

**Diseño e implementación de sistema web gestor de alojamiento con reporte TRA para
prestadores de servicios turísticos en Murillo Tolima**

Jeimy Patricia Popayan Valencia

Asesor:

Omar Leonardo Leyton Mendieta

Universidad Nacional Abierta y a Distancia- UNAD
Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI
Tecnología en Desarrollo de Software

2025

Dedicatoria

A Dios, porque me permitió estar siempre en el lugar correcto, con la gente precisa, en el tiempo indicado, brindándome con cada día de vida la oportunidad de avanzar hacia mis objetivos personales y profesionales. Así mismo, a mi madre que, con su esfuerzo, valentía, disciplina y ejemplo me saco adelante, convirtiéndose en mi inspiración, mi guía y mi apoyo.

Agradecimientos

Este proyecto de grado no habría culminado con éxito de no ser por el apoyo intelectual y humano de muchas personas. Expreso mi agradecimiento a la Universidad Nacional Abierta y a Distancia por abrirme sus puertas y formarme académicamente bajo su modelo flexible e incluyente. Del mismo modo, al tutor Omar Leonardo Leyton Mendieta por su orientación crítica, disposición constante, paciencia y acompañamiento en desarrollo del proyecto. También a los prestadores de servicios turísticos de Murillo Tolima, por compartir la información necesaria para realizar la investigación. Finalmente, a mi familia y amigos que me acompañaron y apoyaron con su granito de arena en cada paso de mi formación, brindándome sus consejos, palabras de aliento, paciencia y amor incondicional.

Resumen

Este proyecto propone diseñar e implementar un sistema web para la gestión del servicio de alojamiento turístico como herramienta indispensable para los pequeños y nuevos empresarios en el municipio de Murillo Tolima, zona que se destaca por su creciente desarrollo y que requiere por parte de los comerciantes cumplir con la normatividad vigente en cuanto al registro y reporte de los visitantes a través el sistema TRA (Tarjeta de Registro de Alojamiento) del Ministerio de Comercio, Industria y Turismo.

El soporte teórico y de diseño se basa en la aplicación de tecnologías de software como PHP, JavaScript, HTML5, Bootstrap 5 y bases de datos relacionales MySQL, utilizando un proceso de desarrollo secuencial que garantiza un progreso organizado y controlado. Los propósitos incluyen mejorar la automatización de tareas, optimizar la gestión de información y desarrollar un sistema accesible y efectivo.

El método propuesto abarca etapas detalladas de recopilación y análisis de información, diseño, implementación y pruebas. Se espera como resultado la entrega de un sistema funcional que mejore la gestión de registros y la creación del reporte TRA, beneficiando tanto a los prestadores de servicios turísticos como al Ministerio, al facilitar la supervisión del sector en la región.

Palabra claves: Sistema web, software, TRA (Tarjeta de Registro de Alojamiento), prestadores de servicios turísticos.

Abstract

This project proposes to design and implement a web-based system for managing tourist accommodation services as an essential tool for small and new entrepreneurs in the municipality of Murillo, Tolima, an area notable for its growing development and requiring merchants to comply with current regulations regarding the registration and reporting of visitors through the TRA (Accommodation Registration Card) system of the Ministry of Commerce, Industry, and Tourism.

The theoretical and design support is based on the application of software technologies such as pure PHP, JavaScript, HTML5, Bootstrap 5, and MySQL relational databases, utilizing a sequential development process that ensures organized and controlled progress. The objectives include improving task automation, optimizing information management, and developing an accessible and effective system.

The proposed method encompasses detailed stages of information collection and analysis, design, implementation and testing. The expected result is the delivery of a functional system that improves record management and the creation of the TRA report, benefiting both tourism service providers and the Ministry by facilitating oversight of the sector in the region.

Keywords: Web system, software, TRA (Accommodation Registration Card), tourism service providers.

Tabla de Contenido

Introducción	17
Planteamiento del Problema	18
Objetivos.....	19
Objetivo General.....	19
Objetivos Específicos.....	19
Justificación	20
Alcance	22
Límites del Proyecto	22
Marco Referencial.....	23
Antecedentes	23
Marco Conceptual.....	24
Ingeniería de Requerimientos	24
Establecimientos de Alojamientos Turísticos	25
Tarjeta de Registro de Alojamiento -TRA	25
Sistemas de Gestión	25
Marco Teórico	26
Metodología Híbrida Cascada y Agile Scrum	26
Diseño del Sistema.....	27
Lenguajes de Programación y Herramientas de Desarrollo.....	28
Bases de Datos Relacionales.....	29
Servidor Local y Entorno de Desarrollo	29
Impacto y Pertinencia	30

Bases Legales.....	30
Metodología.....	32
Fase 1: Identificación de los Requerimientos.....	32
Fase 2: Diseño.....	33
Fase 3: Implementación.....	34
Fase 4: Pruebas.....	34
Fase 1: Identificación de Requerimientos y Necesidades.....	35
Etapas de Construcción del Instrumento: Encuesta.....	36
Identificación del Objetivo del Estudio.....	37
Desglose del Objetivo en Dimensiones e Indicadores.....	38
Elaboración de Secuencia.....	39
Introducción y Solicitud de Colaboración.....	39
Selección o Filtro del Respondente.....	39
Preguntas Relacionadas con Objetivos.....	40
Preguntas Demográficas.....	41
Agradecimiento e Informe sobre Posible Supervisión.....	41
Datos de Identificación, como Nombre y Dirección, para Posterior Supervisión.....	42
Redacción de Preguntas.....	42
Identificación y Redacción de Alternativas de Respuesta.....	45
Identificación de Pases y Dependencias.....	52
Redacción de Apoyos y Tarjetas.....	52
Redacción de Instrucciones.....	53
Instrucciones Generales para la Encuesta.....	53

Instrucciones Específicas	54
Evaluación del Instrumento	54
Diseño de la Muestra	56
Análisis y Lectura de Datos	56
Análisis de las Respuestas	56
Conclusiones del Análisis de Resultados	70
Identificación de Requerimientos	71
Requerimientos de Descripción General	71
Usuarios Finales.....	72
Restricciones	72
Requerimientos de Interfaces.....	72
Interfaz de Usuario.....	73
Interfaz de Hardware.....	73
Interfaz de Software	74
Interfaces de Comunicaciones	74
Requerimientos Funcionales	74
Requerimientos no Funcionales	75
Casos de Uso.....	77
CU-01 Usuario Administrador	77
CU-02 Usuario Recepcionista.....	78
Escenarios de Casos de Uso.....	79
Escenario 01 Caso de Uso Actores Administrador y Recepcionista.....	79
Escenario 02 Caso de Uso Actor Administrador.....	80

Escenario 03 Caso de Uso Actor Administrador.....	81
Escenario 04 Caso de Uso Actor administrador	82
Escenario 05 Caso de Uso Actores administrador y Recepcionista.....	83
Fase 2 - Diseño.....	86
Base Estructural	86
Modelado de Contexto.....	87
Modelado Dirigido por Datos	87
Modelo de la Estructura de Datos.....	99
Arquitectura del Sistema Web.....	101
Consideraciones de Seguridad y Calidad del Software	102
Directrices de Accesibilidad Web WCAG	103
Prototipos de Interfaz de usuario	104
Fase 3 - Implementación.....	115
Implementación del Código para el Sistema	116
Planificación de los Sprints.....	116
Primer Sprint: Arquitectura y Autenticación.....	117
Sprint Review - Primer Sprint.....	119
Segundo Sprint: Módulos Críticos del Sistema	119
Sprint Review - Segundo Sprint	121
Tercer Sprint: Carga Dinámica de Reservas	122
Sprint Review - Tercer Sprint	124
Cuarto Sprint: Check-out y TRA	124
Sprint Review - Cuarto Sprint.....	127

Quinto Sprint: Optimización del Sistema	127
Sprint Review - Quinto Sprint	129
Fase 4 - Pruebas	130
Pruebas Unitarias	131
Prueba Unitaria Estructura del Proyecto	132
Prueba Unitaria RecepcionesModel.....	133
Pruebas Unitaria Controlador Recepción.....	135
Pruebas de Integración	136
Pruebas de Integración Estructura del Proyecto	136
Pruebas de Integración RecepcionesModel	137
Pruebas de Integración Controlador Recepción.....	139
Conclusión General de las Pruebas Unitarias y de Integración	141
Prueba Piloto.....	142
Fase 1: Definir el Alcance de Prueba Piloto	143
Fase 2: Determinar las Categorías de Usabilidad a Probar.....	144
Fase 3: Diseñar Cuestionario de Evaluación	145
Fase 4: Ejecución de la Prueba Piloto.....	147
Fase 5: Análisis de los Resultados	154
Conclusiones.....	156
Recomendaciones	159
Referencias Bibliograficas	160
Apéndices.....	165

