

**UNIVERSIDAD NACIONAL DE INGENIERÍA
RECINTO UNIVERSITARIO “SIMÓN BOLÍVAR”
FACULTAD DE ELECTROTECNIA Y COMPUTACIÓN**



TRABAJO MONOGRÁFICO

“Desarrollo del Sistema Administrativo para el Hotel Paraíso
Condominiuns (SAPCON)”

ELABORADO POR:

Br. Luis Carlos Flores Baltodano (2013-61015)

Br. Rommel Humberto Pérez Gómez (2012-41604)

TUTOR:

Ing. Rhene Gonzalo Hernández Berroteran

PARA OPTAR AL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

Managua, noviembre del año 2021

AGRADECIMIENTO

Primeramente, doy infinitas gracias a Dios por permitirme llegar tan lejos en mi vida y poder cumplir una de las metas más importantes.

A mi familia por ser un pilar tan importante en mi formación académica, ya que sin su ayuda esto no sería posible.

A mi tutor Ing. Rhené Gonzalo Hernández Berroterán quien me ayudo con sus sugerencias y recomendaciones para el desarrollo de esta tesis monográfica.

DEDICATORIA

Dedicamos este trabajo a nuestros padres ya que ellos ha sido el pilar más importante en nuestra formación académica, siempre demostraron su amor y su apoyo incondicional en todo momento, aconsejándonos y guiándonos por el camino correcto, regalándonos ánimos cuando estos estaban bajos y siempre demostrándonos que el sacrificio verdadero siempre es recompensado de la mejor manera.

A nuestras familias en general, porque nos han brindado su apoyo incondicional y por compartir con nosotros buenos y malos momentos.

Índice

Capítulo I: Descripción General	1
<i>I. INTRODUCCIÓN</i>	1
<i>II. ANTECEDENTES DEL PROBLEMA</i>	3
<i>III. JUSTIFICACIÓN</i>	4
<i>IV. OBJETIVOS</i>	5
<i>4.1 OBJETIVO GENERAL:</i>	5
<i>4.2 OBJETIVOS ESPECIFICOS:</i>	5
Capítulo II: Marco Teórico.	6
<i>V. MARCO TEÓRICO</i>	7
<i>5.17 Modelos de desarrollo del software.</i>	11
<i>5.18 Metodología de desarrollo.</i>	11
<i>5.13.1 Tipo de metodología.</i>	12
<i>5.13.2 Fases del Modelo.</i>	12
<i>Fase de concepción o inicio:</i>	12
<i>Fase de elaboración:</i>	13
<i>Fase de construcción:</i>	13
<i>Fase de transición:</i>	13
Capítulo III: Diseño Metodológico y Estudio de Factibilidad.....	14
<i>VI. DISEÑO METODOLÓGICO</i>	15
<i>6.1 Etapa 1 Fase de concepción o inicio del sistema administrativo para el hotel Paraíso Condominiuns.</i>	16
<i>6.2 Etapa 2. Fase de elaboración del sistema.</i>	17
<i>6.3 Etapa 3. Fase de construcción del sistema.</i>	18
<i>6.4 Etapa 4. Fase de transición del sistema.</i>	19
<i>VII. ESTUDIO DE FACTIBILIDAD</i>	21
<i>7.1 Factibilidad Operativa.</i>	21
<i>7.2 Factibilidad Técnica.</i>	22
<i>7.2.1 Requerimientos de software</i>	23
<i>7.2.2 Requerimientos de Hardware.</i>	24
<i>7.3 Factibilidad Económica</i>	25

7.4 Aspectos Legales	28
Capítulo IV: Análisis del Sistema.....	29
VIII. ANÁLISIS DEL SISTEMA.....	30
8.1. Requerimientos del sistema.....	30
Desarrollo del Sistema.....	30
Análisis.....	31
Diseño del Sistema.....	33
Diseño de Base de Datos.....	33
Interfaces	34
Programación del Sistema	34
8.2.1 Casos de Uso del sistema.....	35
• Módulo de Administración.....	35
• Módulo de Reservacion.....	37
Sección de Voucher	38
• Módulo de Cliente	39
• Módulo de Habitación.....	41
8.2.2 Plantilla de Casos de Uso del sistema	42
Capítulo V: Diseño y Modelado del Sistema.....	50
IX. DISEÑO DEL SISTEMA	51
9.1. Diagrama Entidad Relación	51
9.2. Diagrama de Actividades	52
9.2.1 Ingreso al sistema.....	52
Capítulo VI: Codificación, Pruebas e Implementación del Sistema.....	55
X. CODIFICACIÓN DEL SISTEMA.....	56
10.1 Arquitectura del Sistema.....	56
10.2 Estructura del Sistema	57
XI. DESARROLLO DE PRUEBAS FUNCIONALES.....	58
XIII.IMPLEMENTACIÓN DEL SISTEMA	69
Capítulo VII: Conclusiones y Recomendaciones	72
XIII. CONCLUSIONES Y RECOMENDACIONES	73
13.1 Conclusiones	73
13.2 Recomendaciones	74
XV. ANEXOS.....	78

14.1. Glosario	78
14.2. Manual de usuario Diseño de Entradas	79

Capítulo I: Descripción General

I.INTRODUCCIÓN

El presente documento detalla una propuesta para el análisis y desarrollo de un sistema administrativo, que optimice los procesos de registros y manejo de servicios en el Hotel Paraíso Condominiuns. Para entendimiento del mismo, se lo ha dividido en los siguientes capítulos:

CAPÍTULO I denominado “DESCRIPCION GENERAL”, identifica el problema que actualmente se puede observar en la realización de los procesos de registro y manejo inherentes al hotel Paraíso Condominiuns.

CAPÍTULO II denominado “MARCO TEÓRICO”, muestra las investigaciones previas que sirven de soporte para el desarrollo de la investigación, además la información de estudios similares anteriormente realizados, así como los aspectos conceptuales que sustentan el tema en general, y el conjunto de conceptos y fundamentos teóricos que han sido analizados en base al problema establecido.

CAPÍTULO III denominado “DISEÑO METODOLOGICO Y ESTUDIO DE FACTIBILIDAD”, define el tipo de investigación que ha sido desarrollada, el tratamiento de los procesos que señala la modalidad de investigación, asimismo se presenta el tipo de análisis de los datos según el tipo de investigación.

CAPÍTULO IV denominado “ANALISIS DEL SISTEMA”, determina una extensa información sobre el tema investigado en donde se detalla conceptos, métodos y la forma de programar, además se muestra paso a paso el desarrollo de la aplicación y las etapas que se cumplieron de acuerdo a metodología seleccionada.

CAPÍTULO V denominado “DISEÑO Y MODELADO DEL SISTEMA”, determina una extensa información sobre el tema investigado en donde se detalla específicamente, métodos y la forma en la que se diseñó el sistema, además se muestra paso a paso el modelado y las etapas para elaborar este mismo.

CAPÍTULO VI denominado “CODIFICACION, PRUEBAS E IMPLEMENTACION DEL SISTEMA”, determina y detalla finalmente los resultados de la codificación del sistema, además de detallar de forma clara las pruebas que se realizaron para posteriormente implementar el mismo.

CAPÍTULO VII denominado “CONCLUSIONES Y RECOMENDACIONES”, expone de forma clara y concisa las consideraciones más relevantes que se han obtenido al finalizar el proyecto, además se indican recomendaciones para futuras modificaciones.

II. ANTECEDENTES DEL PROBLEMA

El Hotel Paraíso Condominiuns en sus principios era un hotel pequeño de cuatro habitaciones con un concepto diferente, esta novedad se basa en el sistema de Condominiuns que consiste en el alquiler de sus espacios similar a un apartamento donde el cliente es dueño temporal de la unidad. Contaba con un personal reducido de cuatro miembros y una cartera de servicios reducidos.

Con el paso del tiempo el Hotel Paraíso Condominiuns ha venido creciendo cada vez más y gracias a esto ha tenido que aumentar su equipo de personal para responder a la demanda y así mismo ha crecido la cartera de servicios que ofrece el hotel, aumentado de esta manera la cantidad de clientes, viéndose en la necesidad de aumentar su capacidad de alojamiento hasta un total de dieciséis habitaciones con vistas a futuro de seguir creciendo en infraestructura y más servicios.

Es importante mencionar que desde su inicio el Hotel Paraíso Condominiuns ha gestionado su área administrativa, registro de clientes, facturación y cobros y el control de servicios de manera manual, registrando en facturas membretadas que posteriormente son archivadas; y en la actualidad esto les presenta un gran problema ya que la cantidad de facturas que han acumulado a lo largo del tiempo y la nueva cartera de servicios facturados, les dificulta la manera de llevar todas las gestiones.

El trabajo presentado en este documento se debe tomar como una propuesta a desarrollarse. A través de este sistema web se pretende mejorar la calidad y efectividad de la gestión de registros, administración de los recursos humanos, facturación, cobros y monitoreo de los servicios, ayudando a realizar de mejor manera la gestión administrativa de los procesos y servicios antes mencionados en el Hotel Paraíso Condominiuns.

III. JUSTIFICACIÓN

Con el paso del tiempo el Hotel Paraíso Condominiuns ha crecido considerablemente, aumentando su personal requerido, la cantidad de servicios y así mismo la cantidad de habitaciones.

La gran competencia en el sector hotelero y las exigencias del mercado, hacen imprescindible estar siempre al día con los avances tecnológicos, sobre todo los referidos al software de gestión. Por eso, disponer de un sistema web que constantemente este evolucionando y adoptando las diferentes necesidades que esta empresa pueda solicitar en el transcurso del tiempo es una necesidad ineludible, por tanto, con este sistema que automatizara los aspectos ya mencionados, ayudará en gran manera tanto al cliente como al trabajador como tal, que se encargará de llevar la administración de los diferentes servicios que este ofrece.

Gracias a los factores expuestos anteriormente el Hotel Paraíso Condominiuns desea abandonar su metodología tradicional de llevar las gestiones de registros, facturación, control de servicios y cobros de manera manual.

La implementación de este sistema administrativo genera muchos beneficios, como la automatización de los procesos del hotel, desde las reservas de clientes, hasta la gestión de ventas y servicios, facturación y cobros, recursos humanos, mantenimiento, calidad.

Con la implementación del sistema administrativo se podrá tener un control total de todos los procesos mencionados. El sistema generará informes sobre las instalaciones disponibles, estados de cuentas y servicios, además permitirá concentrar toda la información del hotel y ser resguardada de manera segura, la cual estará disponible en todo momento para su respectivo análisis y contribuir en la toma de decisiones.

IV. OBJETIVOS

4.1 OBJETIVO GENERAL:

Desarrollar un sistema administrativo para la gestión de los procesos y servicios del hotel “Paraíso Condominiuns”, mediante herramientas y metodologías basadas en tecnología web.

4.2 OBJETIVOS ESPECIFICOS:

1. Analizar los requerimientos y alcances del sistema administrativo del Hotel Paraíso Condominiuns.
2. Diseñar el modelo seleccionado del sistema administrativo sobre la base de los requerimientos identificados utilizando la metodología RUP y la herramienta de Lenguaje de Modelado Unificado (UML).
3. Codificar el sistema administrativo usando el lenguaje de programación Python y el gestor de bases de datos MySQL
4. Implantar el sistema administrativo y realizar los ajustes correspondientes de acuerdo a los resultados que fueron obtenidos de esta inserción.

Capítulo II: Marco Teórico.

V. MARCO TEÓRICO

Explicaremos en esta parte, para una mayor comprensión del presente protocolo, los conceptos básicos de los sistemas de información, así mismo las herramientas y metodologías, que serán utilizadas para el desarrollo del sistema de información gerencial.

5.1 Sistemas de información: Es un conjunto de elementos que interactúan entre sí con un fin común; que permite que la información esté disponible para satisfacer las necesidades en una organización, un sistema de información no siempre requiere contar con recurso computacional, aunque la disposición del mismo facilita el manejo e interpretación de la información por los usuarios. (Somerville P, 2008)

Un Sistema de Información realiza cuatro actividades básicas:

- **Entrada de información:** proceso en el cual el sistema toma los datos que requiere.
- **Almacenamiento de información:** puede hacerse por computadora o archivos físicos para conservar la información.
- **Procesamiento de la información:** permite la transformación de los datos fuente en información que puede ser utilizada para la toma de decisiones
- **Salida de información:** es la capacidad del sistema para producir la información procesada o sacar los datos de entrada al exterior.

5.2 Entorno de desarrollo: El entorno de desarrollo es un conjunto de procedimientos y herramientas que se utiliza, para desarrollar un código fuente o programa. Este término, se utiliza a veces, como sinónimo de entorno de desarrollo integrado (IDE), que es la herramienta de desarrollo de software utilizado para escribir, generar, probar y depurar un programa. También proporcionan a los desarrolladores una interfaz. (Somerville P, 2008)

5.3 Visual Code: Visual Studio Code, Es un editor de código fuente desarrollado por Microsoft para Windows , Linux y macOS . Incluye soporte para depuración, control de Git integrado, resaltado de sintaxis, finalización de código inteligente, fragmentos de código y refactorización de código. También es personalizable, de modo que los usuarios pueden cambiar el tema del editor, los métodos abreviados de teclado y las preferencias. Es gratuito y de código abierto. (Somerville P, 2008)

Por esa razón que utilizaremos el (IDE) Visual Studio Code por combinar el código con la interfaz de usuario optimizada siendo un editor moderno con asistencia y navegación de código enriquecido y una experiencia de depuración integrada, sin la necesidad de un IDE completo.

5.4 Lenguaje de programación Python: Python es un lenguaje sencillo y rápido de aprender. Su sintaxis es parecida a escribir cualquier texto en inglés, pero con la potencia de sus principales competidores en el Backend. (Pressman Roger, 2010)

Utilizaremos el lenguaje Python debido a una de sus características principales el cual es permitir el dar acceso a todas las librerías disponibles y así poderlas integrar al código de manera sencilla, esta combinación de Python Django permite desarrollar muy rápido para la web.

5.5 Django: Django es un framework de desarrollo Web que ahorra tiempo y hace que el desarrollo Web sea divertido. Utilizando Django se puede crear y mantener aplicaciones Web de alta calidad con un mínimo esfuerzo. (Pressman Roger, 2010)

Se implementará Django como framework para evitar lo conocido como código espagueti, ya se tiene claro que los sistemas administrativos necesitan una capa lógica para el manejo de los datos (backend) y una capa visual para la interacción con los usuarios.

5.6 Frameworks: Se define framework como un marco de trabajo que se utiliza para desarrollar un software determinado. Si se lleva este concepto al mundo de la creación de páginas web, se puede decir que un framework es un entorno de trabajo que hace mucho más fácil la programación o desarrollo de una web. (Hernandez U, 2017)

5.7 MYSQL: es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL). (Margaret Rouse, 2015).

5.8 Modelo – Vista – Controlador (MVC): es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. (Hernandez U, 2017)

Se trata de un modelo muy maduro y que ha demostrado su validez a lo largo de los años en todo tipo de aplicaciones, y sobre multitud de lenguajes y plataformas de desarrollo.

