



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN
GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID**

Raúl Antonio Girón Álvarez

Asesorado por el Ing. Jorge Luis Arroyo Pérez

Guatemala, junio de 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN
GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

RAÚL ANTONIO GIRÓN ÁLVAREZ
ASESORADO POR EL ING. JORGE LUIS ARROYO PÉREZ

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, JUNIO DE 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Raúl Eduardo Ticún Córdova
VOCAL V	Br. Henry Fernando Duarte García
SECRETARIA	Inga. Lesbia Magalí Herrera López

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Alfredo Enrique Beber Aceituno (a. i.)
EXAMINADOR	Ing. Cesar Augusto Fernández Cáceres
EXAMINADOR	Ing. Pedro Pablo Hernández Ramírez
EXAMINADOR	Ing. Edgar Josué González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN
GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 15 de noviembre de 2015.



Raúl Antonio Girón Álvarez

Guatemala 30 de Marzo de 2016

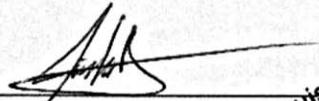
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Le saludo cordialmente, esperando que tenga éxito en sus labores. El motivo de la presente es para informar que he finalizado la asesoría del trabajo de graduación: "IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID" del estudiante de Ingeniería en Ciencias y Sistemas Raúl Antonio Girón Álvarez, con carnet 200914937, aprobando el contenido del mismo.

Sin otro particular, me es grato suscribirme.

F:



Ing. Jorge Arroyo
Industria la Popular, S.A.
jlarroyop@gmail.com
Celular: 45270422.

Jorge Luis Arroyo Pérez
Ingeniero en Ciencias y Sistemas
Colegiado 14320



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 30 de Marzo de 2016

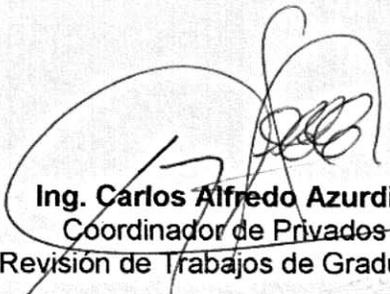
Ingeniero
Marlon Antonio Pérez Türk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante RAÚL ANTONIO GIRÓN ÁLVAREZ con camé 200914937, titulado: "IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID", y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

I
N
G
E
N
I
E
R
Í
A

E
N

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

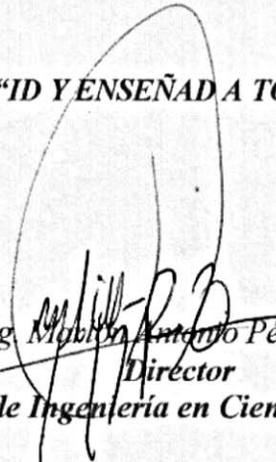
UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN
CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID”**, realizado por el estudiante RAÚL ANTONIO GIRÓN ÁLVAREZ, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑADA TODOS”

Ing.  Pérez Türk
Director

Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 08 de junio de 2016

Universidad de San Carlos
de Guatemala

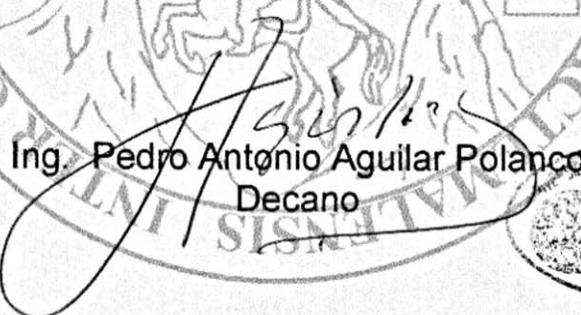


Facultad de Ingeniería
Decanato

Ref.DTG.D.290.2016

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **IMPLEMENTACIÓN DE UN SISTEMA COMPLETO DE HOSPEDAJES PARA HOTELES EN GUATEMALA, ACCESIBLE DESDE DISPOSITIVOS CON PLATAFORMA ANDROID**, presentado por el estudiante universitario: **Raúl Antonio Girón Álvarez**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.


Ing. Pedro Antonio Aguilar Polanco
Decano



Guatemala, junio de 2016

/cc

ACTO QUE DEDICO A:

Dios

Por llevarme de su mano y no dejarme caer ni vencer por ningún obstáculo que se me presentó en la carrera. Por darme sabiduría e inteligencia.

Mis padres

Carmen Anabella Álvarez de Girón, que ha estado presente en todas las etapas de mi vida. Especialmente en toda mi carrera universitaria, desvelándose conmigo, haciéndome mi comida, apoyándome incondicionalmente en todo lo que es de bien para mí. Mi padre Luis Alfonso Girón, por sus invaluable consejos que siempre buscan el bien para mí, por la seguridad que me da al respaldarme en todos los actos que realizo.

Mi prometida

Marilyn Lucila Mejía Tal, por su apoyo a lo largo de mi carrera, por buscar que siempre sea una mejor persona y brindar su amor sin importar la circunstancia.

Mi hermana

Ana Luisa Girón Álvarez, por ser para mí un ejemplo a seguir, y demostrarme que la carrera universitaria es importante. Por su apoyo intelectual y consejos durante toda mi carrera.

Mi hermano

Luis Francisco Girón Álvarez, por su respaldo y apoyo durante toda mi carrera universitaria.

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por ser una maravillosa casa de estudios, por ser quien me brindó basto conocimiento.

Facultad de Ingeniería

Por ser un ente en búsqueda de la excelencia que me formó como profesional.

**Mis amigos de la
Facultad**

Oscar Vásquez, Walter Morales, Marilyn Laínez, quienes formaron parte de mi vida en esta carrera universitaria.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VII
LISTA DE SÍMBOLOS	IX
GLOSARIO	XI
RESUMEN.....	XIII
OBJETIVOS.....	XV
INTRODUCCIÓN.....	XVII
1. MARCO TEÓRICO.....	1
1.1. Aplicación móvil.....	1
1.1.1. Evolución	1
1.1.2. Beneficios de las aplicaciones para dispositivos móviles	2
1.1.3. Entendiendo a los usuarios de aplicaciones móviles	3
1.1.3.1. Proximidad.....	3
1.1.3.2. Cierre	4
1.1.3.3. Continuidad.....	5
1.1.3.4. Figura y fondo.....	6
1.1.3.5. Similitud	6
1.1.3.6. Usabilidad	7
1.1.3.7. Accesibilidad.....	8
1.1.3.8. Personalización de los criterios de búsqueda.....	8
1.1.4. PhoneGap Framework.....	9
1.1.4.1. App utilizando tecnología web	10

	1.1.4.2.	¿Por qué utilizar PhoneGap?	11
	1.1.4.3.	HTML.....	12
	1.1.4.4.	JavaScript.....	13
	1.1.4.5.	CSS.....	15
	1.1.4.6.	JQuery.....	16
1.2.		Métodos de comunicación por internet.....	17
	1.2.1.	Web service.....	17
	1.2.2.	RESTFul Web Service.....	19
	1.2.3.	JAVA RESTful Web Service.....	20
	1.2.4.	Servidor de aplicaciones	22
	1.2.5.	Glassfish.....	23
1.3.		Dadabik.....	23
	1.3.1.	¿Qué es DaDaBIK?.....	24
	1.3.2.	¿Porqué utilizar DaDaBIK para implementación de una página de administración?.....	24
	1.3.3.	Configuración básica de DaDaBIK.....	25
	1.3.4.	Listado de precios (Pricing).....	27
1.4.		Metodología de desarrollo (proceso de software)	28
	1.4.1.	SCRUM.....	29
		1.4.1.1. Actividades de SCRUM.....	29
	1.4.2.	Beneficios y costos de procesos ágiles.....	31
2.		DESCRIPCIÓN DEL PROBLEMA Y PLANTEAMIENTO DE LA SOLUCIÓN	33
	2.1.	Descripción del problema.....	33
	2.2.	Antecedentes de la solución	34
		2.2.1. Despegar.com.....	34
		2.2.2. Sistemas A la Medida para Hoteles (reserva e información).....	35

2.3.	Planteamiento de la solución.....	36
2.3.1.	Alcances.....	36
2.3.2.	Alcances y límites de la aplicación de administración.....	38
2.3.3.	Descripción de la aplicación de administración	39
2.3.3.1.	Pantalla de Log In.....	39
2.3.3.2.	Pantalla habitaciones.....	40
2.3.3.2.1.	Funciones principales de habitaciones	40
2.3.4.	Alcances y límites de la aplicación para dispositivo Android.....	41
2.3.5.	Descripción de la aplicación Android.....	41
2.3.5.1.	Pantalla de inicio.....	42
2.3.5.2.	Action Bar de la aplicación móvil	43
2.3.5.3.	Pantalla ingreso dirección.....	44
2.3.5.4.	Pantalla de filtros	45
2.3.5.5.	Pantalla listado de establecimientos encontrados.....	46
2.3.5.6.	Pantalla datos de establecimiento	48
2.3.5.7.	Pantalla listado de habitaciones	50
2.3.5.8.	Pantalla de favoritos	52
2.3.5.9.	Pantalla de promociones	53
3.	DISEÑO DE LA SOLUCIÓN.....	55
3.1.	Base de datos.....	55
3.1.1.	Diagrama Entidad Relación	55
3.1.1.1.	Objetos principales de la base de datos.....	55

3.1.1.2.	Diagrama completo Entidades-Relaciones.....	57
3.2.	Arquitectura del sistema.....	58
3.2.1.	Arquitectura física.....	59
3.2.2.	Arquitectura lógica del sistema.....	60
3.2.2.1.	Capas del sistema.....	60
3.2.2.2.	Diagrama de comunicación lógica de componentes.....	62
3.3.	Diseño de comunicación entre componentes.....	62
3.3.1.	Descripción de RestFul Services.....	63
3.3.2.	Listado y significado de servicios a utilizar	63
3.4.	Usabilidad	66
3.4.1.	Mensajes	66
3.4.2.	Características de los dispositivos móviles	66
3.5.	Diseño de la aplicación de administración utilizando DaDaBIK	67
3.5.1.	Log In	67
3.5.2.	Visualización de habitaciones	68
3.5.3.	Cambiar estado a habitaciones	70
4.	DOCUMENTACIÓN DEL SISTEMA	71
4.1.	Manual de usuario aplicación administración.....	71
4.1.1.	Proceso de cambiar estado a habitación	71
4.2.	Manual del usuario aplicación android	74
4.2.1.	Proceso de búsqueda de hotel.....	74
4.2.2.	Ver favoritos	80
4.2.3.	Crear favorito.....	82
4.2.4.	Uso del Action Bar.....	83

CONCLUSIONES 85
RECOMENDACIONES 87
BIBLIOGRAFÍA..... 89

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Ejemplo de proximidad.....	4
2.	Ejemplo de cierre	5
3.	Ejemplo de continuidad.....	5
4.	Ejemplo de figura y fondo.....	6
5.	Ejemplo de similitud	7
6.	Clase Persona–Recurso a Acceder como REST	21
7.	Clase PersonaResource–Clase para Acceder como REST	22
8.	Parámetros básicos de configuración	26
9.	Parámetros de configuración	27
10.	Listado de precios y características	28
11.	Actividades Scrum.....	30
12.	Despegar.com.....	35
13.	Diagrama de la solución completa	37
14.	Prototipo pantalla de inicio	43
15.	Prototipo del Action Bar.....	44
16.	Pantalla de ubicación	45
17.	Pantalla de filtros.....	46
18.	Pantalla listado establecimientos encontrados.....	47
19.	Pantalla datos del establecimiento	49
20.	Mostrar tarifas en pantalla de datos	50
21.	Pantalla listado de habitaciones.....	51
22.	Galería de fotos por habitación	52
23.	Pantalla de favoritos.....	53

24.	Pantalla de promociones	54
25.	Diagrama entidad relación	58
26.	Arquitectura física del sistema	59
27.	Diagrama de capas del sistema.....	61
28.	Arquitectura lógica del sistema	62
29.	Pantalla de Log In Administración Hotel	68
30.	Pantalla inicial de sistema administración.....	69
31.	Pantalla con listado de habitaciones.....	69
32.	Pantalla cambio de estados habitaciones.....	70
33.	Paso 1. Cambiar estado habitación	71
34.	Paso 2. Cambiar de estado habitación	72
35.	Paso 3. Cambiar de estado habitación	72
36.	Paso 4. Cambio de estado habitación	73
37.	Selección categoría de establecimiento.....	74
38.	Ingreso de ubicación deseada	75
39.	Selección de filtros.....	76
40.	Listado de establecimientos aplicación.....	77
41.	Pantalla datos del establecimiento en aplicación.....	78
42.	Pantalla con listado de habitaciones.....	79
43.	Habitación galería	80
44.	Pantalla cualquiera con botón favoritos	81
45.	Visualización de favoritos	82
46.	Pantalla crear favorito	83
47.	Action bar de App	84

LISTA DE SÍMBOLOS

Símbolo	Significado
API	Application Programming Interface
DaDaBIK	Herramienta auxiliar en desarrollo de software
<i>Host</i>	Hospedaje de servicio o archivos
REST	Representational State Transfer
Service	Servicio
Ws	<i>Web service</i> (servicio web)

GLOSARIO

Android	Es un sistema operativo basado en el <i>kernel</i> de Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, y también para relojes inteligentes, televisores y automóviles.
Framework	Marco de Trabajo. Es un conjunto estandarizado de conceptos, prácticas y criterios. Referente a informática como conjunto de herramientas previamente desarrolladas que pueden ser utilizadas en proyectos futuros.
Gamification	Es el uso de elementos de juegos y técnicas de diseño de juegos en contextos que no son juegos.
Internet	Es un conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, lo cual garantiza que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial.
Servidor	Es una aplicación en ejecución, es decir, un software capaz de atender las peticiones de un cliente y devolverle una respuesta en concordancia. Se pueden ejecutar en cualquier tipo de computadoras.

Servidor físico	Son computadoras dedicadas en las cuales se ejecutan servidores, es decir, software servidor. Un mismo servidor puede proveer múltiples servicios y tener varios servidores en funcionamiento.
Sistema operativo	Es un conjunto de programas de un sistema informático que gestiona los recursos de hardware y provee servicios a los programas de aplicación.
Sockets	Es un concepto abstracto por el cual dos programas situados en computadoras distintas pueden intercambiar flujo de datos, generalmente de manera fiable y ordenada.
Web	Sistema de distribución de documentos de hipertexto o hipermedios interconectados y accesibles vía internet.
Web service	Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como internet.

RESUMEN

La solución desarrollada busca ayudar a las personas a poder encontrar de forma fácil, rápida y cómoda un lugar para descansar, accediendo a esta información desde un dispositivo con plataforma Android. Un lugar como hotel, posada o incluso un autohotel. Se le permite al usuario ver la información completa de cada lugar de hospedaje, y también se le muestra una amplia, actualizada y verídica galería de fotos. Todo esto por medio de una búsqueda de dirección o detectando ubicación a través de GPS del dispositivo. Esta solución le garantiza al usuario que el lugar que está viendo en su dispositivo móvil realmente existe y es tal como se muestra en las fotos.