Lista de Tablas

Tabla 1 <i>Dimensiones e Indicadores</i>	38
Tabla 2 <i>Requerimientos Funcionales</i>	74
Tabla 3 <i>Requisitos no Funcionales</i>	76
Tabla 4 <i>Escenario 01 Actor Administrador y Recepcionista</i>	79
Tabla 5 <i>Escenario 02 Actor Administrador</i>	80
Tabla 6 <i>Escenario 03 Actor Administrador</i>	81
Tabla 7 <i>Escenario 04 Actor Administrador</i>	82
Tabla 8 <i>Escenario 05 Actor Administrador Recepcionista</i>	83
Tabla 9 <i>Product Backlog</i>	116
Tabla 10 <i>Spring Backlog – Primer Sprint</i>	117
Tabla 11 <i>Spring Backlog – Segundo Sprint</i>	119
Tabla 12 <i>Sprint Backlog – Tercer Sprint</i>	122
Tabla 13 <i>Sprint Backlog – Cuarto Sprint</i>	124
Tabla 14 <i>Sprint Backlog – Quinto Sprint</i>	127
Tabla 15 <i>Tipos de Pruebas para Software</i>	130
Tabla 16 <i>Resultados de Pruebas Unitarias Archivos de Estructura del Proyecto</i>	132
Tabla 17 <i>Resultados de Pruebas Unitarias RecepcionesModel</i>	133
Tabla 18 <i>Resultados de Pruebas Unitarias Controlador Recepción</i>	135
Tabla 19 <i>Resultados de Pruebas de Integración Archivo Query</i>	136
Tabla 20 <i>Resultados de Prueba de Integración RecepcionesModel</i>	137
Tabla 21 <i>Resultados de Pruebas de Integración Controlador Recepción</i>	139
Tabla 22 <i>Tareas de la Prueba Piloto</i>	144

Lista de Figuras

Figura 1 <i>Pasos para Diseñar el Instrumento</i>	35
Figura 2 <i>Etapas para la Construcción de Preguntas del Instrumento</i>	37
Figura 3 <i>Evaluación por Experto</i>	55
Figura 4 <i>Resultados - Pregunta 1</i>	57
Figura 5 <i>Resultados - Pregunta 2</i>	57
Figura 6 <i>Resultados - Pregunta 3</i>	58
Figura 7 <i>Resultados - Pregunta 4</i>	59
Figura 8 <i>Resultados - Pregunta 5</i>	60
Figura 9 <i>Resultados - Pregunta 6</i>	61
Figura 10 <i>Resultados - Pregunta 7</i>	62
Figura 11 <i>Resultados - Pregunta 8</i>	63
Figura 12 <i>Resultados - Pregunta 9</i>	64
Figura 13 <i>Resultados - Pregunta 10</i>	65
Figura 14 <i>Resultados - Pregunta 11</i>	66
Figura 15 <i>Resultados - Pregunta 12</i>	66
Figura 16 <i>Resultados - Pregunta 13</i>	67
Figura 17 <i>Resultados - Pregunta 14</i>	68
Figura 18 <i>Resultados - Pregunta 15</i>	69
Figura 19 <i>Resultados - Pregunta 16</i>	70
Figura 20 <i>CU Usuario Administrador</i>	77
Figura 21 <i>CU Usuario Recepcionista</i>	78
Figura 22 <i>Contexto del Sistema Web</i>	87
Figura 23 <i>Diagrama Flujo de Actividad Inicio de Sesión</i>	88

Figura 24 <i>Diagrama Flujo de Actividad Registrar Usuario Empleado</i>	89
Figura 25 <i>Diagrama Flujo de Actividad Registrar Tipos de Alojamiento</i>	90
Figura 26 <i>Diagrama Flujo de Actividad Registrar Tipos de Habitación</i>	91
Figura 27 <i>Diagrama Flujo de Actividad Registrar Tarifas de Alojamiento</i>	92
Figura 28 <i>Diagrama Flujo de Actividad Registrar Tarifas del Parqueadero</i>	93
Figura 29 <i>Diagrama Flujo de Actividad Registrar Reserva para Huésped Principal</i>	94
Figura 30 <i>Diagrama Flujo de Actividad Registrar Huésped Acompañante</i>	95
Figura 31 <i>Diagrama Flujo de Actividad Traslado de Habitación</i>	96
Figura 32 <i>Diagrama Flujo de Actividad Check-out para los Huespedes</i>	97
Figura 33 <i>Diagrama Flujo de Actividad Generación de la TRA</i>	98
Figura 34 <i>Diseño Entidad - Relación</i>	99
Figura 35 <i>Modelo - Vista - Controlador</i>	101
Figura 36 <i>Prototipo del Formulario Inicio de Sesión</i>	105
Figura 37 <i>Prototipo del Área de Trabajo “Empleado”</i>	106
Figura 38 <i>Prototipo del Formulario Registro de Usuario Empleado</i>	107
Figura 39 <i>Prototipo del Área de Trabajo “Habitaciones”</i>	108
Figura 40 <i>Prototipo del Formulario Registro de Tipo de Habitaciones</i>	108
Figura 41 <i>Prototipo de Área de Trabajo “Alojamientos”</i>	109
Figura 42 <i>Prototipo del Formulario Registro de Servicios de Alojamiento</i>	109
Figura 43 <i>Prototipo del Área de Trabajo “Establecimiento”</i>	110
Figura 44 <i>Prototipo del Área de Trabajo “Tarifas”</i>	110
Figura 45 <i>Prototipo del Formulario Registro de Tarifas de Habitaciones</i>	111
Figura 46 <i>Prototipo del Área de Trabajo “Reporte TRA”</i>	111

Figura 47 <i>Prototipo del Área de Trabajo “Gestión de Reservas”</i>	112
Figura 48 <i>Prototipo del Formulario Registro de Reservas</i>	113
Figura 49 <i>Prototipo del Formulario de Registro Acompañante</i>	113
Figura 50 <i>Prototipo del Formulario Finalizar Reserva</i>	114
Figura 51 <i>Prototipo del Formulario Cambio de Habitación</i>	114
Figura 52 <i>Elementos de Scrum</i>	115
Figura 53 <i>Edición de Formulario Usuario Empleado</i>	118
Figura 54 <i>Edición de Formulario Tipos de Alojamientos</i>	118
Figura 55 <i>Edición de Formulario Tipo de Habitación para Alojamientos</i>	120
Figura 56 <i>Edición de Formulario Tarifas</i>	121
Figura 57 <i>Edición de la Reserva para Huésped Principal</i>	123
Figura 58 <i>Edición del Huésped Acompañante</i>	123
Figura 59 <i>Validación Registro Limite de Acompañantes</i>	125
Figura 60 <i>Descarga del Reporte TRA</i>	126
Figura 61 <i>Reporte TRA</i>	126
Figura 62 <i>Cambiar Reservas de Habitación</i>	128
Figura 63 <i>Disponibilidad de Habitaciones</i>	129
Figura 64 <i>Fases de la Prueba Piloto</i>	143
Figura 65 <i>Categorías de Usabilidad para la Prueba Piloto</i>	145
Figura 66 <i>Formato de la Encuesta para la Evaluar la Prueba Piloto</i>	146
Figura 67 <i>Prueba Piloto Establecimiento 1 Usuario Recepcionista</i>	147
Figura 68 <i>Prueba Piloto Establecimiento 2 Usuario Admin-Recepcionista</i>	148
Figura 69 <i>Prueba Piloto Establecimiento 3 Usuario Recepcionista</i>	148

Figura 70 <i>Prueba Piloto Establecimiento 3 Usuario Recepcionista</i>	149
Figura 71 <i>Respuesta Evaluación de la Prueba Piloto Usuario 1 Hotel Suarez</i>	150
Figura 72 <i>Respuesta Evaluación de la Prueba Piloto Usuario 2 Hotel Chalet</i>	151
Figura 73 <i>Respuesta Evaluación de la Prueba Piloto Usuario 3 Hostal Casa Celeste</i>	152
Figura 74 <i>Respuesta Evaluación de la Prueba Piloto Usuario 4 Hostal Casa Celeste</i>	153
Figura 75 <i>Resultados de la Prueba Piloto</i>	154

Lista de Apéndices

Apéndice A *Manual de Usuario*..... 165

Apéndice B *Resumen Analítico Especializado RAE* 178

Introducción

Mediante la aplicación de principios de ingeniería de software y programación orientada a objetos, se estructurará y desarrollará un sistema web diseñado y destinado a la gestión de alojamientos turísticos, enfocado en resolver algunas necesidades que enfrentan los prestadores de servicios turísticos en Murillo, Tolima, relacionadas con la gestión de huéspedes, reservas, tarifas, servicios, reportes y cumplimiento de la normatividad del Ministerio de Comercio, Industria y Turismo sobre la Tarjeta de Registro de Alojamiento (TRA).

La propuesta tiene como objetivo principal crear un sistema web que sirva como herramienta de apoyo para cumplir con la normatividad vigente, además de la gestión de los servicios de los alojamientos, mediante un sistema intuitivo, accesible y adaptado a las condiciones de los usuarios finales. El desarrollo del sistema web se basa en tecnologías como PHP, MySQL y Bootstrap 5, aplicando el patrón de diseño Modelo-Vista-Controlador (MVC) para garantizar una estructura modular y escalable.

La estrategia de trabajo emplea la combinación del modelo en cascada para las etapas iniciales, como la recolección y análisis de requisitos, con metodologías ágiles durante la implementación y pruebas, asegurando flexibilidad y cumplimiento de plazos. Sin embargo, las limitaciones incluyen la dependencia de los usuarios a equipos con especificaciones mínimas y la conectividad restringida en la región. Este trabajo contribuye al fortalecimiento del sector turístico, promoviendo una transición tecnológica inclusiva y eficiente.