5.9 Modelo: contiene una representación de los datos que maneja el sistema, su lógica de negocio, y sus mecanismos de persistencia. (Hernandez U, 2017)

5.10 Vista: o interfaz de usuario, que compone la información que se envía al cliente y los mecanismos interacción con éste. (Hernandez U, 2017)

5.11 Controlador: que actúa como intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a las necesidades de cada uno. (Hernandez U, 2017)

5.12 Bootstrap 4: es un framework desarrollado y liberado por Twitter que tiene como objetivo facilitar el diseño web. Permite crear de forma sencilla webs de diseño adaptable, es decir, que se ajusten a cualquier dispositivo y tamaño de pantalla y siempre se vean igual de bien. Es Open Source o código abierto, por lo que lo podemos usar de forma gratuita y sin restricciones. (María, 2016)

5.13 JavaScript: es un lenguaje de programación interpretado, se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. (Gauchat J, 2012)

5.14 HTML: Html (HyperText Markup Language, version 5) es la quinta revision del lenguaje HTML. Esta versión (aun en desarrollo), y que con CSS3, define los nuevos estándares de desarrollo Html, rediseñando el código para resolver problemas y actualizándolo así a nuevas necesidades. No se limita a solo crear nuevas etiquetas o atributos, sino que incorpora muchas características nuevas y proporciona una plataforma de desarrollo de complejas aplicaciones web (mediante las Apis). (Gauchat J, 2012)

5.15 JSON: significa **J**ava **S**cript **O**bject **N**otation, es un lenguaje independiente, auto descriptivo y fácil de entender, con formato de intercambio de datos ligero entre el front-end y back-end es decir, intercambio de datos entre la aplicación y el servidor. (Gauchat J, 2012)

5.16 OpenSSL: es un juego de herramientas robusto y con todas las funciones para los protocolos de Seguridad de la capa de transporte (TLS) y la Capa de sockets seguros (SSL). También es una biblioteca de criptografía de uso general. En este caso será utilizado para la encriptación de información sensible que sea proporcionado por el usuario como, por ejemplo: usuarios, contraseñas y otros tipos de datos importantes que sean considerados por la empresa. El framework de PHP Laravel utiliza dicho método de encriptación y gracias a esto proporciona encriptación AES-256 y AES-128, además todos los valores cifrados se firmarán utilizando un código de autenticación de mensaje (MAC) para que su valor subyacente no se pueda modificar una vez cifrado. (Lopez Rosciano, 2015)

5.17 Modelos de desarrollo del software.

Una metodología de desarrollo de software se refiere a un marco de trabajo que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información. (Lopez Rosciano, 2015)

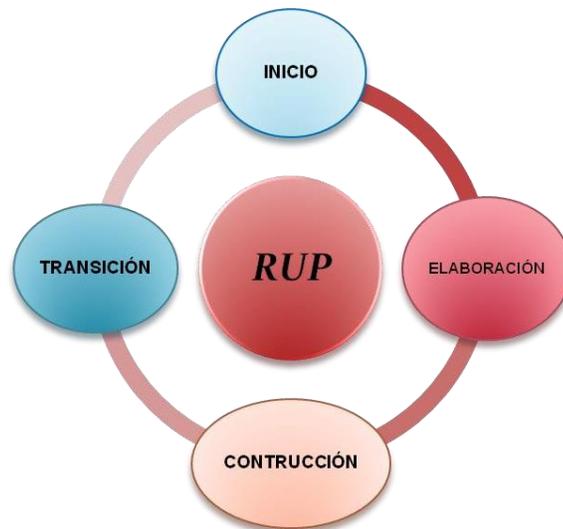
Entre los modelos de procesos del software están, el lineal secuencial, construcción de prototipos, Desarrollo Rápido de Aplicaciones (DRA), modelos evolutivos (incremental, espiral WINWIN y desarrollo concurrente). También existe el desarrollo basado en componentes y el modelo de métodos formales. (Lopez Rosciano, 2015)

5.18 Metodología de desarrollo.

Se trata del proceso cuya finalidad es desarrollar Reservations o soluciones para un cliente o mercado en particular, teniendo en cuenta factores como los costes, la planificación, la calidad y las dificultades asociadas. A todo esto, es a lo que denominamos metodologías de desarrollo de software. Es decir, se trata del proceso que se suele seguir a la hora de diseñar una solución o un programa específico. Tiene que ver, por tanto, con la comunicación, la manipulación de modelos y el intercambio de información y datos entre las partes involucradas. (Lopez Rosciano, 2015)

5.13.1 Tipo de metodología.

El modelo de desarrollo RUP sigue una serie de etapas de forma sucesiva, cada fase representa un ciclo de desarrollo en la vida de un Reservacion de software.



Fases del modelo en cascada.

Fuente: (Lopez Rosciano, 2015)

5.13.2 Fases del Modelo.

El modelo de desarrollo RUP sigue una serie de etapas de forma sucesiva, la etapa siguiente empieza cuando termina la etapa anterior.

Las fases que componen el modelo son las siguientes:

Fase de concepción o inicio:

Esta fase tiene por finalidad definir la visión, los objetivos y el alcance del proyecto, tanto desde el punto de vista funcional como del técnico, obteniéndose

como uno de los principales resultados una lista de los casos de uso y una lista de los factores de riesgo del proyecto. El principal esfuerzo está radicado en el Modelamiento del Negocio y el Análisis de Requerimientos. Es la única fase que no necesariamente culmina con una versión ejecutable

Fase de elaboración:

Esta fase tiene como principal finalidad completar el análisis de los casos de uso y definir la arquitectura del sistema, además se obtiene una aplicación ejecutable que responde a los casos de uso que la comprometen. A pesar de que se desarrolla a profundidad una parte del sistema, las decisiones sobre la arquitectura se hacen sobre la base de la comprensión del sistema completo y los requerimientos (funcionales y no funcionales) identificados de acuerdo al alcance definido.

Fase de construcción:

Esta fase está compuesta por un ciclo de varias iteraciones, en las cuales se van incorporando sucesivamente los casos de uso, de acuerdo a los factores de riesgo del proyecto. Este enfoque permite por ejemplo contar en forma temprana con versiones del sistema que satisfacen los principales casos de uso. Los cambios en los requerimientos no se incorporan hasta el inicio de la próxima iteración.

Fase de transición:

Esta última fase se inicia con una versión “beta” del sistema y culmina con el sistema en fase de producción.

Capítulo III: Diseño Metodológico y Estudio de Factibilidad.

VI. DISEÑO METODOLÓGICO

La tesis desarrollada contemplará las fases de la metodología RUP, ya que nos proporciona un entorno de proceso configurable basado en estándares que permiten estructurar de forma clara las diversas fases del proyecto que se diferencian claramente entre sí. Como cada una de las fases concluye con un hito, el proceso de desarrollo es muy fácil de comprender. El punto clave del modelo reside en la documentación de todos y cada uno de los pasos de proceso o etapas que se deberán ejecutar siempre y cuando el anterior haya concluido y gracias a esto nos dará una mejor percepción de las fases o procesos a seguir para el buen desarrollo del proyecto.

En el siguiente apartado se dará a conocer la metodología, procesos y herramientas que se utilizarán para el desarrollo de este proyecto. Se describirán los procesos que intervinieron en todo el desarrollo del sistema.

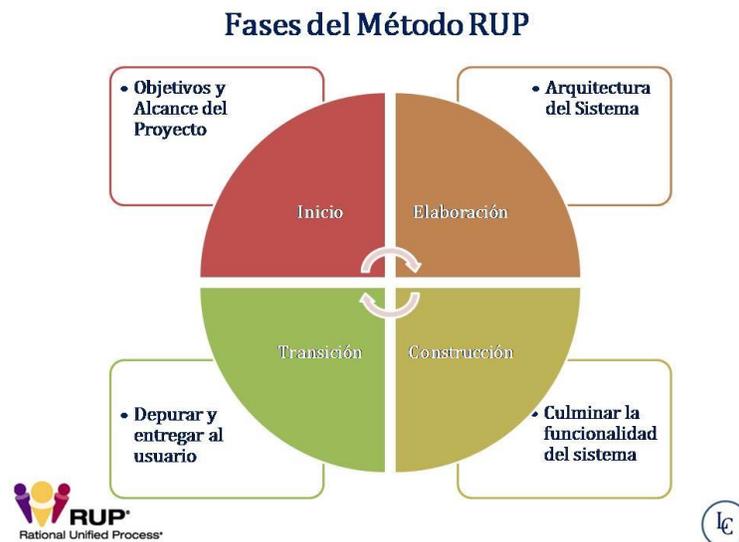


Ilustración 1. Metodología RUP aplicada al proyecto y sus etapas requeridas.

6.1 Etapa 1 Fase de concepción o inicio del sistema administrativo para el hotel Paraíso Condominiuns.

Durante esta fase de inicio se definirá el modelo del negocio y el alcance del proyecto. Se identificarán todos los actores y Casos de Uso, y se diseñarán los Casos de Uso más esenciales (aproximadamente el 20% del modelo completo). Se desarrolla, un plan de negocio para determinar que recursos deben ser asignados al proyecto.

Los objetivos de esta fase son:

- Establecer el ámbito del proyecto y sus límites.
- Encontrar los Casos de Uso críticos del sistema, los escenarios básicos que definen la funcionalidad.
- Mostrar al menos una arquitectura candidata para los escenarios principales.
- Estimar el coste en recursos y tiempo de todo el proyecto.
- Estimar los riesgos, las fuentes de incertidumbre.

Los resultados de la fase de inicio deben ser:

- Un documento de visión: Una visión general de los requerimientos del proyecto, características clave y restricciones principales.
- Modelo inicial de Casos de Uso (10-20% completado).
- Un glosario inicial: Terminología clave del dominio.
- El caso de negocio.
- Lista de riesgos y plan de contingencia.
- Plan del proyecto, mostrando fases e iteraciones.
- Modelo de negocio, si es necesario
- Prototipos exploratorios para probar conceptos o la arquitectura candidata.

Al terminar la fase de inicio se deben comprobar los criterios de evaluación para continuar:

- Todos los interesados en el proyecto coinciden en la definición del ámbito del sistema y las estimaciones de agenda.
- Entendimiento de los requisitos, como evidencia de la fidelidad de los Casos de Uso principales.
- Las estimaciones de tiempo, coste y riesgo son creíbles.
- Comprensión total de cualquier prototipo de la arquitectura desarrollado.
- Los gastos hasta el momento se asemejan a los planeados.

6.2 Etapa 2. Fase de elaboración del sistema.

El propósito de esta fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los mayores riesgos. En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio. También debe demostrarse que se han evitado los riesgos más graves. Posteriormente crear la base de datos que trabajará en conjunto con el sistema.

Los objetivos de esta fase son:

- Definir, validar y cimentar la arquitectura.
- Completar la visión.
- Crear un plan fiable para la fase de construcción. Este plan puede evolucionar en sucesivas iteraciones. Debe incluir los costes si procede.
- Demostrar que la arquitectura propuesta soportará la visión con un coste razonable y en un tiempo razonable

Al terminar deben obtenerse los siguientes resultados:

- Un modelo de Casos de Uso completa al menos hasta el 80%: todos los casos y actores identificados, la mayoría de los casos desarrollados.

- Requisitos adicionales que capturan los requisitos no funcionales y cualquier requisito no asociado con un Caso de Uso específico.
- Descripción de la arquitectura software.
- Un prototipo ejecutable de la arquitectura.
- Lista de riesgos y caso de negocio revisados.
- Plan de desarrollo para el proyecto.
- Un caso de desarrollo actualizado que especifica el proceso a seguir.
- Un manual de usuario preliminar (opcional).

Los criterios de evaluación de esta fase son los siguientes:

- La visión del Reservación es estable.
- La arquitectura es estable.
- Se ha demostrado mediante la ejecución del prototipo que los principales elementos de riesgo han sido abordados y resueltos.
- El plan para la fase de construcción es detallado y preciso. Las estimaciones son creíbles.
- Todos los interesados coinciden en que la visión actual será alcanzada si se siguen los planes actuales en el contexto de la arquitectura actual.
- Los gastos hasta ahora son aceptables, comparados con los previstos.

6.3 Etapa 3. Fase de construcción del sistema.

La finalidad principal de esta fase es alcanzar la capacidad operacional del Reservación de forma incremental a través de las sucesivas iteraciones. Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del Reservación.

Los objetivos concretos incluyen:

- Minimizar los costes de desarrollo mediante la optimización de recursos y evitando el tener que rehacer un trabajo o incluso desecharlo.
- Conseguir una calidad adecuada tan rápido como sea práctico.
- Conseguir versiones funcionales (alfa, beta, y otras versiones de prueba) tan rápido como sea práctico.

Los resultados de la fase de construcción deben ser:

Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación)

- Arquitectura íntegra (mantenida y mínimamente actualizada)
- Riesgos Presentados Mitigados
- Plan del Proyecto para la fase de Transición.
- Manual Inicial de Usuario (con suficiente detalle)
- Prototipo Operacional – beta
- Caso del Negocio Actualizado

Los criterios de evaluación de esta fase son los siguientes:

- El Reservación es estable y maduro como para ser entregado a la comunidad de usuario para ser probado.
- Todos los usuarios expertos están listos para la transición en la comunidad de usuarios.
- Son aceptables los gastos actuales versus los gastos planeados

6.4 Etapa 4. Fase de transición del sistema.

La finalidad de la fase de transición es poner el Reservación en manos de los usuarios finales, para lo que se requiere desarrollar nuevas versiones actualizadas del Reservación, completar la documentación, entrenar al usuario en el manejo del Reservación, y en general tareas relacionadas con el ajuste, configuración, instalación y facilidad de uso del Reservación.

Algunas de las cosas que puede incluir esta fase:

- Prueba de la versión Beta para validar el nuevo sistema frente a las expectativas de los usuarios
- Funcionamiento paralelo con los sistemas legados que están siendo sustituidos por nuestro proyecto.
- Conversión de las bases de datos operacionales.
- Entrenamiento de los usuarios y técnicos de mantenimiento.
- Traspaso del Reservacion a los equipos de marketing, distribución y venta.

Los principales objetivos de esta fase son:

- Conseguir que el usuario se valga por sí mismo.
- Un Reservación final que cumpla los requisitos esperados, que funcione y satisfaga suficientemente al usuario.

Los resultados de la fase de transición son:

- Prototipo Operacional
- Documentos Legales
- Caso del Negocio Completo
- Línea de Base del Reservación completa y corregida que incluye todos los modelos del sistema
- Descripción de la Arquitectura completa y corregida
- Las iteraciones de esta fase irán dirigidas normalmente a conseguir una nueva versión.

Los criterios de evaluación de esta fase son los siguientes:

- El usuario se encuentra satisfecho.
- Son aceptables los gastos actuales versus los gastos planificados.

VII. ESTUDIO DE FACTIBILIDAD

En el estudio de la Factibilidad Económica, determinamos el presupuesto de costos de los recursos técnicos, humanos y materiales tanto para el desarrollo como para la implantación del Sistema.

Además, nos ayudará a realizar el análisis costo-beneficio de nuestro sistema, el mismo que nos permitirá determinar si es factible a desarrollar económicamente el proyecto.

La factibilidad se apoya en 3 aspectos básicos, y estos son:

- Operativo.
- Técnico.
- Económico.

A continuación, se desarrollará el análisis y estudio de factibilidad correspondiente a este trabajo de investigación.

7.1 Factibilidad Operativa

La Factibilidad Operativa se refiere a todos aquellos recursos donde interviene algún tipo de actividad (Procesos), depende de los recursos humanos que participen durante la operación del proyecto. Durante esta etapa se identifican todas aquellas actividades que son necesarias para lograr el objetivo, se evalúa y determina todo lo necesario para llevarla a cabo.

El proyecto que se propone es asequible para ser respaldado por los beneficios que se obtendrían como acción de la implementación de este. No obstante, podría generar inquietud dentro de los usuarios que lo utilizaran por ser algo nuevo para ellos, por esto mismo es importante la realización de la capacitación de los usuarios que serían concurrentes en la manipulación con el sistema y acaparar todas o una mayoría de las sugerencias que estos puedan hacer con referente al mismo.

Sobre el personal que usara el sistema, se considerará que tengan los siguientes atributos:

- Alto conocimiento en los procesos de reservación, manejo de precios y administración del hotel paraíso condominiuns.
- Conocimiento básico en el manejo de una computadora.

