



UNIVERSIDAD JOSÉ ANTONIO PÁEZ

**APLICACIÓN WEB Y MÓVIL PARA EL
CONTROL Y SEGUIMIENTO DE
CONSUMOS Y SERVICIOS DE LOS
CLIENTES DEL HOTEL GARDEN
VALENCIA EN EL ESTADO
CARABOBO.**

Autor: Manríquez Arturo
C.I. V- 16.595.864

Mendoza Wilmer
C.I. V- 20.292.080



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN COMPUTACIÓN**

**APLICACIÓN WEB Y MÓVIL PARA EL CONTROL Y
SEGUIMIENTO DE CONSUMOS Y SERVICIOS DE LOS
CLIENTES DEL HOTEL GARDEN VALENCIA EN EL ESTADO
CARABOBO.**

**Trabajo de grado presentado como requisito para optar al título de
INGENIERO DE COMPUTACIÓN.**

Autores: Manríquez, Arturo

C.I. V-16.595.864

Mendoza, Wilmer:

C.I. V-20.292.080

Tutor: Ing. Requena, Ifigenia

San Diego, Juniode 2017



Universidad José Antonio Páez
Facultad de Ingeniería

FI-TG-2017-1CR-054

Valencia, 13 de Enero de 2017.

Ciudadanos:

Mendoza Wilmer

C.I. 20.292.080

Manriquez Arturo

C.I. 16.595.864

Presente.-

Cumplo con informarle que la Comisión de Trabajo de Grado y Pasantías de la Facultad de Ingeniería en su reunión N° 1-2017 de fecha 13/01/2017 aprobó el proyecto de trabajo de grado titulado **“APLICACIÓN WEB Y MOVIL PARA EL CONTROL Y SEGUIMIENTO DE CONSUMOS Y SERVICIOS DE LOS CLIENTES DEL HOTEL GARDEN VALENCIA EN EL ESTADO CARABOBO.”** presentado por ustedes como requisito para optar al título de Ingeniero en Computación.

Se ratifica la designación de la Ing. Ifigenia Requena, C.I. 10.543.541 y la Ing. Alicia Pizzella, C.I. 4.598.880 como Tutotes Académicos que lo asesorarán en el desarrollo de este proyecto.

Atentamente.



Prof. Marlene Zambrano
Decana (Encargada) de la Facultad de Ingeniería
(CU502 de fecha 11/10/2016)



c. c. Coordinación de Pasantías y Trabajo de Grado (2).
Archivo.

MEZ:jp



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN COMPUTACIÓN**

ACEPTACIÓN DEL TUTOR

Quien suscribe, Ingeniero Ifigenia Requena, portador(a) de la cédula de identidad N° 10.543.541, en mi carácter de tutor del trabajo de grado presentado por el(los) ciudadano(s) Manriquez Arturo y Mendoza Wilmer, portador(es) de la cédula de identidad N° 16.595.864 y 20.292.080, (respectivamente), titulado: **“APLICACIÓN WEB Y MOVIL PARA EL CONTROL Y SEGUIMIENTO DE CONSUMOS Y SERVICIOS DE LOS CLIENTES DEL HOTEL GARDEN VALENCIA EN EL ESTADO CARABOBO”**, presentado como requisito parcial para optar al título de Ingeniero en Computación, considero que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del jurado examinador que se designe.

En San Diego, a los 08 días del mes de Junio del año dos mil diecisiete.

Ifigenia Requena Negrón de Guacarán
C.I.: 10.543.541

AGRADECIMIENTOS

Agradezco a Dios por haberme mantenido firme y fuerte en los momentos ms difíciles de esta meta que decidí alcanzar.

Agradezco a mis padres y hermana por haberme apoyado en todo momento, por nunca perder la fe en mí y apostar siempre el éxito de mi carrera universitaria.

De igual manera agradezco a mi esposa e hijos por ser mis motores en todo momento, la razón por la cual seguir luchando y tratando de ser mejor cada día

A la Universidad José Antonio Páez por brindarme la oportunidad de modelar mi conocimiento para ser un profesional, así como deseo agradecer cordialmente a todos mis profesores durante el transcurso de mi carrera como estudiante de la institución

Agradezco por igual a nuestra tutora de trabajo de grado, Ing. Ifigenia Requena, por su cariño, esfuerzo, comprensión, orientación, y paciencia; quien aportó sus conocimientos, sus experiencias y su carisma, para lograr la culminación de este proyecto de estudios con éxito.

A mi compañero de TEG Wilmer Mendoza y su Hermano Wilfredo Mendozapor mostrar gran profesionalismo en los momentos requeridos. En general a todos mis compañeros de promoción y de clases, por creer en mí, y haberme brindado su apoyo durante esta carrera. ¡Gracias muchachos!

ARTURO MANRIQUEZ

AGRADECIMIENTOS

Primeramente, a Dios por haberme permitido llegar hasta este punto y haberme dado la fortalezanecesaria para poder cumplir con esta meta

Le doy gracias a mi madre, padre y hermanos por haberme apoyado en todo momento, por sus consejos, sus valores, por la motivación constante que me ha permitido ser una persona de bien, pero más que nada, por su amor.

A la Universidad José Antonio Páez por brindarme la oportunidad de modelar mi conocimiento para ser un profesional, así como también deseo agradecer cordialmente a todos mis profesores durante el transcurso de mi carrera como estudiante de la institución.

Agradezco por igual la confianza, apoyo y dedicación de nuestra tutora de TEG, Ing. Ifigenia Requena, por su esfuerzo, comprensión, orientación, y paciencia; quien aportó sus conocimientos, sus experiencias y su carisma, para lograr culminar este proyecto de estudios con éxito.

A mi compañero de trabajo de grado, Arturo Manríquez por su constancia, dedicación y compromiso para poder culminar con éxito este proyecto y en general a todos los compañeros con quien compartí clases, por creer en mí, y haberme brindado su apoyo durante esta carrera.

WILMER MENDOZA

DEDICATORIA

A mis padres que me apoyaron en todo momento, además de ser mi soporte y mi guía en cada una de mis decisiones.

A mis hermanos por estar siempre conmigo y brindarme apoyo en todo momento

A mis familiares que me han apoyado y animado en todo momento, dándome el impulso necesario para llegar hasta este punto.

A mi tutora.

Ing. Ifigenia E. Requena por haber sido factor clave en el logro de esta meta y por su apoyo incondicional a lo largo de este trayecto.

WILMER MENDOZA

DEDICATORIA

ADios.

Por haberme permitido llegar hasta este punto y haberme dado salud para lograr mis objetivos, además de su infinita bondad y amor.

A mis padres y hermana.

Por haberme apoyado en todo momento, por sus consejos, sus valores, por la motivación constante que me ha permitido ser una persona de bien, pero más que nada, por su amor incondicional.

A mi esposa.

Por haber sido pilar fundamental en la realización de este trabajo al darme fuerzas cada vez que me sentía abatido, por sus palabras de aliento cuando no encontraba una solución aparente, por sus consejos siempre pertinentes, por estar y cuidar de nuestros hermosos hijos el tiempo que estuve ausente preparando este trabajo.

A mis hijos.

Les dedico especialmente este trabajo a David Alejandro Manriquez y Melany Sofía Manriquez. Ustedes son mi fuerza, mi aliento, mi motivo, mi razón, el porqué de lo que hago. Siempre me esforzaré para ser su mejor ejemplo hijos, los amo.

A mis familiares.

A mi tía Lourdes, a mi tío Nelson, a mi tío José Luis, a mi primo Jean Carlos y a todos aquellos que participaron directa o indirectamente en la elaboración de este trabajo.

A mis profesores.

Ing. Ifigenia E. Requena, por su gran apoyo y motivación para la culminación de nuestros estudios profesionales y para la elaboración de este trabajo, así como también a todos aquellos que participaron directa o indirectamente en la consecución de esta meta.

A mis amigos.

Que nos apoyamos mutuamente en nuestra formación profesional y que, hasta la fecha, seguimos siendo amigos: Wilmer Mendoza, Miguelangel Hurtado, Francesco de Benedetti, Luis Gonzales, Alejandro Conde, Luis López y Giuseppe Vetri.

ARTURO MANRIQUEZ

INDICE GENERAL	
CONTENIDO	Pág.
INDICE DE TABLAS.....	xii
INDICE FIGURAS	xiii
RESUMEN.....	xiii
INTRODUCCIÓN	1
I EL PROBLEMA.....	3
1.1 Planteamiento del Problema.....	3
1.2 Formulación del problema	5
1.3 Objetivos de la Investigación	5
1.3.2 Objetivos Específicos.....	6
1.4 Justificación de la Investigación	6
1.5 Alcance de la Investigación	7
II MARCO TEORICO	8
2.1 Antecedentes	8
2.2 Bases Teóricas.....	9
2.2.1 Sistemas de Información	9
2.2.2 Metodología de desarrollo de software	11
2.2.3 SCRUM.....	12
2.2.4 Frameworks en el desarrollo de software.....	14
2.2.5 Laravel	15
2.3 Definición de Términos Básicos	16
III MARCO METOLOGICO	19
3.1 Tipo de Investigación	19
3.2 Diseño de la Investigación	19
3.3 Nivel de Investigación.....	21
3.4 Población y Muestra.....	21
3.5 Técnicas e instrumentos de recolección de datos.....	21
3.6 Fases Metodológicas	22
IV RESULTADOS	25
4.1 Fase I: Diagnóstico de la situación actual en el Hotel Garden Valencia	25
4.1.1. Actividad I: Instrumentos de recolección de datos.	25
4.1.2. Actividad II: Aplicación de los instrumentos de recolección de datos.	26
4.2 Fase II: Determinación del Funcional de la Aplicación Web y Móvil	28
4.2.1. Actividad I: Definición de requerimientos de funcionabilidad.....	28
4.3 Fase III: Diseño de Módulos de Programación de la Aplicación.....	30
4.3.1 Actividad I: Modelado de casos de uso.....	30
4.3.2. Actividad II: Descripción de casos de uso.	33

4.3.3. Actividad III: Modelo entidad-relación de la base de datos.	45
4.3.4. Actividad IV: Modelo lógico de la base de datos	47
4.3.5. Actividad V: Descripción de la arquitectura del sistema	47
4.3.6. Actividad VI: Diseño de interfaces sistema Web y Móvil.....	48
4.3.7 Actividad VII: Generación del Mapa de Navegación.	49
4.4 Fase IV: Desarrollo	50
4.5. Fase V: Pruebas.....	58
V CONCLUSIONES Y RECOMENDACIONES.....	67
5.1 Conclusiones.....	67
5.2 Recomendaciones.....	68

ÍNDICE DE TABLAS

TABLA	Pág.
Cuadro 1. Guía de Entrevista	26
Cuadro 2. Guía de Entrevista	27
Cuadro 3. Requerimientos Funcionales del Sistema Web y Móvil	29
Cuadro 4. Requerimientos No Funcionales del Sistema Web y Móvil	30
Cuadro 5. Actores Vs Acciones en el Sistema	31
Cuadro 6. Iniciar Sesión	33
Cuadro 7. Administrar Empleados.	33
Cuadro 8. Administrar Clientes.	35
Cuadro 9. Gestionar Clientes.	35
Cuadro 10. Administrar Reservación.	36
Cuadro 11. Gestionar Reservación.	37
Cuadro 12. Administrar Consumos.	38
Cuadro 13. Gestionar Consumos.	39
Cuadro 14. Administrar Mantenimientos.	40
Cuadro 15. Gestionar Mantenimientos.	41
Cuadro 16. Generar Reportes.	42
Cuadro 17. Ver Reservación.	43
Cuadro 18. Ver Consumos.	44
Cuadro 19. Prueba: Ver Reservación.	58
Cuadro 20. Prueba: Registrar Reservación.	59
Cuadro 21. Prueba: Modificar Reservación.	60
Cuadro 22. Prueba: Eliminar Reservación.	61
Cuadro 23. Prueba: Ingreso de Cliente.	61
Cuadro 24. Prueba: Modificación de información de cliente.	62
Cuadro 25. Prueba: Eliminación de información de cliente.	63
Cuadro 26. Ingreso al sistema administrativo.	63
Cuadro 27. Prueba: Inicio de sesión desde la aplicación móvil.	64
Cuadro 28. . Prueba: Edición de orden de servicio desde la app. móvil.	65
Cuadro 29. Prueba: Creación de orden de consumo desde la app. móvil.	66
Cuadro 30. Prueba: Borrado de orden de consumo desde la app. móvil.	67

ÍNDICE FIGURAS

FIGURA	Pág.
Figura 1. Caso de Uso: Administrador.	
Figura 2. Caso de Uso: Empleado.	
Figura 3. Caso de Uso: Cliente.	
Figura 4. Modelo Entidad-Relación.	
Figura 5. Modelo Lógico de la base de datos.	
Figura 6. Arquitectura de Software.	
Figura 7. Mapa de Navegación.	
Figura 8. Página Inicio de Sesión.	
Figura 9. Inicio del área administrativa.	
Figura 10. Vista de Manejo de Reservaciones.	
Figura 11. Formulario de Registro de Empleados	
Figura 12. Formulario de Registro de Clientes.	
Figura 13. Formulario de Ingreso al Área de Clientes.	
Figura 14. Vista de menú principal de sistema móvil.	
Figura 15. Lista de Manejo de Ordenes de Servicio.	
Figura 16. Vista de Edición de Ordenes de Consumo Sistema Móvil.	
Figura 17. Vista de Nueva Orden de Consumo Sistema Móvil.	
Figura 18. Lista de manejo de Ordenes de Consumo Sistema Móvil.	
Figura 19. Vista de Fin de Sesión del Sistema Móvil.	



**REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD JOSÉ ANTONIO PÁEZ
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN COMPUTACIÓN**

**APLICACIÓN WEB Y MÓVIL PARA EL CONTROL Y
SEGUIMIENTO DE CONSUMOS Y SERVICIOS DE LOS
CLIENTES DEL HOTEL GARDEN VALENCIA EN EL ESTADO
CARABOBO.**

Autor: Manríquez, Arturo y Mendoza, Wilmer.

Tutor: Ing. Ifigenia Requena.

Fecha: Junio 2017.