Dado que es una solución completa, esta le permite al administrador en turno del hotel actualizar y alimentar los datos de su hotel en tiempo real y de una forma fácil e intuitiva. El usuario podrá ver que habitaciones están disponibles en el establecimiento, podrá ver fotos de las mismas y podrá reservarlas. Se presentan todas y cada una de las funcionalidades de la aplicación explicándolas detalladamente de dos formas, técnica y amigable al usuario. Asimismo, se detallan todos los requisitos necesarios para que la aplicación pueda ejecutarse sin ninguna contrariedad.

Por último, se proporciona un listado de conclusiones obtenidas luego de recibir la retroalimentación correspondiente directamente del usuario, se colocaron recomendaciones que deben ser tomadas en cuenta al querer implementar más funcionalidades a la solución.

OBJETIVOS

General

Desarrollar un sistema de hospedajes completo, proporcionando al usuario una aplicación para dispositivos móviles, que le permita encontrar fácil y rápidamente un conjunto actualizado, y verificado de establecimientos de hospedaje, a los cuales podrá acceder para ver todas sus características y galería de fotos. De igual forma, proporcionar una herramienta de alimentación de datos y estados a los establecimientos de hospedaje para que estos puedan estar actualizando continuamente su información.

Específicos

1. Desarrollar una App que le permita al usuario encontrar hospedaje desde su dispositivo Android.
2. Desarrollar una aplicación de alimentación de datos para que sea utilizada por el establecimiento de hospedaje.
3. Proporcionar una App actualizada e intuitiva que sea de fácil uso.
4. Implementar una arquitectura fiable que soporte eficientemente al sistema completo en todas sus necesidades.
5. Implementar una solución escalable, para lograr agregar funcionalidades de una manera simple y rápida.

INTRODUCCIÓN

Guatemala es un país rico en flora y fauna, un país que presenta muchas y variadas experiencias en todo el territorio nacional, desde antigüedades coloniales como el Castillo de San Felipe, pasando por las Georginas en Huehuetenango, pudiendo mencionar al lago más hermoso del mundo como lo es el lago de Atitlán y mencionando algo tan destacado lo que muchos han llamado el “Disneyland de Centro América”, Xetulul y Xocomil, en Retalhuleu. Existe bastante diversidad de lugares turísticos en Guatemala, devocionales como el Cristo Negro de Esquipulas o bien las majestuosas ruinas mayas de Tikal, se podría continuar describiendo lugares turísticos que son visitados por muchas personas extranjeras o también nacionales.

Dado que Guatemala es un país rico en turismo, es necesario que también adelante un paso hacia la tecnología y provea a los turistas no solo una majestuosidad y hermosura en flora y lugares turísticos, asimismo provea una herramienta que promueva el bienestar de todos en los diferentes lugares.

Es por esto que se pensó en el proyecto del desarrollo de una APP que sea accesible gratuitamente por todas las personas que desean visitar a Guatemala. En el presente documento se muestra una visión inicial del proyecto a desarrollar, el cual consiste en el desarrollo de una APP para Android que le permita a cualquier viajero que posea un *smartphone* localizar cualquier tipo de hospedaje, y visualizar toda la información necesaria para decidir que hospedaje escoger, tal como lo es contenido de fotos del hotel de las habitaciones e incluso saber si hay habitaciones disponibles.

Esta aplicación no solo ayuda a viajeros dentro del país, también provee una comodidad y facilidad a aquellas personas que desean pasar momentos a solas en lugares dentro de la ciudad de Guatemala en los conocidos y famosos moteles, lo cuales también se incluyen dentro de una sección de la aplicación.

1. MARCO TEÓRICO

La base para entender el desarrollo y la implementación de la aplicación es un conjunto de conceptos teóricos que se deben tomar en cuenta y que se describen a los largo del presente capítulo.

1.1. Aplicación móvil

Una aplicación para dispositivos móviles o también conocida como aplicación móvil, es una aplicación de software diseñada para ejecutarse sobre dispositivos móviles tales como los teléfonos inteligentes (*smartphone*) y las tabletas (*tablet*). Por lo general, estas aplicaciones pueden ser descargadas desde una plataforma de distribución al dispositivo móvil.

1.1.1. Evolución

Las aplicaciones para dispositivos móviles fueron ofrecidas inicialmente para productividad general y para recuperación de información, incluyendo correo electrónico, calendario, contactos, mercado de valores e información del clima. Sin embargo, la demanda pública y la disponibilidad de herramientas de desarrollo trajeron una expansión muy rápida de video juegos, redes sociales, servicios de localización geográfica, banca en línea, compras en línea y reservación de boletos. Un estudio realizado por COM Score, una compañía norteamericana dedicada al análisis de internet, ha reportado que la popularidad de estas aplicaciones ha continuado creciendo vertiginosamente. El estudio reportó que durante el primer trimestre de 2012, el 51,1 % de

suscriptores utilizó internet a través de las aplicaciones para estos dispositivos y el 49,8 % a través otros dispositivos como las computadoras.

1.1.2. Beneficios de las aplicaciones para dispositivos móviles

Existen numerosos beneficios que tienen sentido para desarrollar una aplicación para dispositivos móviles, dentro de los cuales se pueden mencionar los siguientes.

- Utilización de las características nativas: siempre será fácil aumentar las funcionalidades que provee una aplicación móvil. Utilizando características como el acelerómetro, el giroscopio, la cámara digital y otras propias de estos dispositivos.
- Contenido persistente local: permitiendo utilizar la aplicación aun cuando no se disponga de una conexión a internet, almacenando configuraciones tales como las preferencias de los usuarios.
- Mejor experiencia: los usuarios buscan aplicaciones que tengan una interfaz gráfica, consistente con el resto de aplicaciones en su dispositivo y con la plataforma donde se ejecute.
- Facilidad de descubrimiento: la mayoría de usuarios utilizan motores de búsqueda para encontrar aplicaciones para su dispositivo móvil. Este motor de búsqueda está incorporado en las herramientas que instala la plataforma de distribución.

- Mayor retroalimentación del usuario: los usuarios proveen información acerca de la aplicación, dándole un puntaje determinado y emitiendo sus comentarios sobre lo que piensan de la aplicación, en la plataforma de distribución.

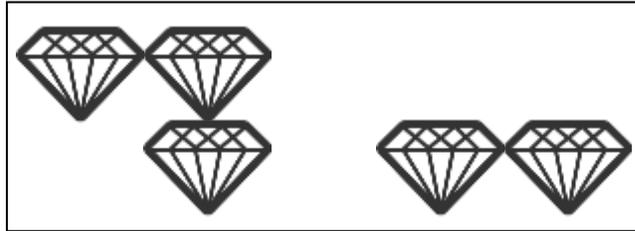
1.1.3. Entendiendo a los usuarios de aplicaciones móviles

La psicología de Gestalt ha tenido una influencia considerable en el diseño de interfaces de usuario, describiendo como la mente humana percibe y organiza visualmente los datos. Estos principios se refieren a teorías de percepción visual desarrollados por el psicólogos alemanes durante la década de 1920. De acuerdo a estos principios, cada estímulo cognitivo es percibido por los usuarios en su forma más simple. Los principios clave incluyen la proximidad, el cierre, la continuidad, la figura y el fondo, y la similitud.

1.1.3.1. Proximidad

Los usuarios tienen la tendencia de agrupar objetos. Los elementos colocados cercanamente uno de otro, son percibidos en grupos. Tal y como se muestra en la figura 1, las personas verán un grupo de tres diamantes y un grupo de 2 diamantes. Muchas partes más pequeñas pueden formar un agujero.

Figura 1. **Ejemplo de proximidad**



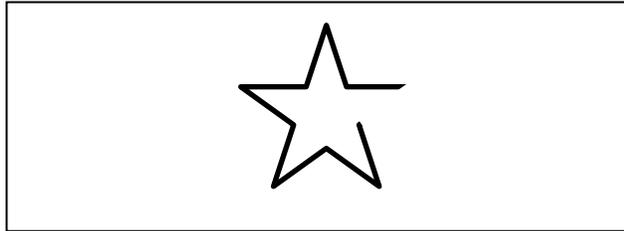
Fuente: elaboración propia, empleando Adobe Photoshop.

Los íconos que realizan tareas similares pueden ser organizados por categorías utilizando el principio de proximidad. Colocar texto descriptivo junto a gráficos para que el usuario pueda entender la relación entre estos gráficos y los objetos textuales.

1.1.3.2. Cierre

Si se muestra lo suficiente de una figura, las piezas faltantes son completadas por la mente humana. En la percepción de espacios sin cierre, los usuarios completan un patrón llenando la información que falta. Por ejemplo, en la figura 2, se ilustra este concepto, donde las personas reconocen una estrella incluso si la figura no está completa.

Figura 2. **Ejemplo de cierre**



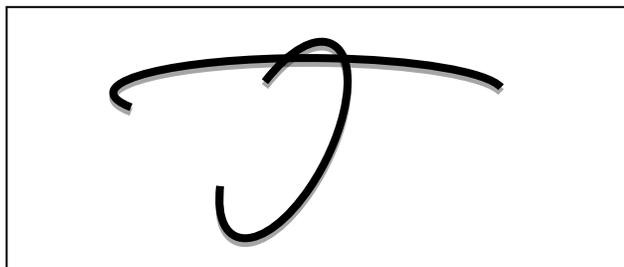
Fuente: elaboración propia, empleando Adobe Photoshop.

Se puede aprovechar el concepto de cierre creando íconos con una silueta primaria fuerte sin sobrecargarlos de detalles.

1.1.3.3. **Continuidad**

El ojo del usuario seguirá un objeto percibido como continuo. Cuando la continuidad ocurra, los usuarios están obligados a seguir un objeto hacia otro porque su enfoque lo guiará en la dirección que ya están observando. Por ejemplo la figura 3 muestra un trazo horizontal diferente de un trazo circular, inclusive, dan la idea de que estos trazos se entrelazan.

Figura 3. **Ejemplo de continuidad**



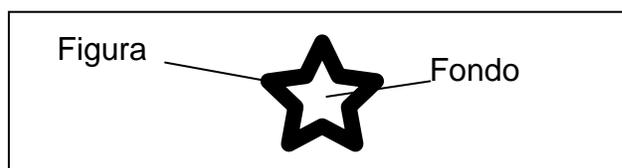
Fuente: elaboración propia, empleando Adobe Photoshop.

Las transiciones suaves visibles pueden conducir a los usuarios a través de una aplicación móvil, tales como cuando un botón activa un formulario y este aparece con una transición con efecto de aparición gradual y se desactiva con una transición de desaparición gradual.

1.1.3.4. Figura y fondo

Una figura como una letra sobre una página está rodeada por un espacio en blanco, este espacio en blanco es el fondo. Por ejemplo, en la figura 4, la figura es un estrella y el fondo es el espacio que la rodea.

Figura 4. **Ejemplo de figura y fondo**



Fuente: elaboración propia, empleando Adobe Photoshop.

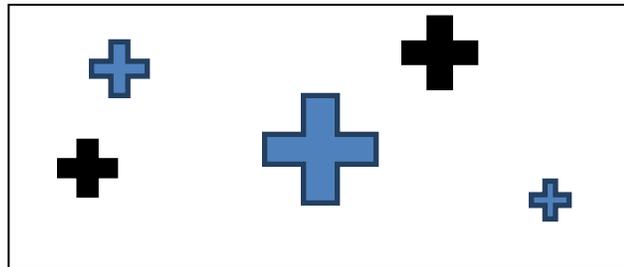
Los diseños complejos pueden jugar con la línea que divide la figura y el fondo, pero la rapidez en las interfaces para aplicaciones móviles a veces puede resultar frustrante. Los controles primarios y el contenido principal de la aplicación deben mantener una separación distinguible entre figura y fondo.

1.1.3.5. Similitud

Los elementos similares son agrupados de una forma casi automatizada, de acuerdo a una fuerte percepción visual del color, forma, tamaño y otros

atributos. Por ejemplo, en la figura 5, percibiendo la similitud, los objetos no similares resaltan.

Figura 5. **Ejemplo de similitud**



Fuente: elaboración propia, empleando Adobe Photoshop.

Es importante que en una interfaz visual, los usuarios no se confundan al relacionar todos los componentes mostrados en toda el área. La disposición debería fomentar la correcta agrupación de las ideas y de los objetos.

1.1.3.6. Usabilidad

Es un factor muy importante a tomar en cuenta cuando se desarrolla una aplicación para dispositivos móviles porque determina si un usuario seguirá utilizando la misma o la desinstalará. Por ejemplo, si una opción no puede ser encontrada fácilmente o su texto es muy pequeño y difícil de leer, indica que la aplicación no es usable. Para determinar la usabilidad de una aplicación hay que recordar que está dirigida hacia una audiencia en específico, y por lo tanto se debe elaborar una estrategia donde se suplan las necesidades de ese grupo. También hay que recordar que el modo de entrada del dispositivo es una pantalla táctil y no se deben diseñar interfaces con una gran cantidad de

componentes y de opciones, en una misma pantalla porque esto tiende a provocar frustración en los usuarios.

1.1.3.7. Accesibilidad

Una aplicación que es fácil de usar para las personas con un pobre o disminuido sentido de la visión, destreza limitada o con alguna discapacidad, será más fácil de usar para las demás personas. La accesibilidad entonces, es una forma de alcanzar más usuarios, así como también una forma de mantener a los usuarios existentes. Resulta imprescindible para la accesibilidad de una aplicación tomar en cuenta el comportamiento total, la navegación y enlaces, el diseño del contenido de una pantalla, la definición de la página y el modo de entrada.

1.1.3.8. Personalización de los criterios de búsqueda

Un factor importante en el desarrollo de un producto de información, software o bien en este caso una aplicación para dispositivo móvil, es la personalización de los criterios del usuario. Este factor se basa en la frase “*A wealth of information creates a poverty of attention*”. En estos días la información se encuentra altamente y rápidamente disponible, esto mismo causa sobrecarga de información no deseada.

El problema en estos días no es encontrar la información o acceder a la misma, el problema es la sobrecarga de información, demasiada información es mostrada siendo mucha de esta no deseada. El valor real producido por la aplicación es proveer información bien localizada y filtrada, comunicándole al usuario únicamente lo que le es útil, se debe recordar que el software es un

bien de experiencia, el valor de la aplicación lo da el usuario, el cual debe sentirse confortable con la información mostrada. El reunir información de una mejor forma respecto a los gustos o tendencias particulares de los clientes permitirá que la aplicación sea altamente personalizable y más valiosa para cada usuario, la aplicación utilizará este factor para poder arraigarse más al usuario. Esto lo hará filtrando información de 3 rubros, cuando las categorías son:

- Precio
- Ubicación
- Tipo de establecimiento

Estas tres categorías permitirán al usuario adecuarse más a sus necesidades y con base en minería de datos la aplicación le recomendará establecimientos la siguiente vez, esto para evitar sobrecarga de información.