Planteamiento del Problema

En Colombia, el Ministerio de Comercio, Industria y Turismo exige a todos los establecimientos como hoteles, apartahoteles, hostales, centros vacacionales, campamentos, glampings, refugios, albergues, viviendas turísticas, etc., la emisión de la Tarjeta de Registro de Alojamiento (TRA) de cada uno de sus huéspedes. Este proceso es fundamental para el control y la legalidad del sector turístico en el país. Sin embargo, en muchas zonas turísticas, especialmente en áreas rurales, de difícil acceso o en proceso de desarrollo turístico se presentan dificultades de conectividad, acceso tecnológico, capacitación y formación en servicios turísticos, representando obstáculos significativos por superar para cumplir con las normativas y exigencias del servicio.

En Murillo Tolima, los establecimientos turísticos carecen de herramientas tecnológicas que les permitan gestionar de manera eficiente el registro de sus huéspedes y la generación de la TRA debido a su limitada infraestructura tecnológica. Esto no solo dificulta la operatividad diaria, sino que también los expone a sanciones e infracciones legales por parte del Estado al no cumplir con las regulaciones vigentes.

El problema radica en la necesidad de desarrollar un sistema web amigable que permita a los establecimientos turísticos tener una herramienta para gestionar y registrar a sus huéspedes, y generar la TRA, asegurando la sincronización y reporte de los datos ante el Ministerio de Comercio, Industria y Turismo de manera fácil y oportuna.

En conclusión, la pregunta central que motiva esta propuesta es: ¿Cómo lograr construir un sistema web que permita la gestión del alojamiento y la generación del reporte TRA como herramienta para los prestadores de servicios turísticos en Murillo Tolima, facilitando el cumplimiento de las normas legales y mejorando la eficiencia operativa del sector?

Objetivos

Objetivo General

Diseñar e implementar un sistema web para la gestión del alojamiento como herramienta para los prestadores de servicios turísticos cumpliendo la normatividad vigente TRA en el municipio de Murillo Tolima.

Objetivos Específicos

Identificar los requerimientos y necesidades de los prestadores de servicios de alojamiento turístico conforme al contexto regional y a la normatividad vigente.

Diseñar el sistema web para la gestión y control de visitantes, y la generación del reporte TRA conforme la normatividad vigente.

Implementar el sistema web para la gestión de visitantes.

Probar el sistema en un entorno real en la prestación del servicio de alojamiento turístico.

Justificación

El desarrollo de este proyecto permitirá a los prestadores de servicios turísticos tener una herramienta web de fácil uso, flexible, adaptable y gratuita, para gestionar el alojamiento y el reporte TRA, supliendo las necesidades administrativas y generando un aporte para mejorar el servicio, tomar decisiones y cumplir las normas legales.

La ejecución del proyecto permite aplicar conocimientos de desarrollo de software para resolver problemas reales en el contexto turístico. El diseño e implementación del sistema permitirá integrar principios de ingeniería de software, experiencia de usuario y gestión de bases de datos en una solución concreta, fomentando el desarrollo de competencias técnicas avanzadas y aplicadas.

Para el desarrollo del proyecto se dispone del recurso humano y fuentes de información suficientes para la cuidadosa planificación y selección de tecnologías accesibles y probadas como PHP, JavaScript, MySQL y Bootstrap 5. La combinación de las metodologías en cascada y agile asegura un desarrollo organizado y controlado, lo que garantiza la entrega de un producto funcional dentro del plazo establecido. Además, la captura de información precisa de las necesidades del sector por medio de encuestas asegura que la solución desarrollada esté alineada con las demandas reales de los prestadores de servicios turísticos en la región. Este proyecto servirá como un caso de estudio para futuras investigaciones en la optimización de sistemas de gestión en sectores rurales con limitaciones tecnológicas.

Los beneficiarios directos de este proyecto son los prestadores de servicios turísticos en el municipio de Murillo Tolima, al mejorar su infraestructura tecnológica automatizando el proceso de registro y reporte TRA, facilitando la operatividad diaria de estos establecimientos, reduciendo el riesgo de sanciones legales promoviendo una imagen positiva en la región.

Finalmente, la realización de este estudio ofrece beneficios significativos a nivel personal, profesional y disciplinario. Para el desarrollador involucrado, el proyecto representa una oportunidad de crecimiento al mejorar las competencias en tecnologías web y gestión de proyectos de software. Profesionalmente, el estudio contribuye al desarrollo de soluciones tecnológicas aplicadas a sectores con necesidades específicas, como el turismo en zonas rurales, lo que fortalece la capacidad de los desarrolladores para abordar desafíos similares en el futuro. Desde una perspectiva disciplinaria, el proyecto se posiciona como un caso de estudio valioso que demuestra la aplicabilidad de tecnologías modernas en la resolución de problemas concretos, aportando al avance del conocimiento en el ámbito del desarrollo de software.

Alcance

El proyecto se enfoca en el desarrollo de un sistema web funcional para la gestión del servicio de alojamiento turístico en el municipio de Murillo, Tolima. Los alcances se centran en la creación de funcionalidades para registrar, editar y consultar información relacionada con huéspedes y reservas, así como la generación del reporte TRA en formato EXCEL exigido por el Ministerio de Comercio, Industria y Turismo. Además, el sistema permitirá automatizar la información de alojamientos, habitaciones y tarifas mejorando la gestión del establecimiento. El sistema será diseñado bajo principios de usabilidad y eficiencia, utilizando tecnologías ampliamente adoptadas y que integran fácilmente el backend y frontend, lo que permitirá su adaptabilidad en entornos similares de pequeñas empresas turísticas.

Límites del Proyecto

Este proyecto se limita a la gestión interna básica de los servicios de alojamiento turístico, sin incluir módulos y funcionalidades comerciales como reservas en línea por parte de los huéspedes o integración con plataformas externas de turismo, facturación y contabilidad. Tampoco implementa notificaciones automáticas por correo electrónico o SMS, pasarelas de pago en línea, análisis de datos ni reportes estadísticos, centrándose exclusivamente en procesos definidos en el alcance buscando garantizar la estabilidad del sistema web, facilitando su futura ampliación de forma modular y escalada. Finalmente, el sistema está enfocado exclusivamente en el contexto del municipio de Murillo, por lo que no se contemplaron necesidades específicas de otras regiones.

Marco Referencial

Antecedentes

Hasta la fecha se identifican múltiples empresas desarrolladoras de software que permiten la gestión del alojamiento y el reporte TRA , pero con costo considerable y enfocadas a prestadores de servicios turísticos grandes, los cuales no son aptos para nuevos y pequeños prestadores de servicios turísticos que no cuentan con los recursos, dimensión, experiencia, conectividad e infraestructura adecuada, caso presente en sectores alejados o en evolución turística, como es el caso de la mayoría de los prestadores de servicios turísticos de Murillo Tolima. Entre los softwares que se destacan hasta el momento se encuentran los siguientes:

EasyTRA: Plataforma colombiana pensada y creada por proveedores de servicios turísticos como forma de ayuda para cumplir con la obligación del reporte TRA ante el ministerio, que funciona como “software hotelero básico para recopilar fácilmente la información de tus huéspedes y enviarla directamente al sistema de la Tarjeta de Registro de Alojamiento – TRA” (EasyTRA, 2025). Aunque la plataforma no es un sistema PMS, aprovecha la API que dispone el gobierno para estos, haciendo que los mismos huéspedes realicen el registro de la información de la reserva desde un enlace que proporciona EasyTRA, sin gestionar huéspedes ni alojamientos. Su servicio se enfoca para establecimientos que no cuenten con sistemas PMS, pero si con conexión constante a internet y además que estén registrados en Airbnb o Booking, ofreciendo diferentes planes de pago de acuerdo con la cantidad de alojamientos con los que cuente el proveedor de servicios;

DigiTRA: Herramienta digital colombiana dirigida para prestadores de servicios de alojamientos que se identifiquen dentro de uno de los tres los grupos denominados por el ministerio, como grupo 2 - prestadores usuarios directos conectados y “anfitriones de renta corta.

Automatiza el check-in, reportes obligatorios y gestión de seguros” (DigiTRA, 2025), con el fin de sincronizar por medio de la API para el grupo 1 – prestadores con PMS, los datos que los mismos huéspedes digitan en un formulario que facilita la propia plataforma. No gestiona alojamientos, ni huéspedes. Ofrece diferentes planes anuales e integración con Airbnb y Booking.

Chekin: Software español que funciona como gestor de reservas, alojamientos y huéspedes, creado para automatizar el registro y reporte de información como la TRA y SIRE ante el gobierno colombiano. Sus servicios están dirigidos para grandes prestadores de servicios de alojamiento como hoteles, villas, apartamentos, campings, que cuenten con sistema PMS, ofreciendo integración con los mismos y almacenamiento en la nube. Su modo de registro de huéspedes se basa en generar enlaces que contienen los formularios para obtener los datos y enviarlos automáticamente a la plataforma del gobierno. (Chekin, 2025).

Marco Conceptual

Ingeniería de Requerimientos

La ingeniería de requerimientos es una etapa fundamental dentro del desarrollo de software ya que “es el proceso de interactuar con los participantes del sistema para descubrir sus requerimientos” (Briano, 2023, p. 81), con el fin de identificar, analizar, documentar y gestionar las necesidades y expectativas de los usuarios respecto al sistema que se va a construir. Esta fase ayuda a tener una visión clara de lo que el usuario final necesita, permitiendo diseñar soluciones que realmente resuelvan sus problemas y cumplan con los objetivos del proyecto. Además, una buena gestión de requerimientos reduce errores y mejora la calidad final del software.