Para el personal que dará mantenimiento del sistema se desea los siguientes atributos:

- Manejo del lenguaje de Programación Python utilizando el patrón de arquitectura Model – Template –View (MTV).
- Manejo del Framework para Python Django en su versión 3.1.3
- Manejo de Bootstrap 4 para la maquetación de la interfaz de usuario.
- Manejo de HTML 5, CSS y JS.
- Manejo del gestor de base de datos MySQL.
- Capacidad de análisis y comprensión de problemas.

7.2 Factibilidad Técnica

El desarrollo se trató de un sistema de administración, tomando en cuenta el alcance del propio negocio, sus costos son bajos y la necesidad de equipo potente es innecesaria, en el hotel paraíso condominiuns se cuenta con estos equipos, por lo que no habría necesidad de adquirirlos se debe de concluir indicando si es factible o no desarrollar técnicamente el proyecto.

7.2.1 Requerimientos de software

Recursos	Características
Análisis del sistema	Visual Paradigm, Bizagui
Desarrollo	<ul style="list-style-type: none">• Django Framework versión 3.1.3• Mysql Server
Diseño y Lenguajes de Programación	Frontend <ul style="list-style-type: none">• Bootstrap 4• HTML 5• CSS• JavaScript• Python
Base de Datos	Mysql
Documentación del Proyecto	Microsoft Office 2019 (Word, Excel, Power Point)
Sistema Operativo	Windows 10
Otros	GitHub (control de versiones)

Tabla 1. Requerimientos Generales de Software.

7.2.2 Requerimientos de Hardware

Requerimientos recomendados para los usuarios.

Recursos	Características
Procesador	Intel Core i3 3410 a 2.60 GHz o superior
Memoria RAM	4 GB o superior
Monitor	21 '' o superior con una resolución 1366 x 720 (para poder trabajar cómodamente con las tablas de información)

Tabla 2. Requerimientos Generales de software

Requerimientos del Servidor donde estará alojado el sistema

Recursos	Características
Procesador	Procesador a 1.4 Ghz o superior
Memoria RAM	2 GB o superior
Versión de PHP	El servidor donde estará alojado el sistema debe soportar el lenguaje de programación PHP en su versión 7 o superior.
Ancho de Banda	2 mb/s Para una buena fluidez al momento de cargar y enviar la información

Tabla 3. Requerimientos Generales de Hardware para el servidor

Recursos para los desarrolladores

Recursos	Características
Procesador	Intel Core i3 6400
Memoria RAM	4 GB a 2400 Mhz
Monitor	21 '' con una resolución 1680 x 1050

Tabla 4. Requerimientos de Hardware para el Desarrollo

7.3 Factibilidad Económica

Consiste en identificar y ordenar todos los ítems de inversiones, costos e ingresos que puedan deducirse de los estudios previos

- Costo en desarrollo

El costo en los pagos de desarrollo está basado en sondeos realizados en empresas de desarrollos de software en Nicaragua y promedios de salarios.

- Recursos Humanos

Recurso	Cant.	Función	Salario Mensual	Duración	Sub Total
<u>Analista/Programador</u>	1	<ul style="list-style-type: none"> • Análisis y diseño de la estructura interna del sistema. • Diseñar y codificar los distintos módulos requeridos para el sistema. • Implementar estándares de seguridad para el acceso y resguardo de la información del sistema. 	\$ 1500.00	6 meses	\$ 9000.00
<u>Encargado de Pruebas</u>	1	<ul style="list-style-type: none"> • Evaluación de los procesos de funcionamiento del sistema 	\$300.00	6 meses	\$ 1,800.00

Tabla 5. Costo en Desarrolladores

- Costo de Hardware utilizado en el desarrollo

Recurso	Cantidad	Característica	Coste
PC	1	<ul style="list-style-type: none"> • Procesador Intel core i3 8400 • Disco duro SSD 480 GB • Disco duro SATA 500 GB • Memoria RAM 8 GB 1600 Mhz • Tarjeta Gráfica RTX NVIDIA 1050 • Windows 10 Pro • Monitor DELL 21" 	U\$ 850.00
Periféricos	2	Mouse y Teclado Logitech	U\$ 65.00
DD Externo	1	Disco Duro externo 1 TB	U\$ 80.00
TOTAL:			U\$ 995.00

Tabla 6. Costo en Hardware de Desarrollo

El desarrollador cuenta con recursos tecnológicos para el desarrollo del sistema por lo que no se requerirá de una inversión inicial en la adquisición de este recurso.

- Costo de Hardware en Servidor de Aplicaciones

Actualmente el hotel paraíso condominiuns no cuenta con servidores donde pueda ser alojado dicho sistema, debido al alto costo que conlleva adquirir y realizar mantenimiento a este equipo, la empresa optará por la compra/suscripción de un Hosting el cual servirá para alojar el sistema.

Recurso	Cantidad	Característica	Costo
Hosting	1	Espacio que tendrá como propósito alojar el sistema web.	\$8.00
SSL	1	Certificado de seguridad para el dominio donde se podrá acceder al sistema en línea	\$12.00

Dominio	1	Dirección URL por el cual se podrá acceder al sistema a través de un navegador de internet	\$ 30.00
TOTAL:			\$50.00

Tabla 7. Costo en Hardware para servidores

Nota: los precios reflejados anteriormente son de carácter anual, ya que se realiza una suscripción para poder alojar el sistema a un servidor, y dicha suscripción se deberá renovar de manera anual, mientras que el pago del SSL es único.

- Costo de Software para desarrollo e implementación.

Para el desarrollo del sistema se realizará adquiriendo licencias de distintos softwares para su debido funcionamiento.

Recurso	Cantidad	Característica	Costo
Sistema Operativo	1	Windows 10 Pro (Desarrollo)	U\$ 286.00
Ofimática	1	Microsoft Office 2019	U\$ 164.00
Herramientas de Desarrollo	-	<ul style="list-style-type: none"> • Django 3.1.3 • Mysql Server versión 5.7. 24 • Framework Bootstrap 4 	Gratis
Otras Herramientas	-	<ul style="list-style-type: none"> • GitHub • 	Gratis
TOTAL:			U\$ 450.00

Tabla 8. Costo de software para Desarrollo e Implementación.

Nota: se utiliza Windows como sistema operativo para el desarrollo e implementación del proyecto debido a que dicho sistema operativo es el ambiente de ejecución que posee la empresa en sus equipos actualmente.

Costo Total.

A continuación, se realizará un resumen de los costos para el desarrollo e implementación del sistema, dando como resultado el costo estimado para el proyecto.

Recurso	Costo
Costo de Desarrollo	U\$ 10,800
Costo de Software	U\$ 450.00
Costo de Hardware	U\$ 995.00
Costo de Servidor (primer año)	U\$ 50.00
Total, estimado	U\$ 12,295.00

Tabla 9. Costo Total del proyecto.

7.4 Aspectos Legales

Para el desarrollo del sistema se requiere de algunos licenciamientos explicados en la sección de Factibilidad Económica, todas las licencias necesarias serán adquiridas por el hotel paraíso condominiuns para su debida implementación.

Por lo tanto, no se quebrantarán leyes que haga que el sistema sea sensible a fraudes o inconvenientes legales, por ello el desarrollo del sistema se considera viable en materia legal.

Capítulo IV: Análisis del Sistema.

VIII. ANÁLISIS DEL SISTEMA

8.1. Requerimientos del sistema.

Desarrollo del Sistema

El proceso para el desarrollo del Sistema del sistema administrativo para el hotel paraíso Condominiuns, se implementó la metodología de desarrollo de ciclo de vida del modelo RUP. Este modelo de ciclo de vida se basó en la filosofía de construir con la finalidad de asegurar la obtención de un software de alta calidad que satisfagan la necesidad de los usuarios finales dentro de un calendario y tiempo predecible, las funcionalidades del programa a medida que interactúa el programador y usuario del sistema.

Dicho método consta de cuatro etapas esenciales, con las cuales se llevó a cabo el desarrollo del software. Las fases que se ejecutaron a lo largo del proceso detallan la manera como se dieron cumplimiento a los objetivos específicos hasta alcanzar el objetivo general de la investigación con respeto al cronograma de desarrollo de la tesis.

Análisis

En esta fase los autores recopilaron la información relevante que fue utilizada para el desarrollo del sistema.

En esta fase se logró determinar los siguientes factores:

- La Unidad administrativa no tiene a su disposición un sistema flexible que pueda acoplarse a las nuevas necesidades que se requieran del lado del usuario.
- La información de las actividades y procesos realizados en el hotel paraíso no se encuentra organizada y clasificada adecuadamente.
- Algunas actividades del hotel paraíso como reportes de reservación lo realizan de forma manual
- EL proceso de reservación está muy obsoleto debido a la versión de escritorio con la que cuentan no ha podido ser actualizada hasta el día de hoy, de acuerdo a las nuevas necesidades de los administradores lo realizan manualmente en fichas estudiantiles y archivadas en folder.

Requisitos funcionales y no funcionales del software

Requerimientos funcionales
<ul style="list-style-type: none">• Registrar la información del tipo de reservación.• Registrar la información de los clientes• Efectuar cambios de tipos de reservaciones, permitiendo que el usuario pueda realizarlas según su comodidad.• Ingresar y almacenar información de los clientes• Realizar consultas• Generación de los reportes correspondientes• Permitir envíos al correo electrónico de los clientes de comprobante de pago, permitiendo así poder visualizar si el cliente lo desea, su estado de cuenta.• Gestión de pago para el área administrativa• Permitir la visualización de un calendario de eventos, para poder llevar la administración de la disponibilidad de eventos del hotel
Requerimientos no funcionales
<ul style="list-style-type: none">• Tiempo de respuestas en pocos segundos, al momento de realizar una operación.• La interfaz que se implementará será web.• El motor de base de base de datos que se utilizará será MySQL• El sistema será multiusuario.• Se utilizará Python en la creación del programa informático.• Almacenamientos y respaldos periódicos de la información de forma que pueda ésta pueda ser recuperada en caso de fallos del sistema o pérdidas de información por accidentes no previstos.

Diseño del Sistema

Con la información obtenida en la primera fase y luego de haber analizado detalladamente todos los procesos existentes y aquellos que se debería incorporar en el sistema, los autores precedieron a la elaboración de los diagramas de casos

Diseño de Base de Datos

El gestor de base de datos a utilizarse es MySql ya que es uno de los gestores de base de datos que se adapta más al funcionamiento, adaptabilidad del sistema administrativo, además que unos de los objetivos del proyecto es poder almacenar toda la información ingresada en el sistema.

Para el desarrollo del sistema se crearon un total de 10 tablas. Cada tabla tiene su propia clave principal que es el campo o conjunto de campos que identifica de manera única cada registro, siendo en la mayor parte de las tablas las letras Id_ nombre de la tabla, por ejemplo: Id reserva en la tabla.

Entre las tablas se encuentran:

- cliente
- empleado
- reservación
- tours
- habitación
- pago
- login
- pedidos
- Reservations
- salón de eventos

Interfaces

El sistema en general, se manejó con interfaces que cumplen con uno de los objetivos primordiales, el cual es facilitar al usuario la interacción y fácil uso del mismo.

La interfaz en forma general es de sentido amigable para la persona encargada de administrar el sistema, en la parte lateral derecha de la pantalla principal se ubicó un menú mediante que funciona de enlace para acceder a los otros formularios. Cada formulario tenía asociados controles básicos como groupbox, buttons, label, textbox, listbox, combobox, datagridview, Menústrip, Tab control, entre otros que sirvieron de complemento para manipular y visualizar con éxito la información.

Programación del Sistema

El lenguaje de programación utilizado para desarrollar el sistema fue Python ya que es un lenguaje orientado al desarrollo de aplicaciones web con excelente integración de base de Datos MySql.

En esta etapa en donde se desarrolló toda la lógica del programa, se procedió a introducir el código fuente, se elaboró la estructura de datos y objetos encargados para manipulación de los datos existente, así como el procesamiento de la información ingresada, se realizó de tal forma que existiera consistencia en los mismos, permitiendo que los datos que se ingresen, así como lo que se extraen de la base de datos sean consistentes y precisos.

8.2.1 Casos de Uso del sistema

Un caso de uso es una descripción de los procesos de un sistema desde el punto de vista del usuario.

- **Módulo de Administración.**

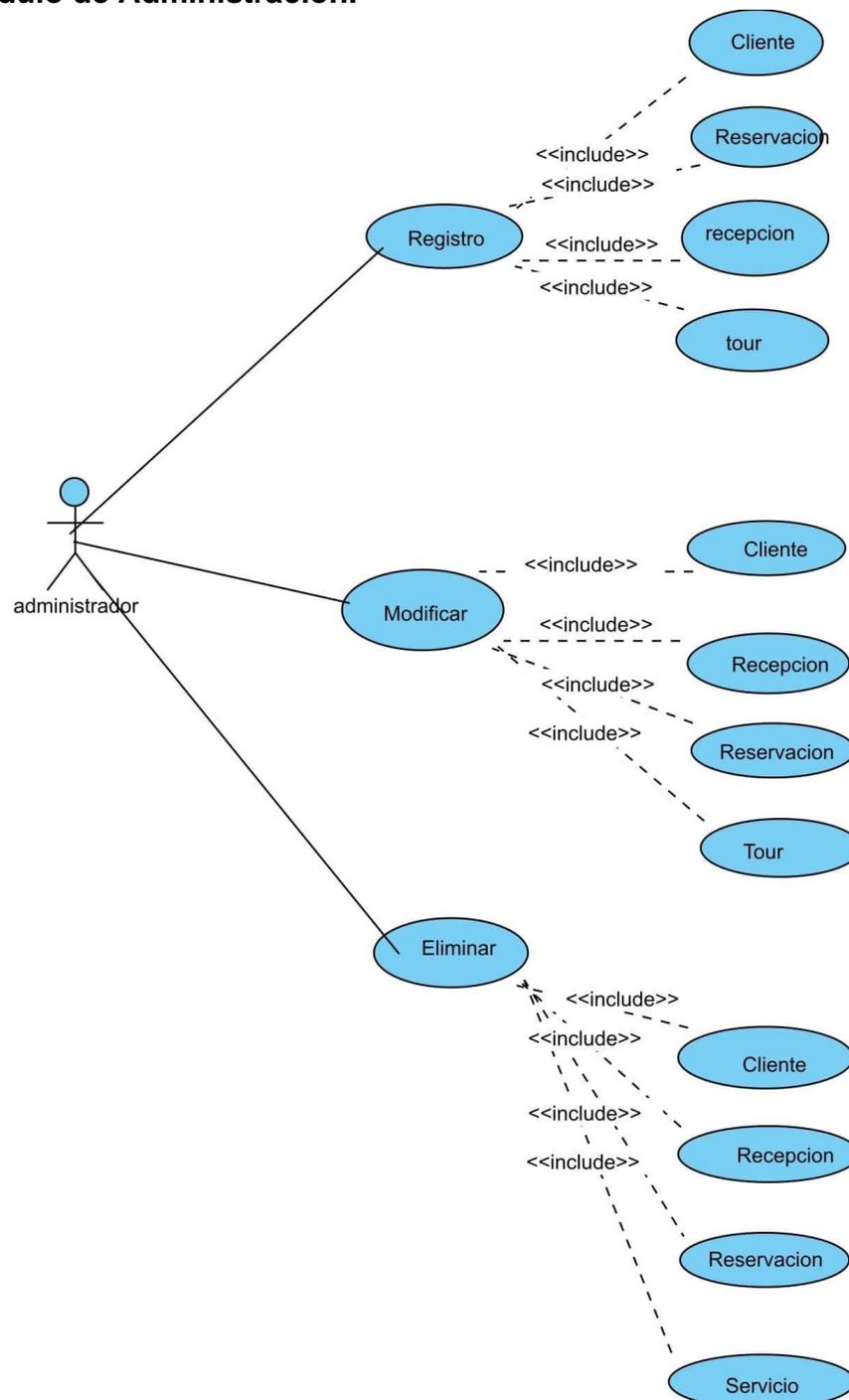


Ilustración 4. Caso de Uso Módulo de Administración 01.