1.1.4. PhoneGap Framework

Es un *framework* de desarrollo, libre y de código abierto que permite crear aplicaciones móviles empleando API web ya estandarizadas. Básicamente este *framework* les permite a los programadores desarrollar aplicaciones para dispositivos móviles utilizando tecnologías web como lo son JavaScript, HTML5 y CSS3.

Este *framework* trabaja en una capa superior a las aplicaciones nativas del dispositivo. PhoneGap permite un desarrollo de plataforma cruzada para poder evitar el lenguaje de desarrollo nativo de cada plataforma móvil. Este *framework* utiliza API bindings para poder acceder a cada componente del

dispositivo móvil como los sensores, contactos, información de la red entre otros.

1.1.4.1. App utilizando tecnología web

El poder utilizar tecnologías web para desarrollar aplicaciones móviles provee una enorme flexibilidad y ventajas. La principal y más sobresaliente es el expertise del desarrollador, existen más desarrolladores especializados y enfocados en desarrollo web que en desarrollo móvil, de igual forma con tecnologías web en la actualidad se tiene una amplia gama de complementos los cuales son perfectamente usados en estas aplicaciones. La movilidad y flexibilidad que se posee con este tipo de aplicaciones es enorme, estas aplicaciones pueden ser soportados y ejecutados en cualquier plataforma porque son programadas en un lenguaje intermedio.

Las aplicaciones web desarrolladas con tecnologías web e implementando el *framework* PhoneGap poseen los siguientes componentes básicos:

- **Config.xml:** es un archivo que provee la información de la app y especifica parámetros que afectan la forma en que esta va a trabajar y funcionar. Este archivo se basa en estándares W3C.
- **Index.html:** la aplicación en sí se implementa como una página web, por lo cual se encuentra por *default* este archivo que hace referencias a CSS, JavaScript, imágenes, archivos de cualquier tipo y cualquier otro recurso necesario.

- **WebView:** es donde se ejecuta la aplicación, esta se encuentra entre el native application wrapper que permite distribuir la aplicación. Este componente le provee a la aplicación la interfaz gráfica.
- **API de híbrido:** un *plugin* que le permite comunicarse a PhoneGap con los componentes nativos de la plataforma.

Al momento de desarrollar una aplicación móvil con tecnologías web se tiene 2 flujos de trabajo básicos para poder realizarlo.

- **Cross-Platform (CLI):** este flujo de trabajo permite ejecutar la aplicación en el mayor número de sistemas operativos posibles. Esta forma de desarrollo se centra en el uso de la herramienta córdova también conocida como CLI. Esta herramienta provee la interfaz necesaria para implementar *plugins* en la aplicación.
- **Platform-centered:** este flujo de trabajo es el que debemos utilizar si se quiere enfocar el desarrollo para una única plataforma.

1.1.4.2. ¿Por qué utilizar PhoneGap?

La tendencia actual es utilizar lenguajes intermedios para el desarrollo de aplicaciones que se utilizarán en dispositivo móviles. Estos lenguajes intermedios son los que no son exclusivos de la plataforma nativa del dispositivo sobre el cual se está utilizando la aplicación.

Básicamente la implementación de una aplicación tiene dos vías, utilizando implementación Nativa y No Nativa (como PhoneGap). A continuación se expondrán ciertas ventajas de utilizar una implementación No Nativa:

- Movilidad Multiplataforma: la ventaja más notoria es poder utilizar el mismo código base para todas las plataformas soportadas por PhoneGap, pudiendo utilizar el mismo código para Android, Iphone, BlackBerry, Windows Phone, entre otros. Se dice mismo código base ya que para algunas plataformas es necesario realizar mínimas modificaciones.
- Desarrollador Web familiarizado: la curva de aprendizaje de desarrollo de una App será mínima en este caso, ya que no debe aprender la plataforma nativa.
- PhoneGap: es el *framework* que más plataformas móviles soporta. Además de funcionar en Iphone, Ipad y Android también funciona en Palm, symbian, Windows Phone y BlackBerry.
- Provee enorme libertad de desarrollo, es muy fácil de desarrollar para las personas con altos conocimientos de HTML y JavaScript.
- Es libre y de código abierto.
- Es el *framework* que posee mejor adaptación a los controles y aspectos nativos de todas las plataformas.

1.1.4.3. HTML

Es el lenguaje con el que se definen las páginas web. HTML son las siglas de HyperText Markup Language. Esto se refiere a un lenguaje de marcas para poder desarrollar páginas web. HTML sirve para definir la estructura básica y un código para la definición de cómo se mostrará cierto

contenido en una página web, dicho contenido puede ser textos, imágenes, etc.

Este lenguaje se basa en las referencias. Es decir cuando se quiere añadir cualquier elemento externo a la página dentro de esta misma se debe realizar una referencia a la ubicación del elemento externo, dicha referencia se hace en forma de texto. Es por esto que la página HTML solo posee texto, una página HTML es solo texto con referencias a otros elementos, el encargado de interpretar y armar todos los elementos en la misma página es el navegador.

Se puede visualizar el lenguaje, HTML, como la lengua materna del navegador. Gracias a este lenguaje se puede presentar información en internet. El navegador interpreta el lenguaje HTML que está en la página y como resultado muestra lo que se puede visualizar.

Actualmente emergió una nueva versión de este, llamado HTML5, el cual se ha convertido en el mejor de los lenguajes web por ser bastante semántico, adaptable, flexible, multiplataforma y sumamente escalable. Este es con el cual se desarrollará la aplicación de PhoneGap. A continuación se detallan ventajas del HTML5:

- Es nativo, no necesita *plugins* de terceros. OpenSource.
- Semántico, posee etiquetas para clasificar y ordenar en distintos niveles y con varias estructuras el contenido.

1.1.4.4. JavaScript

Es un lenguaje de programación interpretado. Es un lenguaje que utiliza el paradigma de POO (Programación orientada a objetos). Este lenguaje es

clasificado como débilmente tipado y dinámico. Este lenguaje normalmente se implementa en el lado del cliente, aunque puede ser entendido en el lado del servidor. Es importante que entendamos que JavaScript no es Java y no tiene nada que ver con este último, es muy común confundirlos o bien pensar que son lo mismo.

Las principales características de este lenguaje de programación son:

- **Objetual:** esto significa que utiliza objetos en su totalidad. Los objetos que utiliza JavaScript son *arrays* asociativos a los cuales se puede acceder mediante la notación del punto.
- **Débilmente tipado:** una característica de todos los lenguajes *scripting*. Donde el tipo de dato que se guarda en las variables está asociado al valor no a la variable.

Las principales ventajas de este lenguaje de programación son:

- Realización de validaciones del lado del cliente, cualquier tipo de validación en ingreso de datos puede ser validada sin necesidad de llamadas al servidor, lo cual hace aplicaciones más livianas y eficientes.
- Creación de efectos dinámicos. El uso de JavaScript para hacer aplicaciones más agradables al usuario es algo muy común. Esto mejora la experiencia del usuario.
- Alta compatibilidad con la mayoría de los navegadores modernos incluyendo iPhone, móviles y PS3.

1.1.4.5. CSS

Es un lenguaje para hojas de estilos. Creado para controlar todo el aspecto y presentación de los documentos electrónicos que son previamente definidos con HTML o cualquiera de sus extensiones. CSS son las siglas en inglés de Cascading Style Sheets.

Utilizar CSS es la mejor manera de separar los contenidos y la presentación de las páginas web. Para poder crear páginas web complejas, dinámicas y agradables a la vista es imprescindible la utilización de este lenguaje. Básicamente separa la definición de los contenidos y la definición de su aspecto gráfico, lo cual provee varias ventajas ya que obliga a crear documentos HTML con significado completo, es decir, semánticamente correctos.

El uso del CSS mejora la accesibilidad al documento ya que minimiza la complejidad del mantenimiento, permitiendo visualizar un mismo documento en múltiples dispositivos.

Para crear una página web se utiliza lo siguiente:

- HTML/XHTML para crear o marcar los contenidos. Es decir la estructura del documento. Designar la función de cada elemento dentro de la página.
- CSS para definir el aspecto de cada uno de los elementos que se preestablecieron en el HTML/XHTML.

Funcionamiento del CSS: este lenguaje permite separar completamente los contenidos de la página y la información respecto a su aspecto. Se escribe

en una zona especial en la cual se agrega toda la información relacionada con los estilos de la página. Se pueden establecer estilos sin ensuciar el HTML.

1.1.4.6. JQuery

Es un conjunto de *scripts* llamado biblioteca en el cual este conjunto de *scripts* están escritos en el lenguaje JavaScript. Básicamente esta biblioteca facilita y simplifica la manera en la que se interactúa con los documentos HTML/XHTML, también la forma en la que se manejan los eventos, animaciones, se manipula el DOM y hasta la forma de interacción utilizando AJAX.

Esta librería es un software libre y de código abierto. En su totalidad JQuery provee funcionalidades implementadas en JavaScript que sin utilizar esta librería se requeriría demasiado trabajo y más líneas de código.

En su naturaleza, JQuery es un archivo de código JavaScript que tiene funcionalidades comunes de DOM, eventos, animaciones y AJAX. La función principal y más ventajosa de esta librería es que permite cambiar el contenido de una página web sin necesidad de recargarla, utilizando la manipulación de DOM y peticiones AJAX. Algunas ventajas de utilizar JQuery son:

- Uso sencillo y fácil de las funciones contenidas en la misma
- Agregar *plugins* de forma sencilla
- Soporte constante y rápido
- Excelente integración con AJAX
- Flexible y rápido para desarrollo

JQuery es una de las mejores librerías en el mundo de JavaScript. Es la librería más utilizada en este lenguaje de programación.

1.2. Métodos de comunicación por internet

La comunicación entre aplicaciones, el envío, recepción y el canal que utiliza la información, son componentes sumamente importantes en la implementación de un sistema informático. Dada la naturaleza de la solución que se plantea para los establecimientos de hotelería en Guatemala, es necesario conocer que tecnología se utilizará para comunicación entre los componentes del sistema.

1.2.1. Web service

Es una tecnología que permite el intercambio de datos entre aplicaciones en una forma estandarizada permitiendo integrar aplicaciones de diferente tipo, este intercambio de datos se da sin necesidad de conocer los detalles de los diferentes sistemas de información, de hecho los diferentes sistemas de información que se intercomunican por medio de *web service* pueden estar implementados en diferentes lenguajes de programación.

Los servicios web se implementan mediante el uso de los siguientes componentes:

- XML: usado para describir los datos, el contenido en sí. Extensible Markup Language es una especificación creada por W3C, esta especificación permite a los programadores crear *tags* propios para habilitar definiciones, transmisiones y varias interpretaciones de datos entre sistemas informáticos.

- SOAP: es el encargado de la transferencia de datos. Simple Object Access Protocol es un protocolo para manejo de mensajes desarrollado en XML que se utiliza para codificar datos de los requerimientos de los Web services. Los mensajes SOAP son independientes de la plataforma ya que pueden ser transportados por protocolos que funcionan en internet como el HTTP y el SMTP.
- WSDL: el cual describe los servicios disponibles y los procedimientos o funciones disponibles dentro de estos. Web Services Descripción Language es un lenguaje especificado en XML que se utiliza para definir los *web services* como conjunto de puntos finales de comunicación. Es un estándar de uso público.
- UDDI: se encarga de conocer los servicios disponibles, este trabaja sobre los protocolos de internet. Universal Description, Discovery and Integration es un directorio distribuido que funciona en la web, este directorio permite a las compañías publicar sus servicios para que otras compañías los conozcan y los utilicen.

Los servicios web es una tecnología que funciona a través de la red, esta tecnología no interactúa directamente con los usuarios, ya que lo que se busca es que se comuniquen diferentes sistemas informáticos sin necesidad de intervención humana, es por esto que los *web services* no tienen una interfaz gráfica para ser utilizados.

La implementación de servicios web no está relacionada con ningún sistema operativo o lenguaje de programación. Programas de diferentes lenguajes pueden conversar de una forma simple y eficiente utilizando servicios web. De hecho esta tecnología permite comunicar aplicaciones de diferentes

orígenes. Los servicios web no necesitan utilizar navegadores ni HTML para poder funcionar.

1.2.2. RESTFul Web Service

Los RESTFul web Service tienen la misma naturaleza y definición que los *Web service*. Básicamente la diferencia es el REST lo cual se explicará a continuación. Un *web service* normal se comunica utilizando SOAP como encargado de la transferencia de datos.

REST, Representational State Transfer es decir, la transferencia de estado representacional es una adopción en la web actualmente, ya que es una alternativa más simple respecto a SOAP y a los servicios web que están basados en WSDL.

REST define un conjunto de principios arquitectónicos con los cuales se implementan los servicios, el foco principal de estos principios son los recursos del sistema. Los principios de REST son:

- Utilizar métodos HTTP explícitamente: asociación 1 a 1 entre operaciones crear, leer, actualizar y borrar respecto a los métodos HTTP. La asociación es la siguiente:
 - POST: para crear un recurso en el servidor
 - GET: para obtener un recurso
 - PUT: para cambiar estado de un recurso o bien modificarlo
 - DELETE: Para eliminar el recurso
- No mantienen estado: esto se refiere a cuando se realiza una petición. Básicamente es enviar una petición completa que incluya todos los datos

necesarios para cumplir el pedido para que de esta manera aunque se comuniquen muchos servidores se pueda redireccionar y gestionar la carga sin mantener el estado local entre peticiones.

- Muestra URI con forma de directorios: cuando una aplicación desea acceder a un recurso la URI es la que define lo intuitivo que tan intuitivo es el *web service* REST. Las URI de los servicios deben hablar por si solas hasta el punto que adivinarlas debe ser lógico y sencillo.
- Puede transferir XML, JSON o ambos, pero principal y más comúnmente JSON: la forma en la que se representa un recurso representa el estado actual del mismo y sus atributos en el momento que el cliente hizo la petición. Se puede decir que la representación del recurso son fotos del recurso en dicho diferencial de tiempo.

REST es la mejor solución para diseñar servicios web más independientes, y también una excelente solución para exponer los recursos del sistema con un API REST, ya que se puede proporcionar la información a distintas aplicaciones con formatos diferentes. Los servicios REST son unos grandes facilitadores de integración de aplicaciones en sistemas informáticos complejos, ya que el uso de estos facilita y agiliza la comunicación.

1.2.3. JAVA RESTful Web Service

En sí un JAVA RESTful *web service* es un servicio que permite tener un buen y funcional diseño de servicios, ya que se puede separar clara y lógicamente como se va a acceder a los recursos. Los servicios Web Java utilizan las características JavaEE6. Estos servicios deben ejecutarse en un servidor de aplicaciones, el que se utilizará será Glassfish.