Establecimientos de Alojamientos Turísticos

Se denomina establecimientos de alojamientos turísticos a “los establecimientos de comercio que brindan el servicio de alojamiento turístico con oferta permanente” (MINTIC, 2023, p. 1). Forman parte de esta categoría los establecimientos como hoteles, cruceros con servicios de alojamiento, apartahoteles, hostales, centros turísticos, glampings, refugios, albergues y espacios habilitados para acampar, así como cualquier otro sitio que brinde hospedaje de forma continua.

Tarjeta de Registro de Alojamiento -TRA

Según Tramincit (2025) la tarjeta de registro de alojamiento TRA “es la herramienta donde se registra la información sobre los huéspedes nacionales y extranjeros que tenga el prestador de los servicios de alojamiento turístico, y es prueba del contrato de hospedaje”, de acuerdo con el artículo 22 de la Ley 2068 de 2020. La exigencia de este requisito aplica de manera obligatoria a los diferentes tipos de alojamientos turísticos, incluidas las viviendas turísticas. Esta medida busca, entre otros fines, mejorar la calidad de los datos estadísticos del sector, apoyar el diseño de políticas públicas efectivas y promover condiciones de mayor seguridad entre el turista y el sistema PSAT.

Sistemas de Gestión

Son plataformas de software utilizados para la administración y automatización de los procesos de negocios de las empresas. El objetivo es facilitar el control de actividades, como la administración de inventarios, ventas, reservas, clientes, personal, contabilidad, entre otras, dependiendo del tipo de organización, simplificando “los procesos operativos, productivos y burocráticos de una organización” (Siigo, 2018). Entre las características más importantes de

estos softwares se incluyen la integración con otros sistemas, escalabilidad, usabilidad, el soporte técnico, flexibilidad y personalización, así también como el análisis y generación de reportes.

Marco Teórico

El desarrollo de un sistema web gestor de alojamiento y huéspedes requiere una sólida comprensión de diversos conceptos y tecnologías fundamentales en el ámbito del desarrollo de software y la gestión de bases de datos. A continuación, se describen las bases teóricas que sustentan la propuesta de diseño e implementación de dicho sistema, detallando los lenguajes de programación, herramientas y principios que guiarán el proyecto.

Metodología Híbrida Cascada y Agile Scrum

Denominado ScrumFall por su combinación de Agile: Scrum con cascada (Waterfall), busca como metodología para desarrollo de software, aprovechar las fortalezas de ambos métodos para adaptarse a proyectos que requieren tanto planificación detallada como flexibilidad en la ejecución.

La metodología en cascada es un proceso lineal donde cada paso se hace uno después del otro, sin saltarse ni regresar, dividido en las siguientes fases:

Fase 1- Análisis de requisitos: esta consiste en especificar y clasificar los requisitos para software que proporciona el usuario final.

Fase 2 – Diseño: consiste en crear la estructura del sistema, tanto a nivel general como técnico.

Fase 3 – Implementación: consiste en la escritura del código según los prototipos diseñados.

Fase 4 – Pruebas: consiste en verificar que es software funcione sin errores.

Por su parte, el enfoque Scrum se basa en ciclos cortos entre 1 a 4 semanas, donde se planea y ejecuta tareas establecidas dando como resultado una funcionalidad que se puede utilizar. El proceso scrum se compone de las siguientes fases:

Revisión de planes de versión: también conocido como planificación del sprint, es la etapa donde se define las tareas a realizar.

Sprint: es la etapa de desarrollo iterativo.

Sprint Review: es la etapa donde se entrega el trabajo terminado de cada sprint para revisión y comparación con la tarea asignada.

Cierre: Se revisan y consolidan los resultados obtenidos del todo el proyecto para distribución.

Al combinarse estas dos metodologías se “produce una calidad de producto estable, ya que ambos modelos de proceso comparten diferentes fases del desarrollo” (Yahaya, y Maidin, 2022). Para las fases iniciales del proyecto (análisis y diseño) se recomienda un enfoque más riguroso y ordenado, y para la implementación (codificación) y pruebas se recomienda la forma iterativa y colaborativa de scrum, respondiendo mejor a los cambios durante el proceso. Por lo tanto, muchas organizaciones han optado por aplicar esta metodología en sus desarrollos de software con gran éxito en proyectos grandes y escalables, realizando una rápida creación de documentación y entrega producto aprovechando la facilidad para cambiar requisitos.

Diseño del Sistema

El diseño del sistema “se centra en crear una solución que cumpla con los requerimientos identificados” (Briano, 2023, p. 97) dividiendo el problema a resolver en fragmentos más pequeños para que este sea más comprensible y fácil de desarrollar. Como resultado se obtiene una visión más precisa para definir el tipo de arquitectura, los tipos de diagramas de actividades

para diseñar el flujo de acciones de cada componente, los diagramas de entidad-relación (ERD) y las maquetas de la interfaz de usuario.

Lenguajes de Programación y Herramientas de Desarrollo

PHP: Según la documentación oficial “PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular, especialmente adecuado para el desarrollo web, que puede incrustarse en HTML” (PHP.net, 2021). Es un lenguaje de programación del lado del servidor muy utilizado para aplicaciones web dinámicas e interactivas. La elección de PHP para este proyecto se justifica por su facilidad de integración con bases de datos relacionales como MySQL, su alta compatibilidad con servidores web como Apache, y su capacidad para manejar formularios, sesiones y cookies, elementos esenciales en la gestión de usuarios y datos en sistemas web.

JavaScript: De acuerdo con Developer.mozilla (2024) “JavaScript es un lenguaje de programación del lado del cliente que permite la creación de interfaces de usuario interactivas y la manipulación dinámica del Document Object Model (DOM)”. En este proyecto, JavaScript se utilizará para mejorar la experiencia de usuario (UX) mediante la validación de formularios, la creación de componentes dinámicos, y la comunicación asíncrona con el servidor a través de AJAX (Asynchronous JavaScript and XML).

HTML5 y CSS3: HTML5 (Hypertext Markup Language) es la versión más reciente del estándar HTML y “proporciona la estructura semántica del contenido web, mejorando la accesibilidad y el rendimiento en diferentes dispositivos” (Saavedra, 2023). CSS (Cascading Style Sheets) se utiliza para la presentación visual, lo que "permite crear interfaces estéticamente agradables y adaptativas para pantallas de diferentes tamaños" (Pérez, 2019). En este proyecto,

HTML5 y CSS se combinarán para estructurar y estilizar la interfaz del sistema, asegurando una experiencia de usuario coherente y accesible en distintos dispositivos.

Bootstrap 5: Es un framework frontend de código abierto que permite crear interfaces web responsivas y optimizadas para todo tipo de dispositivos. En su quinta versión agrega utilidades que simplifican el diseño adaptativo. Según W3Schools (2024), “Bootstrap incluye plantillas de diseño basadas en HTML y CSS para tipografía, formularios, botones, tablas, navegación, modales, carruseles de imágenes y muchos otros, así como complementos de JavaScript opcionales”. En este proyecto, Bootstrap 5 se implementará para garantizar la consistencia visual y adaptabilidad de la interfaz de usuario en múltiples resoluciones y dispositivos, mejorando la experiencia de usuario final.

Bases de Datos Relacionales

MySQL: es un sistema de gestión de bases de datos relacional (RDBMS) que utiliza el lenguaje SQL (Structured Query Language) para la gestión de datos. “Es conocido por su eficiencia, seguridad y escalabilidad, siendo una elección popular para aplicaciones web que requieren una gestión robusta de datos” (Oracle, 2024). En este proyecto, MySQL será el motor de base de datos encargado de almacenar la información relacionada con alojamientos, huéspedes y reportes TRA, permitiendo consultas rápidas y precisas.

Servidor Local y Entorno de Desarrollo

XAMPP: es una distribución de Apache que incluye MySQL, PHP, y Perl, proporcionando un entorno de desarrollo local completo para la creación y prueba de aplicaciones web. XAMPP se selecciona para este proyecto debido a su “facilidad de configuración y su capacidad para simular un entorno de producción, permitiendo a los

desarrolladores probar el sistema en un servidor local antes de su despliegue en un entorno de producción real” (Apachefriends, 2024).

Impacto y Pertinencia

La combinación de estas tecnologías permite la creación de un sistema web robusto, eficiente y escalable que responde a las necesidades específicas del sector turístico en Murillo, Tolima. El enfoque en el uso de tecnologías ampliamente adoptadas y probadas, como PHP, JavaScript, MySQL y Bootstrap 5, asegura que el sistema no solo cumpla con los requisitos actuales, sino que también esté preparado para adaptarse a futuras demandas y expansiones.

Este proyecto es relevante en términos técnicos y tiene un gran impacto en la profesionalización y modernización del sector turístico local, contribuyendo a un desarrollo sostenible y a la mejora continua de los servicios ofrecidos.

Bases Legales

El marco legal que respalda este proyecto incluye:

Ley 2068 de 2020: modifica la Tarjeta de Registro Hotelero por la Tarjeta de Registro de Alojamiento, estableciendo la obligación de llevar un registro de los huéspedes.