RESUMEN

El presente estudio planteó como objetivo general, el desarrollo de una aplicación web y móvil para el control y seguimiento de consumos y servicios de los clientes del Hotel Garden Valencia en el estado Carabobo. Las faltas de controles en los procesos previamente descritos generaban conflictos de tipo administrativos debido a las discrepancias en las ordenes de consumo presentadas mensualmente por la franquicia del restaurant y el hotel. En el mercado actual no existe una solución informática que satisfaga las carencias que tiene el hotel en cuanto al seguimiento de dichos procesos, por lo que se decidió desarrollar un software que supliera las necesidades previamente mencionadas. Luego de recolectar toda la información necesaria, se comenzó con el proceso de desarrollo de dicha herramienta utilizando la metodología Scrum como elemento principal para la organización de este proyecto, ya que su naturaleza permite aplicarla a pequeños equipos de desarrollo, elevando los estándares de calidad del producto final y aumentando la rapidez en los tiempos de entrega del mismo. Una vez completado el desarrollo de esta herramienta informática se procedió a efectuar las pruebas pertinentes, documentando los resultados y generando las conclusiones y recomendaciones del caso.

Descriptor: Desarrollo, Web, Móvil, Solución Informática, Herramienta, Scrum.

INTRODUCCIÓN

Hotel Garden Valencia inició sus actividades en la ciudad de Valencia del Estado Carabobo, con el fin de proveer servicios de hospedaje a Personal Ejecutivo, Federaciones de Atletas, Comitivas Políticas y Personalidades del Arte y Espectáculo. Ofreciendo paquetes de estadías atractivos para las empresas, así como también añadiendo servicios especiales como taxis ejecutivos para traslados desde el Aeropuerto Internacional de Valencia hacia el hotel y viceversa. Con el tiempo fue expandiendo su abanico de clientes, permitiendo así que particulares tuvieran acceso al disfrute de estas instalaciones y servicios añadidos que el hotel ofrecía.

Como resultado de esta decisión tomada por parte de la directiva del hotel, las reservaciones y estadías aumentaron considerablemente, esto por supuesto implica una mayor cantidad de trabajo y mayor procesamiento de información, las prácticas habituales para manejar estos procesos fueron quedándose obsoletas a medida que el tiempo pasaba y volviéndose insuficientes, además, todo esto por una simple razón, la cantidad de trámites y procesos que se debían llevar a cabo. Había mucho retraso en la información y la directiva del hotel debía tomar una decisión al respecto.

La decisión es tomada y somos contactados para llevar a cabo el estudio de los procesos que hacen parte del día a día del hotel, en aras de automatizar la máxima cantidad ellos y proveer resultados confiables y a tiempo, que ayuden a la toma acertada de decisiones y por consiguiente, el crecimiento saludable de la empresa. Para cumplir con las normativas establecidas por la Universidad José Antonio Páez para la presentación de un Trabajo de Grado y orientar la investigación, la estructura de este trabajo se desarrolla en cuatro capítulos, los cuales se describen a continuación:

Capítulo I: El trabajo inicia con la descripción y formulación del problema, seguidamente de la presentación de los objetivos que guiarán la investigación, luego se procede a dar la justificación, alcance y limitaciones de la misma.

Capítulo II: Se introduce los antecedentes e investigaciones que tratan el mismo problema o se relacionan con la presente investigación. Se explican las bases teóricas que son necesarias para la elaboración de esta investigación y en última instancia se definen los términos básicos.

Capítulo III: Se describe todo lo referente al marco metodológico, como lo son el tipo y diseño de la investigación a utilizar para cumplir los objetivos del trabajo, el nivel de investigación, fuentes, técnicas e instrumentos de recolección de datos, conjuntamente con la población y muestra. También se detallan las fases metodológicas diseñadas para cumplir con los objetivos específicos del trabajo.

Capítulo IV: Una vez aplicados los instrumentos de recolección de información, se procedió a realizar el tratamiento correspondiente para el análisis de los resultados, por cuanto la información que arrojará será la que indique las conclusiones a las cuales llega la investigación.

Capítulo V: En este capítulo, se presentan las conclusiones y recomendaciones donde se expresan nuestras sugerencias para la aplicación de la investigación.

CAPÍTULO I

EL PROBLEMA

1.1 Planteamiento del Problema

La ciudad de Valencia es la capital y ciudad más poblada del Estado Carabobo, situada en la Región Central del país. Es conocida como Capital Industrial de Venezuela debido a que alberga una cantidad significativa de zonas Industriales del país. De igual forma, se ha convertido en un centro de inversión, objeto de proyectos como el World Trade Center Valencia y el Complejo Isla Multiespacio. En 2015 el GaWC (Globalization and World Cities, en español, Globalización y Ciudades del Mundo), incluyó a la ciudad de Valencia en su listado, siendo ésta y Caracas las únicas ciudades de Venezuela en el ranking, clasificándola como una Ciudad de Suficiencia. Como consecuencia, es periódicamente visitada por un número considerable de personas interesadas en expandir sus negocios, ofrecer sus productos o servicios y concretar nuevos proyectos, lo que hace de esta ciudad un mercado bastante atractivo para la empresa hotelera.

En base a lo expuesto anteriormente, empresas que realizan negocios en esta ciudad, envían a sus representantes desde el interior o exterior del país, haciendo necesaria la existencia de lugares donde pernoctar. Como consecuencia de esta situación se produce un apreciable desarrollo de la industria hotelera en esta ciudad. Advirtiéndose un nicho de mercado, Hotel Garden Valencia (HGV) inició servicios contando con una gran capacidad y comodidades para todos sus huéspedes, atención personalizada y el mejor trato al cliente.

A partir del año 2015, la cantidad de huéspedes que hacen uso frecuente de las instalaciones del Hotel Garden Valencia(HGV), ha aumentado en alrededor de un 25% según cifras suministrados por el departamento de mercadeo del hotel. Dicha circunstancia ha ocasionado un aumento considerable del flujo de información a ser procesado por el personal que labora en el hotel. Dicha circunstancia ha ocasionado un

aumento considerable del flujo de información a ser procesado por el personal que labora en el hotel.

Actualmente, Hotel Garden Valencia (HGV), procesa toda esta data de forma manual, haciendo uso de fichas para el registro de consumos y servicios de los huéspedes. Este mecanismo de trabajo ralentiza los procesos, y es susceptible a errores o pérdida de datos. Sin mencionar la ausencia de un sistema de respaldo para dichas fichas, lo cual compromete la integridad de la información sensible del Hotel.

Los consumos involucran cargos al huésped por concepto de comidas o bebidas en el restaurant del hotel (Rest-Service, Servicio de Restaurante) y consumo de comidas o bebidas en la habitación (Room-Service, Servicio a la Habitación).

Los servicios ofrecidos por el hotel a sus huéspedes, configuran parte fundamental de la estadía de los mismos, Reservación, Check-In(Chequeo al Entrar), Check-Out(Chequeo al Salir), limpieza oportuna de la habitación, cambio de juego de toallas, configuración de aire acondicionado, reposición de artículos de nevera ejecutiva e higiene, entre otros, deben ser gestionados de manera automática y eficiente por el personal calificado para ello.

La gestión de consumos y servicios en el Hotel Garden Valencia se lleva a cabo de forma manual debido a que no se dispone de una plataforma tecnológica capaz de integrar dichos procesos de forma automática, eficaz y eficiente.

La importancia de poseer la información actualizada en tiempo real se hace cada vez más necesaria, saber si una habitación se encuentra reservada(ocupada por un huésped) o está disponible(lista para ser ocupada), si se encuentra en mantenimiento o está fuera de servicio, es esencial al momento de atender un potencial huésped en la recepción.

Actualmente, el supervisor de mantenimiento revisa cada cierto tiempo las fichas de todas las habitaciones en búsqueda de las que estén marcadas para servicio de mantenimiento, ya sea para una reposición de inventario de nevera ejecutiva o para limpieza y cambio de toallas. Este procedimiento se torna ineficiente ya que la información no llega a sus manos de manera actualizada, en muchos casos, para

cuando llega la información, no hay tiempo de ejecutar el mantenimiento, debido a que el huésped regresó a su habitación. Éste debe estar informado en cualquier momento sobre las habitaciones que están desocupadas, ya sea porque no están alquiladas o porque el huésped no se encuentra, para así programar eficaz y eficientemente el servicio a las mismas atendiendo al lema frecuentemente practicado en hotelería que reza “El servicio de limpieza es Invisible”.

El huésped tiene la posibilidad de ordenar cualquier consumo tanto en el Restaurant del Hotel(Rest-Service), como desde la comodidad de su habitación(Room-Service). Estas actividades son registradas por parte del restaurant(Franquicia separada del hotel) utilizando comandas en papel, las cuales posteriormente son entregadas a la recepción del hotel para ser cargadas al consumo de la habitación. Al finalizar cada semana, la administración del hotel en conjunto con la administración del restaurant coteja sus respectivas comandas, éstas, en múltiples ocasiones no coinciden, generando conflictos en los montos a pagar por parte del Hotel hacia el Restaurant.

La aplicación web y móvil para el control y seguimiento de consumos y servicios de clientes para el Hotel Garden Valencia (HGV) en el Estado Carabobo determinará las características, requisitos funcionales y no funcionales necesarios para el desarrollo del mismo, así como también los nuevos procesos para implementar buenas prácticas para el desarrollo de las distintas actividades llevadas a cabo por el equipo de trabajo del Hotel Garden Valencia (HGV).

1.2 Formulación del problema

Según lo explicado anteriormente se hace la siguiente pregunta: ¿Qué componentes funcionales debe abarcar una solución automatizada que permita el registro y control oportuno de consumos y servicios ofrecidos al huésped?

1.3Objetivos de la Investigación

1.3.1 Objetivo General

Desarrollar una aplicación web y móvil para el control y seguimiento de consumos y servicios de los clientes del Hotel Garden Valencia en el Estado Carabobo.

1.3.2 Objetivos Específicos

- Diagnosticar la situación actual respecto de los procesos de control y seguimiento de los consumos y servicios del Hotel Garden Valencia, a fin de establecer especificaciones del modelo conceptual de la aplicación a desarrollar.
- Determinar el modelo funcional de la aplicación web y móvil a través de los requerimientos funcionales y no funcionales, definidos en el modelo conceptual.
- Diseñar cada uno de los módulos de programación de la aplicación web y móvil para satisfacer los requisitos determinados aplicando la metodología SCRUM
- Evaluar el funcionamiento de cada uno de los componentes de la aplicación web y móvil con los casos de prueba iniciales.

1.4 Justificación de la Investigación

La ejecución de este proyecto se fundamenta en la problemática existente en Hotel Garden Valencia (HGV), de no poseer un herramienta informática que se ajuste a sus necesidades, careciendo así de integración y control de procesos tales como Reservas, Check-In, Check-Out y Consumos, así como también de reportes de utilidad tales como Reportes de consumos por habitación, Reportes de alquiler de habitaciones, Reporte general de consumos y servicios por habitación, que facilitan el trabajo al personal que integra el equipo de trabajo del Hotel Garden Valencia(HGV).

La solución de sistemas propuesta contará con la cualidad de poder ser utilizado tanto por el personal del Hotel como por los clientes allí hospedados, con funcionalidades interactivas que le permitirán a los mismos customizar su estadía en el hotel en casi cualquier aspecto que deseen, desde la comida que desean para cenar, hasta los grados de temperatura que tendrá el aire acondicionado, apalancándonos de las nuevas tecnologías móviles que tenemos a disposición con los Smartphones (Teléfonos Inteligentes) que se han vuelto tendencia en los últimos años.

La solución planteada contará con variedad de funciones, que permitirán así un robusto manejo de información. Procesos principales como registro de clientes, asignación de habitaciones, generación de reservas, Check-In y Check-Out, hará de este aplicativo una herramienta fundamental para el desempeño de las actividades del Hotel Garden Valencia (HGV).

1.5 Alcance de la Investigación

La Aplicación Web y Móvil será una herramienta ideal para casi cualquier tipo de Hotel o Cadena Hotelera Nacional o Internacional, salvando las limitaciones de idioma por los momentos.

La Aplicación Web y Móvil propuesta tendrá dentro de sus principales funciones la Reserva de Habitaciones, módulos para ejecución de Check-In y Check-Out, estatus completo de las habitaciones en tiempo real, registro de Consumos tanto en habitación como en el restaurante para ser cargado a una habitación en específico, entre otras funciones.

Se aplicará la metodología de desarrollo de software SCRUM tomando en cuenta todas sus fases y pasos a seguir para llevar a cabo las metas planteadas. El desarrollo de esta herramienta se llevará a cabo utilizando los siguientes lenguajes de programación:HTML5, CSS3 Y JAVASCRIPT para el Frontend de la aplicación web, para la aplicación móvil se utilizará SWIFT. Para el Backend que compartirán ambas aplicaciones se utilizara el reconocido Framework de PHP Laravel, y por último, como motor de base de datos MySQL.

CAPÍTULO II

MARCO TEÓRICO

2.1 Antecedentes

Para el desarrollo de este proyecto se han tomado como referencia trabajos de grado realizados previamente por distintos autores, para así poder cumplir exitosamente los objetivos planteados con anterioridad. A continuación, se dará una breve descripción de los mismos:

En el proyecto realizado por Villasana E. y Villaquirán F. (febrero, 2016) titulado **“Desarrollo de un sistema web para el proceso administrativo en los talleres de latonería y pintura multiservicios Latón Car, C.A”** el cual fue presentado para optar al título de Ingeniero de Computación en la Universidad José Antonio Páez San Diego Venezuela. En este trabajo de grado se utiliza SCRUM como metodología de desarrollo de software haciendo énfasis en la asignación de los roles y en cada una de las etapas de los Sprints, los cuales tienen gran importancia para el cumplimiento de las metas propuestas en este proyecto.

Por otro lado Culqui E.(Abril, 2015) en su trabajo titulado **“Sistema web para el registro de reservaciones y control de hospedaje en el Hotel Acapulco de la ciudad de Ambato”**, el cual fue presentado para optar al título de ingeniero en Sistemas Computacionales e Informáticos en la Universidad Técnica de Ambato Tungurahua Ecuador, El objetivo de este proyecto era el desarrollar un sistema web que ayudara a optimizar los procesos de reservación de habitaciones, optimización de sus registros y un mejor control de las operaciones del restaurant de dicho hotel. Este trabajo de grado muestra cual es la manera más óptima de llevar a cabo cada uno de estos procesos, también muestra cómo se debe realizar un sistema utilizando el patrón de diseño Modelo, Vista, Controlador(MVC).