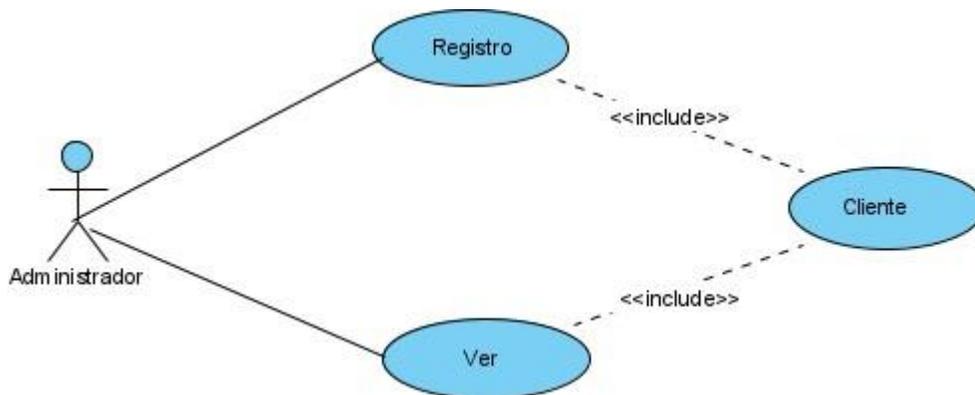
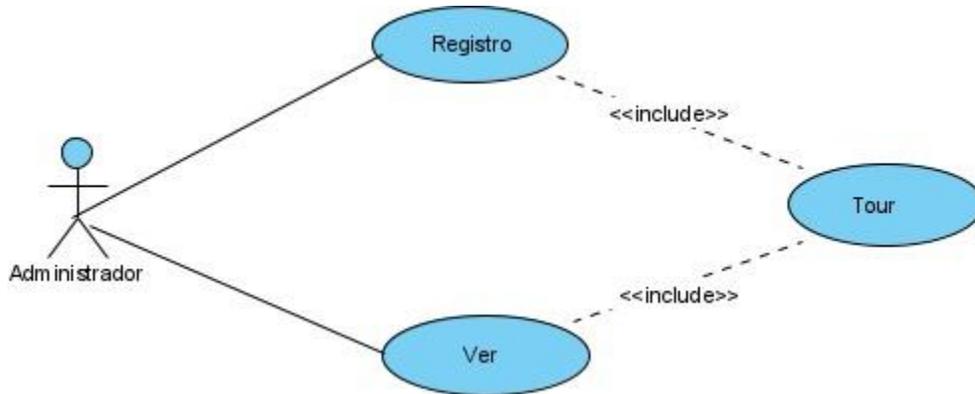
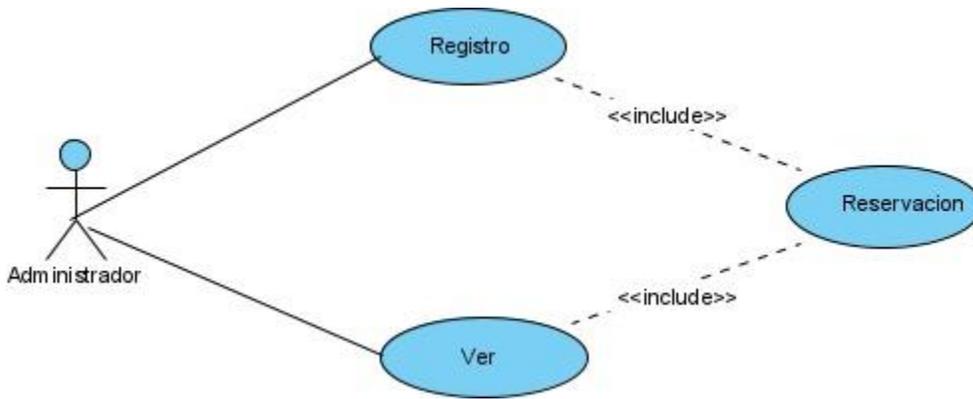


Ilustración 5. Caso de Uso Módulo de Administración 02.

- **Módulo de Reservacion**

Sección de Reservacion.

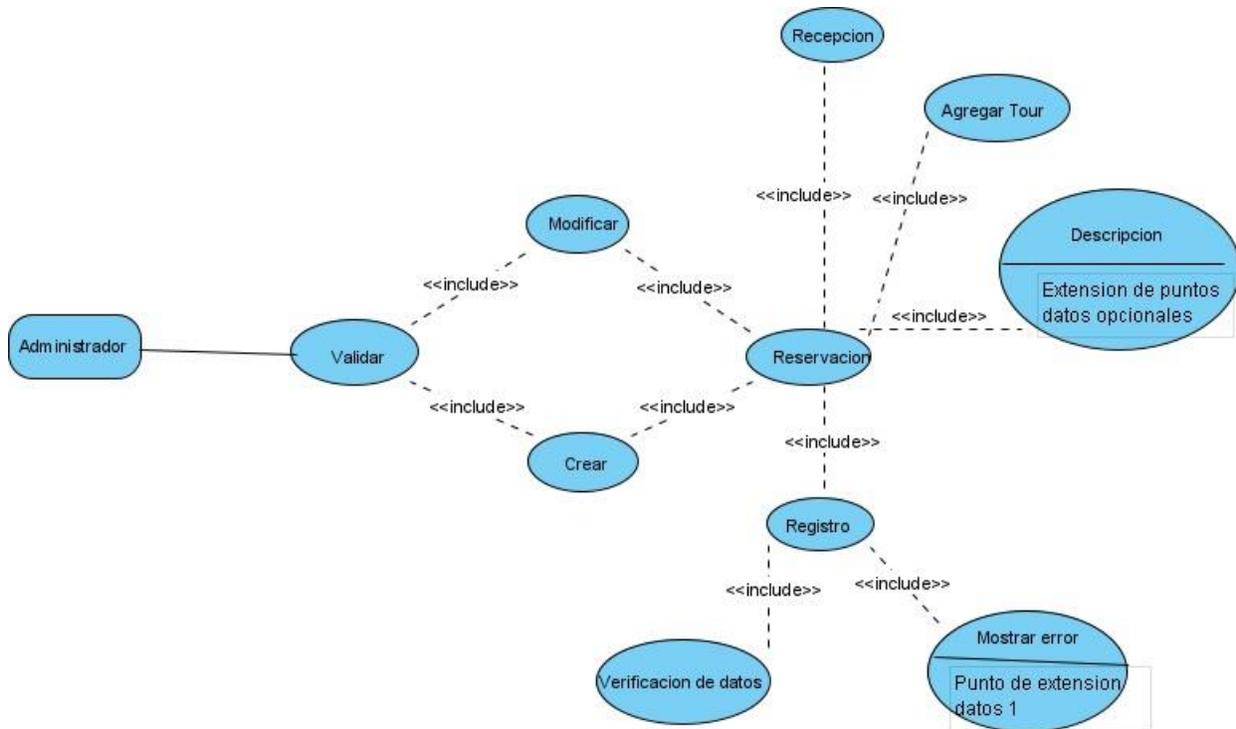


Ilustración 6.Caso de Uso Módulo de Compras 01.

Sección de Voucher

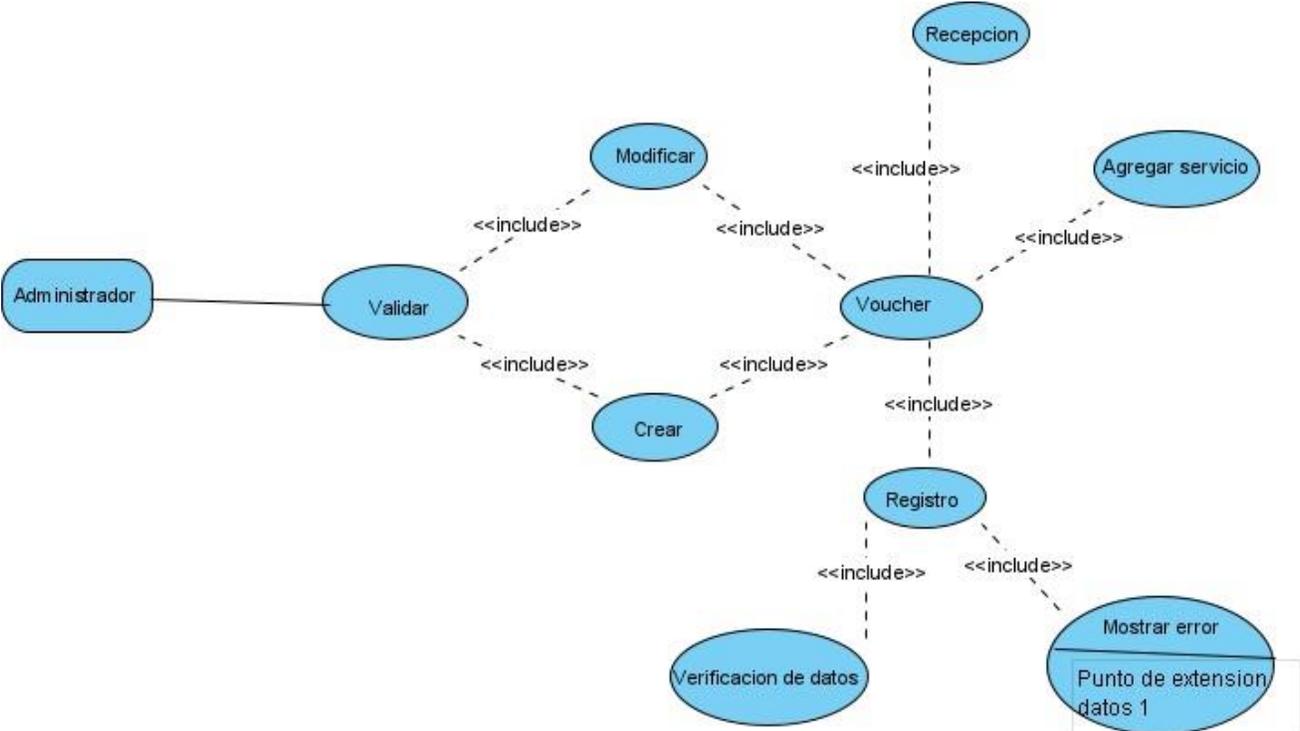


Ilustración 7.Caso de Uso Módulo de Compras 02.

Sección de Tour.

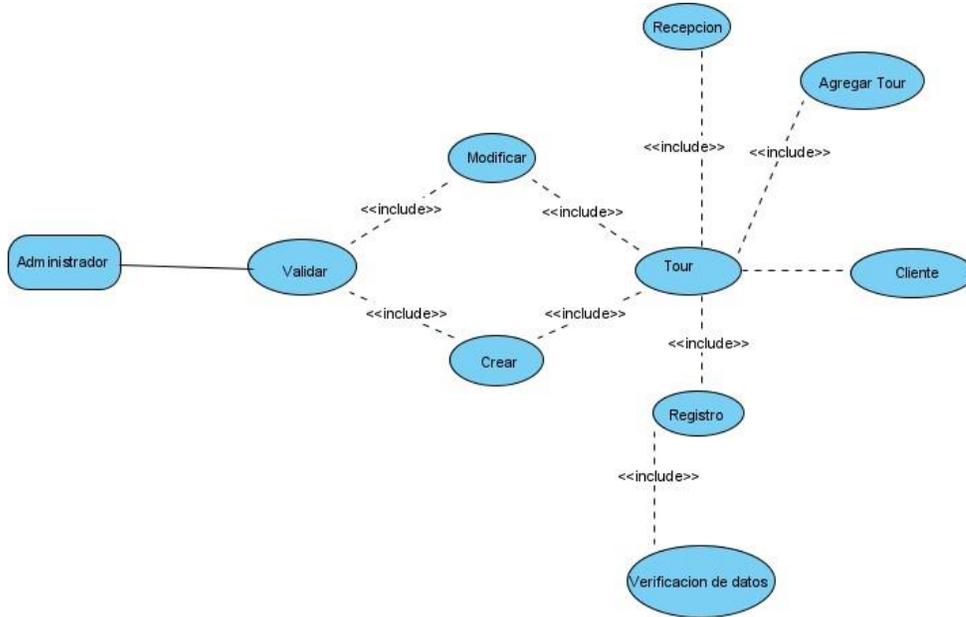


Ilustración 8. Caso de Uso Módulo de Compras 03.

• Módulo de Cliente

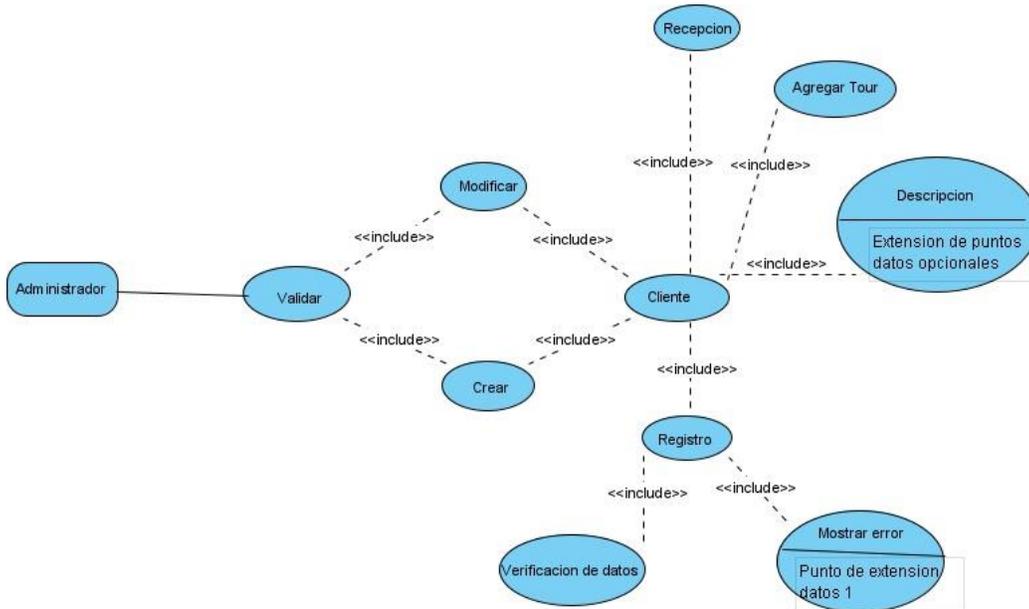


Ilustración 9. Caso de Uso Módulo de Ventas 01

Sección Venta

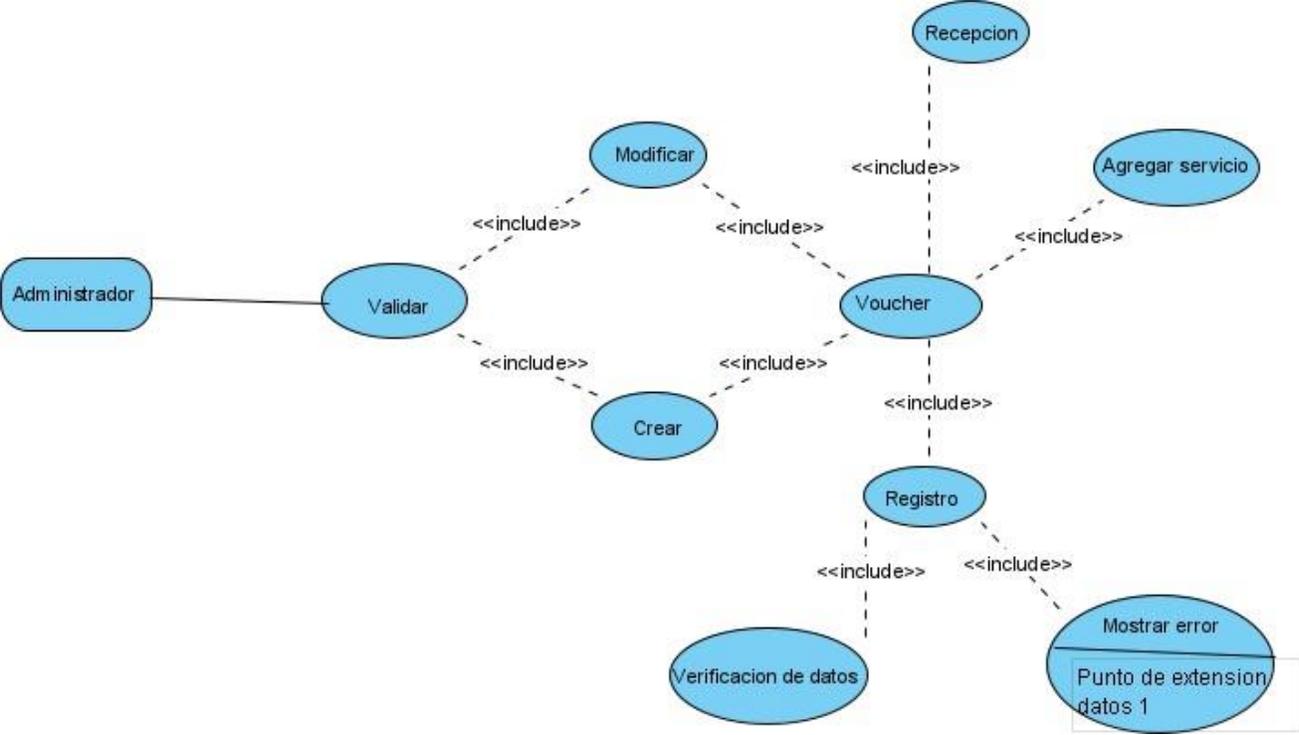


Ilustración 10.Caso de Uso Módulo de Ventas 02.