Resolución 700 de 2021: implementa la TRA y define los grupos de prestadores de servicios de alojamiento turístico, destacando la necesidad de un sistema que permita a aquellos sin conexión a internet e infraestructura tecnológica registrar la información de sus huéspedes.

Resolución 409 de 2022: modifica parcialmente la Resolución 700, ampliando el plazo para la implementación de la TRA y reduciendo los campos de información requeridos, lo que facilita el proceso de registro para los prestadores en áreas desconectadas.

Artículo 71 de la Ley 300 de 1996, modificado por el artículo 28 de la Ley 2068 de 2020: establece las infracciones que cometerá el prestador de servicios turísticos por no reportar la información de los huéspedes.

Ley 300 de 1996, Artículo 72, Modificado por el artículo 29 de la Ley 2068 de 2022: establece las sanciones que deberá pagar el prestador de servicios turísticos si comete alguna de las infracciones por no reportar la TRA.

Metodología

Este proyecto aplicado enmarca diferentes líneas de investigación: de acuerdo con el propósito es una investigación aplicada porque busca resolver un problema concreto y práctico desde su uso como el desarrollo de nuevos productos y tecnologías; en cuanto a los medios usados para obtener datos se denomina investigación documental y de campo, ya que se apoya fuentes bibliográficas, documentos, encuestas y observaciones; considerando el nivel de conocimientos que se adquieren se clasifica como descriptiva y exploratoria dado que recolecta, analiza, evalúa y profundiza en áreas de la programación.

La construcción del sistema web gestor de alojamiento y huéspedes empleado el enfoque metodológico híbrido, incluye las siguientes fases al proyecto:

Fase 1: Identificación de los Requerimientos

La identificación de los requerimientos funcionales y no funcionales del sistema, se implementará el proceso de Ingeniería de Requisitos, utilizando la metodología de recolección de datos denominada encuesta como herramienta clave. Esta será diseñada específicamente para captar las necesidades de los prestadores de servicios turísticos en relación con la gestión de huéspedes y la generación del reporte TRA.

Los datos obtenidos se analizarán para identificar patrones y necesidades comunes entre los usuarios. Para realizar este proceso eficazmente, se aplicará el análisis de frecuencia para cuantificar la repetición de respuestas en preguntas cerradas o de opción múltiple mediante tablas y gráficos de barras dentro de las encuestas aplicadas. Esta técnica es útil para identificar los requerimientos funcionales más solicitados por los usuarios estableciendo una priorización efectiva que se debe ofrecer y los requerimientos no funcionales, que incluyen aspectos como la seguridad, rendimiento y usabilidad.

Para el proceso de validación de los requisitos se implementarán escenarios y casos de uso simulando situaciones reales de uso del sistema, haciendo posible verificar si los requerimientos funcionales y no funcionales responden correctamente a las expectativas. Esta etapa es crítica, ya que cualquier error en los requerimientos puede derivar en retrasos durante el desarrollo o incluso después de que el sistema entre en funcionamiento, debido a la necesidad de modificar tanto el diseño como la implementación del sistema (Briano, 2023).

Fase 2: Diseño

Se implementará la técnica del prototipado de interfaz de usuario utilizando tecnologías como HTML5, CSS3 y Bootstrap 5. Estos prototipos permiten visualizar y refinar la interfaz del sistema web desde etapas tempranas, asegurando que se ofrezca "una experiencia de usuario óptima en dispositivos móviles y de escritorio" (Nielsen, 2020). para que el producto final cumpla con las expectativas y necesidades del usuario.

Para representar la lógica y la estructura de datos que se implementará en MySQL, se elaborarán diagramas de actividades UML y diagramas entidad-relación (ERD). Estos diagramas son esenciales para visualizar las interacciones y relaciones entre los diferentes elementos del sistema, facilitando la comprensión del proceso de desarrollo.

Además, el diseño se fundamentará en el patrón arquitectónico Modelo-Vista-Controlador (MVC), que permite la separación de responsabilidades y también la escalabilidad del software. Esta elección permite una gestión eficiente de las interacciones entre los componentes del sistema, garantizando que cada uno cumpla con su función específica sin interferencias.

Fase 3: Implementación

Considerando la flexibilidad del marco scrum y el tiempo (5 semanas) establecido en el cronograma, cada sprint para el desarrollo del sistema web tendrá una duración de una semana (5 días hábiles, 5 horas diarias), para un total de 5 sprint, produciendo una versión funcional del sistema, que incluirá módulos específicos como la gestión de reservas, registros de tipos de alojamientos, habitaciones, tarifas y la generación del reporte TRA, utilizando lenguaje de programación PHP para el backend y JAVASCRIPT para el frontend. Además, con el fin de proteger los datos, se aplican criterios de seguridad como encriptado de contraseñas y consultas preparadas SQL. Este enfoque iterativo permite al desarrollador recibir retroalimentación temprana y realizar ajustes necesarios antes de avanzar a nuevas funcionalidades (Schwaber y Sutherland, 2020, p.3).

Fase 4: Pruebas

La fase inicial de pruebas contempla la ejecución de pruebas unitarias con el fin de verificar el funcionamiento de componentes individuales del programa, asegurando que operen correctamente, seguidamente se ejecutan las de pruebas de integración, para evaluar la comunicación entre estos módulos garantizando que el flujo de datos entre las distintas capas del sistema (modelo, vista, y controlador) se realice de manera correcta y sin interrupciones (Briano, 2023, pp. 140 - 141).

Finalmente, se hará una prueba de piloto con usuarios finales, para validar que el sistema cumpla con las expectativas de usabilidad, funcionamiento y requerimientos específicos, de los prestadores de servicios turísticos. Este tipo de pruebas es crucial luego de la fase de implementación, ya que permiten obtener retroalimentación directa de los usuarios que interactuarán diariamente con el sistema (Briano, 2023, p. 142).

Fase 1: Identificación de Requerimientos y Necesidades

Las encuestas como técnica de investigación, se utiliza comúnmente para recolectar datos e información para conocer comportamientos y tomar decisiones respecto al objetivo de estudio. Los instrumentos para realizar dicha técnica son: cuestionarios, entrevistas, observaciones, etc. Así pues, para llevar a cabo esta técnica, la figura 1 ilustra que, se requiere seguir un proceso lineal, el cual ha sido adaptado en área de interés por diferentes organizaciones y equipos de investigación a lo largo de la historia, convirtiéndose en una herramienta estandarizada para la humanidad. La manera de proceder para la realización de una encuesta es la siguiente:

Figura 1

Pasos para Diseñar el Instrumento



Nota. Autoría Propia

Cabe mencionar que existen encuestas con diferentes enfoques: descriptivo, exploratorio, transversal, longitudinal, por mencionar algunos. Según López (1998), “las encuestas exploratorias permiten identificar las características generales o dimensiones del problema, así como establecer hipótesis y alternativas de trabajo” (p. 6). En este sentido, para efectos del presente proyecto de investigación se busca comprender el contexto actual de los prestadores de

servicios turísticos, y obtener datos clave que permitan fundamentar el desarrollo de la solución tecnológica adecuada.

Etapas de Construcción del Instrumento: Encuesta

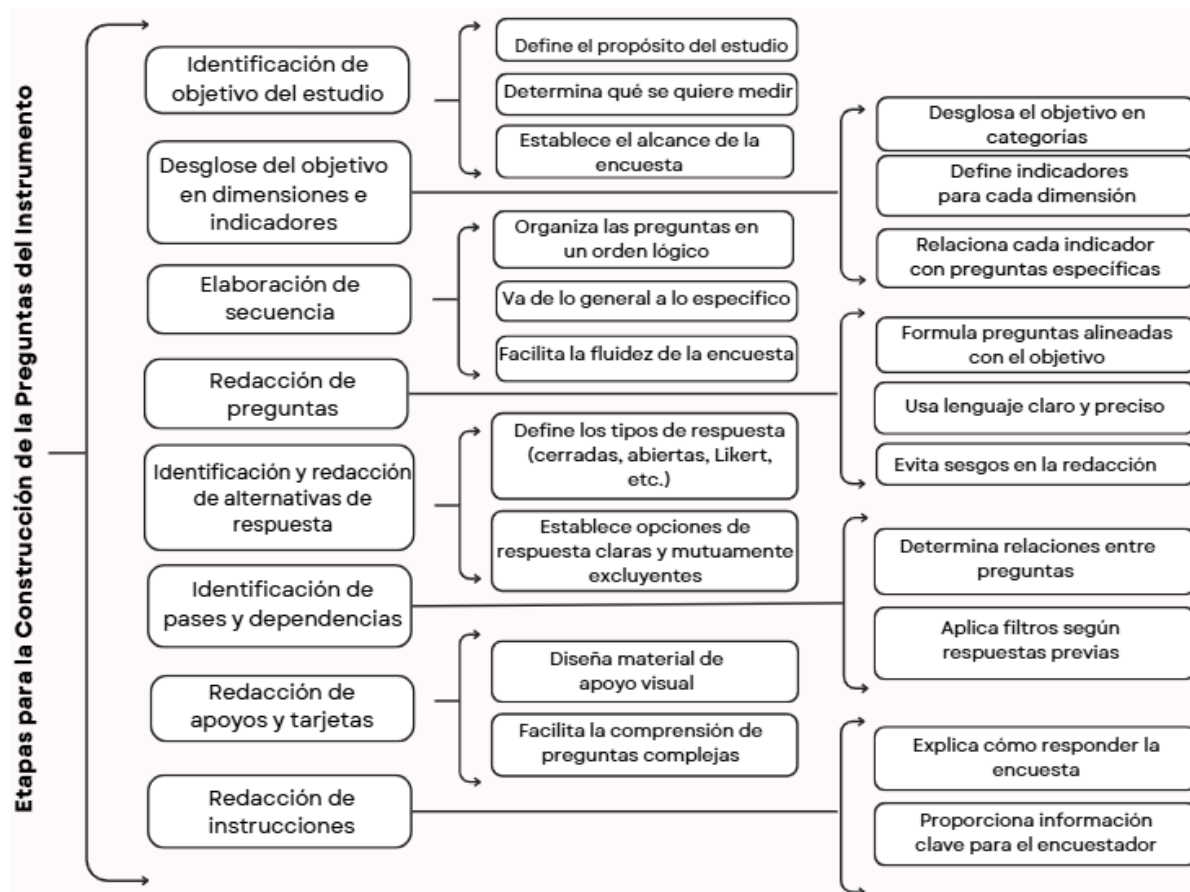
Mencionados antes algunos de los diferentes tipos de instrumentos, se ha seleccionado el cuestionario para el levantamiento de características y requerimientos del sistema web gestor de alojamientos en Murillo Tolima, dado que, permite llegar a más participantes, no es costoso, la recolección de información es organizada, facilitando el análisis porque los datos serán cuantificables. Las preguntas y opciones de respuesta estarán formuladas puntualmente para identificar las limitaciones del sistema web gestor.