A continuación se detallan y explican los elementos principales de un servicio web JAVA. En el siguiente código se expondrá en un RESTful Service de los recursos “Persona”.

Figura 6. **Clase Persona–Recurso a Acceder como REST**

Primer Componente la Clase Persona:

```
public class Persona {  
  
    public String nombre;  
    public String apellido;  
    public int edad;  
    public int peso;  
  
    public Persona() {  
    }  
    public Persona(String nombre, String apellido, int edad, int peso) {  
        this.nombre = nombre;  
        this.apellido = apellido;  
        this.edad = edad;  
        this.peso = peso;  
    }  
}
```

Segundo Componente La clase que expondrá en forma REST el recurso persona.

Fuente: elaboración propia, empleando HTML 5.

Figura 7. **Clase PersonaResource–Clase para Acceder como REST**

```
@Stateless
@Path("/persona")
public class PersonaResource {

    @GET
    @Produces("application/json")
    public String persona(@QueryParam("id") int id){
        Persona n=new Persona();
        n.nombre="Raul Antonio";
        n.apellido="Giron Alvarez";
        n.edad=id;
        n.peso=280;
        Gson g=new Gson();
        return g.toJson(n);
    }
}
```

Si se analizan los componentes principales son :

- @Stateless: esta anotación define que la clase ser un EJB.
- @Path("/persona"): esta anotación indica que este recurso será accedido desde la ruta "/persona" vía web.
- @GET: esta anotación indica que dicho método va a ser de lectura para obtener una persona.
- @Produces("application/json"): esta anotación indica que dicho método (función) produce como resultado un JSON.

Fuente: elaboración propia, empleando HTML 5.

1.2.4. **Servidor de aplicaciones**

Es un servidor que permite procesar datos de una aplicación cliente. Básicamente es un servidor que se encuentra en una red de computadoras y que ejecuta ciertas aplicaciones. Un servidor de aplicaciones es un programa o

bien un conjunto de programas que permite brindar servicios de determinada aplicación a las computadoras clientes.

Ventajas de los servidores de aplicaciones:

- Mejora *performance* en grandes aplicaciones ya que limita el tráfico de la red solamente al tráfico de la capa de presentación.
- Los cambios en la configuración de una aplicación almacenada en este es centralizada ya que únicamente una vez se realiza esto.
- Dado que está centralizada la capa de aplicaciones, las actualizaciones de estas están garantizadas para los usuarios.

1.2.5. Glassfish

Es un servidor de aplicaciones que fue desarrollado por Sun Microsystems enfocado en la plataforma Java EE. A continuación se muestran algunas de las principales funcionalidades de Glassfish:

- Servidor JEE
- Servidor Web
- Contenedor de EJB
- Servidor de *web services*
- Balanceo de carga y colas de espera para concurrencia

1.3. Dadabik

Para un desarrollo ágil y eficiente existen gran cantidad de herramientas que auxilian a la implementación de una solución informática, estas

herramientas se pueden dividir en varios tipos, pero las herramientas que más tiempo reducen y riesgos al mismo tiempo son los generadores de código.

Para la implementación de este sistema informático es necesario una aplicación de administración para cada establecimiento y esta aplicación debe poder ser accesible desde cualquier ubicación, en cualquier momento y desde cualquier dispositivo, es por esto que se tomó la decisión de realizarla web y por medio de un generador de código llamado Dadabik.

1.3.1. ¿Qué es DaDaBIK?

Es una aplicación bastante popular desarrollada en PHP que permite crear *front-end* bastante personalizable sin necesidad de codificar una sola línea. Básicamente es un generador de código que permite personalizarlo a un alto nivel, es de los mejores generadores de código en la actualidad.

Una de las grandes ventajas de DaDaBIK respecto a otros generadores de código es que no produce directamente un *script* de PHP, ya que producir un *script* de PHP no permitiría que se este mismo se actualizara cuando se realizan cambios al esquema de la base de datos. Por esta razón, cuando se realizan modificaciones realmente se realizan directamente en la capa de abstracción, por lo cual se puede actualizar el esquema de base de datos en cualquier momento.

1.3.2. ¿Porqué utilizar DaDaBIK para implementación de una página de administración?

Esta aplicación se acopla a las necesidades de una página para administración ya que da acceso a un esquema de base de datos de una forma

fácil, rápida y altamente personalizable por parte del administrador del sistema. A continuación se presentan las principales ventajas de esta aplicación:

- Alto nivel de personalización. Para cada campo de una tabla en la base de datos se puede realizar lo siguiente:
 - El campo de la base de datos puede estar incluido o excluido en los diferentes formularios que provee la herramienta como buscar/insertar/actualizar y *datagrids*.
 - Se puede personalizar el texto.
 - Se puede seleccionar el formato que deseemos para el contenido.
 - Se puede colocar *Dropdown* menú, con base en valores posibles de una relación de la base de datos.

- Características que facilitan la adaptación a cualquier sistema o aplicación de administración:
 - Personalización y creación de *datagrid*
 - Personalización y creación de formularios
 - Personalización de *datagrid* utilizando CSS
 - Subir y bajar archivos
 - Visualizar relaciones Maestro-Detalle
 - Autenticación
 - Personalización de los permisos con alta granularidad
 - Exportar datos a CSV
 - Alertas de *email*

1.3.3. Configuración básica de DaDaBIK

Para la configuración de DaDaBIK se debe modificar el archivo de configuración que este trae y copia al equipo cuando se instala. Este archivo se

encuentra en la dirección: /include/config.php. Al abrir el archivo se puede configurar los parámetros que se desean, a continuación se escriben los principales parámetros que se deben configurar:

Figura 8. **Parámetros básicos de configuración**

```
// dbms type ('mysql' or 'postgres' or 'sqlite')
$dbms_type = 'mysql';

// DBMS server host
$host = 'mysql115.000webhost.com'; // the name or the IP of the host

// database name
$db_name = 'a9973043_as02'; // for SQLite not only the name but the file name

// database user
$user = 'a9973043_as02'; // this user must have select, insert, update, delete, and execute permissions

// database password
$pass = 'ingeniero7'; // for sqlite this parameter is not needed

// DaDaBIK complete url (e.g. http://www.mysite.com/john/dadabik/)
$site_url = 'http://proyectoas02.freeiz.com/as/program_files/';
```

Fuente: elaboración propia, empleando HTML 5.

Como se puede ver en la figura 8 se debe configurar de la siguiente forma:

- `$dbms_type`: indica que DBMS utiliza mysql, postgres o sqlite.
- `$host`: se debe colocar la Ip o el nombre del *host* (dominio) donde está la base de datos.
- `$db_name`: nombre de la base de datos.
- `$user`: usuario de la base de datos con permisos deseados.
- `$pass`: contraseña del usuario de la base de datos.
- `$site_url`: es la dirección completa del sitio donde se encuentra la aplicación de DaDaBIK.

Figura 9. **Parámetros de configuración**

```
$site_path = 'as/program_files/';

// timezone, specify here your timezone (a li
$timezone = 'America/Guatemala';

$secret_key = 'ingeniero7';

$dadabik_session_name = 'as02';

// prefix to use for the name of the tables c
$prefix_internal_table = 'dadabik_'; // you c

// choose the front-end language ('english',
$language = 'spanish';
```

Fuente: elaboración propia, empleando HTML 5.

Los parámetros que se ven en la figura 9 se deben configurar así:

- `$site_path`: es la dirección respecto al *host*, básicamente es la dirección de `site_url`, pero sin el dominio del equipo.
- `$timezone`: especificar el *timezone* para la aplicación.
- `$secret_key`: utilizada para la autorización de las *cookies*.
- `$dadabik_session_name`: muy importante cuando se tienen más aplicaciones de DaDaBIK para que las variables de sesión empiecen con este prefijo.
- `$prefix_internal_table`: prefijos de tablas internas de DaDaBIK.
- `$language`: lenguaje para la aplicación.

1.3.4. **Listado de precios (pricing)**

DaDaBIK es un software por el cual se debe de pagar, a continuación se muestra el listado de precios y algunas características respecto a la versión.

Figura 10. Listado de precios y características

	Basic	Pro	Enterprise
Choose your Version	€15 (\$20)	€50 (\$69)	€160 (\$220)
GENERAL FEATURES			
Forms generation	✓	✓	✓
Datagrid generation	✓	✓	✓
CSV export	✗	✓	✓
E-mail alert	✓	✓	✓
File upload	✓	✓	✓
Front-end in 17 languages	✓	✓	✓
Wordpress integration	✗	✓	✓
UTF-8 support	✓	✓	✓
Full, unencrypted source code	✓	✓	✓
Unlimited number of applications	✓	✓	✓
SUPPORTED DBMS			
MySQL support	✓	✓	✓
PostgreSQL support	✓	✓	✓
SQLite support	✓	✓	✓
PERMISSIONS-RELATED			
Users and User Groups	✓	✓	✓
Authentication	✓	✓	✓
Secure password storage	✓	✓	✓
Granular permissions manager	✗	✓	✓
Records Ownership	✗	✗	✓

Fuente: *Dadabik*. http://www.dadabik.org/index.php?function=show_pricing. Consulta: 10 de marzo de 2015.

1.4. Metodología de desarrollo (proceso de software)

En la actualidad el proceso de software es algo sumamente importante que se debe tener en cuenta en la implementación de cualquier sistema

informático, esto porque la implementación de un sistema es una obra de ingeniería.

La ingeniería de software es un enfoque sistemático, disciplinado, cuantificable para el desarrollo, operación y mantenimiento del software. Utilizar un correcto proceso de software es una tarea vital para cualquier ingeniero de software, es por esto que este sistema no se implementará de forma rústica, empírica o artesanal, la implementación de este sistema se basará en un proceso de software preestablecido.

1.4.1. SCRUM

Es una metodología ágil que recibe su nombre por una jugada de *rugby* en el cual todos los delanteros de cada lado van juntos en una formación cerrada, y luchan por ganar posesión de la pelota usando sus pies. Básicamente *scrum* es un proceso que sigue un "caos organizado".

Esta metodología se basa en la noción que el proceso de desarrollo de software es impredecible y complicado, por lo cual puede ser definido únicamente por un conjunto poco rígido de actividades.

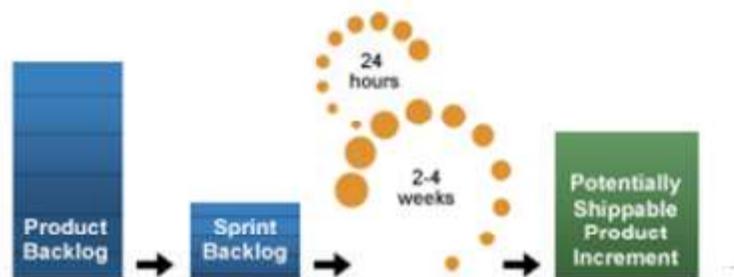
Scrum adopta una estrategia de desarrollo incremental en lugar de utilizar una planificación y ejecución completa del software, esto por lo mencionado con anterioridad ya que no se puede predecir el software.

1.4.1.1. Actividades de SCRUM

Scrum posee actividades principales las cuales se detallan a continuación:

- Creación de Backlog: al inicio del proyecto se crea una lista de lo que el cliente quiere y necesita.
- Ejecución de Sprint: los *sprints* son ciclos de 2 a 4 semanas en los cuales se procede a la implementación.
- Features definidas: cada *sprint* toma un conjunto de características del *backlog* para desarrollo.
- Se tienen equipos de entre 6 y 9 miembros que se enfocan en un área de trabajo.

Figura 11. **Actividades Scrum**



Fuente: elaboración propia, empleando Adobe Photoshop.

Otras actividades necesarias para Scrum son las siguientes:

- Daily Scrum meeting: el *scrum master* es responsable de conducir esta reunión, en esta reunión se permiten únicamente 3 preguntas:
 - ¿Qué ítems han sido completados desde el último *scrum meeting*?
 - ¿Qué cosas se han descubierto y necesitan ser resueltas?
 - ¿Qué nuevas asignaciones hacen sentido completar para el equipo hasta la siguiente *scrum meeting*?

- Demostración al cliente: al final de cada *sprint* se conduce a una demostración para el cliente con el propósito de demostrar lo que se ha alcanzado, dar a los desarrolladores sensación de logro, asegurar que el software ha sido integrado y probado apropiadamente, asegurar que se ha realizado progreso real en el proyecto.

1.4.2. Beneficios y costos de procesos ágiles

A continuación se detallan los principales beneficios de emplear procesos ágiles:

- Receptivo al cliente, siempre hay software demostrable.
- La estimación se facilita más cada ciclo.
- Se realizan pruebas a fondo.
- Motiva a los desarrolladores ya que en cada demostrable tienen satisfacción de logro.

Los principales costos de emplear procesos ágiles son:

- Dificultad en estimar todo el proyecto
- Dificultad para integración de nuevos participantes

2. DESCRIPCIÓN DEL PROBLEMA Y PLANTEAMIENTO DE LA SOLUCIÓN

El presente sistema debe de solucionar un problema del cual se ha hecho mención previamente en el documento, en el presente capítulo se describirá el problema real y como se solucionará con el Sistema de Hospedajes.

2.1. Descripción del problema

Guatemala es un país con diversidad de lugares turísticos para visitar, esto hace a Guatemala un perfecto destino para cualquier viajero o aventurero extranjero. Dado esta fortaleza del país se debe invertir más en el trato a los turistas y en la información proporcionada a los mismos. Pero no solo turistas extranjeros también muchos guatemaltecos visitan las riquezas naturales y lugares turísticos con grandes atracciones. Es por esto que en la actual era de la información debe hacer que esté disponible y a la mano de cualquiera que tenga tecnología a su alcance.

Sin embargo, con tantos lugares turísticos y atracciones naturales que posee Guatemala, no existe una aplicación o sistema de información que guíe y provea información verídica y actualizada a los turistas en todo el país, hacia lugares de hospedaje seguros. Muchas veces se oyen testimonios en los que se desea reservar, ver disponibilidad y hasta comparar los diferentes establecimientos de hospedaje en los lugares turísticos de Guatemala, pero no existe más que comparaciones por testimonios de amigos y conocidos.

Es por esto que surge la necesidad de un sistema que provea al turista toda la información necesaria acerca de los establecimientos de hospedaje en Guatemala, en todos sus municipios.

2.2. Antecedentes de la solución

En la actualidad encontrar un proyecto o sistema que no se haya implementado es una tarea bastante complicada. Sin embargo, el presente sistema parece ser único en su naturaleza, siendo esta una aplicación de hoteles sin importar su categoría o métrica en estrellas. Incluyendo cualquier hotel en incluso motel dentro de Guatemala.