- **Módulo de Habitación**

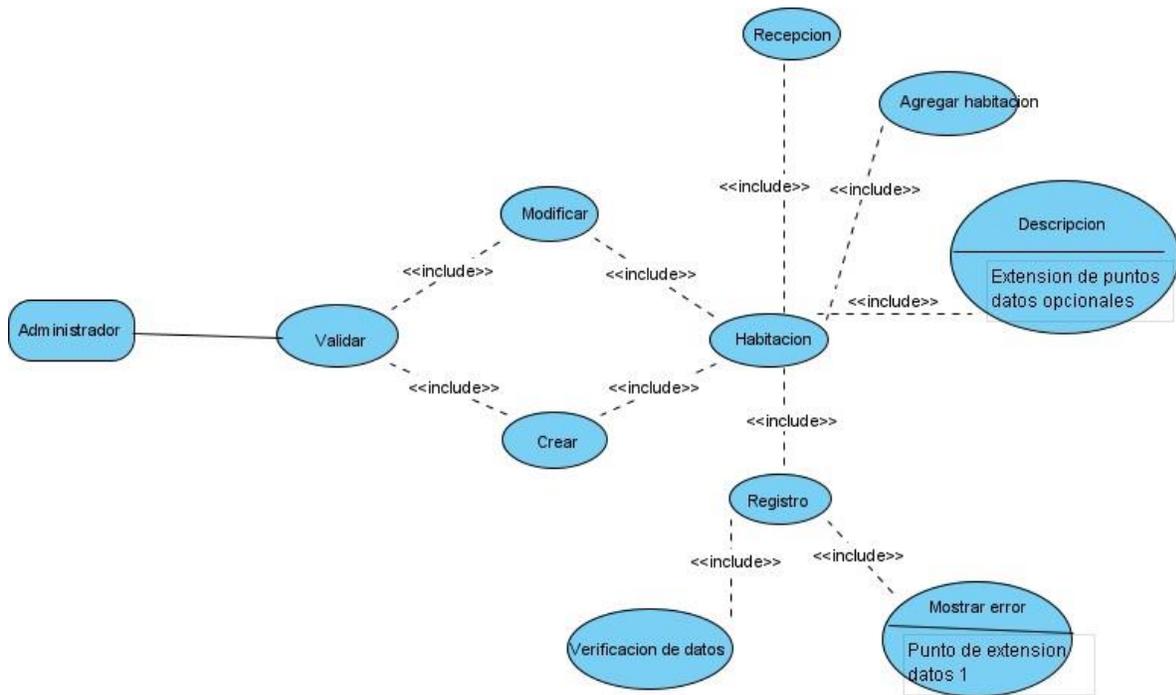


Ilustración 11.Caso de Uso Módulo de Seguridad 01.

8.2.2 Plantilla de Casos de Uso del sistema

A continuación, se desarrollarán de manera detallada los procesos más importantes del sistema haciendo uso de una plantilla para los casos de uso.

Caso de Uso 001 – Registrar Reservación	
Descripción:	Registrar Reservación
Autor:	Rommel Humberto Pérez Gómez
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Registrar la información necesaria para añadir una nueva reservación a la base de datos.
Pre – Condición:	Los usuarios deberán estar Autenticados en el sistema.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Reservación, luego en el sub menú seleccionar Reservar. • El sistema cargara una tabla con el listado de las reservaciones que hayan sido registrados previamente. • Dar Click en el botón agregar reservación. • El sistema mostrara un formulario de registro con la información requerida para registrar una nueva Reservación. • El sistema validará la información y si todo está correcto procederá a registrar la nueva Reservación a la base de datos.
Post – Condición:	Se registro una nueva Reservación a la base de datos de manera correcta.

Tabla 20.Caso de uso 004.

Caso de Uso 002 – Editar Reservación	
Descripción:	Editar Reservación
Autor:	Rommel Humberto Pérez Gómez
Fecha:	15/09/2020
Autores:	Administrador.

Propósito:	Actualizar la información de una Reservación que haya sido previamente registrado en la base de datos.
Pre – Condición:	Los usuarios deberán estar Autenticados en el sistema.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Reservación, luego en el sub menú seleccionar Reservaciones. • El sistema cargara una tabla con el listado de las Reservaciones que hayan sido registrados previamente. • Dar Click en el botón editar ubicado en la parte derecha de Reservación. • El sistema mostrara un formulario de edición con la información previamente registrada de la Reservación seleccionado. • El sistema validará la información y si todo está correcto procederá a actualizar la Reservación en la base de datos.
Post – Condición:	Se actualizo la Reservación en la base de datos de manera correcta.

Tabla 21.Caso de uso 005.

Caso de Uso 003 – Eliminar Reservación	
Descripción:	Eliminar Reservación
Autor:	Rommel Humberto Pérez Gómez
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Dar de baja o Eliminar lógicamente los registros de una Reservación.
Pre – Condición:	Los usuarios deberán estar autenticados en el sistema, la reservación a eliminar no debe estar vinculado a ningún otro registro.

Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción reservación, luego en el sub menú seleccionar reservación. • El sistema cargara una tabla con el listado de las Reservaciones que hayan sido registrados previamente. • Dar Click en el botón eliminar ubicado en la parte derecha de reservación. • El sistema mostrara una alerta al usuario preguntando si de verdad desea eliminar la reservación. • El sistema validará la información y si todo está correcto procederá a eliminar la reservación en la base de datos.
Post – Condición:	Se eliminó la reservación en la base de datos de manera correcta.

Tabla 22.Caso de uso 006.

Caso de Uso 004 – Registrar Cliente	
Descripción:	Registrar Cliente
Autor:	Luis Carlos Flores Baltodano
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Registrar la información necesaria para un nuevo Cliente en la base de datos.
Pre – Condición:	Los usuarios deberán estar Autenticados en el sistema.

Flujo Normal:

- Seleccionar en el menú la opción Cliente.
- El sistema cargara una tabla con el listado de los Clientes que hayan sido registrados previamente.
- Dar Click en el botón agregar Cliente.
- El sistema mostrara un formulario de registro con la información requerida.
- El sistema validará la información y si todo está correcto procederá a registrar el nuevo Cliente en la base de datos.

Post – Condición:

Se registro un nuevo Cliente en la base de datos de manera correcta.

Tabla 23.Caso de uso 007.

Caso de Uso 005 – Editar Cliente	
Descripción:	Editar Cliente
Autor:	Luis Carlos Flores Baltodano
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Actualizar la información de un Cliente en la base de datos.
Pre – Condición:	Los usuarios deberán estar Autenticados en el sistema.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Cliente. • El sistema cargara una tabla con el listado de los Clientes que hayan sido registrados previamente. • Dar Click en el botón editar ubicado en la parte derecha de Cliente. • El sistema mostrara un formulario de edición con la información previamente registrada del Cliente seleccionado. • El sistema validará la información y si todo está correcto procederá a actualizar la información del Cliente en la base de datos.
Post – Condición:	Se actualizo la Cliente en la base de datos de manera correcta.

Tabla 24.Caso de uso 008.

Caso de Uso 006 – Eliminar Cliente	
Descripción:	Eliminar Cliente
Autor:	Luis Carlos Flores Baltodano
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Dar de baja o Eliminar lógicamente los registros de un Cliente.

Pre – Condición:	Los usuarios deberán estar autenticados en el sistema, el Cliente a eliminar no debe estar vinculado a ningún otro registro.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Cliente. • El sistema cargara una tabla con el listado de los Clientes que hayan sido registrados previamente. • Dar Click en el botón eliminar ubicado en la parte derecha del Cliente. • El sistema mostrara una alerta al usuario preguntando si de verdad desea eliminar el Cliente. • El sistema validará la información y si todo está correcto procederá a eliminar el Cliente en la base de datos.
Post – Condición:	Se elimino el Cliente en la base de datos de manera correcta.

Tabla 25.Caso de uso 009.

Caso de Uso 007 – Registrar Habitación	
Descripción:	Registrar Habitación.
Autor:	Luis Carlos Flores Baltodano
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Registrar la información necesaria para poder agregar una nueva Habitación a la base de datos.
Pre – Condición:	Los usuarios deberán estar Autenticados en el sistema.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Habitación • El sistema cargara una tabla con el listado de las Habitaciones que hayan sido registrados previamente. • Dar Click en el botón agregar Habitación.

	<ul style="list-style-type: none"> • El sistema mostrara un formulario con la información requerida para registrar una nueva Habitación. • El sistema validará la información y si todo está correcto procederá a registrar la nueva Habitación en la base de datos.
Post – Condición:	Se registro una nueva Habitación en la base de datos de manera correcta.

Tabla 26.Caso de uso 010.

Caso de Uso 008 – Editar Habitación	
Descripción:	Editar Habitación.
Autor:	Luis Carlos Flores Baltodano
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Actualizar la información de una Habitación registrada previamente en la base de datos.
Pre – Condición:	Los usuarios deberán estar Autenticados en el sistema.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Habitación • El sistema cargara una tabla con el listado de las Habitaciones que hayan sido registrados previamente. • Dar Click en el botón editar ubicado en la parte derecha de la Habitación. • El sistema mostrara un formulario de edición con la información registrada previamente de la Habitación seleccionada. • El sistema validará la información y si todo está correcto procederá a actualizar la Habitación en la base de datos.
Post – Condición:	Se actualizo la Habitación en la base de datos de manera correcta.

Tabla 27.Caso de uso 011.

Caso de Uso 009 – Eliminar Habitación	
Descripción:	Eliminar Habitación
Autor:	Luis Carlos Flores Baltodano
Fecha:	15/09/2020
Autores:	Administrador.
Propósito:	Dar de baja o Eliminar lógicamente los registros de una Habitación.
Pre – Condición:	Los usuarios deberán estar autenticados en el sistema, la Habitación a eliminar no debe estar vinculado a ningún otro registro.
Flujo Normal:	<ul style="list-style-type: none"> • Seleccionar en el menú la opción Habitaciones. • El sistema cargara una tabla con el listado de las Habitaciones que hayan sido registrados previamente. • Dar Click en el botón eliminar ubicado en la parte derecha de la Habitación. • El sistema mostrara una alerta al usuario preguntando si de verdad desea eliminar la Habitación. • El sistema validará la información y si todo está correcto procederá a eliminar la Habitación en la base de datos.
Post – Condición:	Se elimino la Habitación en la base de datos de manera correcta.

Tabla 28.Caso de uso 012.

Capítulo V: Diseño y Modelado del Sistema.

IX. DISEÑO DEL SISTEMA

El objetivo de esta sección es explicar de manera clara como estará definida la estructura interna del sistema y del entorno tecnológico que le dará soporte, con esta información se generan todas las especificaciones de construcción o codificación del sistema.

9.1. Diagrama Entidad Relación

Un diagrama o modelo Entidad-Relación es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información, así como sus interrelaciones y propiedades.

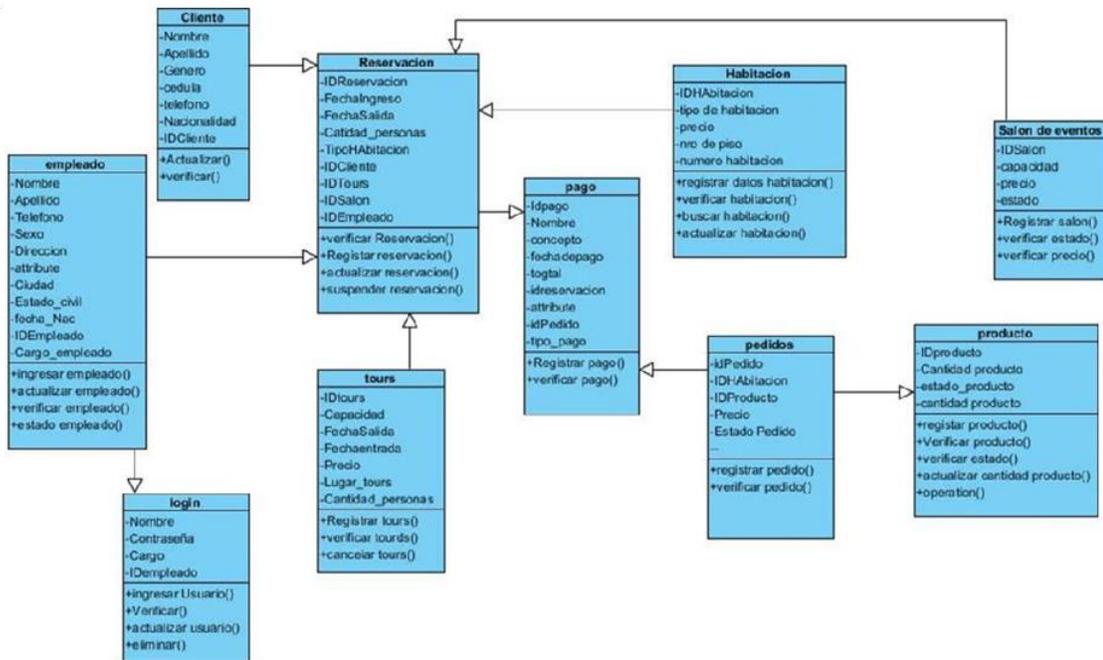


Ilustración 15. Diagrama de Entidad Relación.

9.2. Diagrama de Actividades

Son considerados diagramas de comportamiento del sistema porque describen lo que debe suceder en el sistema que se está modelando.

A continuación, se presentarán los diagramas de actividades de los módulos más importantes del sistema.

9.2.1 Ingreso al sistema

- **Actores:** Administrador, Facturador, Comprador y Encargado de Cliente.