2.2 Bases Teóricas

Para el desarrollo de la investigación es necesario describir los distintos fundamentos relacionados al problema investigado, esto proporcionara una visión amplia de los conceptos utilizados por los investigadores para consolidar su proyecto. Según Arias (2012), “Las bases teóricas implican un desarrollo amplio de los conceptos y proposiciones que conforman el punto de vista o enfoque adoptado, para sustentar o explicar el problema planteado” (p.107).

2.2.1 Sistemas de Información

Según Pablos, López, Agius, Romo, Medina (2012), definen un sistema de información como un conjunto de recursos técnicos, humanos y económicos, interrelacionados dinámicamente y organizados en torno al objetivo de satisfacer las necesidades de información de una organización empresarial para la gestión y la correcta adopción de decisiones.

Los sistemas de información se encargan básicamente de cuatro (4) actividades: entrada de información, almacenamiento, procesamiento, y salida de información.

- **Entrada de información:** Es el proceso mediante el cual el Sistema de Información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos.
- **Almacenamiento:** El almacenamiento es una de las actividades o capacidades más importantes que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior.
- **Procesamiento de Información:** Es la capacidad del Sistema de Información

para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecida. Estos cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados. Esta característica de los sistemas permite la transformación de datos fuente en información que puede ser utilizada para la toma de decisiones.

- **Salida de Información:** La salida es la capacidad de un Sistema de Información para sacar la información procesada o bien datos de entrada al exterior.
 - o Cualquier sistema de información va pasando por una serie de fases a lo largo de su vida. Su ciclo de vida comprende una serie de etapas entre las que se encuentran las siguientes:
- **Planificación:** En la fase inicial se realizan las primeras tareas del proyecto incluyen actividades tales como la determinación del ámbito del proyecto, la realización de un estudio de viabilidad, el análisis de los riesgos asociados al proyecto, una estimación del coste del proyecto, su planificación temporal y la asignación de recursos a las distintas etapas del proyecto.
- **Análisis:** La etapa de análisis en el ciclo de vida del software corresponde al proceso mediante el cual se intenta descubrir qué es lo que realmente se necesita y se llega a una comprensión adecuada de los requerimientos del sistema (las características que el sistema debe poseer).
- **Diseño:** En la fase de diseño se han de estudiar posibles alternativas de implementación para el sistema de información que hemos de construir y se ha de decidir la estructura general que tendrá el sistema (su diseño arquitectónico). El diseño de un sistema es complejo y el proceso de diseño ha de realizarse de forma iterativa.
- **Implementación:** Para la fase de implementación se ha de seleccionar las herramientas adecuadas, un entorno de desarrollo que facilite el trabajo y un lenguaje de programación apropiado para el tipo de sistema que se vaya a

construir. La elección de estas herramientas dependerá en gran parte de las decisiones de diseño que se hayan tomado hasta el momento y del entorno en el que el sistema deberá funcionar.

- **Pruebas:** La etapa de pruebas tiene como objetivo detectar los errores que se hayan podido cometer en las etapas anteriores del proyecto (y, eventualmente, corregirlos). El objetivo de las pruebas es encontrar dichos errores antes de que el usuario final del sistema pueda toparse con ellos. De hecho, una prueba es un éxito cuando se detecta un error (y no al revés, como nos gustaría pensar).
- **Instalación o despliegue:** De cara a su instalación, se ha de planificar el entorno en el que el sistema debe funcionar, tanto hardware como software: equipos necesarios y su configuración física, redes de interconexión entre los equipos y de acceso a sistemas externos, sistemas operativos (actualizados para evitar problemas de seguridad), bibliotecas y componentes suministrados por terceras partes etc.
- **Mantenimiento:** La etapa de mantenimiento consume típicamente del 40 al 80 por ciento de los recursos de una empresa de desarrollo de software. De hecho, con un 60 por ciento de media, es probablemente la etapa más importante del ciclo de vida del software. Dada la naturaleza del software, que ni se rompe ni se desgasta con el uso.

En referencia a lo anteriormente dicho, la principal razón de esta información es que nos puede orientar a los principales beneficios de los sistemas de información, por consiguiente, también los futuros beneficios de la automatización de los procesos mencionados en el capítulo anterior.

2.2.2 Metodología de desarrollo de software

Una metodología de desarrollo de software se refiere al marco de trabajo que es utilizado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información. Una gran variedad de estos marcos de trabajo ha evolucionado a

través de los años, cada uno con sus propias debilidades y fortalezas reconocidas, una metodología de desarrollo de software no necesariamente se adapta a los requerimientos de todos los proyectos, cada una de las metodologías disponibles se adapta mejor a un tipo de proyecto en específico, basándonos en ciertas especificaciones técnicas, organizacionales, del proyecto y del equipo.

Por otra parte, la clasificación de metodologías no es una tarea simple, esto se debe a la gran cantidad de propuestas que existen, sus diferentes alcances, la información disponible y el nivel de detalle de cada una de las metodologías. Las metodologías pueden ser clasificadas en dos grupos: Metodologías Estructuradas y Metodologías Orientadas a Objetos.

Otras metodologías, denominadas Metodologías Ágiles, están más orientadas a la generación de código con ciclos muy cortos de desarrollo, enfocadas a equipos de desarrollo pequeños, y realizan especial énfasis en el trabajo en equipo e involucrar activamente al cliente en el proceso. Esta última, es donde se encuentra clasificada la metodología escogida para este trabajo de grado, la metodología de desarrollo ágil.

2.2.3 SCRUM

Según www.proyectosagiles.org, Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En esta metodología se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

La metodología Scrum se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.

El trabajo como tal, es realizado en pequeñas iteraciones de tiempo, las cuales usualmente tienen un rango de una semana hasta un mes completo. Durante cada iteración, un grupo auto-organizado, multifuncional, realiza todo el trabajo (ejemplo, diseño, codificación, y pruebas) para producir características completas y funcionales que puedan pasar a producción.

Por lo general, la cantidad de trabajo en la lista de características o backlog de un producto es mucho mayor al que se puede realizar por el equipo en una iteración corta.

La mejor manera para trabajar con la metodología Scrum es definiendo dos pilares fundamentales como lo son los roles y las actividades.

- **Los Roles**

En cada esfuerzo de desarrollo de *Scrum* consiste en uno o más equipos de *Scrum*, cada uno conformado con los siguientes roles: ProductOwner, ScrumMaster, y el equipo de desarrollo.

El **ProductOwner** (Dueño del Producto) representa la voz del cliente. Se asegura de que el equipo Scrum trabaje de forma adecuada desde la perspectiva del negocio. El ProductOwner escribe historias de usuario, las prioriza, y las coloca en el ProductBacklog.

El Scrum es facilitado por un **ScrumMaster**, cuyo trabajo primario es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El

ScrumMaster no es el líder del equipo (porque ellos se auto-organizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga.

Por último, el equipo de desarrollo, que tiene la responsabilidad de entregar el producto. Es recomendable un pequeño equipo de 3 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc.).

También en Scrum existen Roles Auxiliares, estos roles son aquellos que no tienen un rol formal y no se involucran frecuentemente en el "proceso Scrum", sin embargo, deben ser tomados en cuenta.

Un aspecto importante de una aproximación ágil es la práctica de involucrar en el proceso a los usuarios, expertos del negocio y otros interesados ("stakeholders"). Es importante que ellos participen y entreguen retroalimentación con respecto a la salida del proceso a fin de revisar y planear cada sprint.

- **Actividades**

- Planificación de la iteración (Sprint Plannig).
- Ejecución de la iteración(Sprint).
- Reunión diaria de sincronización del equipo(ScrumDaily meeting).
- Demostración de los requisitos completados(Sprint Review).
- Retrospectiva(Sprint Retrospective).
- Refinamiento de la lista de requisitos y cambios en el proyecto.

2.2.4 Frameworks en el desarrollo de software

Framework se define como entorno o trabajo de ambiente para el desarrollo óptimo de una aplicación, dependiendo del lenguaje, normalmente integra componentes que facilitan el desarrollo de aplicaciones, como el soporte de programa, bibliotecas, plantillas y más".

Los framework tienen como objetivo principal ofrecer una funcionalidad definida, auto contenida, siendo construidos usando patrones de diseño, y su característica principal es su alta cohesión y bajo acoplamiento. Para acceder a esa

funcionalidad, se construyen piezas, objetos, llamados objetos calientes, que vinculan las necesidades del sistema con la funcionalidad que este presta. Esta funcionalidad, está constituida por objetos llamados fríos, que sufren poco o ningún cambio en la vida del framework, permitiendo la portabilidad entre distintos sistemas. Frameworks conocidos que se pueden mencionar por ejemplo son Spring Framework, Hibernate, donde lo esencial para ser denominados frameworks es estar constituidos por objetos casi estáticos con funcionalidad definida a nivel grupo de objetos y no como parte constitutiva de estos, por ejemplo, en sus métodos, en cuyo caso se habla de un API o librería.

Por otra parte, los frameworks tienen a disposición una gran variedad de herramientas que apoyan el desarrollo de sitios web dinámicos, las cuales cuentan con bibliotecas de clases que permiten establecer conexiones y accesos a bases de datos mediante un servidor, vinculando un manejador de bases de datos, y facilitando estructuras para plantillas.

La gestión de sesiones y usuarios y comúnmente facilitan la reutilización de código, lo que a su vez reduce el tiempo de codificación puesto a que dispone del código funcional en cualquier momento de la fase de desarrollo, además de que permite la refacción del software, la cual consiste en reescribir el código sin alterar la funcionalidad del sistema con la finalidad de optimizarlo.

2.2.5 Laravel

Según Wikipedia. “Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el código espagueti”.

Laravel promete llevar el lenguaje PHP a un nuevo nivel, la sencillez de su desarrollo se debe fundamentalmente a su patrón de diseño MVC(Modelo-Vista-Controlador) tradicional, a su expresiva sintaxis, a sus generadores de código, y a su ORM (Objet-Relational Mapping, Mapeo de Objeto-Relacional) incluido en un paquete llamado Eloquent.

Haciendo referencia a lo mencionado, se adiciona que este mismo incluye de paquete un sistema de procesamiento de plantillas llamado Blade. Este sistema de plantillas favorece un código mucho más limpio en las Vistas, además de incluir un Sistema de Caché que lo hace mucho más rápido.

Los Sistemas de Cache, evitan el tener que procesar el código una y otra vez en cada petición. Para lo cual, estos sistemas generan versiones estáticas en memoria o disco duro con archivos que corresponden a peticiones previamente procesadas. Y con esta técnica se logra mejorar el rendimiento de la aplicación.

Los Layouts en Blade, son archivos de texto plano que contiene todo el HTML de la página con etiquetas que representan elementos o zonas a incluir en el Layout, o vistas parciales como se conocen en otros Frameworks en PHP. Sin embargo, en Blade estos elementos incrustados se organizan en un sólo archivo. Esta es una idea muy interesante de Laravel que mejora la organización de las vistas y su rendimiento. Sobre todo, cuando las vistas pueden llegar a ser muy complejas incluso con elementos anidados.

Con la presente información se permite conocer a fondo la herramienta usada en el desarrollo de la aplicación web, con el fin de aprovechar todas las funciones y ventajas ofrecidas por el framework Laravel como lo son los sistemas caché, el diseño HTML con el uso de Blade y su patrón de diseño MVC, y a su vez saber su correcto funcionamiento, consiguiendo como resultado una aplicación robusta.

2.3 Definición de Términos Básicos

Ajax: acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web, aunque es confundido por muchos como un lenguaje de programación, y tiene la finalidad de crear aplicaciones interactivas en donde interviene un servidor. Las aplicaciones que hacen uso de esta técnica son ejecutadas en el navegador del usuario, y a su vez mantiene una comunicación asíncrona con el servidor en segundo plano. La principal ventaja de esta técnica es

que permite interactuar y realizar cambios sobre la aplicación sin recargarla, lo que se convierte en mayor velocidad de respuesta y usabilidad de la aplicación.

Automatización: Conversión de determinados procesos corporales o psíquicos en automáticos o involuntarios.

Base de datos: Base de Datos es básicamente un sistema para llevar registros, es decir, es un sistema cuya finalidad general es almacenar información y permitir a los usuarios recuperar y autorizar peticiones de información. (C. J. Date ,2001).

HTML: Del acrónimo HyperTextMarkupLanguage, que traducido al español significa lenguaje de marcas de hipertexto. Son documentos que contienen texto plano organizado mediante etiquetas y usan la extensión .html.

IDE: Entorno integrado de desarrollo (IntegratedDevelopmentEnvironment). Aplicación compuesta por un conjunto de herramientas útiles para un programador. Un entorno IDE puede ser exclusivo para un lenguaje de programación o bien, poder utilizarse para varios. Suele consistir de un editor de código, un compilador, un debugger y un constructor de interfaz gráfica GUI. (Leandro Alegsa, 2010).

Interfaz:Es un término que procede del vocablo inglés interface. En informática, esta noción sirve para señalar a la conexión que se da de manera física y a nivel de utilidad entre dispositivos o sistemas.

iOS:es un sistema operativo móvil de la multinacional Apple Inc. Originalmente desarrollado para el iPhone (iPhone OS), después se ha usado en dispositivos como el iPod touch y el iPad. No permite la instalación de iOS en hardware de terceros.

JavaScript: Es un lenguaje de programación interpretado que se ejecuta del lado del cliente, es decir, en el navegador del usuario, aportando interactividad a la página, mejorando la interfaz de usuario y las páginas web dinámicas.

jQuery: Es una librería de *JavaScript* creada con la intención de hacer la programación en *JavaScript* más fácil. Una librería de *JavaScript* (jQuery) es un conjunto de códigos en lenguaje JavaScript que permite simplificar tareas complicadas y resolver problemas de compatibilidad entre navegadores web. En otras

palabras, jQuery resuelve los dos principales problemas de *JavaScript*: complejidad y compatibilidad.

Optimizar: Conseguir que algo llegue a la situación óptima o de los mejores resultados posibles.

Proceso: Conjunto de las fases sucesivas de un fenómeno natural o de una operación artificial.

PHP: PHP es el acrónimo de HipertextPreprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación. Un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente.

Swift: Swift es un lenguaje de programación multiparadigma creado por Apple enfocado en el desarrollo de aplicaciones para iOS y Mac OS X. Fue presentado en la WWDC 2014.