2.2.1. Despegar.com

Es un sistema web que posee la misma idea, pero se diferencia en que en este sistema se postean muchas ofertas de los diferentes hospedajes, aún si el cliente no desea verlas. Parece bastante parecida, pero no lo es ya que al ingresar y buscar hoteles en Guatemala, únicamente muestra hoteles de 4 estrellas para arriba, cuando uno desea encontrar un hotel en la ciudad de Guatemala le muestra Camino Real, Vista Real y Grand Tikal Futura. Pero no le muestra los hospedajes cómodos y agradables que hay en muchas partes de Guatemala. Este sistema web no toma en cuenta auto hoteles y moteles del país.

Figura 12. Despegar.com



Fuente: *Hoteles*. www.despegar.com. Consulta: marzo de 2015.

2.2.2. Sistemas A la Medida para Hoteles (reserva e información)

En la actualidad muchos hoteles tienen un sistema de reserva y *website* o inclusive App para dar a conocer el hotel. Sin embargo, la diferencia entre el sistema que se plantea y estos sistemas son los siguientes:

- El sistema a desarrollar es genérico personalizable y acoplable a cualquier establecimiento de hospedaje, creando un estándar de administración de establecimientos de hospedaje.
- Los sistemas a la medida tienen costo de US\$ 10 000 o más. El sistema a implementar se distribuirá con costo mensual significativo.
- Solo hoteles con mucho dinero e ingresos altos pueden obtener estos sistemas desarrollados a la medida. El sistema a implementar permitirá que todos los establecimientos de hospedaje de Guatemala sin importar su tipo, obtener y adquirir estos servicios de tecnología.

2.3. Planteamiento de la solución

A continuación se explicará el planteamiento de la solución.

2.3.1. Alcances

El proyecto que se desarrolla es una solución informática completa, es un sistema completo.

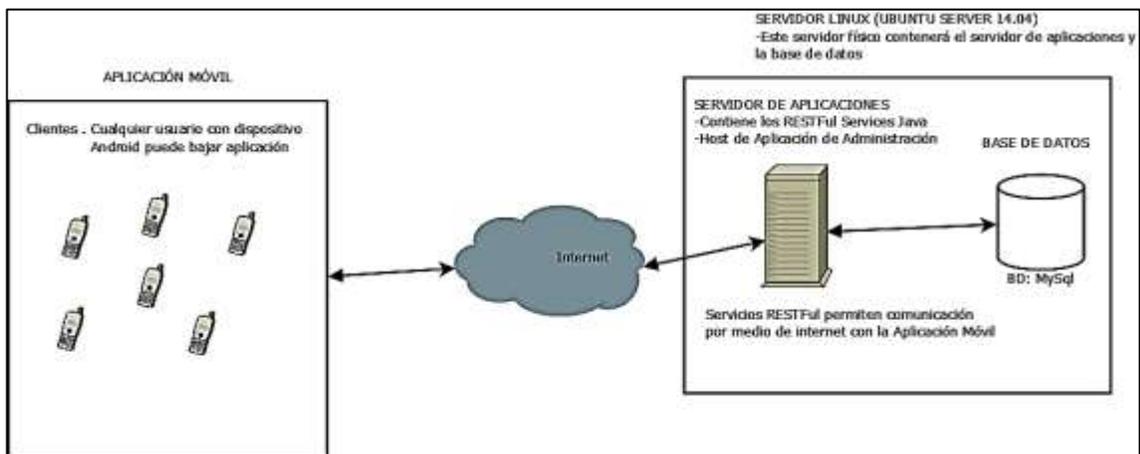
Los componentes tecnológicos que serán incluidas para esta solución son los siguientes:

- **Aplicación móvil (desarrollada con PhoneGap).**
Es la aplicación móvil que podrá ser descargada por cualquier usuario que tenga un dispositivo Android. Estará disponible en Kunustore como primer punto. Esta aplicación es gratuita totalmente.
- **Aplicación Administración de Hotel (generada con DaDaBIK)**
Esta aplicación la cual es aplicación web, es a la que los administradores o personal en turno de cada hotel accederá para retroalimentar el sistema. Esta aplicación tiene como objetivo principal el indicar que habitaciones están disponibles y cuales no lo están.
- **Capa de Datos (MySQL)**
Es la base de datos que contendrá toda la información de la aplicación. El DBMS a utilizar es MySQL.

- Capa de servicios (RESTFul Java)
Es la capa de comunicación y publicación de datos, por medio de esta capa se podrán comunicar la aplicación móvil y la aplicación web de administración. La función principal de esta capa es la de hacer disponible los recursos de la base de datos.

El funcionamiento y relación de estos componentes se presentan en el siguiente diagrama.

Figura 13. Diagrama de la solución completa



Fuente: elaboración propia, empleando Adobe Photoshop.

Los componentes físicos que serán incluidos en esta solución son:

- Servidor físico:
Este contendrá un servidor de aplicaciones y el servidor de base de datos. Las características de este servidor son:
 - Procesador: Corei3
 - RAM: 8Gb

- Sistema operativo: ubuntu 14.04
- Servidor de BD: MySql 5.1.49
- Servidor de aplicaciones: Glassfish 4.0

2.3.2. Alcances y límites de la aplicación de administración

Las características técnicas de esta son:

- Aplicación web
- Lenguaje de implementación: PHP
- *Framework* utilizado para agilizar proceso: DaDaBIK

Esta aplicación es la que accederá el personal en turno del hotel, para poder actualizar el estado de las habitaciones del establecimiento. Las opciones serán las siguientes:

- Habitaciones
 - Ver estado (ocupado, libre o reservada)
 - Ver detalle habitaciones
 - Ocupar habitación
 - Desocupar habitación
 - Ver datos de reserva
 - Modificar habitación, modificar precios, fotos y otros datos
 - Agregar promociones
- Log In (ingreso al sistema)

Para esta aplicación existirán 2 tipos de usuario:

- **Administrador**
Es el usuario que podrá modificar datos, precios y fotos de las habitaciones. Es el único que puede crear las promociones del establecimiento que serán vistas en la aplicación.
- **Operador**
Es el usuario que podrá ver estado y cambiar estado de las habitaciones. También tiene acceso a ver información de reservas.

2.3.3. Descripción de la aplicación de administración

Esta aplicación es a la cual accederá el personal del hotel, esta aplicación podrá ser accedida desde cualquier dispositivo computadora o dispositivo móvil. Esta aplicación tendrá como principal objetivo alimentar el sistema el estado de las habitaciones. Los estados pueden ser:

- **Ocupado:** estado que indica que la habitación se encuentra ocupada y otro cliente no puede hacer uso de la misma. En este estado el cliente que la está ocupando esta dentro de la habitación.
- **Libre:** estado que indica que la habitación se encuentra disponible y que otro cliente puede hacer uso de la misma.

Esta aplicación es sumamente intuitiva. Con base en esta aplicación se alimenta el sistema de información. El uso de esta no debe ser impedimento.

2.3.3.1. Pantalla de Log In

Esta es la pantalla de ingreso al sistema, únicamente los usuarios registrados podrán ingresar. Dependiendo del usuario le mostrará los datos del establecimiento.

2.3.3.2. Pantalla habitaciones

Esta pantalla le muestra al usuario un listado de habitaciones y la información básica de la misma. Al lado izquierdo de cada habitación aparecen iconos para mantenimiento del sistema al usuario administrador se le muestran iconos de Ver Detalles, Editar y Eliminar. El usuario operador únicamente puede modificar el estado de las habitaciones.

2.3.3.2.1. Funciones principales de habitaciones

A continuación se muestran las funcionalidades que tendrán los usuarios del establecimiento de hospedaje en la pantalla de habitaciones.

- Funciones del usuario operador:
 - Visualizar Habitaciones del Hotel: en el listado de habitaciones al lado izquierdo de cada habitación se le mostrará la opción de “Cambiar Estado” y “Ver Detalles de Habitación”.
 - Cambiar Estado: esta función se le muestra en el lado izquierdo de cada habitación, al ingresar a esta opción el usuario podrá modificar el estado de la habitación. Los posibles estados son Ocupado y Libre.
 - Ver Detalles de Habitación: esta función se le muestra en el lado izquierdo de cada habitación, al ingresar a esta opción el usuario podrá visualizar la información completa de la habitación, incluyendo tarifa y fotos.
- Funciones del usuario administrador:
 - Ver Detalles de Habitación: esta función se le muestra en el lado izquierdo de cada habitación, al ingresar a esta opción el usuario

- podrá visualizar la información completa de la habitación, incluyendo tarifa y fotos.
- Cambiar Tarifa de habitaciones: a cada habitación podrá cambiarle la tarifa o bien al tipo de habitación se le cambia y todas la habitaciones cambian.
 - Agregar promociones del hotel: el administrador podrá agregar promociones que se mostraran en la aplicación.

2.3.4. Alcances y límites de la aplicación para dispositivo Android

Esta es una App que puede ser descargada gratuitamente por cualquier persona inicialmente desde Kunustore. Puede ser instalada en cualquier dispositivo que tenga como sistema operativo Android. Es una aplicación súper intuitiva que cuenta únicamente con 3 pasos. Las características técnicas de esta aplicación son las siguientes:

- Framework de Desarrollo: PhoneGap
- Tecnologías: Web, Javascript y css

2.3.5. Descripción de la aplicación Android

Esta aplicación móvil le permite al usuario encontrar en cualquier parte de Guatemala establecimientos de hospedajes. Como se mencionó anteriormente se podrá bajar de Kunustore y posteriormente (fuera de los alcances de este trabajo) de Google playstore. Esta aplicación le permite al usuario como primer punto escoger el tipo de establecimiento que desea como hotel, posada, motel, auto hotel o pensión, luego de esto tiene 3 pasos básicos e intuitivos los cuales son:

- Ingresar Dirección: en este paso puede ingresar manualmente la dirección o bien la opción "Encontrar Cerca" que utilizará el GPS.
- Filtros: este segundo paso puede filtrar por tarifas o bien más filtros, o bien ningún filtro.
- Ver Listado de Establecimientos y seleccionar uno. Este listado es con base en los filtros. De este listado selecciona el deseado y se le muestra los detalles del establecimiento.

2.3.5.1. Pantalla de inicio

Esta pantalla es la que el usuario visualiza al abrir la aplicación. Es opcional registrarse a la aplicación por esta razón la pantalla *default* es la pantalla de inicio. En esta pantalla se le mostrará el primer gran filtro de la aplicación. El primer paso en el que debe seleccionar el tipo de establecimiento que se desea buscar. Ejemplo: hotel, motel, autohotel, posada. Este primer gran filtro es para que se le personalice más la información al usuario.

A continuación se muestra un prototipo de esta pantalla de inicio.

Figura 14. **Prototipo pantalla de inicio**



Fuente: elaboración propia, empleando Adobe Photoshop.

2.3.5.2. **Action Bar de la aplicación móvil**

Para esta aplicación se utilizará un patrón denominado Toolbar o barra de acciones. Esta barra de acciones consta de una barra horizontal que muestra varias opciones únicamente con íconos, que representan las acciones más utilizadas dentro de la aplicación. Estas acciones son los 3 pasos que se vieron previamente.

Por lo cual existirá un ícono por cada paso. Los pasos se llamarán de la siguiente forma:

- Pantalla principal
- Ubicación
- Filtros
- Selección establecimiento

Las barras de acción en una aplicación por lo general muestran en la parte inferior de la pantalla, sin embargo, esta puede ser incluida también en la parte superior de la pantalla. Esta barra debe mostrar íconos que sean reconocibles y fáciles de recordar, o también mostrar etiquetas que describan la acción del ícono.

Figura 15. **Prototipo del Action Bar**



Fuente: elaboración propia, empleando Adobe Photoshop.

2.3.5.3. Pantalla ingreso dirección

Este es el segundo paso del intuitivo proceso de búsqueda, en esta pantalla es donde el usuario le indica a la aplicación en donde quiere buscar hoteles, se debe recordar que puede ser que el usuario este viendo previo al viaje o bien ya estando en el mismo, es por esto que se le muestran 2 opciones:

- **Encontrar Cerca:** esta opción utiliza el GPS del dispositivo para obtener la dirección en donde se encuentra, y con base en estos datos seguirá el proceso. Este botón es simple, se da un toque al mismo y este empieza a buscar y reconocer en donde está el usuario.
- **Ingresar Dirección:** si el usuario desea buscar en determinada dirección, esta función es utilizada si en caso el usuario no se encuentra cerca o bien desea conocer el estado de algún establecimiento.

Figura 16. **Pantalla de ubicación**



Fuente: elaboración propia, empleando Adobe Photoshop.

2.3.5.4. Pantalla de filtros

Esta pantalla es opcional, se encuentra en el flujo principal, sin embargo, no es necesario realizar operaciones en esta ya que únicamente es un método más de personalización de la información. Se puede seleccionar el rango de tarifas o bien únicamente se presiona siguiente o cualquier botón del Action Bar.

Esta pantalla actualmente contendrá únicamente el filtro de tarifas por rango. Esto aplica un tercer filtro a la búsqueda ya que el primero es la categoría del establecimiento, a continuación la ubicación y este que es opcional para mostrar hotel de determinado precio.

Figura 17. **Pantalla de filtros**



Fuente: elaboración propia, empleando Adobe Photoshop.

2.3.5.5. Pantalla listado de establecimientos encontrados

Esta pantalla puede ser accedida en cualquier momento. Si el usuario no selecciona ningún dato en los pasos previos los filtros *default* se describen a continuación:

- Categoría (pantalla principal): categoría seleccionada *default* es "Hotel".
- Ubicación (pantalla de ubicación): ubicación seleccionada *default* es Zona 1 de ciudad de Guatemala.
- Tarifas (pantalla de filtros): el rango seleccionado *default* es "Cualquiera", esto significa no importa precios.

Al momento que el usuario accede a esta pantalla sin ningún paso previo se le mostrará un listado de hoteles que se encuentren en la zona 1 de la ciudad de Guatemala y con cualquier monto de tarifa.

Si el usuario accede a esta aplicación por medio de cada paso, se aplicarán los filtros que el usuario seleccione en cada paso. El listado mostrará los siguientes datos por cada establecimiento:

- Nombre completo del establecimiento.
- Calificación del establecimiento. (por medio de estrellas promedio con base en retroalimentación del usuario, para poder realizar calificación el hotel debe proveer un *ticket* con número de autorización. Asignado a dicho establecimiento por RAGATEC).

Figura 18. **Pantalla listado establecimientos encontrados**



Fuente: elaboración propia, empleando Adobe Photoshop.

Como se puede ver en la figura 18, la pantalla de listado establecimientos posee un *table view*, donde se podrán ver todos los establecimientos que cumplen con los filtros seleccionados por el usuario. Al momento de presionar un establecimiento este mostrará la siguiente pantalla que son los datos completos del hotel y su galería de fotos.

Esta pantalla es de las más importantes ya que es en la que el usuario estará más tiempo para encontrar su establecimiento de hospedaje.