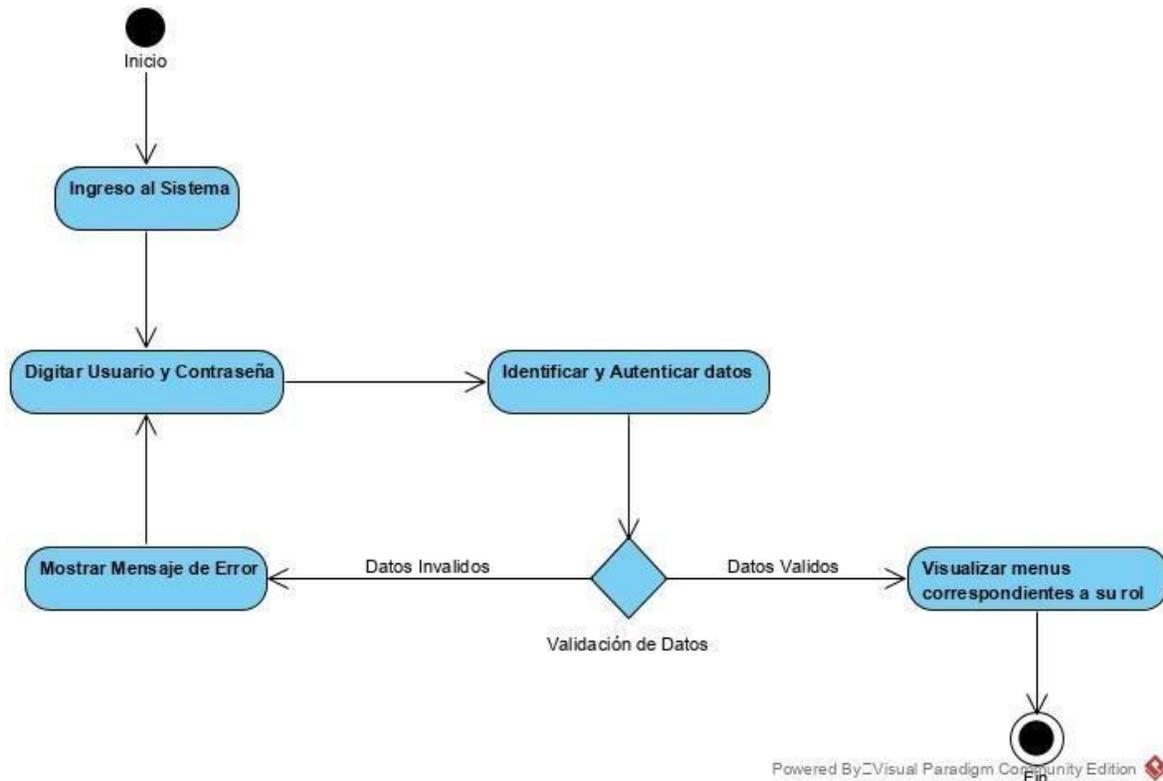


Ilustración 17. Diagrama de Actividades 01 – Ingreso al sistema.

9.2.2 Gestión de usuarios

- **Actores:** Administrador

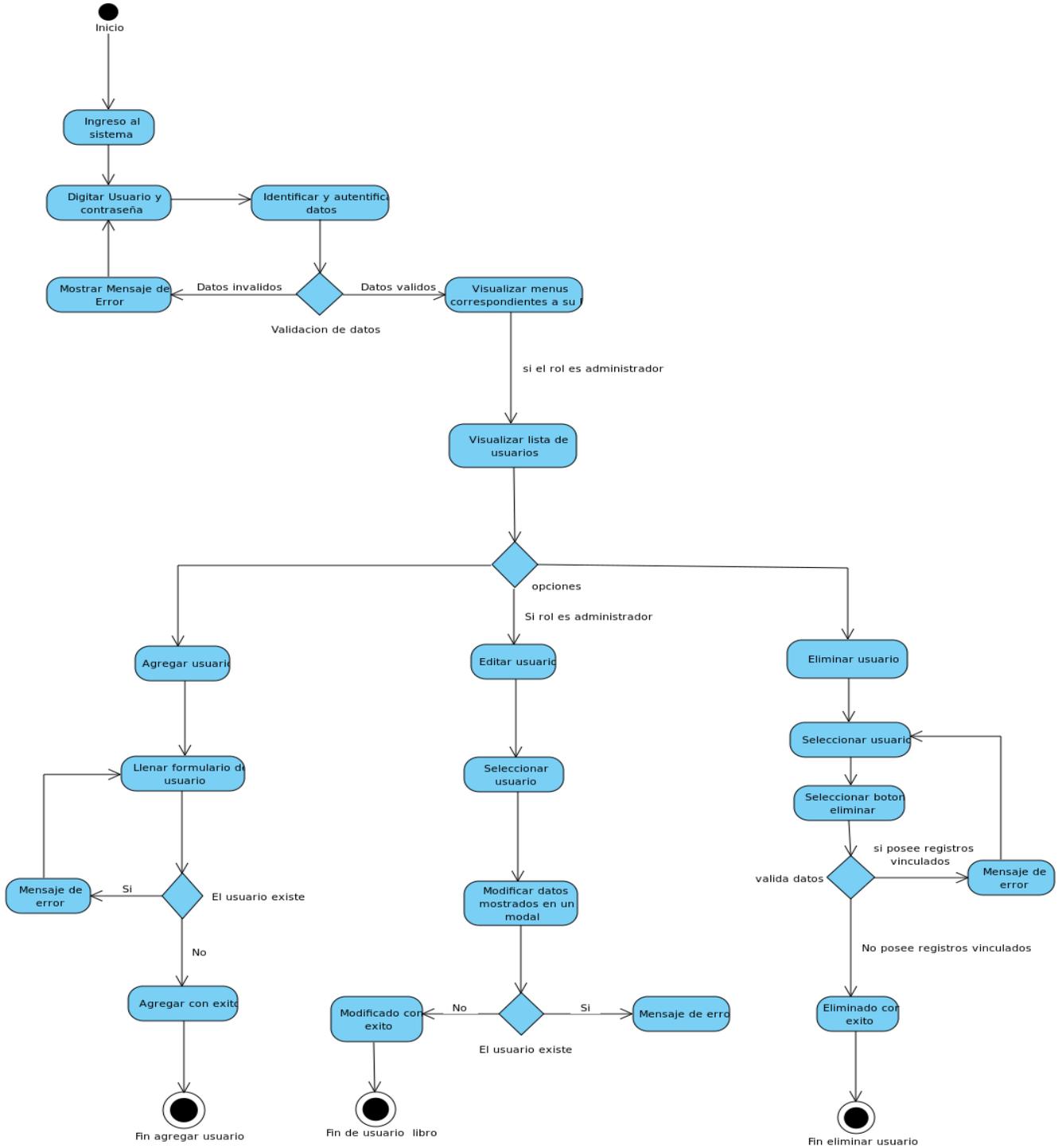


Ilustración 18. Diagrama de Actividades 02 – Gestión de usuarios

9.2.3 Gestión de reportes

- **Actores:** Administrador

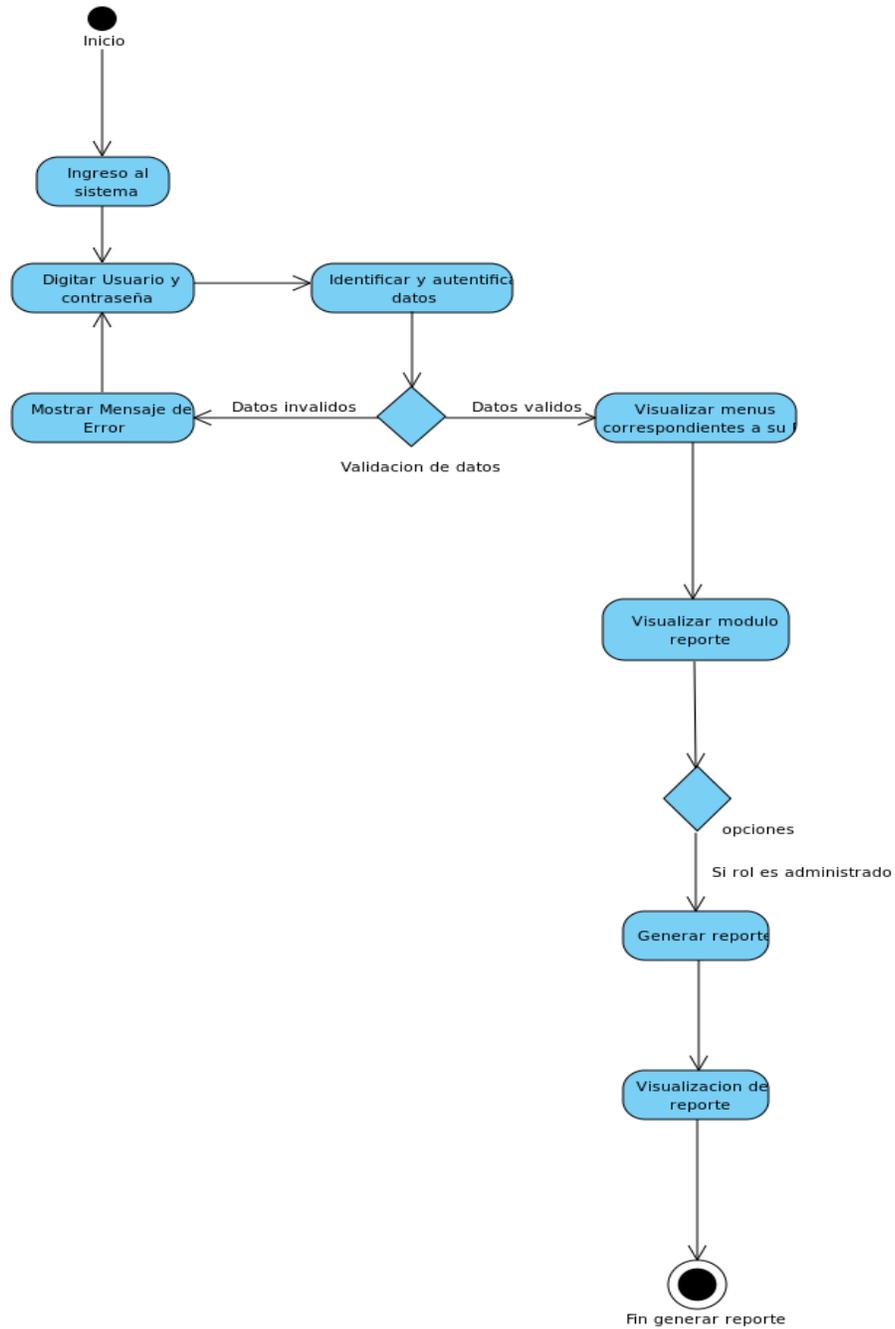


Ilustración 19. Diagrama de Actividades 03 – Gestión de reportes.

Capítulo VI: Codificación, Pruebas e Implementación del Sistema

X. CODIFICACIÓN DEL SISTEMA

Una vez que tenemos un modelado y diseño del sistema podemos comenzar la etapa de codificación, en esta se traducen dichos modelos a un lenguaje de programación en este caso Python con la ayuda del Framework Django en su versión 3.1.3

10.1 Arquitectura del Sistema

Para el desarrollo del sistema se utilizará la arquitectura Model – Template – View, esta nos permite separar los componentes de nuestra aplicación dependiendo de la responsabilidad que tienen, esto significa que cuando hacemos un cambio en alguna parte de nuestro código, esto no afecte otra parte del mismo.

En la actualidad es uno de los patrones de arquitectura de software más usados en el mundo tanto en pequeños como en grandes sistemas de información, en el mundo laboral es casi indispensable el llevarlo a la práctica.

A continuación, se mostrará el proceso completo desde que un usuario solicita acceso a una página, el funcionamiento del sistema es el siguiente:

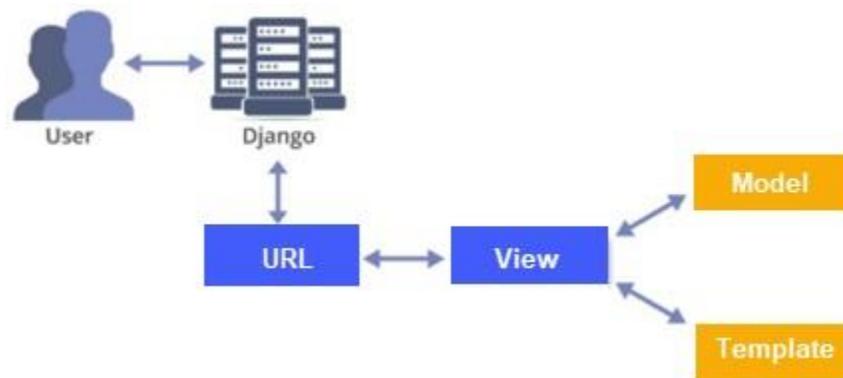


Ilustración 37. Proceso Modelo – Vista – Controlador (MVC).

- El usuario realiza una petición desde el navegador.
- La petición es recibida por el controlador, este controla la lógica de la aplicación.
- El controlador solicita al modelo la información solicitada por el usuario.
- El modelo consulta a la base de datos y retorna la información al controlador.
- El controlador recibe y procesa la información obtenida por el modelo y la envía a la vista.
- La vista presenta en pantalla la información solicitada por el usuario.

10.2 Estructura del Sistema

A continuación, se mostrará la estructura del proyecto, en donde se presentarán los principales elementos requeridos para el funcionamiento del sistema.

Nombre	Descripción
App	Dentro de esta encontramos los modelos y controladores de los respectivos módulos del sistema.
Public	Dentro de esta encontraremos: <ul style="list-style-type: none"> • Css • Js Que forman parte de plugin o controles que ayuden en la interfaz de usuario.
Resource	Se encuentran las vistas de cada uno de los módulos del proyecto, así como el Layout general.
Route	Dentro de este se encuentra la configuración de las distintas rutas que será utilizadas para navegar por cada una de las vistas del sistema.
.Env	Dentro de este se encuentra la configuración del proyecto, así como las credenciales necesarias para conectar el sistema con la base de datos, seleccionada en dicho archivo.

Tabla 39. Estructura del Proyecto.

XI. DESARROLLO DE PRUEBAS FUNCIONALES

Para la realización de las pruebas del sistema se tomó como base la propuesta de estrategias de software del libro ingeniería de software un enfoque práctico de Roger Pressman, el cual propone validar el programa a partir de requerimientos, consecuentemente, utilizando los casos de usos para crear escenarios de pruebas en los cuales se verifica la funcionalidad del software con respecto a la interacción con los usuarios mediante la entrada de datos y la obtención de resultados de manera controlada. Finalmente, todo el proceso anteriormente mencionado es documentado usando una plantilla de casos de prueba.

Caso de Prueba N° 1: Registrar reservación	
Módulo	reservación
Autor	Rommel Humberto Pérez Gómez
Fecha	30/10/2020
Acción	Registrar una nueva reservación en la base de datos.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan registrar nuevas reservaciones correctamente.
Descripción	Se realiza la prueba de registrar una reservación para detectar errores al momento de realizar el proceso.
Criterios de éxito	El sistema envía mensaje de éxito al guardar o mensaje de validación con los campos que no cumplan con las reglas

Criterios de Fallo	establecidas, la lista en pantalla se actualiza con la reservación que acabamos de registrar. El sistema no retorna mensaje de validación ni de guardado, tampoco actualiza la pantalla mostrando la nueva información digitada.
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al formulario de nueva reservación y completar dicho formulario. • Pulsar el botón Guardar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción reservación en el menú principal del sistema. • Pulsar el botón Nuevo reservación. • Completar el formulario con la información requerida. • Pulsar el botón Guardar.
Resultados Obtenidos	Se registro el nuevo departamento de manera correcta.

Tabla 43.Caso de Prueba 001.

Caso de Prueba N° 2: Editar reservación	
Módulo	reservación
Autor	Rommel Humberto Pérez Gómez
Fecha	30/10/2020
Acción	Actualizar la información de una reservación ya registrado en la base de datos.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan actualizar la información de una reservación correctamente.
Descripción	Se realiza la prueba de editar una reservación para detectar errores al momento de realizar el proceso.