Ahora para centrarnos en la construcción del instrumento, se investigó diferentes fuentes bibliográficas sobre metodologías que permitieran la construcción clara y eficiente del cuestionario para la encuesta, entre ellas las propuestas por autores Esther Chiner con su publicación “tema 8 investigación descriptiva mediante encuestas” en el año 2011, Vidal Diaz de Rada y su publicación “Tipos de encuesta y diseños de investigación” en el año 2002, Tabaré Fernández y su artículo “Encuesta noción general, tipos de encuestas y sugerencias para mejorar su calidad” en el año 2014 y Heriberto López Romo “la metodología de encuesta” en el año 1998. Para fines del presente proyecto, se decide aplicar la metodología del último autor, pues esta presenta un enfoque detallado y sistemático para la construcción de instrumentos de recolección de datos, además que comparte los 4 primeros pasos de la encuesta.

Esta metodología se distingue por validar la integridad y consistencia de los datos reunidos por su estructura paso a paso ilustrada en la figura 2.

Figura 2

Etapas para la Construcción de Preguntas del Instrumento



Nota. Autoría propia

Identificación del Objetivo del Estudio

El primer paso en la construcción del instrumento recolector de datos es la identificación y revisión del objetivo del estudio. Este objetivo establece el enfoque, la amplitud y la forma en que se estructurará la encuesta, garantizando que las preguntas formuladas sean pertinentes y alineadas con la investigación. En este sentido, el objetivo a evaluar es el siguiente:

Identificar las características y requerimientos de los prestadores de servicios de alojamiento turístico conforme al contexto local y a la normatividad vigente, como insumos para el diseño y desarrollo del sistema web de alojamiento.

Desglose del Objetivo en Dimensiones e Indicadores

Tras definir el objetivo del estudio, es necesario desglosarlo en dimensiones e indicadores que permitan una evaluación más detallada y estructurada. Este proceso facilita la identificación de los aspectos clave a medir, asegurando que la encuesta recoja información precisa y alineada con el propósito de la investigación. Al segmentar el objetivo en sus componentes esenciales, se garantiza un análisis más claro y enfocado en las variables relevantes para la elaboración del sistema web gestor de alojamientos. Acto continuo, se presenta la tabla 1 que organiza las dimensiones e indicadores derivados del objetivo del estudio.

Tabla 1

Dimensiones e Indicadores

Dimensiones	Indicadores		
Capacidad operativa y servicios ofertados	Capacidad del establecimiento	Tipo de servicios complementarios	Estructura de tarifas
Infraestructura tecnológica	Equipamiento tecnológico		
Habilidades digitales	capacidad del personal para adaptarse al sistema		
Conocimiento y cumplimiento normativo	Conocimiento sobre la normativa TRA	Métodos actuales para generar la TRA	
Percepción del prestador de servicios de alojamiento	Identificar la aceptación de software	Disposición del establecimiento para adoptar el software	
Requerimientos funcionales	Funcionalidades para el software	Flexibilidad para software gestionar el check-out	

Nota. Autoría propia

Elaboración de Secuencia

Una vez definidas las escalas de medición, es fundamental organizarlas en una secuencia lógica para garantizar la coherencia y fluidez de la encuesta. Un orden estructurado facilita la comprensión del respondente, evita confusión y mejora la calidad de las respuestas.

Para ello, se siguen principios como ir “de lo general a lo específico, de lo simple a lo complejo” (López, 1998, p.37) y agrupar preguntas por temas relacionados. Además, la encuesta debe mantener una estructura clara que incluya una introducción, preguntas de filtro, preguntas centrales alineadas con los objetivos de la investigación y un cierre adecuado.

Este paso es clave para optimizar la experiencia del respondente y asegurar que los datos obtenidos sean consistentes y útiles para el análisis. A partir de estos conceptos se procede a elaborar la encuesta:

Introducción y Solicitud de Colaboración

Estimado prestador de servicios turísticos, el objeto de esta encuesta es identificar características y requisitos para la construcción del proyecto “Diseño e implementación de sistema web gestor de alojamiento con reporte TRA para prestadores de servicios turísticos en Murillo Tolima” de la UNAD Líbano.

En cumplimiento de la Ley 1581 de 2012 y el Decreto 1377 de 2013 que refiere la protección de datos personales en Colombia, le informamos que la información recopilada será tratada de manera confidencial y utilizada exclusivamente con fines académicos dentro del desarrollo del proyecto en mención. De igual manera, esta encuesta no solicitará datos personales ni información sensible. Agradecemos de antemano su tiempo y colaboración.

Selección o Filtro del Respondente

¿Cuál es su cargo o función en el establecimiento turístico?

Preguntas Relacionadas con Objetivos

Capacidad operativa y servicios ofertados:

De acuerdo con el tipo de alojamiento prestado, ¿Cuántas camas tiene la habitación más grande? Si presta varios tipos de alojamiento diligencia para cada uno

¿Cuál es la cantidad máxima de personas que pueden alojarse en una habitación?

¿Qué otros servicios se ofrecen en su establecimiento turístico?

¿Qué tipo de tarifas para alojamiento se ofrecen en el establecimiento?

¿Qué tipo de tarifas para parqueadero se ofrecen en el establecimiento?

¿Cuáles son las horas de check-in de su establecimiento?

¿Cuáles son las horas de check-out de su establecimiento?

Infraestructura Tecnológica:

¿Cuenta su establecimiento con computador y acceso a internet?

¿Qué herramienta se utiliza en el establecimiento turístico para el registro de huéspedes?

Habilidades Digitales:

¿Cómo califica la habilidad de los recepcionistas para manejar una computadora o tableta?

Conocimiento de la normatividad:

¿Conoce la normativa vigente sobre la Tarjeta de Registro de Alojamiento (TRA)?

¿Cómo genera actualmente la Tarjeta de Registro de Alojamiento (TRA)?

Percepción de los Prestadores de Servicios

¿Considera importante disponer de un software para el manejo de alojamientos y la generación de la TRA en su establecimiento?

¿En caso de contar con un software gratuito para el manejo de huéspedes y creación del reporte TRA automático, el establecimiento dispondría de un pc para su instalación y funcionamiento?

Requerimientos funcionales:

¿Qué funcionalidades consideran esenciales en un software web de gestión de alojamientos?

Preguntas Demográficas

Las preguntas demográficas no se incluirán en esta encuesta, Dado que el enfoque está en la exploración de los establecimientos turísticos de Murillo Tolima, en relación con la gestión de alojamientos y la generación de la Tarjeta de Registro de Alojamiento (TRA), y no en el perfil sociodemográfico de los encuestados, la recopilación de datos personales como edad, género o nivel educativo no son relevante para el análisis y desarrollo del sistema web gestor de alojamientos. Esto permite centrar la encuesta en información directamente útil para la investigación y evitar preguntas innecesarias que puedan afectar la disposición de los participantes a responder.

Agradecimiento e Informe sobre Posible Supervisión

La encuesta no requiere informe sobre posible supervisión porque no se está aplicando el paso “preguntas demográficas” para recolectar datos del respondiente. Por lo tanto, la verificación individual de respuestas no es necesaria para el análisis de los datos. Sin embargo, se aplica el paso de agradecimiento, como se plantea a continuación, para reconocer y valorar la participación de los encuestados en este proceso:

“Agradecemos sinceramente su participación en esta encuesta. Sus respuestas serán clave para el desarrollo de un sistema web gestor de alojamientos que se ajuste a las necesidades del sector turístico en Murillo. Muchas gracias por su tiempo”.

Datos de Identificación, como Nombre y Dirección, para Posterior Supervisión

Los datos de identificación del respondiente no son relevantes para complementar los requerimientos para el sistema web gestor. Además, garantizar la confidencialidad y anonimato, puede aumentar la disposición a responder con mayor sinceridad y precisión. Asimismo, al no requerir supervisión posterior, se evita la recopilación innecesaria de información que no aporta valor al análisis de los datos informados ni al cumplimiento de los objetivos de la investigación.