Room-Service: El Room Service es considerado como un servicio muy importante dentro de los hoteles el cual consiste en llevar alimentos y bebidas a las habitaciones de los huéspedes. Estos a su vez son una fuente muy importante de ingresos para el hotel.

Check-In: Es el acto de Registrarse en el Hotel para recibir una habitación y poder disfrutar de los servicios de estas instalaciones.

Check-Out: Es el acto de finalización de estadía en el Hotel, el cual culmina con el pago de la factura correspondiente a los servicios provistos por el mismo.

Rest-Service: Es el Servicio de Restaurante del Hotel, las personas pueden bajar a la planta del Hotel, desayunar, almorzar, cenar y disfrutar de Snacks a cualquier hora del día.

CAPÍTULO III

MARCO METOLÓGICO

3.1 Tipo de Investigación

Por la naturaleza propia del proyecto, hace que la investigación entre en la clasificación de proyecto especial, puesto que se desarrollará un plan de trabajo para la elaboración de una aplicación web y móvil para el control seguimiento de consumos y servicios de los clientes del hotel garden, A fin de solventar la problemática detectada en dichas instalaciones. Referente a eso, las Normas de Trabajo de Grado de la Universidad José Antonio Páez (UJAP, julio 2007), nos indica que un Proyecto especial de grado:

“Consistirá en las creaciones tangibles, susceptibles de ser realizadas a problemas demostrados, o que respondan a necesidades o intereses de tipo cultural. Se incluyen en esta categoría los trabajos de elaboración de libros de texto y de materiales de apoyo educativo, el desarrollo de software y hardware, prototipos y productos tecnológicos en general.”
(p.5)

3.2 Diseño de la Investigación

El diseño de investigación constituye el plan general del investigador para obtener respuestas a sus interrogantes o comprobar la hipótesis de investigación. Según Arias (2012) define como investigación de campo a:

“La investigación de campo es aquella que consiste en la recolección de todos directamente de los sujetos investigados, o de la realidad donde ocurren los hechos (datos primarios), sin manipular o controlar variable alguna, es decir, el investigador obtiene la información, pero no altera las condiciones existentes. De allí su carácter de investigación no experimental.”(p.31)

Debido a que en este trabajo se necesita que el investigador recolecte los datos directamente en las instalaciones del Hotel Garden Valencia, para así garantizar la

veracidad y fiabilidad de la información obtenida de los empleados y clientes de dicho

hotel, por esto la investigación encaja en el concepto anteriormente expuesto de investigación de campo.

3.3 Nivel de Investigación

Arias (2012), señala que en un estudio pueden identificarse diversos tipos de investigación, existiendo muchos modelos y diversas clasificaciones, sin embargo, independientemente de la clasificación utilizada “todos son tipos de investigación, y al no ser excluyentes, un estudio puede ubicarse en más de una clase” (p.23).

Según el nivel de investigación, es decir, el grado de profundidad con que se aborda un fenómeno u objeto de estudio, la investigación se enmarcó en una investigación de tipo descriptiva. Según Arias(2012) define como investigación descriptiva a:

“la investigación descriptiva consiste en la caracterización de un hecho, fenómeno, individuo o grupo, con el fin de establecer su estructura o comportamiento. Los resultados de este tipo de investigación se ubican en un nivel intermedio en cuanto a la profundidad de los conocimientos se refiere.”(p.24).

3.4 Población y Muestra

Según Arias (2012), la población, o en términos más precisos población objetivo, es un conjunto finito o infinito de elementos con características comunes para los cuales serán extensivas las conclusiones de la investigación. Esta queda delimitada por el problema y por los objetivos del estudio. (p.81)

La población de la presente investigación está compuesta por el personal del Hotel Garden Valencia, cuyo personal administrativo, técnico y obrero suman 20 personas.

3.5 Técnicas e instrumentos de recolección de datos

Según Arias (2012), un instrumento de recolección de datos es cualquier recurso, dispositivo o formato (en papel o digital), que se utilizar para obtener, registrar o almacenar información (p.68). A lo que se refiere Arias en la cita anterior es que claramente la recolección de datos se debe realizar con la ayuda de

herramientas y procesos preestablecidos con el fin de facilitar el posterior análisis de los datos recolectados.

Para fines de la investigación, la recolección de datos se llevará a cabo mediante la técnica de la entrevista aplicada a los empleados y clientes del hotel. Según Arias (2012), “La entrevista, más que un simple interrogatorio, es una técnica basada en un dialogo o conversación "cara a cara", entre el entrevistador y el entrevistado acerca de un tema previamente determinado, de tal manera que el entrevistador pueda obtener la información requerida” (p.73).

3.6 Fases Metodológicas

La metodología que se usará para el desarrollo del proyecto es la metodología SCRUM, debido a su gran efectividad para obtener resultados a corto plazo. La manera como Scrum está estructurada permite organizar mejor el trabajo y dividirlo en iteraciones de entre una semana y un mes, donde cada iteración debe tener como salida software funcional y de valor para el cliente. Esta metodología está diseñada para trabajar en equipos de hasta siete (7) personas, con roles multifuncionales, por lo que se adapta perfectamente a las exigencias planteadas en este proyecto de investigación.

Fase I: Diagnóstico de la situación actual derivada del control y seguimiento de consumos de servicios del cliente en el Hotel Garden.

Esta fase se basó en el análisis de la situación actual, el estado en el que se encuentra y el rol de cada uno de los actores, que en este caso fueron los empleados y clientes del Hotel Garden Valencia. Para esto, se utilizó la entrevista como técnica de recolección de datos.

Fase II: Determinación del Modelo Funcional de la Aplicación Web y Móvil

En esta fase, los datos obtenidos en la fase de análisis fueron recopilados, y en base a estos resultados y la utilización de técnicas de análisis de datos, se determinó los requerimientos funcionales y no funcionales del sistema.

Fase III: Diseño de Módulos de Programación de la Aplicación

En esta fase los requerimientos fueron la parte más importante para asegurar un diseño que se ajustó de la mejor manera al problema que se planteó. Por tanto, se hizo uso de los requerimientos funcionales y no funcionales obtenidos en la fase anterior.

Cabe destacar que se utilizaron herramientas de documentación como las propuestas por UML: Diagramas de Casos de Uso, Diagramas de Secuencia, descripción de los actores y de los casos de usos, para la creación de una documentación que fundamentó los pasos a seguir en el desarrollo del sistema.

Además, se diseñó las interfaces de usuario y la experiencia de usuario que permitió a los beneficiarios realizar las diferentes actividades que propone el sistema, de la manera más simple y práctica. De igual forma, la experiencia de usuario se diseñó tomando en cuenta el ámbito para el cual fue desarrollado, por lo tanto, se simplificó al máximo sin perder funcionalidad o interacción entre las partes.

Asimismo, en esta fase se diseñó el modelo entidad relación de la base de datos que aloja toda la información del sistema en sus diferentes etapas y niveles. Igualmente, se realizó el modelo lógico de la misma para obtener las relaciones entre los datos en las diferentes tablas que conformaran el sistema.

Fase IV: Desarrollo

El desarrollo del sistema para el control de consumos y servicios en el Hotel Garden Valencia, se realizó en base a los pasos sugeridos por la metodología de desarrollo de software Scrum, ya que según Scrum.org, se adapta perfectamente a equipos pequeños de hasta nueve (9) personas, organizando las tareas de sus integrantes, evaluando sistemáticamente los resultados obtenidos y generando entregas con elevados estándares de calidad.

Fase V: Evaluación del Funcionamiento de cada Componente de la Aplicación

En esta fase se evaluó la funcionalidad de la aplicación mediante la realización de pruebas, para la detección de posibles errores y su posterior corrección.

La realización de pruebas estructurales o de caja blanca fue la principal herramienta para la verificación y correcto funcionamiento del código desarrollado, estas pruebas fueron complementadas con pruebas de caja negra. Igualmente, seguido de estas pruebas se procedió a realizar pruebas unitarias a los módulos completados y finalmente se probó el correcto funcionamiento de todo el sistema mediante una prueba de integración. Esta última se hizo de manera exhaustiva y enfocada principalmente en la verificación de compatibilidades entre módulos, integridad de los datos procesados por cada uno de ellos, así como la fidelidad de los resultados arrojados por los mismos, cotejando cada uno de éstos con resultados provenientes de cálculos manuales u otras herramientas de software destinadas para tal fin.

CAPÍTULO IV

RESULTADOS

4.1 Fase I: Diagnósticode la situación actual derivada del control y seguimiento de consumos de servicios del cliente en el Hotel Garden

A través de la implementación de las diferentes técnicas de recolección de datos mencionadas anteriormente, se logró una idea más clara de cómo se efectúan las diferentes tareas en Hotel Garden Valencia. Una vez conocido a detalle la manera cómo funcionan las diferentes áreas o departamentos del hotel se procedió a generar los diferentes requerimientos funcionales y no funcionales del sistema de acuerdo con las necesidades observadas.

4.1.1. Actividad I: Instrumentos de recolección de datos.

En concordancia con los métodos de recolección de datos descritos anteriormente, nos apalancamos en las entrevistas y cuestionarios para obtener una visión lo bastaste objetiva y clara como para permitirnos el apropiado desarrollo de nuestra investigación. Para ello nos permitimos crear un formato guía para registrar tanto las entrevistas como los cuestionarios en cada departamento del hotel para así determinar los actores que participaran en el sistema. (Ver cuadro 1)

Cuadro 1. Guía de Entrevista

Guía para entrevista al personal administrativo y obrero de Hotel Garden Valencia.
Objetivo de la Entrevista
Determinar los roles y procesos realizados por los empleados entrevistados, como llevan a cabo su trabajo, sus herramientas y sus opiniones con respecto a esto.
Datos del Entrevistado
Nombre:
Departamento:
Cargo:
Puntos a Tratar en la Entrevista
Breve Descripción del cargo:
¿Personal a Cargo?, si lo tiene, que funciones desempeñan:
Dificultades encontradas en el desempeño de su cargo:
Preguntas Adicionales a discreción del Entrevistador:

Fuente: Manríquez, Mendoza (2017)

4.1.2. Actividad II: Aplicación de los instrumentos de recolección de datos.

Una vez diseñadas las técnicas de recolección de datos, se procedió aplicarlas. Se eligió de manera sistemática y ordenada un espacio muestral con los departamentos y los empleados a entrevistar para poder hacer el proceso de recolección de datos más efectivo y no tan extenso. A continuación, se muestra la estructura de una de las entrevistas (Ver Cuadro 2).

Cuadro 2. Guía de Entrevista

Guía para entrevista al personal administrativo y obrero de Hotel Garden Valencia.
Objetivo de la Entrevista Determinar los roles y procesos realizados por los empleados entrevistados, como llevan a cabo su trabajo, opiniones y sugerencias.
Datos del Entrevistado Nombre: Jean Carlos Acevedo Departamento: Mantenimiento Cargo: Supervisor de mantenimiento
Puntos a Tratar en la Entrevista
Breve Descripción del cargo: Se encarga de gestionar procesos de limpieza de las instalaciones del hotel, así como de las habitaciones del mismo, apoyado en los obreros de mantenimiento, se encargan de reponer los insumos necesarios para cada habitación, así como verificar el funcionamiento de los artefactos y equipos instalados en las habitaciones como los A/C, Neveras, Calentadores de Agua, Humidificadores, entre otros.
¿Personal a Cargo?, si lo tiene, que funciones desempeñan: Sí, cuatro personas, que constantemente hacen recorridos a las instalaciones, reponiendo los insumos a las habitaciones y generando reportes de fallas u observaciones de las mismas que luego serán derivados al personal correspondiente
Dificultades encontradas en el desempeño de su cargo: El retardo en la información sobre las habitaciones lo que dificulta la organización del personal para atender todas las necesidades de las habitaciones.
Preguntas Adicionales a discreción del Entrevistador: ¿Cree que la automatización del proceso de información disminuirá los retardos para la ejecución de las actividades en las habitaciones del Hotel? R. Seguro, mientras más rápido llegue la información a mi oficina, más rápido yo acciono.

Fuente: Manríquez, Mendoza (2017)

4.2 Fase II: Determinación del Funcional de la Aplicación Web y Móvil

Las entrevistas aplicadas al personal del hotel, permitió recopilar información sumamente importante para la investigación. Gracias al formato implementado para registrar las entrevistas pudimos obtener de una manera objetiva y clara las fortalezas y debilidades de cada uno de los procesos que se llevan a cabo dentro de las instalaciones del hotel. Con esta información recolectada y categorizada, pudimos detectar los problemas específicos existentes y generar las directrices por las cuales se regirá el sistema a desarrollar, tratando de que los procesos mejorados no sean tan ajenos a los procesos actuales de trabajo.

4.2.1. Actividad I: Definición de requerimientos funcionales y no funcionales del sistema

Una vez culminada la fase de recolección de datos, y el análisis de resultados, determinamos los requerimientos funcionales y no funcionales del sistema. Estos requerimientos nos mostraron el camino a seguir en el diseño del sistema, garantizando que este último estuviera ajustado a las necesidades del cliente (Ver cuadro 3 y 4).

Cuadro 3. Requerimientos Funcionales del Sistema Web y Móvil

Funcionales
Permitir el ingreso al sistema administrativo al empleado (con rol activo dentro del sistema) por medio de un nombre de usuario y contraseña asignada por el administrador. (Web)
Permitir el ingreso al sistema administrativo al usuario(Cliente) por medio de un nombre de usuario y contraseña asignada por el administrador. (Móvil)
Registrar un Cliente (Web)
Registrar una Reservación (Web)
Registrar un Empleado (Web)
Visualizar estatus de una habitación o varias habitaciones. (Web)
Visualizar estatus de una Reservación (Web y Móvil)
Visualizar estatus de mantenimientos ordenados a las habitaciones por el supervisor de mantenimiento. (Web y Móvil)
Visualizar los consumos de una habitación (RestService o RoomService) (Web y Movil)
Generar Reportes (Web)

Fuente: Manríquez, Mendoza (2017)

Cuadro 4. Requerimientos No Funcionales del Sistema Web y Móvil

No Funcionales
Mantener una uniformidad y armonía entre las diversas interfaces que tendrá el sistema.
Aplicar las validaciones respectivas para garantizar un correcto manejo de los datos.
Diseñar una interfaz amigable y de fácil manejo al usuario a fin de hacer más sencilla la navegación dentro del sistema.
Mantener la integridad de los datos que se manejan
Respaldar la base de datos como práctica común dentro del sistema
Utilizar nombres apropiados para cada interfaz, que posean relación con la función que desempeñan.