2.3.5.6. Pantalla datos de establecimiento

Es aquí donde el usuario podrá ver todos los datos de cada establecimiento, en esta pantalla se muestran los datos únicamente de un establecimiento y es del seleccionado en la pantalla de listados de establecimiento o bien de favoritos. Los datos que se mostrarán son los siguientes:

- Nombre del establecimiento (logo del mismo si posee).
- Ubicación, dirección exacta, mapa al establecimiento (ver mapa).
- Teléfonos del establecimiento. *Link* a sitio web.
- Ver tarifas, ver promociones, ver habitaciones.
- Pequeña galería fotográfica con 4 fotografías, entrada principal, parqueo si tiene, recepción y habitación modelo.

Figura 19. **Pantalla datos del establecimiento**



Fuente: elaboración propia, empleando Adobe Photoshop.

Como se puede ver se tendrá una estrella en esquina superior derecha, si el usuario la activa hará de ese establecimiento uno de sus favoritos.

A continuación se describe cada elemento de esta pantalla:

- Ver mapa (Ir A): esta opción al tocarla automáticamente mostrará un mapa de google respecto a la dirección del hotel. Dará la opción de dar que google maps guíe hacia dicha dirección.
- Ver habitaciones: esta opción mostrará la pantalla de listado de habitaciones que tiene en forma matricial las habitaciones del hotel, pudiendo seleccionar una para ver su información.
- Ver promociones: esta opción mostrará la pantalla de listado de promociones, como máximo establecimientos. podrán colocar los 2 promociones.

- Ver tarifas: esta opción abrirá un cuadro de diálogo con las tarifas del establecimiento.

Figura 20. **Mostrar tarifas en pantalla de datos**



Fuente: elaboración propia, empleando Adobe Photoshop.

- Estrella superior derecha: esta opción le permite al usuario agregar el establecimiento que esta viendo como favorito. Para luego acceder a favoritos directamente.

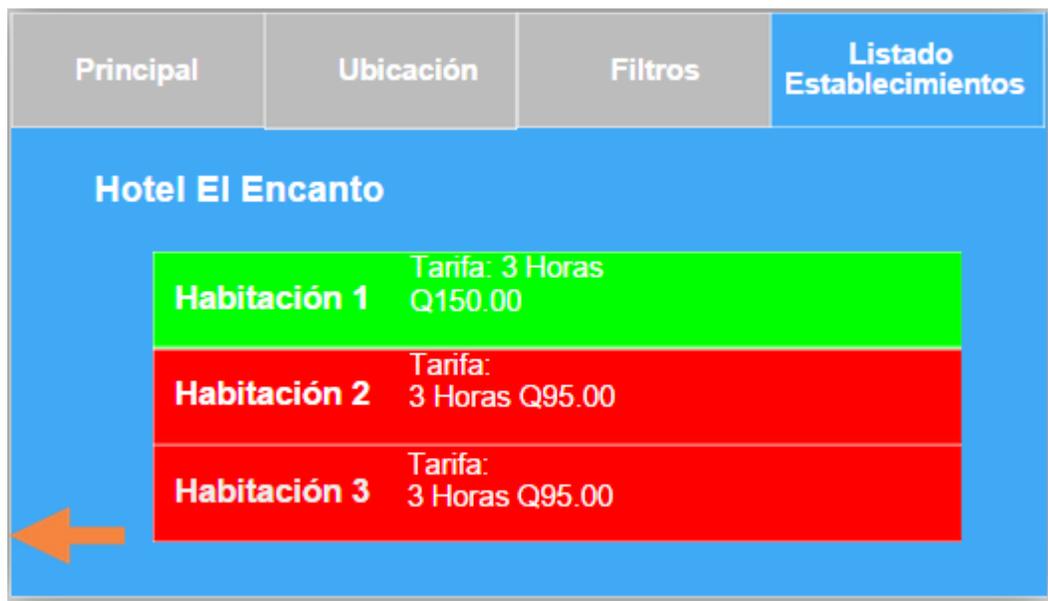
2.3.5.7. **Pantalla listado de habitaciones**

Esta pantalla muestra por número las habitaciones con las que cuenta el establecimiento. Esta pantalla tiene 2 características importantes:

- Muestra en color verde las habitaciones disponibles
- Muestra en color rojo las habitaciones ocupadas

Los elementos del *grid* son las habitaciones que podrán ser visualizadas por el usuario, todos los datos se muestran en el *grid*. Estos datos son tarifa y número de habitación.

Figura 21. **Pantalla listado de habitaciones**



Fuente: elaboración propia, empleando Adobe Photoshop.

Para poder ver fotos de las habitaciones se debe dar un toque encima de dicho elemento para que este muestre una pequeña galería de 3 fotos. A continuación se muestra como:

Figura 22. **Galería de fotos por habitación**



Fuente: elaboración propia, empleando Adobe Photoshop.

2.3.5.8. **Pantalla de favoritos**

Esta pantalla es bastante importante porque hará que la sobrecarga de información, es decir sobrecarga de hoteles, sea descartada ya que el usuario podrá elegir cuales son los hoteles que más frecuenta o bien más le interesan, es un acceso directo a los establecimientos favoritos del usuario.

Esta pantalla es igual al listado de hoteles presentado previamente, la diferencia es que son los establecimientos que el usuario configura. El usuario podrá eliminar los establecimientos de favoritos.

Figura 23. **Pantalla de favoritos**



Fuente: elaboración propia, empleando Adobe Photoshop.

2.3.5.9. **Pantalla de promociones**

Esta pantalla permitirá a los establecimientos de hospedaje mostrar ofertas o paquetes especiales en dicho establecimiento. Un ejemplo de estas promociones puede ser: dar desayuno gratis si se entra antes de las 11:00 horas.

Esta pantalla hará que la oferta del hotel sea más atractiva.

Figura 24. **Pantalla de promociones**



Fuente: elaboración propia, empleando Adobe Photoshop.

3. DISEÑO DE LA SOLUCIÓN

3.1. Base de datos

El almacenamiento de información es el primer componente que se debe diseñar. Es parte vital de el sistema informático. A continuación se describirá todo el proceso del diseño de la base de datos, basado en las necesidades del sistema.

3.1.1. Diagrama Entidad Relación

Para realizar el diseño de la base de datos como primer punto se va a definir las entidades de la base de datos.

3.1.1.1. Objetos principales de la base de datos

- Estado: esta entidad es bastante importante, es la que va a proveer el dato en qué estado se encuentra cualquier entidad de la base de datos. Se toma como entidad ya que se definirán "n" estados. Es importante conceptualizar al estado como un conjunto ya que todas las entidades tendrán un subconjunto de estados diferentes.
- País: entidad que representa el país al cual pertenece un usuario, pero como existen "n" usuarios pueden ser de "m" países, por esta razón se define esta entidad.

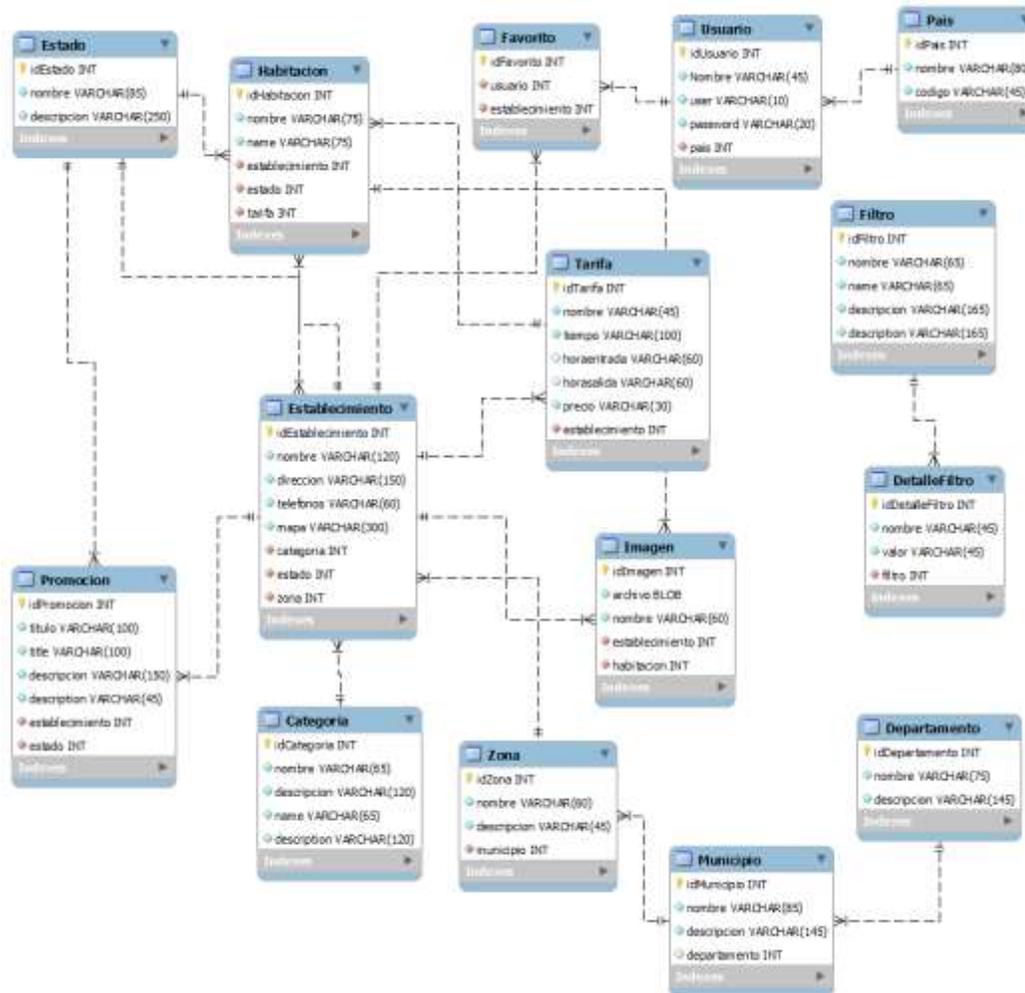
- Usuario: entidad que representa al usuario que ingresa a la aplicación guarda los datos necesarios únicamente.
- Establecimiento: es la entidad principal, el centro de la base de datos, esta entidad representa el establecimiento de hospedaje y todos sus datos.
- Categoría: entidad utilizada para categorizar los establecimientos.
- Habitación: entidad que pertenece a un establecimiento y representa el cuarto o habitación con la cual presta servicio el establecimiento, un establecimiento puede tener "n" habitaciones.
- Tarifa: entidad que representa el precio de las habitaciones y al conjunto de precios de un establecimiento, esta entidad posee datos característicos como los horarios por determinado precio.
- Promoción: dado que los establecimientos tendrán la oportunidad de crear promociones para que los usuarios las observen, esta entidad representará al conjunto de promociones de los diferentes hoteles.
- Imagen: entidad que representa a todas las imágenes que se encuentran en el servidor, útil para poder hacer referencia en establecimientos o habitaciones. Una imagen debe pertenecer a un establecimiento, debe tener un nombre con el cual sabrá dónde mostrarla y por último una habitación, si fuera imagen general la habitación será 0.

- Filtro: entidad utilizada para agregar los diferentes filtros al sistema, es posible que en el futuro se agreguen más que la tarifa, es por esto que se parametrizan los filtros.
- Detalle filtro: entidad para representar un elemento del conjunto de detalles de la entidad filtro.

3.1.1.2. Diagrama completo Entidades-Relaciones

A continuación se muestra el diagrama generado en base a las entidades definidas y las relaciones entre las mismas.

Figura 25. Diagrama entidad relación



Fuente: elaboración propia, empleando Microsoft Word.

3.2. Arquitectura del sistema

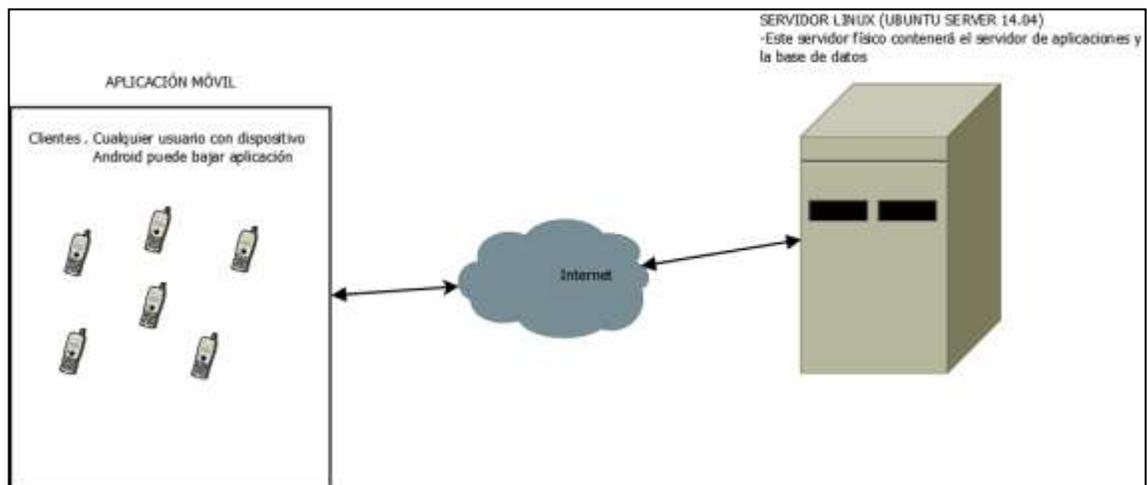
A continuación se mostrará el diseño de más alto nivel de la estructura de un sistema. Se definirá de manera abstracta los componentes lógicos y físicos.

3.2.1. Arquitectura física

La forma en la que se distribuyen los componentes físicos del sistema es la siguiente, los principales componentes son:

- Servidor único: este es una computadora con características técnicas suficientes para llevar a cabo las tareas de procesamiento solicitadas. Dentro de este equipo físico se implementará toda la lógica del sistema.
- Dispositivos móviles: son los dispositivos utilizados por los usuarios para acceder al sistema.

Figura 26. **Arquitectura física del sistema**



Fuente: elaboración propia, empleando Adobe Photoshop.