Redacción de Preguntas

Definen las características y restricciones del sistema que no están relacionadas directamente con las funcionalidades específicas, donde se resalten los requerimientos que se de forma clara que se refieren a las propiedades del sistema, a la fiabilidad y rendimiento.

Pregunta: ¿Cuál es su cargo o función en el establecimiento turístico? Está bien formulada y permite una respuesta objetiva sin inducir al encuestado. no sugiere que una opción sea mejor que otra, y la redacción es imparcial y no influye en la respuesta.

Pregunta: De acuerdo con el tipo de alojamiento prestado, ¿Cuántas camas tiene la habitación más grande? Si presta varios tipos de alojamiento diligencia para cada uno. Permite obtener datos concretos sin ambigüedad.

Pregunta: ¿Cuántas personas máximo pueden alojarse en una habitación o espacio de alojamiento? Permite obtener datos concretos sin ambigüedad.

Pregunta: ¿Qué otros servicios se ofrecen en su establecimiento turístico?

Aunque con lenguaje sencillo, presenta una ambigüedad con “otros servicios” dando por sentado que ya se han mencionado algunos servicios antes, pero no queda claro cuáles. Para usar una lista de servicios como opciones de respuesta la pregunta sería:

Pregunta: ¿Qué servicios adicionales además del alojamiento ofrece su establecimiento?

Así se tendrá más comprensión por parte del encuestado.

Pregunta: ¿Qué tipo de tarifas para alojamiento se ofrecen en el establecimiento?

permite la obtención de datos concretos.

Pregunta: ¿Qué tipo de tarifas para parqueadero se ofrecen en el establecimiento?

permite la obtención de datos concretos sin ser invasivos.

Pregunta: ¿Cuáles son las horas de check-in de su establecimiento?

permite la obtención de datos concretos sin ser invasivos.

Pregunta: ¿Cuáles son las horas de check-out de su establecimiento?

permite la obtención de datos concretos sin ser invasivos.

Pregunta: ¿Cuenta su establecimiento con computador y acceso a internet?

Contiene claridad y precisión, sin inducir una respuesta.

Pregunta: ¿Qué herramienta se utiliza en el establecimiento turístico para el registro de huéspedes?

es directa, usando lenguaje claro y comprensible que permite abarcar diferentes opciones de respuesta.

Pregunta: ¿Cómo califica la habilidad de los recepcionistas para manejar un computador o tablet?

es directa, usando lenguaje claro, comprensible, percibiendo la habilidad de los recepcionistas.

Pregunta: ¿Conoce la normatividad vigente sobre el reporte de la Tarjeta de Registro de Alojamiento (TRA) ante el ministerio de comercio, industria y turismo?

Es directa. No contiene ambigüedades. No sugiere si conoce la normativa es positiva o negativa.

Pregunta: ¿Cómo genera actualmente la Tarjeta de Registro de Alojamiento (TRA)?

El término "genera" es ambiguo, ya que puede interpretarse de distintas formas. Además, si el encuestado no conoce la TRA, la pregunta resultaría irrelevante. Por lo tanto, la redacción adecuada sería

“Si conoce la Tarjeta de Registro de Alojamiento ¿qué método utiliza para reportar la TRA?”

"Método" especifica la acción. No favorece ninguna opción de respuesta y se liga la pregunta a la anterior permitiendo que la encuesta sea lineal.

Pregunta: ¿Considera importante disponer de un software para el manejo de alojamientos y la generación de la TRA en su establecimiento?

Es puntual y evalúa la disposición del respondiente para usar el software, alineándose con el objetivo de la encuesta.

Pregunta: ¿En caso de contar con un software gratuito para el manejo de huéspedes y creación del reporte TRA automático, el establecimiento dispondría de un pc para su instalación y funcionamiento?

Es puntual para conocer la disposición del respondiente de contar con un pc para usar el software, alineándose con el objetivo de la encuesta.

Pregunta: ¿Qué funciones considera importantes incluir en un software para el manejo de alojamientos?

Es directa. Permite conocer directamente las preferencias y expectativas de los usuarios sobre el sistema.

Identificación y Redacción de Alternativas de Respuesta

Redactadas las preguntas, es fundamental definir las alternativas de respuesta para garantizar que la información recopilada sea clara, precisa y cuantificable. Las opciones deben:

Ser claras y mutuamente excluyentes para evitar ambigüedades.

Incluir opciones abiertas cuando sea necesario para respuestas no previstas.

Usar escalas de Likert cuando se evalúen percepciones o niveles de importancia.

Acto continuo, se presentan las preguntas con sus respectivas opciones de respuesta estructuradas siguiendo estos principios.

Selección o filtro del encuestado:

Justificación: La encuesta busca identificar una persona adecuada para el suministro de la información con el fin de lograr confiabilidad y credibilidad de los datos reportados.

¿Cuál es su cargo o función dentro del establecimiento?

Opciones de respuesta:

Propietario

Administrador

Recepcionista

Otro: _____

Capacidad operativa y servicios prestados:

Justificación: Esta pregunta permite establecer la capacidad máxima de camas por habitación que debe manejar el sistema en su etapa de diseño

De acuerdo con el tipo de alojamiento prestado, ¿Cuántas camas tiene la habitación más grande? Si presta varios tipos de alojamiento diligencia para cada uno (Selección múltiple)

Opciones de respuesta:

Alojamientos	/	Camas
Hotel		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Hostal		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Cabaña		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Camping		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Glamping		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Centro vacacional		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Refugio		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Alberge		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Vivienda turística		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Apartahotel		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Camping		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30
Otro		1 - 4 5 - 9 10 - 14 15 - 19 20 - 24 25 - 29 Mas de 30

Justificación: Esta pregunta permite identificar el valor máximo de ocupación de las habitaciones, dato importante dentro del diseño del sistema.

¿Cuántas personas máximo pueden alojarse en una habitación o espacio de alojamiento?

(Selección única)

Opciones de respuesta:

1 - 3

4 - 6

7 - 9

10 - 12

13 - 15

16 - 18

19 - 21

22 - 24

25 - 27

28 - 30

Justificación: Esta pregunta permite identificar y si es el caso, agregar módulos adicionales de servicios ofrecidos dentro del diseño del sistema.

¿Qué servicios adicionales ofrece su establecimiento turístico, además del alojamiento?

(selección múltiple)

Restaurante

Parqueadero

Venta de productos

Piscina

Lavandería

Rutas turísticas

Alquiler (vehículos, caballos, etc).

Otro _____

Justificación: Esta pregunta permite identificar las diferentes modalidades de tarifas utilizadas en el establecimiento para incluirlas dentro del diseño del sistema.

¿Qué tipo de tarifas para alojamiento se ofrecen en el establecimiento? (Selección múltiple)

Por horas

Por días

Por noches

Por personas

Por grupos

Según la temporada (alta, baja, festivos, etc.)

Otro: _____

Justificación: Esta pregunta permite identificar las modalidades de tarifas de parqueadero utilizadas en el establecimiento para incluirlas dentro del diseño del sistema.

¿Qué tipo de tarifas para parqueadero se ofrecen en el establecimiento?

Opciones de respuesta

Por horas

por días

Por noches

Por tipo de vehículo

El parqueadero va incluido en la tarifa

No se ofrece servicio de parqueadero

Otro: _____

Justificación: Esta pregunta permite identificar las modalidades de ingreso de los huéspedes al establecimiento, información requerida dentro del diseño del sistema.

¿Cuáles son las horas de check-in de su establecimiento? (Respuesta abierta)

Justificación: Esta pregunta permite identificar las modalidades de ingreso de los huéspedes al establecimiento, información requerida dentro del diseño del sistema.

¿Cuáles son las horas de check-out de su establecimiento? (Respuesta abierta)

Infraestructura tecnológica:

Justificación: Esta pregunta permite identificar la disponibilidad de los requerimientos tecnológicos dentro del establecimiento, para establecer la viabilidad de la implementación del sistema en el área local.

¿Cuenta su establecimiento con computador y acceso a internet? (Selección única)

Opciones de respuesta:

Sí

No

Solo computador

Solo internet

Justificación: Esta pregunta permite identificar la disponibilidad de los requerimientos tecnológicos dentro del establecimiento, para establecer la viabilidad de la implementación del sistema en el área local.

¿Qué herramienta utiliza en el establecimiento para el registro de huéspedes? (Selección múltiple)

Opciones de respuesta:

Papel

Computador

Celular

Tablet

Ninguno

Otro _____

Habilidades digitales:

Justificación: Esta pregunta permite determinar la habilidad digital de los recepcionistas para establecer la viabilidad de la implementación del sistema en el área local.

¿Cómo califica la habilidad de los recepcionistas para manejar un computador o tablet

(Escala en Likert)

Recepcionista 1	Deficiente	Regular	Bueno	Excelente
Recepcionista 2	Deficiente	Regular	Bueno	Excelente
Recepcionista 3	Deficiente	Regular	Bueno	Excelente
Recepcionista 4	Deficiente	Regular	Bueno	Excelente
Recepcionista 5	Deficiente	Regular	Bueno	Excelente

Cumplimiento normativo:

Justificación: Las siguientes dos preguntas permiten establecer si el prestador de servicios de alojamiento conoce la reglamentación legal TRA para los prestadores de servicios turísticos y sus modalidades de reporte.