Criterios de éxito	El sistema deberá mostrar la opción de modificar solamente a los usuarios con rol Administrador, al realizar la modificación deberá recibir un mensaje de guardado y actualización en pantalla de los datos de la reservación.
Criterios de Fallo	El sistema no retorna mensaje de validación ni de guardado, tampoco actualiza la pantalla mostrando la nueva información digitada.
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al formulario de editar reservación y realizar los cambios necesarios en dicho formulario. • Pulsar el botón Actualizar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción reservación en el menú principal del sistema. • Pulsar el botón Editar al lado de la reservación que desea editar. • Actualizar la información del formulario con la información necesaria. • Pulsar el botón Actualizar.
Resultados Obtenidos	Se actualizo la reservación de manera correcta.

Tabla 44.Caso de Prueba 005.

Caso de Prueba N° 3: Eliminar reservación	
Módulo	reservación
Autor	Rommel Humberto Pérez Gómez
Fecha	30/10/2020
Acción	Eliminar lógicamente una reservación en la base de datos del Sistema.

Objetivo	Verificar que los usuarios con el rol de Administrador puedan eliminar una reservación de la base de datos del sistema.
Descripción	Se realiza la prueba de eliminar una reservación para detectar errores al momento de realizar el proceso.
Criterios de éxito	El sistema deberá mostrar una alerta preguntando al usuario si desea eliminar la reservación. El sistema deberá mostrar una notificación confirmando la eliminación de la reservación seleccionado. El sistema deberá actualizar la lista de las reservaciones.
Criterios de Fallo	El sistema no muestra la alerta de confirmación, tampoco actualiza la lista de las reservaciones ya registrados, no muestra la notificación de confirmación de eliminación de la reservación.
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al módulo de reservación. • Pulsar el botón Eliminar que se encuentra ubicado a la derecha de la reservación que desea eliminar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción reservación en el menú principal del sistema. • Pulsar el botón Eliminar al lado de la reservación que desea eliminar. • El sistema muestra una alerta preguntando al usuario si desea eliminar la reservación. • Pulsar el botón Sí.
Resultados Obtenidos	Se eliminó la reservación de manera correcta.

Tabla 45.Caso de Prueba 006.

Caso de Prueba N° 4: Registrar Habitación	
Módulo	Habitación
Autor	Luis Carlos Flores Baltodano
Fecha	30/10/2020
Acción	Registrar una nueva Habitación en la base de datos.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan registrar una nueva Habitación correctamente.
Descripción	Se realiza la prueba de registrar una Habitación para detectar errores al momento de realizar el proceso.
Criterios de éxito	El sistema envía mensaje de éxito al guardar o mensaje de validación con los campos que no cumplan con las reglas establecidas, la lista en pantalla se actualiza con la Habitación que acabamos de registrar.
Criterios de Fallo	El sistema no retorna mensaje de validación ni de guardado, tampoco actualiza la pantalla mostrando la nueva información digitada.
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al formulario de nueva Habitación y completar dicho formulario. • Pulsar el botón Guardar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción Habitaciones en el menú principal del sistema. • Pulsar el botón Agregar Habitación. • Completar el formulario con la información requerida. • Pulsar el botón Guardar.
Resultados Obtenidos	Se registró la nueva Habitación de manera correcta.

Tabla 46.Caso de Prueba 007.

Caso de Prueba N° 5: Editar Habitación	
Módulo	Habitación
Autor	Luis Carlos Flores Baltodano
Fecha	30/10/2020
Acción	Actualizar la información de una Habitación ya registrado en la base de datos.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan actualizar la información de una Habitación correctamente.
Descripción	Se realiza la prueba de editar una Habitación para detectar errores al momento de realizar el proceso.
Criterios de éxito	El sistema deberá mostrar la opción de modificar solamente a los usuarios con rol Administrador, al realizar la modificación deberá recibir un mensaje de guardado y actualización en pantalla de los datos de la Habitación.
Criterios de Fallo	El sistema no retorna mensaje de validación ni de guardado, tampoco actualiza la pantalla mostrando la nueva información digitada.
Usuario	Administrador
Pre-Condicion	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al formulario de editar Habitación y realizar los cambios necesarios en dicho formulario. • Pulsar el botón Actualizar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción Habitaciones en el menú principal del sistema. • Pulsar el botón Editar al lado de la Habitación que desea editar. • Actualizar la información del formulario con la información necesaria.

Resultados Obtenidos	<ul style="list-style-type: none"> • Pulsar el botón Actualizar.
	Se actualizo la Habitación de manera correcta.

Tabla 47.Caso de Prueba 008.

Caso de Prueba N° 6: Eliminar Habitación	
Módulo	Habitación
Autor	Luis Carlos Flores Baltodano
Fecha	30/10/2020
Acción	Eliminar lógicamente una Habitación en la base de datos del sistema.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan eliminar una Habitación de la base de datos del sistema.
Descripción	Se realiza la prueba de eliminar una Habitación para detectar errores al momento de realizar el proceso.
Criterios de éxito	<p>El sistema deberá mostrar una alerta preguntando al usuario si desea eliminar la Habitación.</p> <p>El sistema deberá mostrar una notificación confirmando la eliminación de la Habitación seleccionado.</p> <p>El sistema deberá actualizar la lista de las Habitaciones.</p>
Criterios de Fallo	El sistema no muestra la alerta de confirmación, tampoco actualiza la lista de las Habitaciones ya registrados, no muestra la notificación de confirmación de eliminación de la Habitación.
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al módulo de Habitaciones. • Pulsar el botón Eliminar que se encuentra ubicado a la derecha de la Habitación que desea eliminar.
	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema

Flujo Normal	<ul style="list-style-type: none"> • Seleccionamos la opción Habitaciones en el menú principal del sistema. • Pulsar el botón Eliminar al lado de la Habitación que desea eliminar. • El sistema muestra una alerta preguntando al usuario si desea eliminar la Habitación. • Pulsar el botón Si.
Resultados Obtenidos	Se eliminó la Habitación de manera correcta.

Tabla 48.Caso de Prueba 009.

Caso de Prueba N° 7: Registrar Cliente	
Módulo	Cientes
Autor	Luis Carlos Flores Baltodano
Fecha	30/10/2020
Acción	Registrar un nuevo Cliente en la base de datos.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan registrar un nuevo Cliente correctamente.
Descripción	Se realiza la prueba de registrar un Cliente para detectar errores al momento de realizar el proceso.
Criterios de éxito	El sistema envía mensaje de éxito al guardar o mensaje de validación con los campos que no cumplan con las reglas establecidas, la lista en pantalla se actualiza con el Cliente que acabamos de registrar.
Criterios de Fallo	El sistema no retorna mensaje de validación ni de guardado, tampoco actualiza la pantalla mostrando la nueva información digitada.
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador.

	<ul style="list-style-type: none"> • Acceder al formulario del nuevo Cliente y completar dicho formulario. • Pulsar el botón Guardar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema. • Seleccionamos la opción Clientes en el menú principal del sistema. • Pulsar el botón Agregar Cliente. • Completar el formulario con la información requerida. • Pulsar el botón Guardar.
Resultados Obtenidos	Se registro el nuevo Cliente de manera correcta.

Tabla 49.Caso de Prueba 010.

Caso de Prueba N° 8: Editar Cliente	
Módulo	Clientes
Autor	Luis Carlos Flores Baltodano
Fecha	30/10/2020
Acción	Actualizar la información de un Cliente ya registrado en la base de datos.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan actualizar la información de un Cliente correctamente.
Descripción	Se realiza la prueba de editar un Cliente para detectar errores al momento de realizar el proceso.
Criterios de éxito	El sistema deberá mostrar la opción de modificar solamente a los usuarios con rol Administrador, al realizar la modificación deberá recibir un mensaje de guardado y actualización en pantalla de los datos de los Clientes.
Criterios de Fallo	El sistema no retorna mensaje de validación ni de guardado, tampoco actualiza la pantalla mostrando la nueva información digitada.
Usuario	Administrador

Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al formulario de editar Cliente y realizar los cambios necesarios en dicho formulario. • Pulsar el botón Actualizar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción Clientes en el menú principal del sistema. • Pulsar el botón Editar al lado de la Cliente que desea editar. • Actualizar la información del formulario con la información necesaria. • Pulsar el botón Actualizar.
Resultados Obtenidos	Se actualizo el Cliente de manera correcta.

Tabla 50.Caso de Prueba 011.

Caso de Prueba N° 9: Eliminar Cliente	
Módulo	Clientes
Autor	Luis Carlos Flores Baltodano
Fecha	30/10/2020
Acción	Eliminar lógicamente un Cliente en la base de datos del sistema.
Objetivo	Verificar que los usuarios con el rol de Administrador puedan eliminar un Cliente de la base de datos del sistema.
Descripción	Se realiza la prueba de eliminar un Cliente para detectar errores al momento de realizar el proceso.
Criterios de éxito	<p>El sistema deberá mostrar una alerta preguntando al usuario si desea eliminar el Cliente.</p> <p>El sistema deberá mostrar una notificación confirmando la eliminación del Cliente seleccionado.</p>

Criterios de Fallo	<p>El sistema deberá actualizar la lista de los Clientes.</p> <p>El sistema no muestra la alerta de confirmación, tampoco actualiza la lista de los Clientes ya registrados, no muestra la notificación de confirmación de eliminación del Cliente.</p>
Usuario	Administrador
Pre-Condiciones	<ul style="list-style-type: none"> • El usuario debe estar autenticado y debe de poseer el rol de Administrador. • Acceder al módulo de Cliente. • Pulsar el botón Eliminar que se encuentra ubicado a la derecha del Cliente que desea eliminar.
Flujo Normal	<ul style="list-style-type: none"> • Iniciar Sesión dentro del Sistema • Seleccionamos la opción Cliente en el menú principal del sistema. • Pulsar el botón Eliminar al lado del Cliente que desea eliminar. • El sistema muestra una alerta preguntando al usuario si desea eliminar el Cliente. • Pulsar el botón Sí.
Resultados Obtenidos	Se eliminó el Cliente de manera correcta.

Tabla 51.Caso de Prueba 012.

XIII.IMPLEMENTACIÓN DEL SISTEMA

Una vez desarrollado el sistema podemos comenzar la implementación en producción del mismo, esta etapa es clave ya por fin se podrán introducir las nuevas soluciones al problema presentado en el comienzo del documento.

Esta fase está íntimamente unida a la formación ya que resulta necesario la capacitación del personal para el uso adecuado de la herramienta desarrollada, es por ello que colocamos en esta parte como soporte las actas de entrega del sistema al representante del hotel paraíso condominiuns, así como la del personal que fue presentado y capacitado para el uso del mismo.

ACTA DE ENTREGA

El Br. Luis Carlos Flores Baltodano y el Br. Rommel Humberto Pérez Gómez, hace entrega formal de la versión 1.0.0 del Sistema administrativo para el hotel Paraíso Condominiuns (SAPCON).

NOMBRE DEL SISTEMA DE INFORMACIÓN	Sistema administrativo para el hotel Paraíso Condominiuns (SAPCON)
VERSIÓN DEL SISTEMA DE INFORMACIÓN	1.0.0
EMPRESA	ParaisoCondominiuns
FECHA DE ENTREGA	25 de octubre del 2020

Entregue Conforme
Br. Rommel Humberto Pérez Gómez

Recibí Conforme

Br. Luis Carlos Flores Baltodano

ACTA DE PRESENTACIÓN Y CAPACITACIÓN

Sistema administrativo para el hotel Paraíso Condominiuns (SAPCON)

N: 001	Fecha: 30/10/2020	Hora Inicio: 1:00 pm	Hora Fin: 6:00 pm	Realizada p o r : Rommel Humberto Pérez Gómez Rommel Humberto Pérez Gómez
Nombre del Sistema de Información: Sistema administrativo para el hotel Paraíso Condominiuns (SAPCON)				N° Versión: 1.0.0
Objetivo: Presentar la versión final del sistema desarrollado para la administración de reservaciones, tours, clientes. Capacitar sobre el uso y control del sistema de administracion				
Nombres		Cargos	Oficinas	Firma
Rommel Humberto Pérez Gómez			N/A	
Luis Carlos Flores Baltodano				
		Administrador	N/A	
Conclusiones: Se concluye que la presentación de la versión 1.0.0 es de satisfacción completa para el usuario, ya que se encuentra la solución a las necesidades solicitadas por los mismos.				

Capítulo VII: Conclusiones y Recomendaciones

XIII. CONCLUSIONES Y RECOMENDACIONES

13.1 Conclusiones

El sistema administrativo implantado bajo las tecnologías web para la administración de los procesos de reservación en el hotel “Paraíso condominiuns del departamento de Granada” ayuda al personal de la comunidad hotelera a realizar tareas de procesos de tipo administrativas en forma ágil, optimizando tiempo y recursos.

Bootstrap es un marco de trabajo de fácil implementación para desarrollar sistemas administrativos en tecnologías web adaptativos, debido a que su utilización no requiere conocimientos amplios en materia de estilos CSS y además dispone de una extensa documentación acerca de la implementación de sus componentes.

La comunicación entre distintas plataformas de desarrollo de software ha dejado de ser un inconveniente, debido a la aparición de las tecnologías web los cuales usan estándares y protocolos muy bien definidos, en el caso del presente proyecto se usó servicios a través de tecnologías de desarrollo web, basados en CRUD, para el desarrollo de creación, actualización y eliminación en los diferentes módulos administrativos desarrollados en el sistema administrativo.

El hotel “Paraíso condominiuns del departamento de Granada” será pionera en integrar sus servicios en la plataforma web, el impacto de esto promoverá el uso de implementar sistemas administrativos en los diferentes hoteles que en el país todavía no han dado este gran paso en la inserción de estas tecnologías para dejar a un lado los procesos manuales que aún se pueden encontrar en muchos lugares de esta índole en el país.

13.2 Recomendaciones

Se le recomienda al hotel paraíso condominiuns los siguientes puntos importantes, que ayudaran a que el sistema siempre este operativo de manera óptima:

- Se recomienda la utilización de la metodología de desarrollo RUP para medianos y pequeños proyectos, debido la secuencia de pasos que se utiliza para el desarrollo y/o mantenimiento en gran cantidad de sistemas, en diferentes áreas de aplicación, diferentes organizaciones, diferentes medios de competencia y en proyectos de tamaños variables (desde el más básico al más complejo), Además de estar basado en un enfoque disciplinado de asignación de tareas y responsabilidades dentro de una organización de desarrollo con la finalidad de asegurar la obtención de un software de alta calidad que satisfaga la necesidad de los usuarios finales dentro de un calendario y tiempo predecible.
- Para el desarrollo de sistemas administrativos se recomienda usar el patrón de arquitectura MTV por las ventajas que tiene su implementación, la separación de la capa de datos, y la lógica de negocio de la interfaz de usuario permiten la reutilización de código y hacen posible un mantenimiento adecuado a lo largo del tiempo.
- Se aconseja implementar un módulo de notificaciones que permitan poder enviar anuncios u ofertas especiales a lo largo del tiempo a todos aquellos clientes que visitaron el hotel, para mantener permanentemente un contacto con ellos y de esta manera mantener informado a estos por si hay ofertas de interés para alguno de ellos.
- Se sugiere el empleo de técnicas de marketing dirigidas a promocionar la aplicación desarrollada hacia los usuarios finales.