Fuente: Manríquez, Mendoza (2017)

4.3 Fase III: Diseño de Módulos de Programación de la Aplicación

Una vez culminada la fase de análisis, y habiendo determinado los requerimientos funcionales y no funcionales del sistema, se procedió a diseñar la estructura basal del sistema, tomando en cuenta primeramente los datos recolectados en fases anteriores, así como también la observación “in-situ” de los procesos llevados a cabos en el hotel. En esta fase, se categorizaron los requerimientos y en base a ello se crearon módulos, se diseñó la forma en que se manejaran los datos en el sistema mediante la creación de un modelo entidad relación y se seleccionó la metodología de desarrollo a utilizar.

4.3.1 Actividad I: Modelado de casos de uso.

Los casos de uso son un diagrama del Leguaje de Modelado Unificado (UML) que permite describir las diferentes opciones que tienen los actores que participan en el sistema.

Cuadro 5. Actores Vs Acciones en el Sistema

Actor	Acciones Disponibles
Administrador	Posee un alto nivel de privilegios, éste puede ejecutar todas las acciones del sistema. Además, como función principal podrá Administrar los usuarios (Empleados) que interactúen en el sistema, Generar todos los reportes permitidos y eliminar información del sistema.
Empleado	Posee un nivel intermedio de privilegios. Como función principal del empleado, se encuentran, gestionar el ingreso y modificar la información de ser necesaria de: clientes, reservaciones, consumos y ordenes de servicio. Además, se le permitirá generar algunos reportes.
Cliente	Posee un acceso limitado dentro del sistema, éste solo visualizará sus consumos y podrá generar ordenes de consumo y mantenimiento a su habitación.

Fuente: Manríquez, Mendoza Wilmer (2017)

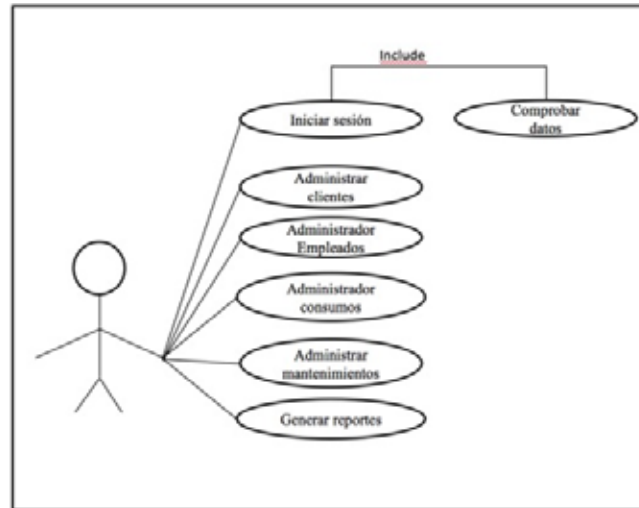


Figura 1. Caso de Uso: Administrador.

Fuente: Manríquez, Mendoza (2017)

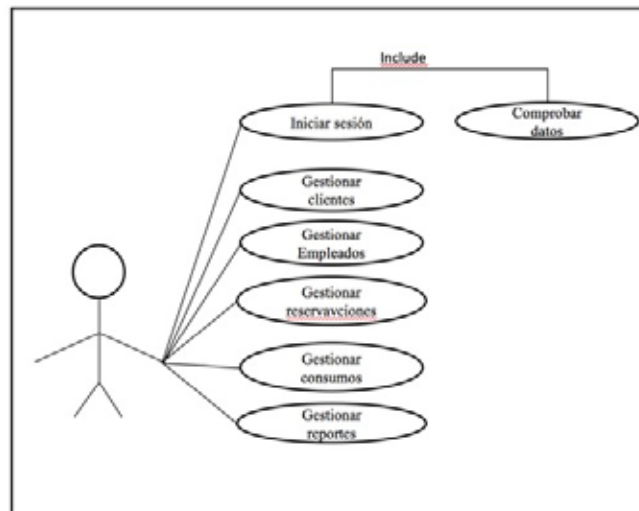


Figura 2. Caso de Uso: Empleado.

Fuente: Manríquez, Mendoza (2017)

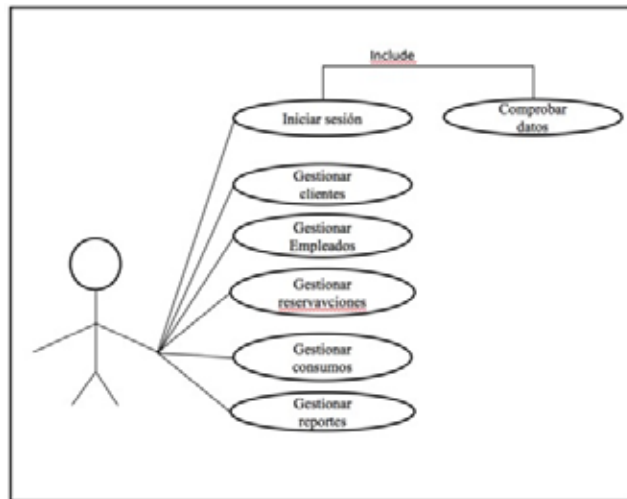


Figura 3. Caso de Uso: Cliente.

Fuente: Manríquez, Mendoza (2017)

4.3.2. Actividad II: Descripción de casos de uso.

A continuación, se muestra la descripción de cada uno de los casos de uso mostrados anteriormente.

Cuadro 6. Iniciar Sesión

Iniciar Sesión	
Actor: Administrador, Empleado, Cliente.	
Objetivos: Ingresar al sistema	
Precondición: Tener un usuario y contraseña asignados por el administrador	
Flujo normal: 1. Ingresar pseudónimo o correo electrónico y contraseña en los campos requeridos. 2. El sistema verifica los datos. 3. Se permite el ingreso al sistema.	Flujo alternativo: 1. Los datos ingresados en los campos son incorrectos. 2. Se muestra un error al usuario. 3. No se permite el ingreso al sistema.
Post-condición: Se accede al sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 7. Administrar Empleados.

Administrar Empleados	
Actor: Administrador	
Objetivos: Registrar Empleados al sistema.	
Precondición: Haber iniciado sesión en el sistema como administrador.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Empleados”. 3. Seleccionar la opción “Nuevo Registro”, “Editar” o “Eliminar”. 4. Ingresar los datos requeridos. 5. Agregar al usuario. 	Flujo alternativo: <ol style="list-style-type: none"> 1. Los datos ingresados son incorrectos o poseen errores. 2. Se muestran los errores al administrador.
Post-condición: Se registra un nuevo Empleado al sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 8. Administrar Clientes.

Administrar Clientes	
Actor: Administrador	
Objetivos: Registrar o Modifica o Elimina la información de un cliente en el sistema.	
Precondición: Haber iniciado sesión en el sistema como administrador y tener al menos un cliente agregado en el sistema.	
Flujo normal: 1. Ir a la sección del menú en Administración. 2. Seleccionar “Clientes”. 3. Seleccionar la opción “Nuevo Registro”, “Editar” o “Eliminar”. 4. Ejecutar la acción.	Flujo alternativo: 1. Seleccionar la opción de “Eliminar”. 2. No se puede eliminar el cliente, porque no se encuentra ninguno asociado. 3. Se muestra un error al usuario.
Post-condición: Se registra o modifica un cliente en el sistema. Se elimina solo si no posee consumos abiertos en el hotel.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 9. Gestionar Clientes.

Gestionar Clientes	
Actor: Administrador, Empleado.	
Objetivos: Registrar o Modificar un cliente en el sistema.	
Precondición: Haber iniciado sesión en el sistema como Administrador o Empleado.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Clientes”. 3. Seleccionar la opción “Nuevo Registro” o “Editar”. 4. Ingresar los datos. 5. Se ejecuta la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 1. Los datos ingresados son incorrectos o poseen errores. 2. Se muestran los mensajes de error.
Post-condición: Se registra un nuevo cliente al sistema o es modificada la información registrada.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 10. Administrar Reservación.

Administrar Reservación	
Actor: Administrador	
Objetivos: Registrar, Modificar o Eliminar la información de una Reservación en el sistema	
Precondición: Haber iniciado sesión en el sistema como administrador y tener al menos una reservación agregada al sistema que esté asociada a un cliente.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Reservaciones”. 3. Seleccionar la opción “Nuevo registro”, “Editar” o “Eliminar”. 4. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 1. Seleccionar la opción de “Eliminar”. 2. No se puede eliminar la reservación, porque no se posee consumos asociados. 3. Se muestra un error al usuario.
Post-condición: Se Registra, Modifica o Elimina la información de una Reservación en el sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 11. Gestionar Reservación.

GestionarReservación	
Actor: Administrador, Empleado	
Objetivos: Registrar o Modificar la información de una Reservación en el sistema	
Precondición: Haber iniciado sesión en el sistema como Administrador o Empleado y tener al menos una reservación agregada al sistema que esté asociada a un cliente.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Reservaciones”. 3. Seleccionar la opción “Nuevo registro”, “Editar”. 4. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 1. Los datos ingresados son incorrectos o poseen errores. 2. Se muestran los mensajes de error.
Post-condición: Se Registra o Modifica la información de una Reservación en el sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 12. Administrar Consumos.

Administrar Consumos

Actor: Administrador, Empleado.	
Objetivos: Registrar, Modificar o Eliminar la información de una orden de Consumo en el sistema	
Precondición: Haber iniciado sesión en el sistema como Administrador o Empleado y tener al menos una reservación agregada al sistema que esté asociada a un cliente para poder asociar un consumo a ésta.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Consumos”. 3. Seleccionar la opción “Nuevo registro”, “Editar” o “Eliminar”. 4. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 1. Seleccionar la opción de “Eliminar”. 2. No se puede eliminar el consumo, porque no se encuentra ningún registro asociado. 3. Se muestra un error al usuario.
Post-condición: Se Registra, Modifica o Elimina la información de una orden de consumo en el sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 13. Gestionar Consumos.

Gestionar Consumos	
Actor: Administrador, Empleado, Cliente	
Objetivos: Registrar o Modificar la información de una orden de Consumo en el sistema	
Precondición: Haber iniciado sesión en el sistema como Administrador, Empleado o Cliente, tener al menos una reservación agregada al sistema que esté asociada a un cliente para poder asociar un consumo a ésta.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Consumos”. 3. Seleccionar la opción “Nuevo registro”, “Editar”. 4. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 1. Los datos ingresados son incorrectos o poseen errores. 2. Se muestran los mensajes de error.
Post-condición: Se Registra o Modifica la información de una orden de consumo en el sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 14. Administrar Mantenimientos.

Administrar Mantenimientos	
Actor: Administrador	
Objetivos: Registrar, Modificar o Eliminar la información de una orden de Mantenimiento en el sistema	
Precondición: Haber iniciado sesión en el sistema como administrador.	
Flujo normal: <ol style="list-style-type: none"> 5. Ir a la sección del menú en Administración. 6. Seleccionar “Mantenimientos”. 7. Seleccionar la opción “Nuevo registro”, “Editar” o “Eliminar”. 8. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 4. Seleccionar la opción de “Eliminar”. 5. No se puede eliminar el Mantenimiento, porque no se encuentra ningún registro asociado. 6. Se muestra un error al usuario.
Post-condición: Se Registra, Modifica o Elimina la información de una orden de mantenimiento en el sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 15. Gestionar Mantenimientos.

Gestionar Mantenimientos	
Actor: Administrador, Empleado, Cliente.	
Objetivos: Registrar o Modificar la información de una Orden de Mantenimiento en el sistema	
Precondición: Haber iniciado sesión en el sistema como Administrador, Empleado o Cliente.	
Flujo normal: <ol style="list-style-type: none"> 5. Ir a la sección del menú en Administración. 6. Seleccionar “Mantenimientos”. 7. Seleccionar la opción “Nuevo registro”, “Editar”. 8. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 3. Los datos ingresados son incorrectos o poseen errores. 4. Se muestran los mensajes de error.
Post-condición: Se Registra o Modifica la información de una orden de mantenimiento en el sistema.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 16. Generar Reportes.

Generar Reportes	
Actor: Administrador	
Objetivos: Generar reportes con los datos del sistema.	
Precondición: Haber iniciado sesión en el sistema como administrador.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir a la sección del menú en Administración. 2. Seleccionar “Reportes”. 3. Seleccionar el tipo de reporte a generar. 4. Ejecutar la acción. 	Flujo alternativo: <ol style="list-style-type: none"> 1. No hay datos en el sistema. 2. No se pueden generar reportes.
Post-condición: Se exportan archivos .xls, .csv o se imprimen los datos.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 17. Ver Reservación.

Ver Reservación	
Actor: Cliente	
Objetivos: Observar el estatus de la Reservación y las características de la misma.	
Precondición: Que el empleado o el administrador lo hayan agregado al sistema como “Cliente” y que posea una reservación en el hotel.	
Flujo normal: 1. Ir al menú y seleccionar la opción “Área de clientes”. 2. Ingresar correo y contraseña. 3. Ingresar a “Ver Reservaciones Activas”.	Flujo alternativo: 1. El cliente no está registrado. 2. Se muestra un mensaje al cliente. 3. Seleccionar “Regístrate aquí” 4. Introducir su correo y cedula en el formulario. 5. Una clave generada de manera aleatoria es enviada a su correo. 6. Puede ingresar a ver su reservación.
Post-condición: Se observa el estado de la reservación.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 18. Ver Consumos.

Ver Consumos	
Actor: Cliente	
Objetivos: Observar los Consumos hechos en el hotel.	
Precondición: Que el empleado o el administrador lo hayan agregado al sistema como “Cliente” y que posea una reservación en el hotel.	
Flujo normal: <ol style="list-style-type: none"> 1. Ir al menú y seleccionar la opción “Área de clientes”. 2. Ingresar correo y contraseña. 3. Ingresar a “Ver Consumos”. 	Flujo alternativo: <ol style="list-style-type: none"> 1. El cliente no está registrado. 2. Se muestra un mensaje al cliente. 3. Seleccionar “Regístrate aquí” 4. Introducir su correo y cedula en el formulario. 5. Una clave generada de manera aleatoria es enviada a su correo. 6. Puede ingresar a ver su consumo.
Post-condición: Se observa el estado de la reservación.	

