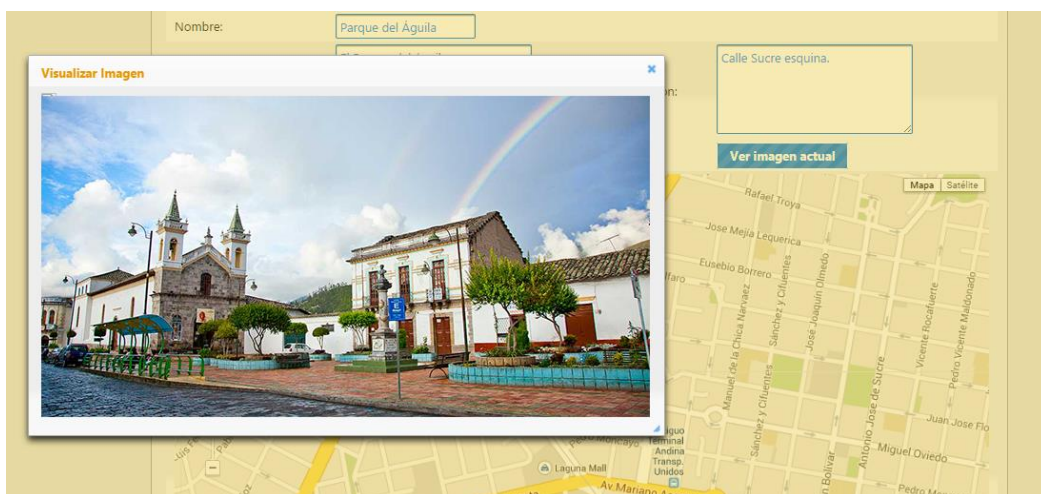
Dirección: Calle Sucre esquina.

Seleccionar Imagen: **Agregar imagen** **Ver imagen actual**




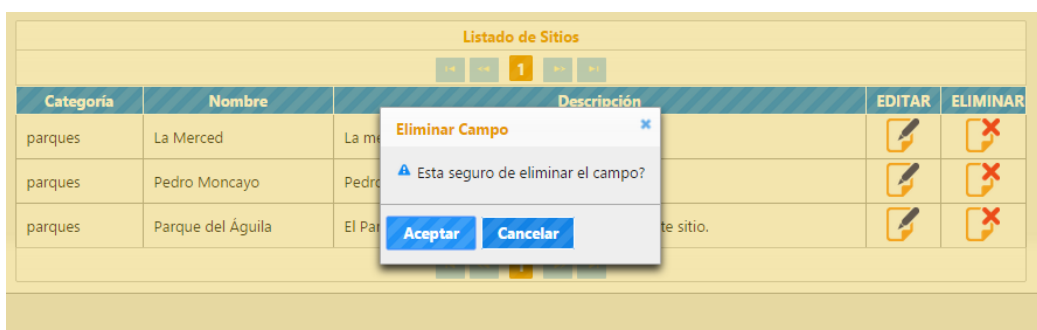
Modificar **Cancelar**

Existe un ligero cambio dentro del formulario, este se refiere al botón **Ver imagen actual**; este botón nos permite visualizar la imagen que pertenece a dicha información actualmente mediante una ventana flotante.

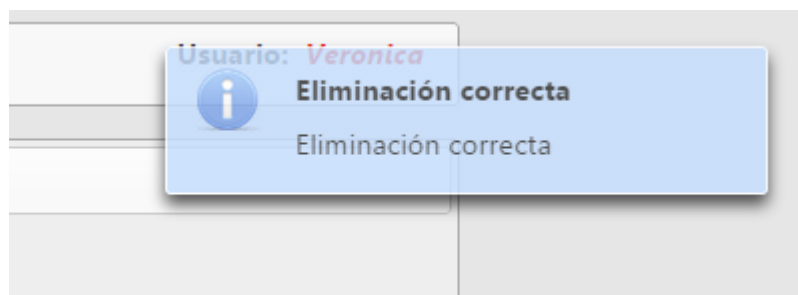


3.5.4 Eliminar atractivos

Dentro de la lista de sitios se encuentra una columna **Eliminar**, si deseamos borrar los datos de aquella información existente nos ubicaremos en la fila correspondiente y pulsaremos la imagen .



Al presionar el botón nos aparecerá una ventana de aceptación de eliminación del campo; si pulsamos **Cancelar** no se realizará ninguna acción, por el contrario al pulsar **Aceptar** dicha información será eliminada y se mostrará un mensaje de éxito.



3.6 Salir del sistema

Dentro de nuestra página principal de usuario, ya sea **Operador** o **Administrador**, se encuentra en la parte superior derecha el botón de **Cerrar Sesión**; con el cual como su nombre lo dice saldremos satisfactoriamente del sistema regresando a la página de inicio.