3.2.2. Arquitectura lógica del sistema

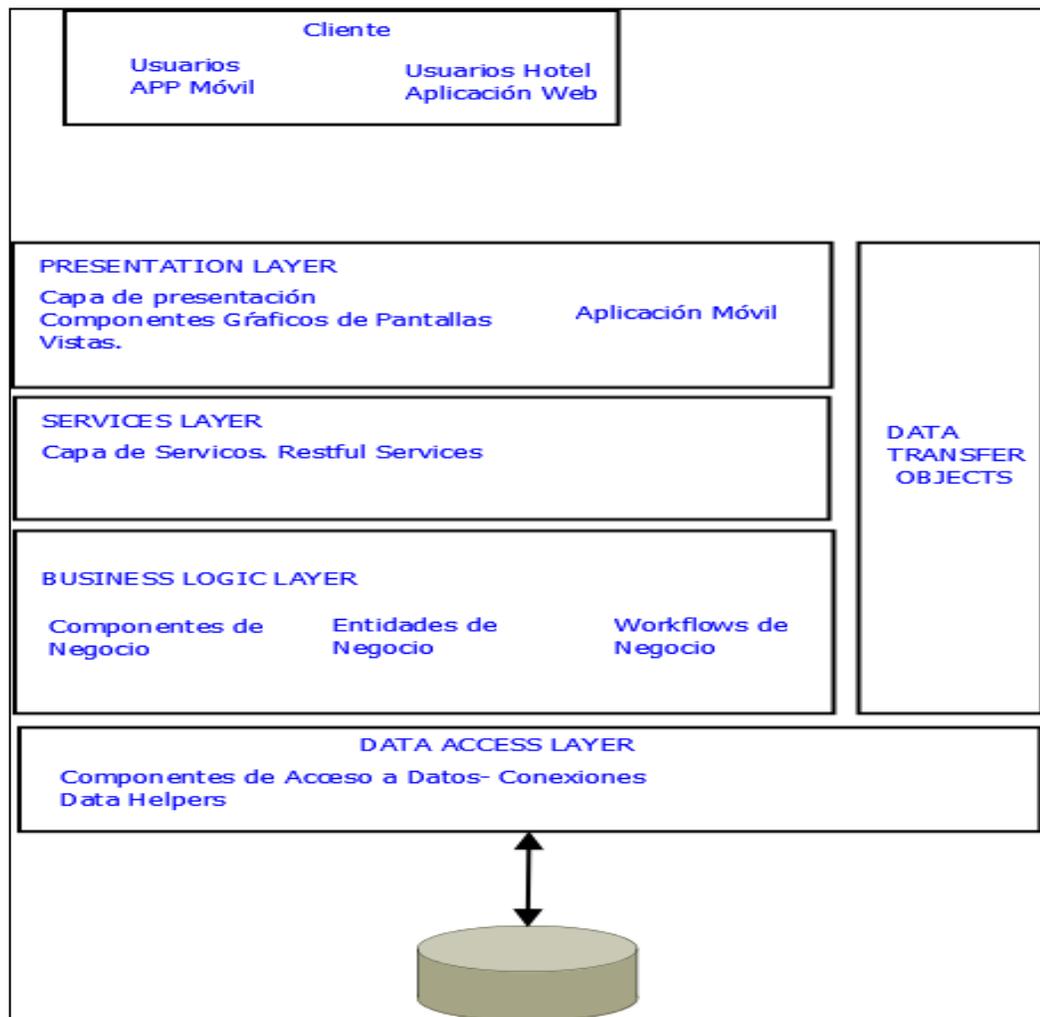
Para un buen desempeño y eficiencia del sistema se utilizará un diseño de 4 capas. Cada capa tiene funciones específicas dentro del sistema y es lo que hará que este sea más eficiente y seguro.

3.2.2.1. Capas del sistema

- DAL (Data Access Layer): esta es la capa de acceso a los datos, es la única capa que tiene acceso a los datos que se encuentran almacenados en la base de datos. Es aquí donde se realiza la conexión a la base de datos y tiene acceso a la misma con privilegios altos. Ninguna capa del sistema puede acceder a la base de datos si no utiliza esta capa.
- DTO (Data Transfer Object): estos son el conjunto de objetos que transporta datos entre los procesos. Se utilizan POJO para realizar esta tarea ya que se utilizará Hibernate. Este conjunto de objetos son los que viajan entre componentes conteniendo información necesaria, se necesita que sean objetos ya que la comunicación se llevará a cabo por medio de formato JSON y dichos objetos son convertidos al formato.
- BLL (Business Logic Layer): esta capa es la encargada de realizar todos los procesos, esta capa se encarga de validaciones, correcciones y CRUD a nivel lógico de todas las entidades. Esta capa no tiene acceso a la base de datos directamente, únicamente a los DTO. Es a esta capa a donde debe de acceder cualquier UI o servicio que necesite de datos o bien de validaciones o procesos.

- Servicios: capa implementada con RESTFul Services java, es la capa que expone y publica los recursos de la base de datos, en esta capa se tiene acceso únicamente al BLL.
- Capa de presentación: esta capa se ve representada por UI Web php y UI Móvil.

Figura 27. Diagrama de capas del sistema

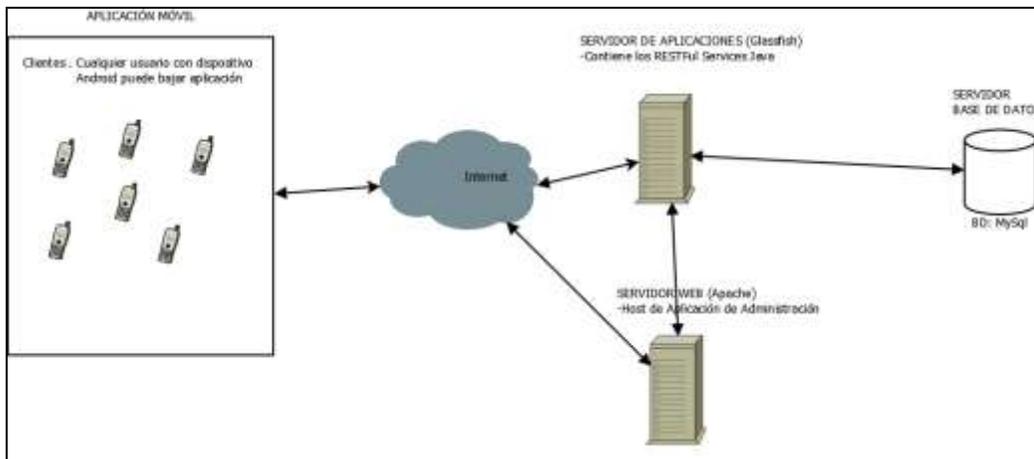


Fuente: elaboración propia, empleando Microsoft Word.

3.2.2.2. Diagrama de comunicación lógica de componentes

Ya definidas las capas del sistema, se debe conocer la forma en la que se van a almacenar en los distintos servidores.

Figura 28. Arquitectura lógica del sistema



Fuente: elaboración propia, empleando Adobe Photoshop.

Se pueden identificar los componentes principales del sistema y donde estarán almacenados estos mismos. Aunque se vea en el diagrama 3 componentes, se debe recordar que los tres servidores estarán en un mismo servidor físico.

3.3. Diseño de comunicación entre componentes

A continuación se describirán las interfaces, formatos y medios utilizados para comunicación entre la aplicación móvil y aplicación web, de igual forma se mostrará explícitamente los servicios a utilizar para llevar a cabo esta tarea.

3.3.1. Descripción de RestFul Services

Por cada entidad que se tiene, se tendrá un RestFul Service que debe implementar. Cada servicio debe tener como nombre el nombre del recursoSv.

Ejemplo:

Recurso: establecimiento

RESTFul Service: establecimiento Sv

Todos los servicios que se implementen en el sistema de hotelería, deben tener como mínimo los siguientes métodos:

- POST: para crear un recurso en el servidor
- GET: para obtener un recurso o varios
- PUT: para cambiar estado de un recurso o bien modificarlo
- DELETE: para eliminar el recurso

3.3.2. Listado y significado de servicios a utilizar

A continuación se detallan los servicios que serán implementados para hacer disponibles los recursos de información que se encuentran en el servidor de base de datos.

- Estado Sv: servicio que proveerá el acceso al recurso "Estado" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).

- HabitaciónSv: servicio que proveerá el acceso al recurso "Habitación" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- PromocionSv: servicio que proveerá el acceso al recurso "Promoción" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- FavoritoSv: servicio que proveerá el acceso al recurso "Favorito" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- UsuarioSv: servicio que proveerá el acceso al recurso "Usuario" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- PaisSv: servicio que proveerá el acceso al recurso "País" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- EstablecimientoSv: servicio que proveerá el acceso al recurso "Establecimiento" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).

- CategoríaSv: servicio que proveerá el acceso al recurso "Categoría" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- TarifaSv: servicio que proveerá el acceso al recurso "Tarifa" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- ImagenSv: servicio que proveerá el acceso al recurso "Imagen" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- FiltroSv: servicio que proveerá el acceso al recurso "Filtro" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).
- DetalleFiltroSv: servicio que proveerá el acceso al recurso "Detalle Filtro" de la base de datos. Se implementan como mínimo los 4 métodos básicos para el recurso, Obtener (GET), Crear recurso (POST), Actualizar (PUT), Eliminar (DELETE).

Cualquier comunicación entre base de datos y el exterior debe ser por medio de estos servicios. Ya que estos están en la cima de las capas del sistema. Al referirse al exterior se hace referencia a:

- Aplicación web de administración
- Aplicación móvil

3.4. Usabilidad

Cuando se diseña una interfaz de usuario para una aplicación destinada a un dispositivo móvil, no basta con buscar que la aplicación sea visualmente atractiva, es sumamente importante que se tome en cuenta cómo va a ser usada la aplicación. En el contexto de aplicaciones para dispositivos móviles, la usabilidad incluye la forma en que se presentan los mensajes, colores adecuados, el soporte para distintas pantallas, las fuentes y tipo de letra, las imágenes y gráficas. Es importante incorporar características propias de los dispositivos móviles como el acelerómetro, giroscopio o bien el GPS. Esta aplicación utiliza el GPS para obtener la ubicación del usuario.

3.4.1. Mensajes

Es importante tener mensajes con el objetivo de orientar al usuario y minimizar los errores que se cometen durante el uso de la aplicación, se muestran mensajes que constan en su mayoría de Tooltip o bien ventanas emergentes que se muestran cuando un evento lo amerite.

3.4.2. Características de los dispositivos móviles

Los dispositivos móviles ofrecen una gran variedad de características que pueden ayudar a mejorar la usabilidad de las aplicaciones, sin embargo, estas características pueden variar dependiendo de la marca de fábrica del dispositivo. Contenidas en estas características se encuentran el sistema de

posicionamiento global (GPS), recepción de radiofrecuencia (RD), vibrador, cámara digital, acelerómetro y giroscopio.

Para la aplicación de hoteles de Guatemala se ha decidido hacer uso de las siguientes características:

- GPS: Global Positional System, en el paso 2 del flujo para buscar hoteles en Guatemala se encuentra la opción "Encontrar Aquí" que es la que utiliza esta característica del dispositivo para ubicarlo geográficamente en Google maps. Es importante que se sepa que el GPS únicamente es el que da las coordenadas del dispositivo y el Google Maps el que se encarga de la interfaz gráfica donde se verá la ubicación en cada diferencial de tiempo.

3.5. Diseño de la aplicación de administración utilizando DaDaBIK

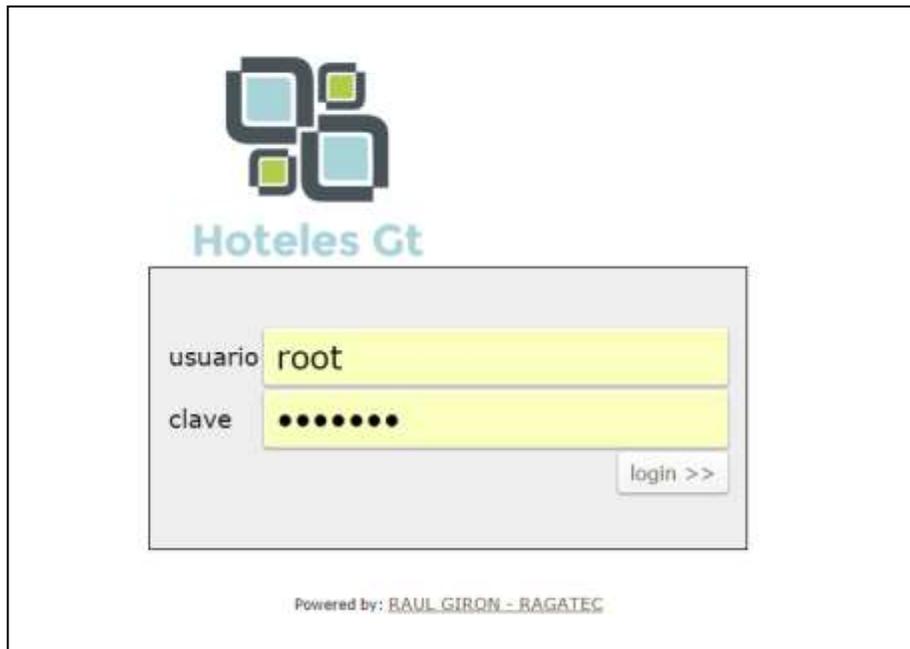
Esta aplicación tiene un diseño estándar, a continuación se muestra la forma en que se muestra la información y se realizan las tareas necesarias.

3.5.1. Log In

Esta pantalla es la que se asegura que solo los usuarios autenticados ingresen al sistema, con base en usuario ingresado le va a mostrar los registros de habitaciones ya que son "n" hoteles. Actualmente solo está implementado en un establecimiento. El usuario y *password* es el siguiente:

- Usuario: casanovagt
- Password: casanovaturno

Figura 29. **Pantalla de Log In Administración Hotel**



Fuente: elaboración personal, empleando Adobe Photoshop.

3.5.2. Visualización de habitaciones

Todas la visualizaciones se realizan de la misma forma, se debe dar *click* en el boton "Ver Habitaciones", en el caso del usuario en turno únicamente le aparece habitaciones ya que solo a esto tiene acceso.

Figura 30. **Pantalla inicial de sistema administración**



Fuente: elaboración propia, empleando Adobe Photoshop.

Figura 31. **Pantalla con listado de habitaciones**



Fuente: elaboración propia, empleando Adobe Photoshop.