Fuente: Manríquez, Mendoza (2017)

4.3.3. Actividad III: Modelo entidad-relación de la base de datos.

Una vez determinados los distintos actores, sus casos de uso y describirlos, se diseñó el modelo entidad-relación de la base de datos del sistema, el cual permitió describir la realidad del destino de los datos dentro del sistema a través de representaciones graficas como lo son, las entidades, relaciones y atributos.

Luego, basándonos en el modelo que se muestra a continuación, se implementara el diseño de la base de datos (Ver Figura 4).

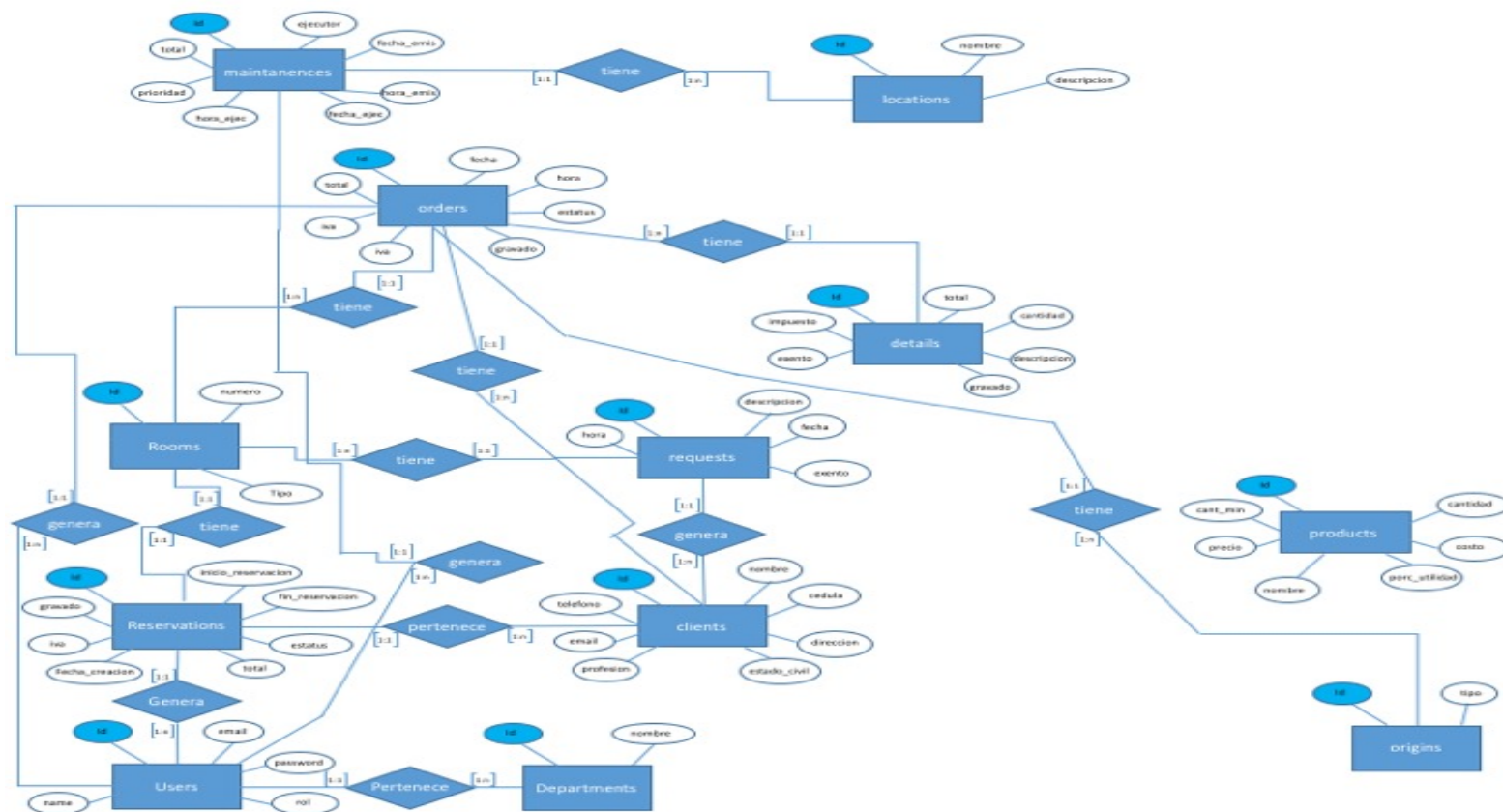


Figura 4. Modelo Entidad-Relación.

Fuente: Manríquez, Mendoza (2017)

4.3.4. Actividad IV: Modelo lógico de la base de datos

El modelo lógico de la base de datos nos permitió especificar los datos que pueden ser procesados en la base de datos por medio de tablas o relaciones. A continuación, se muestra el modelo lógico de la base de datos del sistema:

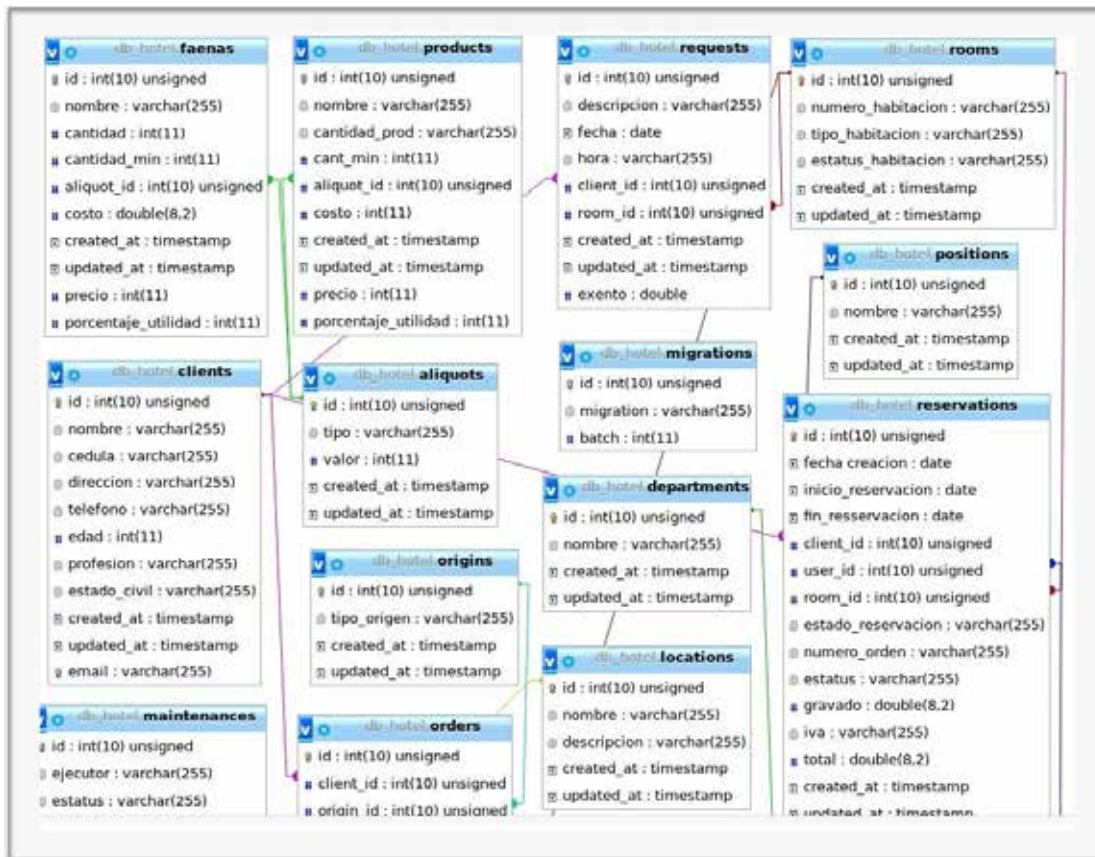


Figura 5. Modelo Lógico de la base de datos.

Fuente: Manríquez, Mendoza (2017)

4.3.5. Actividad V: Descripción de la arquitectura del sistema

La arquitectura de software es un conjunto de patrones que nos proporcionan un marco de referencia necesario para guiar la construcción de un software. Para este proyecto se utilizó el MVC (Modelo-Vista-Controlador) como el patrón de arquitectura de software a utilizar para ser desarrollado. De esta forma se separan en capas abstractas los componentes que conforman el sistema, los cuales son: las

interfaces de usuario, la lógica de programación del sistema y la manipulación de los datos.

Por lo que, el contenido de las vistas como son llamadas las interfaces de usuarios en este tipo de arquitectura, es determinado por los controladores que son los que reciben los datos de provenientes de los modelos, que pueden ser uno o varios según sea el caso, interactuando directamente con la base de datos. separar los procesos de esta manera, hace que el sistema posea un mejor rendimiento y facilita el desarrollo del mismo al momento de agregar otras funcionalidades (escalabilidad) además de poseer un código mucho más ordenado, estético y elegante a la vista.

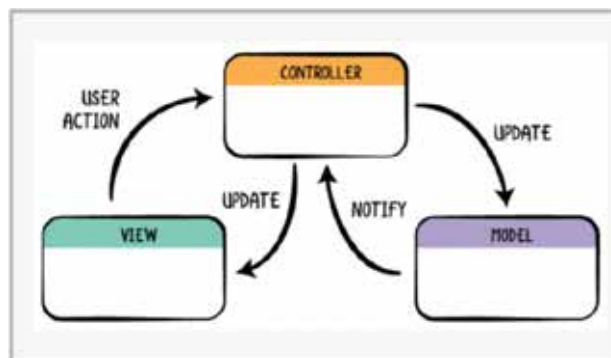


Figura 6. Arquitectura de Software.

Fuente: Peres, R. (2016)

4.3.6. Actividad VI: Diseño de interfaces sistema Web y Móvil

Consideramos el diseño de las interfaces, un paso medular para el desarrollo de nuestro proyecto, debido a que fue aquí donde tratamos de combinar el cumplimiento de todas las necesidades del cliente con la facilidad y entorno amigable del software, de manera que utilizarlo lejos de ser frustrante o dificultoso, fuera ameno y sencillo de usar.

4.3.7 Actividad VII: Generación del Mapa de Navegación.

El mapa de navegación representa a través de un diagrama las diferentes rutas u opciones, a las cuales se puede acceder en el sistema, éste se representa de manera ordenada, leyéndose de izquierda a derecha y de arriba hacia abajo. A continuación, se presenta el mapa de navegación:

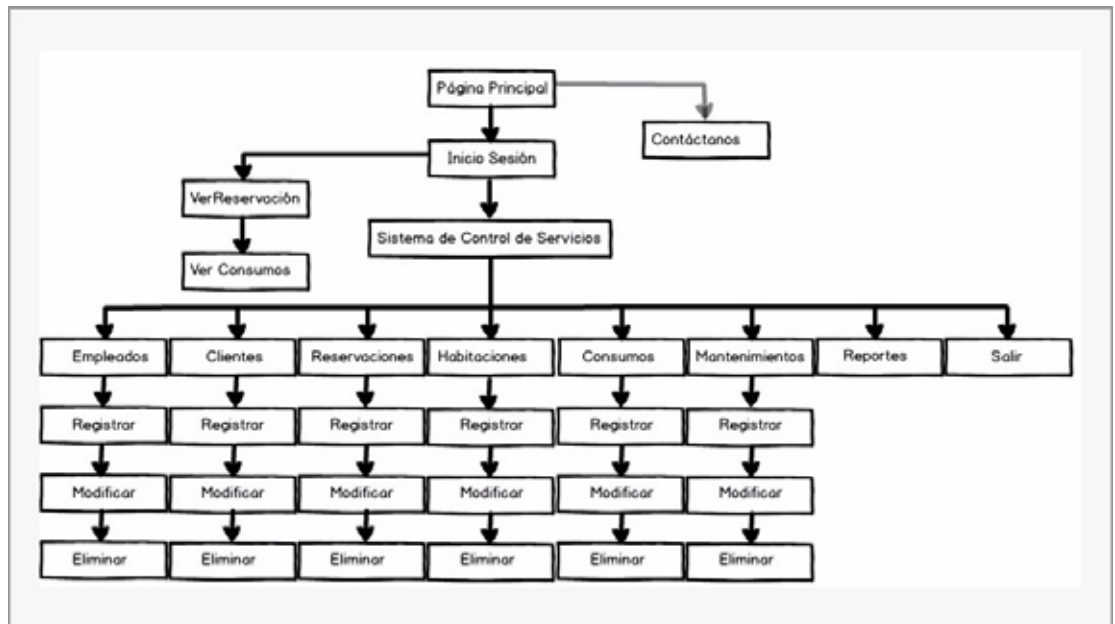


Figura 7. Mapa de Navegación.

Fuente: Manríquez, Mendoza (2017)

4.4 Fase IV: Desarrollo

Una vez reunida y organizada la información obtenida en las fases anteriores, se procedió al desarrollo del sistema el atendiendo principalmente a los requerimientos definidos por medio de las entrevistas y la observación directa. El software se codificó en base a los lenguajes HTML, JavaScript, CSS, PHP, MySQL. Todas estas herramientas aplicadas dentro del marco de trabajo (framework) *Laravel*, como el que se utilizó para crear la arquitectura del sistema y a su vez agilizar el proceso de codificación en base al paradigma antes mencionado de Modelo-Vista-Controlador.