XIV. BIBLIOGRAFÍA

- [1] Kimmel, P., (2008). Manual de UML.
- [2] Pressman, S, Roger., (2010). Ingeniería de software un enfoque práctico.
- [3] Sommerville, Ian., (2011). Ingeniería de software.
- [4] agac1, Anónimo (2008). MARCO TEORICO. México. Recuperado de:
<https://faga.blogcindario.com/2008/10/00012-marco-teorico.html>.
- [5] Santillán García, Arturo. Ibáñez Soto, María. Ortiz González, Nereyda., (2011). LOS SISTEMAS INFORMÁTICOS DE GESTIÓN HOTELERA Y LOS BENEFICIOS DE SU IMPLEMENTACIÓN. Universidad de Málaga, España. Recuperado de: <http://www.eumed.net/rev/turydes/11/sio.html>.
- [6] Rodríguez Martínez, Joaquín., (2013). LOS SISTEMAS DE INFORMACIÓN EN EL SECTOR HOTELERO: UN MODELO DE ÉXITO. Universidad de Girona. Recuperado de:
<https://www.tdx.cat/bitstream/handle/10803/319717/tjmr1de1.pdf?sequence=7>
- [7] Baca Urbina, G. (2014). Administración Informática I. México DF: Grupo Editorial Patria.
- [8] Andreu, R., & Ricart, J. (1996). Estrategias y Sistemas de información. Madrid: McGraw-Hill.
- [9] López Rosciano, R. a., & Pech Montejo, J. A. (2015). Desarrollo de herramientas de gestión d proyectos RUP usando metodología SCRUM + XP: Pruebas. Madrid: Universidad Politécnica de Madrid.
- [10] Eguíluz Pérez, J. (2008). Introducción a JavaScript. Madrid: Creative Commons.

[11] García Saul, M. (2015). La guía definitiva de Django. Boston, Estados Unidos.

[12] Python como backend, Anonimo., (2019). Recuperado de:
<https://platzi.com/backend-python/>.

[13] ¿Que aporta Python para el desarrollo web?, Fenollosa, A., (2018).
Recuperado de: <https://programadorwebvalencia.com/que-aporta-python-para-el-desarrolloweb/>.

[14] Hill, M. (2002). FUNDAMENTOS DE BASES DE DATOS. Cuarta edición.
Madrid. Edificio Valrealty.

[15] Entorno de desarrollos, Anónimo. Recuperado de:
<https://www.arimetrics.com/glosario-digital/entorno-de-desarrollo>.

[16] Sublime HQ Pty Ltd. (2019). Sublime text. Woollahra, Sydney.
Recuperado de: <https://www.sublimetext.com/>.

[17] ¿Que es bootstrap y para qué sirve? Casas, V. (2019). Recuperado de:
<https://www.lucushost.com/blog/que-es-bootstrap>.

[18] El gran libro de HTML5, CSS3 y JavaScript. Gauchat, J, D. (2012).
Barcelona. Gran Via de les Corts Catalanes.

[19] Software hotelero Opera, Eulises Ortiz, A. (2019). Recuperado de:
<https://pcweb.info/software-hotelero-opera-programa-hoteles-sistema-gestionhotelera/>.

[20] Software hotelero Fidelio, Anónimo. (2018). Recuperado de:
<http://www.softwarefidelio20.blogspot.com/>.

[21] Software hotelero Zeus, Cecilia Suarez, M. (2014). Recuperado de:
<https://es.scribd.com/doc/248214603/ZEUS-HOTELERO>.

[22] ¿Que es la modelo template view? Desconocido. (Desconocido).
Recuperado de: <https://si.ua.es/es/documentacion/asp-net-MTV-3/1-dia/modelo-vistacontrolador-MTV.html>

xv. ANEXOS

14.1. Glosario

1. Algoritmo: Es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad. Dados un estado inicial y una entrada, siguiendo los pasos sucesivos se llega a un estado final y se obtiene una solución.

2. API: La interfaz de programación de aplicaciones (API) es el conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas.

3. Sistema: conjunto ordenado de componentes relacionados entre sí, ya se trate de elementos materiales o conceptuales, dotado de una estructura, una composición y un entorno particulares. Se trata de un término que aplica a diversas áreas del saber, como la física, la biología y la informática o computación.

4. IDE: Es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

5. JSON: Acrónimo de JavaScript Object Notation, es un formato de texto ligero para el intercambio de datos.

6. Objeto: Es una unidad dentro de un programa de computadora que consta de un estado y de un comportamiento, que a su vez constan respectivamente de datos almacenados y de tareas realizables durante el tiempo de ejecución.

7. CRUD: CRUD resume las funciones requeridas por un usuario para crear y gestionar datos.

14.2. Manual de usuario

Diseño de Entradas

Los botones estándar para las entradas de la información en el sistema son los siguientes:

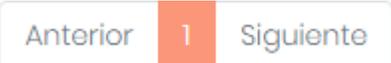
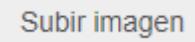


A continuación, se describe cada uno de los botones:

Botón	Descripción
Registrar	Es utilizado para agregar un nuevo registro a la base de Datos
Editar	Es utilizado para modificar un determinado registro de la base de datos
Agregar	Es utilizado para guardar los cambios realizados por los botones de agregar y modificar.
Eliminar	Es utilizado para cancelar cualquier acción realizada dentro del formulario actual
Cancelar	Es utilizado para Cancelar una acción en el sistema

Además de los botones estándar, el sistema también maneja otros botones los cuales se describen a continuación:

Botón	Descripción
	Este botón es utilizado para agregar un nuevo producto
	Este botón es utilizado para guardar una venta
	Estos botones son utilizados para generar un reporte en Excel, pdf o bien imprimir directamente los registros actuales

	Estos botones son los de paginación para movernos a la siguiente página o anterior de los registros que se tienen
	Este botón permite anular un registro
	Estos botones permiten ver tarifas y dar mantenimiento a una habitacion
	Esta opción permite buscar un dato en especifico
	Este botón permite filtrar información del reporte
	Este botón permite imprimir un documento
	Estos botones filtran información especifica
	Este botón permite subir una imagen

A continuación, se muestran algunas de las pantallas de captura de datos del sistema:

Nombre de Entrada:	Datos Generales del cliente
Objetivo:	Almacenar los datos personales de los clientes (as) que forman parte de los inquilinos del hotel
Datos Contenidos:	
Usuario que lo utiliza:	Administrador, Recepcionista
Tablas Accesada	Atributos
Cliente	Código del cliente= id_cliente Título del usuario= Titulo Nombre completo del cliente= Nombre Email del cliente= Email

	<p>Edificador del cliente= Identificador</p> <p>Origen del cliente= Origen</p> <p>País del cliente= País</p> <p>Celular del cliente= Celular</p> <p>Acciones= Acciones</p>
--	--

Visualización en el Sistema:

The screenshot shows the HOTEL Paraiso system interface. At the top, there is a header with the hotel name, a search bar for clients, and the user role 'Administrador'. Below the header is a navigation sidebar with icons for various system modules: Reserva, Recepción, Punto de venta, Inventario, Módulo caja, Egresos, Configuración, Clientes, Reportes, and Administración. The main content area displays a calendar for JUNIO 2021. The calendar grid has columns for days of the month (ma 1 to mi 30) and rows for room numbers (Habitación 102 to 401). A date range from June 25th to 27th is highlighted in yellow, indicating a selected period.

Registro de clientes

Registro de cliente

Inicio > recepción > Procesar

Datos de la habitación

Nombre: Habitación 101 Tipo: Dobles

Detalles: Con cama de 2 plazas, una TV, Cable Estado: DISPONIBLE

DATOS DEL CLIENTE

Tipo Documento: PASAPORTE ON°:

Nombres:

E-mail: Teléfono:

Razón social:

Procedencia:

Factura: Folio de factura:

Nota:

Toallas:

DATOS DEL ALOJAMIENTO

Tarifa:

Cant. de personas:

Estado de pago:

Fecha salida: Hora salida:

Cancelar
Registrar ingreso

DIRECTORIO DEL CLIENTE							+ NUEVO CLIENTE	⚙	✕
N°	Tipo documento	Documento	Nombres completos	Razón social	Dirección	Fecha nac.			
81	PASAPORTE	0					Contactos	Editar	Eliminar
79	PASAPORTE	201	mario sandoval		dd		Contactos	Editar	Eliminar
80	PASAPORTE	233	DSDS		calle la ceiba		Contactos	Editar	Eliminar
77	PASAPORTE	5234384	Lozano Magin Carlos hugo	NULL	xxxx		Contactos	Editar	Eliminar
78	DNI	5298765	José Manuel Rios Montes	NULL	xxxx		Contactos	Editar	Eliminar
74	DNI	44710181	Lozano Magin Carlos hugo	NULL	xxxx	1987-09-18	Contactos	Editar	Eliminar
71	DNI	74045101	Tatiana Abigail Rengifo Uribe	NULL	xxxx		Contactos	Editar	Eliminar

« < 1 2 > »

Listado de clientes registrados

Nombre de Entrada:	Gestionar reservaciones
Objetivo:	Almacenar los datos de las reservaciones del hotel
Datos Contenidos:	
Usuario que lo utiliza:	Administrador, Recepcionista
Tabla Accesada	Atributos
Reservación	Código=id_reservacion Nombre habitación= Habitación Fecha de entrada= init_date Fecha de salida= last_date Cantidad de adultos para la reservación= Adultos Cantidad de niños para la reservación= Niños Incluir desayuno=breackfast Nota= notes

Visualización en el Sistema:

MANTENIMIENTO DE HABITACIONES

REGISTRAR NUEVA HABITACIÓN

Buscar:

N°	NOMBRE	CATEGORÍA	DETALLES				
6	Habitación 101	Dobles	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento
7	Habitación 102	Dobles	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento
8	Habitación 103	Dobles	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento
9	Habitación 104	Dobles	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento
10	Habitación 105	Triple	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento
11	Habitación 106	Dobles	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento
12	Habitación 201	Personal	Con cama de 2 plazas, una TV, Cable	Ver tarifas	Editar	Eliminar	Mantenimiento

« < 1 2 3 4 > »

Registro de habitación

N°	IMAGEN	NOMBRE	EDITAR	ELIMINAR
1		Personal	Editar	Eliminar
2		Dobles	Editar	Eliminar
3		Triple	Editar	Eliminar
4		Cuadruples	Editar	Eliminar
5		Matrimoniales	Editar	Eliminar
6		Simples	Editar	Eliminar

INGRESAR NUEVA HABITACIÓN



Nombre

Categoría

Detalles

Cerrar

Agregar Datos

Registro de reservación

Inicio > recepción > Procesar

Datos de la habitación

Nombre: Habitación 101 Tipo: Dobles

Detalles: Con cama de 2 plazas, una TV, Cable Estado: **DISPONIBLE**

DATOS DEL CLIENTE

Tipo Documento: PASAPORTE ON?: Ingrese el número

Nombres: Ingrese nombres

E-mail: Ingrese Correo Electrónico Teléfono: Ingrese teléfono

Razón social: Ingrese razon social

Procedencia: Ingrese procedencia (No es obligatorio)

Factura: Ingrese factura Folio de factura: Ingrese folio de factura

Nota: Ingrese Nota

Toallas: 1

DATOS DEL ALOJAMIENTO

Tarifa: --- Selecciona ---

Cant. de personas: 1

Estado de pago: --- Selecciona ---

Fecha salida: 2021-06-27 Hora salida: 12:00

Cancelar **Registrar ingreso**

Registro de ventas

Detalle de Venta

+ Agregar productos

Producto	Cant.	Precio	Importe	Quitar
----------	-------	--------	---------	--------

Cancelar Venta

IMPORTANTE

Tipo de cliente: --- Selecciona ---

Tipo Comprobante: TICKET Fecha Comprobante: 26/06/2021

Forma de Pago: CONTADO

Guardar Venta

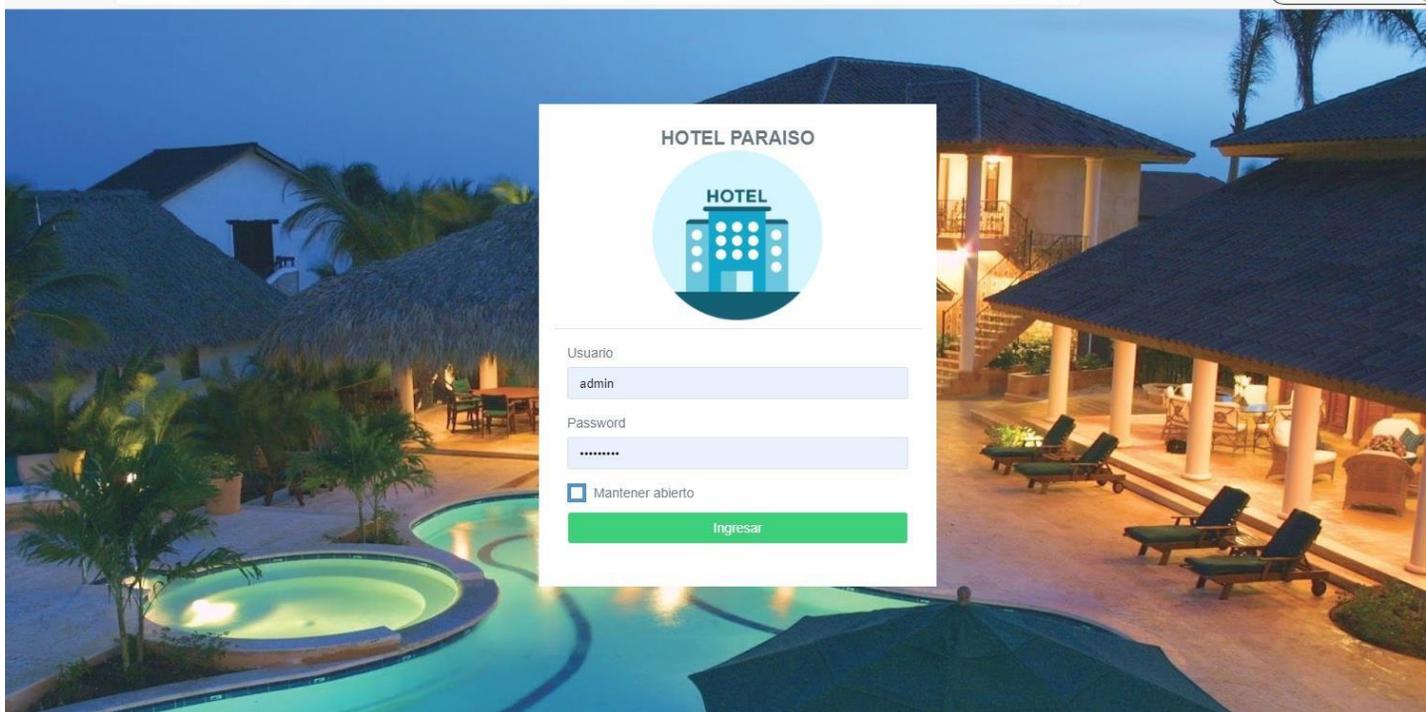
Interfaz principal del sistema

The screenshot shows the main interface of the HOTEL PARAISO system. At the top, there is a header with the hotel name "HOTEL PARAISO", a search bar labeled "Buscar Cliente...", and a user profile icon labeled "Administrador". Below the header is a navigation menu with icons for "Reserva", "Recepción", "Punto de venta", "Inventario", "Módulo caja", "Egresos", "Configuración", "Clientes", and "Reportes". The main content area features a calendar for "JUNIO 2021" with a grid of days from "ma 1" to "mi 30". A list of rooms is displayed on the left side of the calendar, including "Habitación 102" through "Habitación 304". The calendar grid shows a yellow highlight for the 25th and 26th of June, and a blue highlight for the 27th.

Funcionalidad

Acceso al Sistema

Se definió una pantalla en la cual se autenticará el usuario:



Ingreso al sistema

Dependiendo de ello se visualizará un formulario principal que incluye menú



Interfaz Principal

El formulario principal donde se encuentran algunas pestañas con los accesos a otras ventanas con todas las opciones habilitadas o con parte de ellas, por ejemplo: para el subadministrador y el personal se habilitarán ciertas opciones y para administrador incluirá todas ellas, ya que hay ciertos niveles de privilegios para almacenar, modificar, imprimir o eliminar datos, a menos que esté autorizado por encargados del hotel. Las opciones que se presentan en dicho formulario son:

Cliente. - En esta opción el usuario podrá agregar, editar, actualizar la información del cliente. Además de obtener la lista de clientes