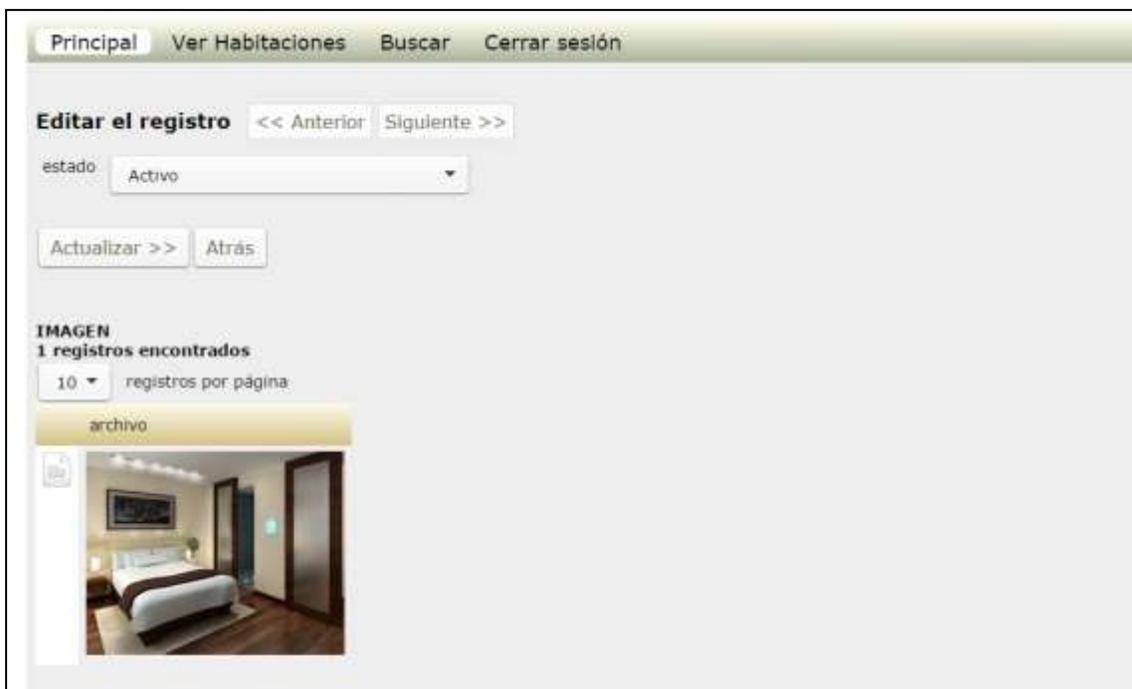
Como se puedes ver se le muestran todas las habitaciones del hotel en el que se logueó, por lo cual puede ver 3 datos en el caso de autohotel Casanova.

Como se puede ver muestra el nombre de la habitación con su número este elemento es bastante importante, cuando se implemente el sistema en un establecimiento deben nombrarse las habitaciones, es un requerimiento mínimo para poder implementarlo.

3.5.3. Cambiar estado a habitaciones

Esta es la función principal del usuario en turno, cambiar el estado, esto se logra dándole *click* al lápiz que se encuentra del lado izquierdo de la aplicación y podrá modificar únicamente el estado de la habitación como se muestra en la siguiente figura.

Figura 32. Pantalla cambio de estados habitaciones



Fuente: elaboración propia, empleando Adobe Photoshop.

4. DOCUMENTACIÓN DEL SISTEMA

A continuación una documentación de cómo utilizar el sistema.

4.1. Manual de usuario aplicación administración

La aplicación puede accederse en la siguiente dirección: <http://adminhotel.dnsd.info/>

4.1.1. Proceso de cambiar estado a habitación

- Acceder con su usuario al sistema

Figura 33. Paso 1. Cambiar estado habitación



Fuente: *Administración*. <http://adminhotel.dnsd.info/>. Consulta: abril de 2015.

- Click en "Ver Habitaciones"

Figura 34. Paso 2. Cambiar de estado habitación



Fuente: elaboración propia, empleando Adobe Photoshop.

- A continuación se le mostrarán las habitaciones de su establecimiento

Figura 35. Paso 3. Cambiar de estado habitación



Fuente: elaboración propia, empleando Adobe Photoshop.

En este paso debe darle *click* en el lápiz a la habitación que desea cambiar de estado.

- Al momento de dar *click* en el lápiz mostrará la siguiente pantalla:

Figura 36. **Paso 4. Cambio de estado habitación**



Fuente: elaboración propia, empleando Adobe Photoshop.

En esta pantalla debe seleccionar el estado deseado del dropdown list. y para finalizar *click* en Actualizar.

4.2. Manual del usuario aplicación android

Para acceder al prototipo funcional de la aplicación favor entrar al siguiente link: https://www.fluidui.com/editor/live/preview/p_1q1IE8nG0SXw2Azd9dCC6KLUBVPSJCzx.1426975399565.

4.2.1. Proceso de búsqueda de hotel

Para este proceso, es un flujo simple y básico, alterable en cualquier momento sin que la aplicación se interrumpa. El flujo básico se muestra a continuación:

- Como primer paso se debe seleccionar la categoría de establecimiento en la pantalla Principal.

Figura 37. Selección categoría de establecimiento



Fuente: elaboración propia, empleando Adobe Photoshop.

- Una vez seleccionada la categoría de establecimiento la aplicación automáticamente guiará al siguiente paso, que es Ingresar la Ubicación, para esto se mostrará la siguiente pantalla.

Figura 38. **Ingreso de ubicación deseada**



Fuente: elaboración propia, empleando Adobe Photoshop.

En esta pantalla usted puede seleccionar cualquiera de las dos opciones:

- Encontrar Cerca: utilizando el GPS de su dispositivo la aplicación buscará su ubicación actual y con base en esta realizará la búsqueda.
- Ingresar la ubicación con una cascada de selección, primero el departamento con base en esto se cargarán municipios y en base a estos las áreas o zonas.

Una vez seleccionada cualquier opción la aplicación automáticamente guiará al siguiente paso que es utilizar filtros.

- Como tercer paso, una vez seleccionada la ubicación, se debe seleccionar un filtro si se desea, este paso es opcional. Sin embargo, si no se desea usar únicamente pasar a la siguiente pantalla.

Figura 39. **Selección de filtros**



Fuente: elaboración propia, empleando Adobe Photoshop.

Como primer versión únicamente se puede filtrar por tarifas, si se desea filtrar por alguno de los valores mostrados puede realizarse, sin embargo, este paso es totalmente opcional, se puede continuar el flujo básico de la aplicación.

Una vez seleccionado el filtro o bien únicamente continuar sin elegir nada, la aplicación mostrará la pantalla con el listado de establecimientos con base en todo lo que ya ingresó con anterioridad.

- Visualizar el listado de establecimientos que cumplen los filtros que se desean y seleccionar la que se quiera.

Figura 40. **Listado de establecimientos aplicación**



Fuente: elaboración propia, empleando Adobe Photoshop.

Si se conoce el nombre se puede ingresar en el campo de búsqueda de lo contrario navegar utilizando su pantalla *touch* a través de los diferentes establecimientos y para seleccionar uno únicamente un *double top*. Al seleccionarlo se visualizarán las siguientes opciones:

Figura 41. **Pantalla datos del establecimiento en aplicación**



Fuente: elaboración propia, empleando Adobe Photoshop.

En esta pantalla se podrán visualizar todos los datos del hotel y si se desea información específica, se puede lograr dando *click* en cada una de las opciones que allí se muestran.

- El siguiente paso que se describe es opcional únicamente si se desea visualizar las habitaciones del establecimiento y ver cuales están ocupadas y cuáles no. Se debe dar *click* en Ver Habitaciones y se le mostrará la siguiente pantalla.

Figura 42. **Pantalla con listado de habitaciones**



Fuente: elaboración propia, empleando Adobe Photoshop.

De esta forma se podrán visualizar las habitaciones, las que se encuentran con fondo verde están disponibles y las que se encuentran en rojo están ocupadas. Para ver una pequeña galería de fotos se puede dar *top* o doble *top* y se mostrarán 3 fotos de dicha habitación.

Figura 43. **Habitación galería**



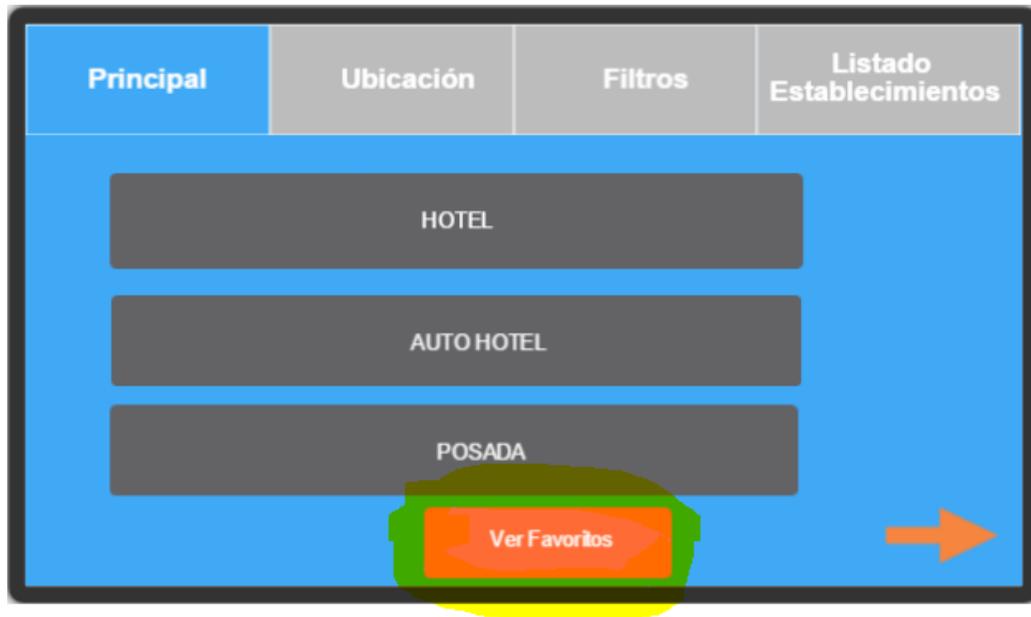
Fuente: elaboración propia, empleando Adobe Photoshop.

Aquí se podrán visualizar las 3 fotos de la habitación. Si se lo desea se puede regresar o presionar cualquier acción botón del Action Bar.

4.2.2. Ver favoritos

Este paso es importante ya que comúnmente los usuarios tienen un hotel o establecimiento que es común visitar, es por esto que desde cualquier pantalla excepto Habitaciones y Listado de Hoteles, se podrá acceder a los favoritos, a continuación se muestra donde está el botón de ver favoritos.

Figura 44. **Pantalla cualquiera con botón favoritos**



Fuente: elaboración propia, empleando Adobe Photoshop.

Todas las pantallas poseen en la misma posición el botón de favoritos, para acceder a dicha pantalla en cualquier momento. Esto se hace para crear una fidelización de los clientes hacia los establecimientos que ya han visitado, este fenómeno de fidelización es con el cual se le hará ver a muchos establecimientos que deben estar dentro de la base de datos de la aplicación.

Al momento de seleccionar la opción favoritos desde cualquier pantalla se mostrará el siguiente listado de hoteles que ya se han hecho favoritos.

Figura 45. **Visualización de favoritos**



Fuente: elaboración propia, empleando Adobe Photoshop.

En esta pantalla se mostrarán los hoteles hechos favoritos por el usuario sin importar si tienen ubicaciones cercanas o muy lejanas, estas estarán juntas sin opciones a filtro.

También se muestra un ícono de "Quitar" al lado izquierdo de cada establecimiento, ya que únicamente se puede tener 10 favoritos el usuario podrá administrar estos quitándolos.

4.2.3. Crear favorito

Dado que muchas personas utilizan los servicios del mismo establecimiento muchas veces, el estar buscándolo nuevamente será un proceso demasiado tedioso y hasta aburrido, pudiendo hacer que el usuario no

desea utilizar la aplicación. Es por esto que se incluye en la pantalla de datos del hotel una estrella del lado derecho.

Al momento de dar *click* en la estrella esta cambiará de color y el establecimiento se convertirá en favorito para dicho usuario.

A continuación se muestra la estrella donde se debe dar un *top* o *double top*.

Figura 46. **Pantalla crear favorito**



Fuente: elaboración propia, empleando Adobe Photoshop.

4.2.4. **Uso del Action Bar**

El Action Bar está presente en todo momento en la aplicación, esto es porque sus opciones puede ser seleccionadas en cualquier momento y son establecidas para mostrar el listado de establecimientos.

Este Action Bar muestra en que paso se está del flujo principal de búsqueda, pero como se mencionó con anterioridad no es necesario hacerlo en orden, siempre y cuando se vea el Action Bar, se podrán presionar su opción y al momento de regresar o presionar por primera vez la opción "Listado de Establecimientos" esta será aplicada.

Figura 47. **Action Bar de App**



Fuente: elaboración propia, empleando Adobe Photoshop.

CONCLUSIONES

1. La aplicación "Hoteles Gt" es una herramienta que puede ser utilizada por cualquier persona que tenga un dispositivo Android y siempre tenga conexión a internet.
2. Todos los datos de los establecimientos que estén registradas dentro del sistema de "Hoteles Gt" son datos verídicos que estarán en constante actualización y verificación.
3. Guatemala como un país con tasa alta de turismo necesitaba este tipo de herramienta, que podrá ser utilizado por extranjeros que vengan a visitar cualquiera de los hermosos lugares turísticos del país.
4. Actualmente, se está en la era de la información, y toda la información incluyendo la de establecimientos de hospedaje debe ser pública y gratuita, accesible por cualquier persona.

RECOMENDACIONES

1. Si se desea tener un buen rendimiento de la aplicación en su dispositivo debe poseer un navegador web actualizado, ya que la aplicación corre sobre dicho proceso a pesar de ser un apk.
2. Si se desea que el establecimiento sea agregado a la base de datos del sistema "Hoteles Gt" ponerse en contacto con: Raúl Girón, celular 5608-0695.
3. Descargar la aplicación de la tienda Kunustore, mientras más usuarios tengan dicha aplicación, más retroalimentación de la misma y más establecimientos querrán formar parte del sistema.

BIBLIOGRAFÍA

1. Colaboradores de GitHub. *Phonegap*. [en línea]. <<https://github.com/phonegap/phonegap/wiki>>. [Consulta: 2 de marzo de 2015].
2. Colaboradores de librosweb.es. *CSS*. [en línea]. <http://librosweb.es/libro/css/capitulo_1.html>. [Consulta: 8 de marzo de 2015].
3. Colaboradores PhoneGap. *PhoneGap Documentation*. [en línea]. <http://docs.phonegap.com/en/edge/guide_overview_index.md.html#Overview>. [Consulta: 12 de marzo de 2015].
4. NISHANT, Vicky. *Returning JSON Object from REST Web Service with Complex Objects*. [en línea]. <<http://stackoverflow.com/questions/11881246/returning-json-object-from-rest-web-service-with-complex-objects>>. [Consulta: 8 de marzo de 2015].
5. RAM, Nelson. *Introducción a los servicios RESTful*. [en línea]. <<http://www.dosideas.com/noticias/java/314-introduccion-a-los-servicios-web-restful.html>>. [Consulta: 5 de marzo de 2015].
6. SILVA, Diego. *RESTful... la forma más ligera de hacer WebServices*. [en línea]. <<http://www.apuntesdejava.com/2010/11/restful-la-forma-mas-ligera-de-hacer.html>>. [Consulta: 5 de marzo de 2015].
7. TACCHINI, Eugenio. *DaDaBIK*. [en línea]. <<http://www.dadabik.org/index.php>>. [Consulta: 10 de marzo de 2015].

8. Wikipedia. *Framework*. [en línea]. <<http://es.wikipedia.org/w/index.php?title=Framework&oldid=80876102>>. [Consulta: 10 de marzo de 2015].

9. _____ *Socket de Internet*. [en línea]. <http://es.wikipedia.org/w/index.php?title=Socket_de_Internet&oldid=80242122>. [Consulta: 5 de marzo de 2015].