A continuación, se muestran algunas descripciones de las vistas del sistema y su estructura. Se observa el diseño de la página de Inicio de sesión en el sistema. (Ver figura 8)

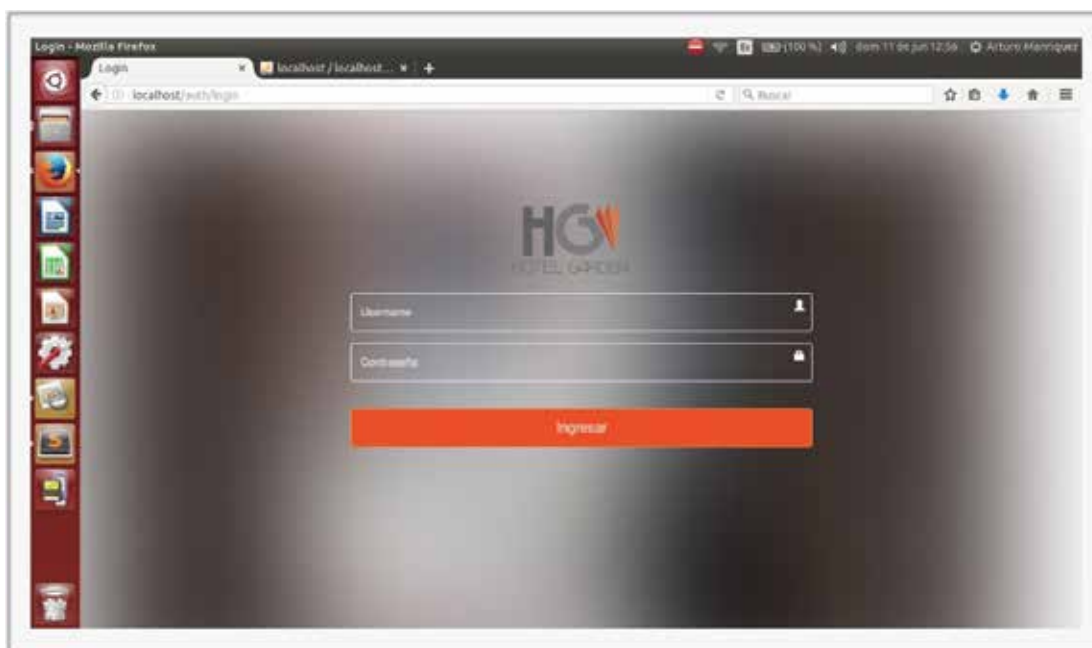


Figura 8. Página Inicio de Sesión.

Fuente: Manríquez, Mendoza (2017)

Como apreciamos en la figura anterior, para poder ingresar al área administrativa del sistema, el empleado debe utilizar su nombre de usuario y su contraseña.

Luego de acceder al área administrativa, se muestra un menú de navegación en la parte superior de la pantalla, con las diferentes opciones y funciones del sistema dejando libre el centro de la página para la ejecución de los diferentes formularios según la acción que se esté ejecutando. (Ver figura 9)

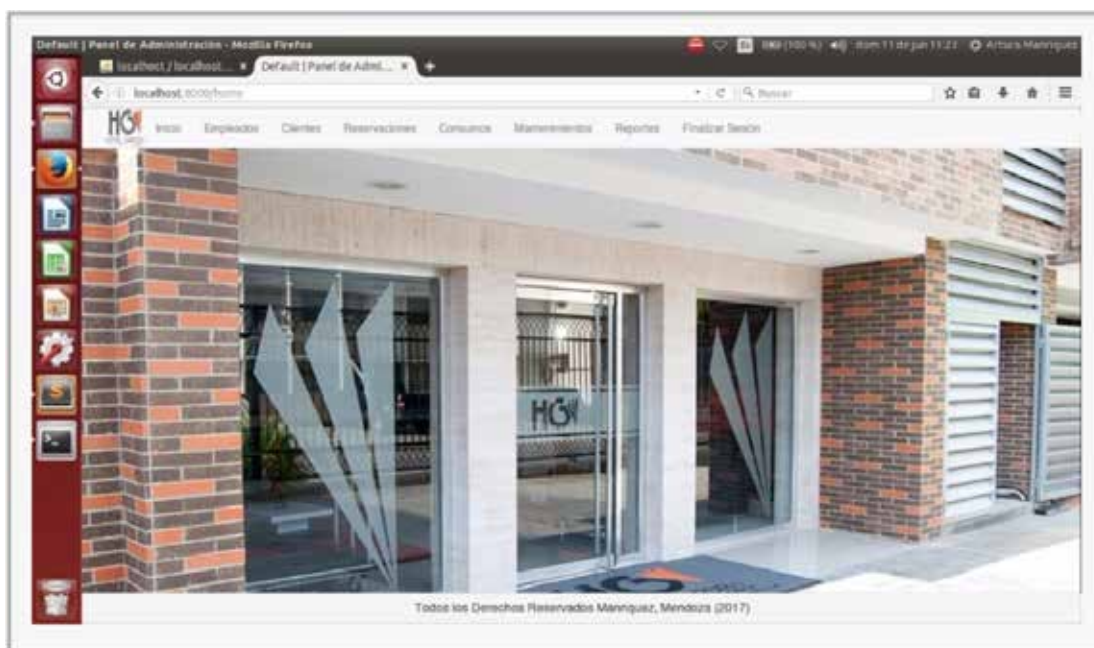
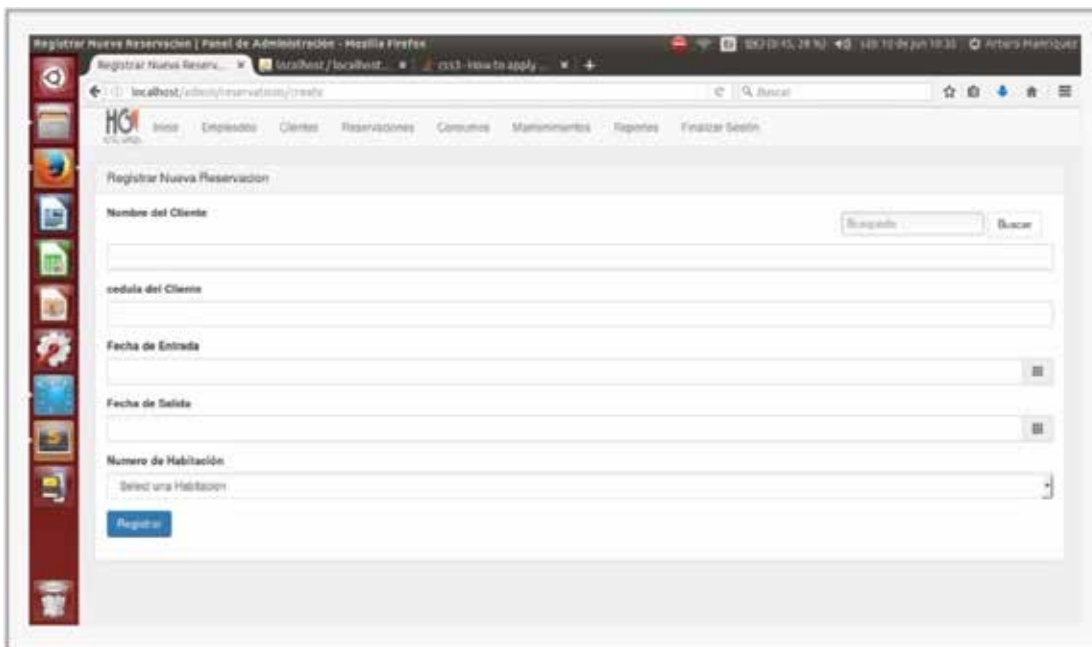


Figura 9. Inicio del área administrativa.

Fuente: Manríquez, Mendoza (2017)

A continuación, se muestran algunas imágenes de los formularios que se manejan en el sistema administrativo. (Ver figuras 10, 11 y 12)



The image shows a screenshot of a web browser displaying a form titled "Registrar Nueva Reservación". The browser's address bar shows the URL "localhost/admin/reservaciones/create". The page has a navigation menu with items: Inicio, Empleados, Clientes, Reservaciones, Compras, Mantenimientos, Reportes, and Finalizar Sesión. The form fields include: "Nombre del Cliente" with a search dropdown and "Buscar" button; "cedula del Cliente"; "Fecha de Entrada" with a calendar icon; "Fecha de Salida" with a calendar icon; and "Numero de Habitación" with a dropdown menu labeled "Selecciona una Habitación". A blue "Registrar" button is at the bottom of the form.

Figura 10. Vista de Manejo de Reservaciones.

Fuente: Manríquez, Mendoza (2017)

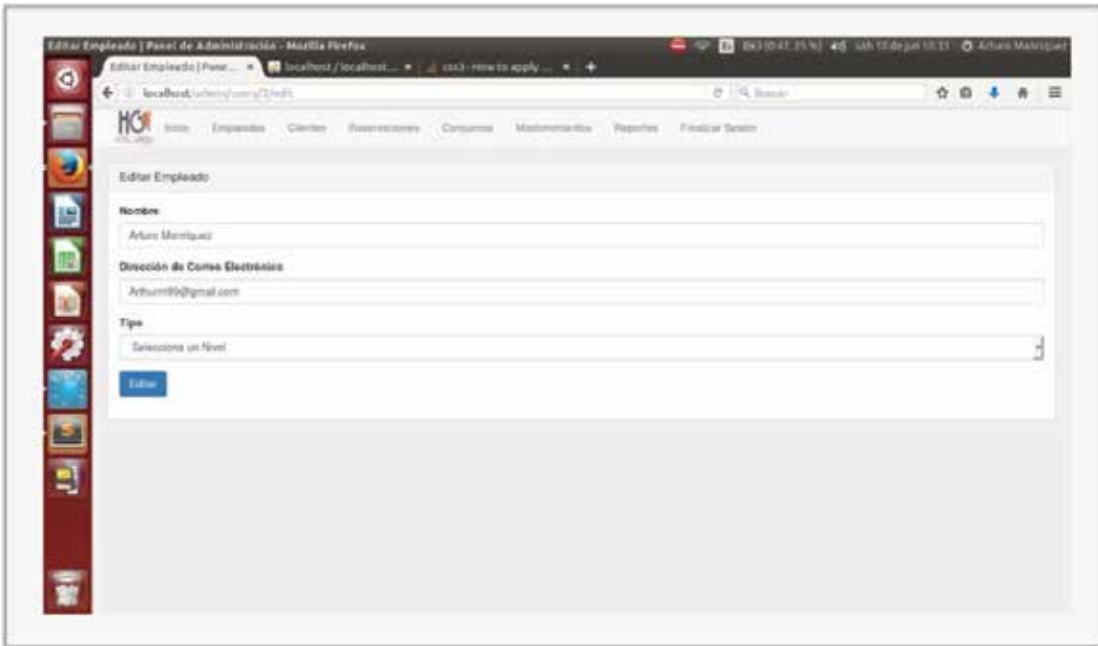


Figura 11. Formulario de Registro de Empleados

Fuente: Manríquez, Mendoza (2017)

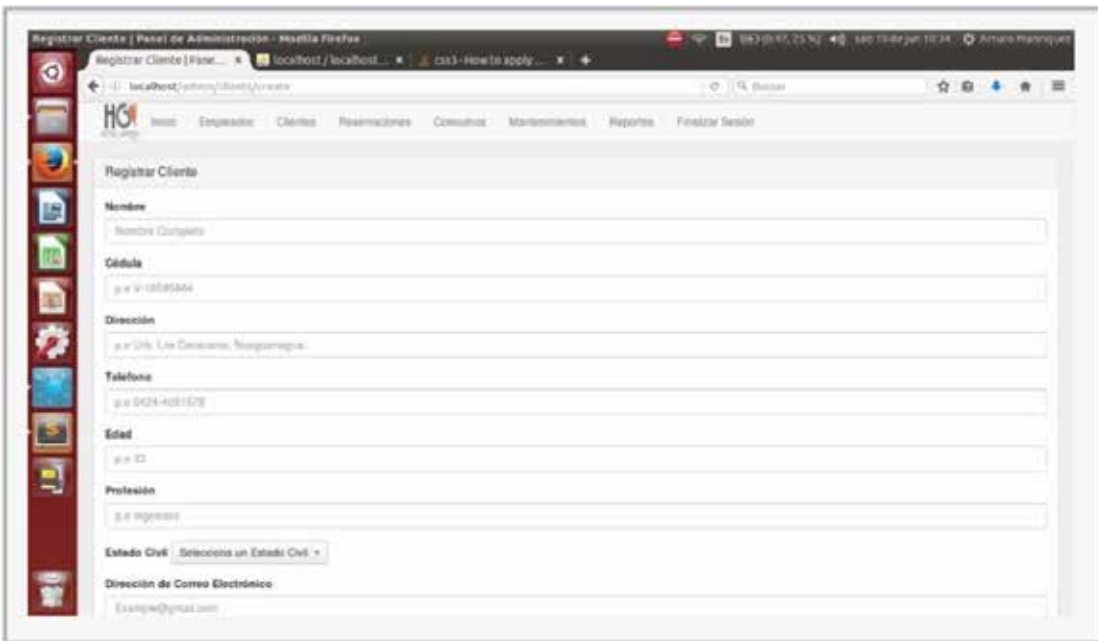


Figura 12. Formulario de Registro de Clientes.

Fuente: Manríquez, Mendoza (2017)

Como fue mencionado anteriormente, el Cliente, una vez registrado por el personal del hotel, tendrá la posibilidad de poder observar tanto sus consumos dentro de las instalaciones, como el estatus de proceso de los mismos, así como también podrá ordenar consumos adicionales y generar ordenes de mantenimiento, que luego serán procesadas por los departamentos respectivos del hotel y todo esto mediante nuestra Aplicación Móvil. A continuación, se muestran imágenes del formulario de Ingreso al Área de Clientes, Manejador de Ordenes de Consumo y Manejador de Ordenes de Mantenimiento. (Ver figuras 13,14, 15, 16,17,18,19)



Figura 13. Formulario de Ingreso al Área de Clientes.

Fuente: Manríquez, Mendoza (2017)

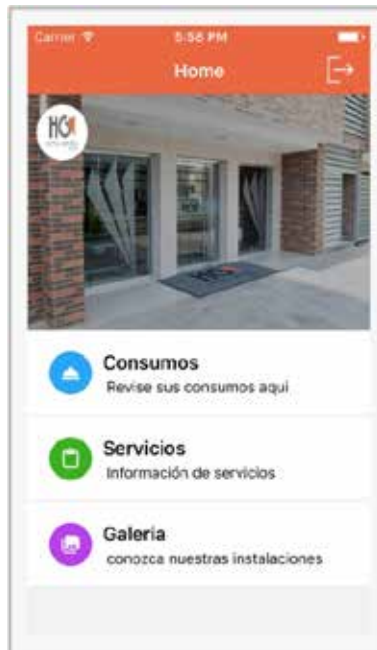


Figura 14. Vista de menú principal de sistema móvil.

Fuente: Manríquez, Mendoza (2017)

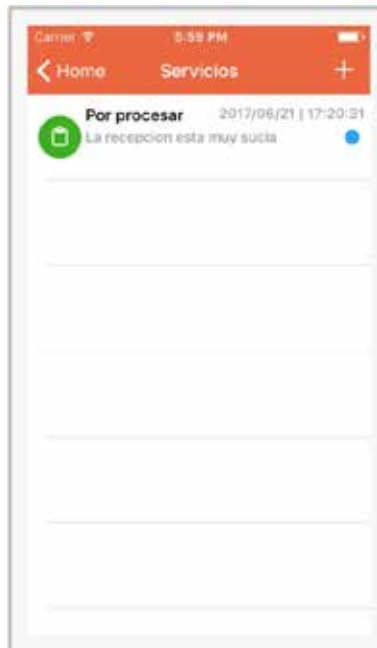


Figura 15. Lista de Manejo de Ordenes de Servicio.

Fuente: Manríquez, Mendoza (2017)

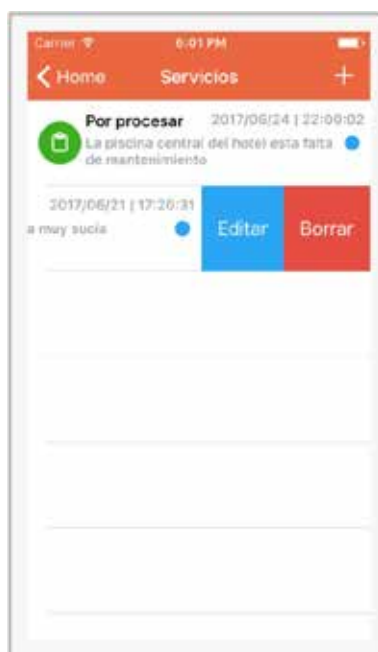


Figura 16. Vista de Edición de Ordenes de Servicios de Sistema Móvil.

Fuente: Manríquez, Mendoza (2017)



Figura 17. Vista de Nueva Orden de Consumo de Sistema Móvil.

Fuente: Manríquez, Mendoza (2017)



Figura 18. Lista de manejo de Ordenes de Consumo de Sistema Móvil.

Fuente: Manríquez, Mendoza (2017)



Figura 19. Vista de Fin de Sesión del Sistema Móvil.

Fuente: Manríquez, Mendoza (2017)

4.5. Fase V: Pruebas

En esta fase, se aplicaron pruebas a los diferentes módulos y funcionalidades del sistema, con la finalidad de que el funcionamiento de éste, según las acciones que sean ejecutadas fuera el correcto. Se llevó a cabo la realización de pruebas de caja negra, así como también, pruebas unitarias y de aceptación, se obtuvo los resultados requeridos luego de realizar las correcciones necesarias. De igual forma, se realizaron pruebas seguridad para asegurar la integridad de los datos procesados por el sistema, siendo este uno de los requerimientos del mismo.

Cuadro 19. Prueba: Ver Reservación.

Ver Reservación	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que el cliente vea la información correcta sobre el proceso de reservación de su habitación.	
Entrada: Usuario y Contraseña.	Caso de uso: Ver Reservación.
Resultado: Se muestra toda la información que al cliente le interesa saber sobre la reservación de su habitación.	
Observaciones: Falta información para mostrar y de interés para el cliente.	
Correcciones: Agregar datos adicionales para ampliar información de la reservación	

Fuente: Manríquez, Mendoza (2017)

Cuadro 20. Prueba: Registrar Reservación.

Registrar Reservación	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que se almacena de forma correcta la información de reservación en el sistema.	
Entrada: Usuario, Contraseña, Cliente.	Caso de uso: Administrar Reservación, Gestionar Reservación.
Resultado: La información de la Reservación es almacenada en la base de datos y puede ser utilizada en otros módulos.	
Observaciones: Si el cliente ya está registrado, no es necesario ingresarlo de nuevo en sistema.	
Correcciones: Se modificó el tipo de dato en varios campos de la tabla “reservations” por no coincidir con los tipos de datos de la tabla “clients”	

Fuente: Manríquez, Mendoza (2017)

Cuadro 21. Prueba: Modificar Reservación.

Modificar Reservación	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que se modifica de forma correcta la información de la Reservación.	
Entrada: Numero de Reservación	Caso de uso: Gestionar Reservación, Administrar Reservación.
Resultado: La información de Reservación es modificada en la base de datos y puede ser utilizada en otros módulos del sistema.	
Observaciones: Campos Fecha de entrada y Fecha de Salida no se modifican al ejecutar la acción de modificación.	
Correcciones: Se verifico el código y se encontró que había un error de nomenclatura en el nombre de las variables, como consecuencia, estos datos no eran almacenados en la base de datos.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 22. Prueba: Eliminar Reservación.

Eliminar Reservación	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que la reservación se elimina del sistema.	
Entrada: Cliente, Nro. de Reservación.	Caso de uso: Gestionar Reservación, Administrar Reservación.
Resultado: La información de Reservación es eliminada del sistema.	
Observaciones: La información de Reservación es eliminada del sistema de manera lógica, por lo que se mantiene disponible en la base de datos.	
Correcciones: Se modificó la manera de hacer el borrado lógico, implementando el uso de la función SoftDelete() de Laravel Collective	

Fuente: Manríquez, Mendoza (2017)

Cuadro 23. Prueba: Ingreso de Cliente.

Ingreso de Cliente	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que se almacena de forma correcta la información del cliente.	
Entrada: Nombre, Apellido, Teléfono, Cedula, Correo y Género.	Caso de uso: Gestionar Cliente, Administrar Cliente.
Resultado: La información del Cliente es almacenada en la base de datos y puede ser utilizada en otros módulos.	
Observaciones: Si el cliente ya está registrado, no es necesario ingresarlo de nuevo en sistema.	
Correcciones: Se implementaron nuevos campos en CRUD y BD para capturar más datos de interés para el hotel.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 24. Prueba: Modificación de información de cliente.

Modificación de información de cliente	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que se modifica de forma correcta la información del cliente.	
Entrada: Cedula del Cliente	Caso de uso: Gestionar Cliente, Administrar cliente.
Resultado: La información del cliente es modificada en la base de datos y puede ser utilizada en otros módulos del sistema.	
Observaciones: Si el cliente está eliminado o suspendido, no aparecerá en el panel de búsqueda.	
Correcciones: Se agregó un método que permitiera filtrar los clientes Activos y suspendidos para mostrar solo los activos en el panel de búsqueda.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 25. Prueba: Eliminación de información de cliente.

Eliminación de información de cliente	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que se elimina de forma correcta la información del cliente en el sistema.	
Entrada: Cedula, Nombre.	Caso de uso: Administrar Cliente.
Resultado: La información del cliente es eliminada exitosamente.	
Observaciones: La información del cliente es eliminada del sistema de manera lógica, por lo que se mantiene disponible en la base de datos.	
Correcciones: Se modificó el método para la eliminación lógica, utilizando el método SoftDelete de Laravel Collective.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 26. Ingreso al sistema administrativo.

Ingreso al sistema administrativo.	
Prueba: Aceptación	Tipo: Caja negra
Descripción: Comprobar que el sistema responde de acuerdo al rol.	
Entrada: usuario, contraseña.	Caso de uso: Iniciar Sesión.
Resultado: El ingreso al sistema es el correcto, y varía según el rol del usuario.	
Observaciones: Cambiar el método de encriptación del Password.	
Correcciones: Se modificó el tipo de encriptación a BCRYPT por ser este más eficiente	

Fuente: Manríquez, Mendoza (2017)

Cuadro 27. Prueba: Inicio de sesión desde la aplicación móvil.

Inicio de sesión desde la aplicación móvil	
Prueba: Aceptación	Tipo: Caja blanca
Descripción: Comprobar que cuando se abra la aplicación esta mande al cliente a la pantalla principal si ya este había iniciado sesión previamente.	
Entrada: Cliente.	Caso de uso: Inicio de sesión.
Resultado: El registro fue modificado exitosamente en la base de datos, mas la información en el sistema no reflejo los cambios.	
Observaciones: Se pudo observar que el problema provenía del controlador de servicios el cual no refrescaba la vista después de haber modificado una orden de servicio.	
Correcciones: Se modifiko el método viewWillAppear del controlador de servicios para que este hiciera de nuevo la petición al servidor luego de que se modifica un registro.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 28. Prueba: Edición de orden de servicio desde la aplicación móvil.

Edición de orden de servicio	
Prueba: Aceptación	Tipo: Caja blanca
Descripción: Comprobar que se modificó de forma correcta la información de la orden de servicio en el sistema.	
Entrada: Cliente.	Caso de uso: Modificar orden de servicio.
Resultado: El registro fue modificado exitosamente en la base de datos, más la información en el sistema no reflejo los cambios.	
Observaciones: Se pudo observar que el problema provenía del controlador de servicios el cual no refrescaba la vista después de haber modificado una orden de servicio.	
Correcciones: Se modificó el método viewWillAppear del controlador de servicios para que este hiciera de nuevo la petición al servidor luego de que se modifica un registro.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 29. Prueba: Creación de orden de consumo desde la aplicación móvil.

Creación de orden de consumo	
Prueba: Aceptación	Tipo: Caja blanca
Descripción: Comprobar que la orden de consumo se genere exitosamente en el sistema.	
Entrada: Cliente.	Caso de uso: Gestionar orden de consumo.
Resultado: El registro fue creado exitosamente en la base de datos, pero cuando el cliente quiere generar una nueva orden de consumo los formularios se muestran con la información de la orden creada anteriormente.	
Observaciones: Se pudo observar que el problema provenía del controlador de creación de consumos el cual no borraba la información de las variables cuando se regresaba a la vista anterior.	
Correcciones: Se modificó el método viewWillDisapear del controlador de creación de consumos para que este borrara toda la información de las variables del formulario antes de que esta vista desaparezca.	

Fuente: Manríquez, Mendoza (2017)

Cuadro 30. Prueba: Borrado de orden de consumo desde la aplicación móvil.

Borrado de orden de consumo	
Prueba: Aceptación	Tipo: Caja blanca
Descripción: Comprobar que la orden de consumo se borra lógicamente de la base de datos.	
Entrada: Cliente.	Caso de uso: Gestionar orden de consumo.
Resultado: La aplicación muestra un mensaje diciendo que la orden de consumo fue borrada exitosamente pero no se borra de la vista de consumos.	
Observaciones: Se pudo observar que el problema provenía del controlador de consumos el cual hacia la petición al servidor con el método post de http y no con el método delete.	
Correcciones: En el controlador de consumos se modificó el método deleteConsumo o para que se hiciera la petición al servidor con el método http delete.	

Fuente: Manríquez, Mendoza (2017)

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

Una vez que fue finalizado el desarrollo del sistema y finalizada la investigación del proyecto para el control y seguimiento de consumos y servicios de los clientes del hotel garden valencia en el estado Carabobo. Se concluye que:

- Se detectaron las principales fallas y problemas en diversos procesos llevados a cabo en el hotel.
- En base a las diferentes problemáticas encontradas, se creó una lista de requerimientos para realizar un sistema basado en la web y sistema móvil, que permitiera controlar y dar seguimiento a consumos y servicios del hotel.
- Con la realización del sistema web y móvil para el control de consumos y servicios en el hotel, se solucionó la problemática planteada en el capítulo I de esta investigación.
- El sistema satisface las necesidades descritas por los empleados y clientes de Hotel Garden valencia, C.A.
- El sistema se encuentra debidamente documentado lo que permite posibles mejoras por parte de la empresa sin mayores dificultades.
- La utilización de la metodología *Scrum* permitió la constante mejora de los requerimientos del sistema, permitiendo crear un sistema más ajustado a las necesidades.

5.2 Recomendaciones

A continuación, se plantean una serie de recomendaciones a fin de mantener el sistema en óptimas condiciones, así como también de mejoras que se pueden realizar para ampliar el alcance del mismo, las cuales no están previstas en esta investigación.

- Utilizar las versiones correspondientes de los lenguajes PHP y MySQL para el servidor web compatible con la aplicación web desarrollada.
- Realizar un respaldo regularmente para evitar la pérdida de datos de forma inesperada de la base de datos.
- Evitar realizar cambios mayores a la interfaz del sistema ya que puede causar una confusión en los usuarios regulares.
- Desarrollar el sistema para otras plataformas como Android, que permita a los usuarios una mayor accesibilidad al sistema.
- Incorporar módulos de Facturación, Cuentas por Cobrar, Cuentas por Pagar y derivados de éstos a fin de agregarle una mayor funcionalidad al sistema.
- Agregar un módulo de reservación para los clientes en la aplicación móvil.
- Agregar notificaciones push a la aplicación web y móvil.

REFERENCIAS

Bibliográficas

Flores, P. (2015). *Desarrollo de un sistema web para el control del mantenimiento preventivo de equipos en línea de producción. Caso estudio: Fábrica Nacional de pañales desechables*. Carabobo: Universidad José Antonio Páez.

Kenneth, S. (2012). *Essential Scrum: A Practical Guide to the Most Popular Agile Process. Primera Edición*. Nueva Jersey: Addison-Wesley.

Arias, F (2012). *El Proyecto de Investigación: Introducción a la metodología científica. Sexta Edición*. Caracas, Venezuela

Villasana E. y Villaquirán F. (2016). *Desarrollo de un sistema web para el proceso administrativo en los talleres de latonería y pintura multiservicios Latón Car, C.A. Carabobo: Universidad José Antonio Páez*

Electrónicas

Ortega, J. (2015). *Paradigma de la programación. (En línea).*

Disponible en: <http://paradigmasiut.blogspot.com/2013/04/metodologia-de-desarrollo-de-software.html>

Ortiz, J. (2012). *Repositorio Digital UTN. (En línea).* Universidad Técnica del Norte.

Disponible en: <http://repositorio.utn.edu.ec/handle/123456789/1078>

Culqui, E. (2015). *Sistema Web para el registro de reservaciones y control de hospedaje en el Hotel Acapulco de la ciudad de Ambato. (En línea)*

<http://repositorio.uta.edu.ec/jspui/handle/123456789/10388>

Wikipedia (2017). *Laravel. (En línea)*

Disponible en: <https://es.wikipedia.org/wiki/Laravel>

Ramon Gonzales (2015). *El Catalejo de Rami: Una mirada al mundo de la web y las tecnologías. (En línea)*

Disponible en: <http://elcatalejoderami.blogspot.com/2015/02/scrum-para-equipos-extra-pequenos.html>

Scrum.Org (2017). *The home of Scrum. Professional Scrum Training, Resources and Certifications to learn and prove your knowledge. (En línea) (En Inglés)*

Disponible en: <https://www.scrum.org/>

Peres, R. (2017). *Model-View-Controller (MVC) in iOS: a Modern Approach. (En línea) (En Inglés).*

Disponible en: <https://www.raywenderlich.com/132662/mvc-in-ios-a-modern-approach>