¿Conoce la normatividad vigente sobre la Tarjeta de Registro de Alojamiento (TRA)?

(Selección única)

Opciones de respuesta:

Sí

No

Si conoce la Tarjeta de Registro de Alojamiento ¿qué método utiliza para reportar la TRA? (Selección única, con opciones específicas)

Opciones de respuesta:

Sistema PMS

Mediante Excel

No genera el reporte TRA

Percepción del prestador de servicios de alojamiento:

Justificación: Esta pregunta permite identificar el interés del prestador de servicios turísticos en el sistema, y de paso su viabilidad como herramienta indispensable.

¿Considera importante disponer de un software para el manejo de alojamientos y la generación de la TRA en su establecimiento?

Si

No

Justificación: Esta pregunta permite identificar la disponibilidad de los dispositivos con que cuenta el establecimiento para el uso del sistema, con el fin de establecer la viabilidad de la implementación.

¿En caso de contar con un software gratuito para el manejo de huéspedes y creación del reporte TRA automático, el establecimiento dispondría de un pc para su instalación y funcionamiento?

Si

No

Tal vez

Requisitos funcionales:

Justificación: Esta pregunta permite identificar funcionalidades adicionales que pueden requerir el sistema en la etapa de diseño

¿Qué funciones considera importantes incluir en un software para el manejo de alojamientos? (Selección múltiple)

Opciones de respuesta

Administración de huéspedes y habitaciones

Contabilidad y facturación

Cálculo de costos del servicio prestado

Estadísticas y reportes diarios y mensuales

Generación del reporte TRA

Identificación de Pases y Dependencias

Según la metodología, “es necesario identificar las dependencias entre preguntas y explicitarlas a través de instrucciones precisas” (López, 1998, p. 37) para garantizar un flujo coherente en la encuesta y evitar que los encuestados respondan preguntas irrelevantes.

Sin embargo, en la presente encuesta, este procedimiento no es requerido, porque todas las preguntas han sido planeadas y redactadas para ser respondidas por todos los participantes sin necesidad de filtros o bifurcaciones.

Redacción de Apoyos y Tarjetas

Como menciona el autor, “casi todos los cuestionarios utilizan apoyos visuales como tarjetas con escalas de respuestas o redacción de conceptos” (López, 1998, p. 37) que faciliten la comprensión y respuesta de los respondientes. Como las preguntas están formuladas de manera clara y directa, sin términos complejos ni escalas extensas que requieran una representación visual adicional no se considera necesario el uso de tarjetas.

Redacción de Instrucciones

El último paso en la construcción del cuestionario es la redacción de instrucciones generales para los respondentes. Siguiendo la metodología, estas indicaciones deben ser claras, detalladas y precisas, garantizando que la aplicación de la encuesta se realice de manera uniforme y sin sesgos en la recopilación de datos.

En este caso, dado que la encuesta ha sido diseñada con preguntas que abarcan los conceptos antes mencionados, no se requiere un conjunto extenso de instrucciones. Sin embargo, es recomendable incluir algunas pautas generales para orientar y asegurar respuestas consistentes:

Instrucciones Generales para la Encuesta

Explicar brevemente a los encuestados el propósito de la encuesta en la siguiente manera:

“El propósito de la encuesta es una exploración general para identificar las características y requerimientos para el desarrollo del proyecto “Diseño e implementación de sistema web gestor de alojamiento con reporte TRA para prestadores de servicios turísticos en Murillo Tolima” de la UNAD Líbano. Esta encuesta no solicitará datos personales ni información sensible”.

Informar el tiempo estimado que le tomara responder a la encuesta:

“El tiempo estimado que le tomara responder es aproximadamente 3 a 5 minutos”.

Indicar a los encuestados las opciones de respuesta que tiene las preguntas:

“Algunas preguntas permiten seleccionar solo una respuesta, mientras que otras permiten seleccionar múltiples respuestas”.

Asegurar a los encuestados que las respuestas serán tratadas en absoluto anonimato y utilizadas únicamente con fines investigativos.

Instrucciones Específicas

Leer cada pregunta con atención antes de responder.

Si la pregunta no aplica a su caso puede dejar el campo vacío.

En preguntas con opción “Otro”, especifique la respuesta en el espacio proporcionado o en su defecto dejar el campo vacío.

Evaluación del Instrumento

Una vez concluidas las etapas de construcción del instrumento, las preguntas resultantes deben validarse por expertos relacionados con la docencia o la investigación, para verificar si cumplen con las condiciones y pertinencia del objetivo investigado. Este paso es importante y no puede omitirse porque “La evaluación previa permite que se hagan las correcciones necesarias antes de aplicar el cuestionario en su forma definitiva” (Bravo et al. 2019).

La evaluación aplicada al instrumento de la figura 3 por parte del experto, demuestra que este cumple a cabalidad con los criterios de la metodología. El experto considera que las preguntas son claras, relevantes y no presentan ambigüedades; además señala que presenta el flujo correcto, facilitando la comprensión de los encuestados, por lo tanto, no siguiere eliminar o reorganizar preguntas. Así mismo no se han sugerido observaciones adicionales de mejora, lo que respalda la validación del instrumento para su aplicación en el campo.

Figura 3

Evaluación por Experto

ENCUESTA DE EVALUACIÓN DEL INSTRUMENTO DE CUESTIONARIO	
Estimado/a participante,	
Agradecemos su colaboración en la evaluación del instrumento de entrevista utilizado en nuestra investigación. Sus respuestas nos ayudarán a mejorar la calidad y efectividad del instrumento. La información proporcionada será utilizada exclusivamente con fines de validación y mejora de este.	
DATOS GENERALES	
1.	Nombre (opcional): <u>Omar Leonardo Leyton Mendieta</u>
2.	Cargo o rol en la investigación: <u>Asesor</u>
3.	Experiencia en investigación (años): <u>5</u>
SECCIÓN 1: CLARIDAD Y COMPRENSIÓN	
4.	¿Las preguntas del instrumento son claras y comprensibles?
	<input checked="" type="checkbox"/> Totalmente de acuerdo
	<input type="checkbox"/> De acuerdo
	<input type="checkbox"/> Neutral
	<input type="checkbox"/> En desacuerdo
	<input type="checkbox"/> Totalmente en desacuerdo
5.	¿Considera que las preguntas pueden ser interpretadas de manera ambigua?
	<input type="checkbox"/> Sí
	<input checked="" type="checkbox"/> No
	En caso afirmativo, indique cuáles y cómo podrían mejorarse:
<hr/>	
SECCIÓN 2: RELEVANCIA Y ADECUACIÓN	
6.	¿Las preguntas del instrumento son relevantes para los objetivos de la investigación?
	<input checked="" type="checkbox"/> Totalmente de acuerdo
	<input type="checkbox"/> De acuerdo
	<input type="checkbox"/> Neutral
	<input type="checkbox"/> En desacuerdo
	<input type="checkbox"/> Totalmente en desacuerdo
7.	¿Cree que hay preguntas innecesarias o que podrían eliminarse?
	<input type="checkbox"/> Sí
	<input checked="" type="checkbox"/> No
	Si su respuesta es afirmativa, indique cuáles:
<hr/>	
SECCIÓN 3: FORMATO Y ESTRUCTURA	
8.	¿La estructura y secuencia de las preguntas facilitan el flujo de la entrevista?
	<input checked="" type="checkbox"/> Totalmente de acuerdo
	<input type="checkbox"/> De acuerdo
	<input type="checkbox"/> Neutral
	<input type="checkbox"/> En desacuerdo
	<input type="checkbox"/> Totalmente en desacuerdo
9.	¿Cree que se debería reorganizar alguna sección?
	<input type="checkbox"/> Sí
	<input checked="" type="checkbox"/> No
	Si su respuesta es afirmativa, sugiera mejoras:
<hr/>	
SECCIÓN 4: PERTINENCIA Y EXHAUSTIVIDAD	
10.	¿Considera que las preguntas cubren de manera suficiente todos los aspectos relevantes del tema?
	<input checked="" type="checkbox"/> Sí
	<input type="checkbox"/> No
	Si su respuesta es negativa, indique qué aspectos podrían incluirse:
<hr/>	
SECCIÓN 5: COMENTARIOS Y SUGERENCIAS	
11.	¿Tiene alguna otra sugerencia o comentario para mejorar el instrumento de entrevista?
	<u>ninguna</u>
Agradecemos su tiempo y colaboración en esta evaluación.	

Nota. Autoría propia

Diseño de la Muestra

Desde una perspectiva metodológica el diseño muestral no es viable cuando se aplica a toda población. Como señala (Lopez, 1998) “El valor del muestreo radica en la posibilidad de conocer el comportamiento de una población infinita, a partir de un subconjunto” (p. 13). En consecuencia, respaldado por el enfoque exploratorio, para la aplicación del instrumento en este proyecto no es necesario diseñar una muestra dado que la población a encuestar es pequeña y disponible (15 prestadores de servicios turísticos); y lo que se busca con ello identificar parámetros clave, requerimientos y expectativas para el desarrollo del sistema web gestor de alojamientos, mas no hacer inferencias ni investigación cuantitativa.

Análisis y Lectura de Datos

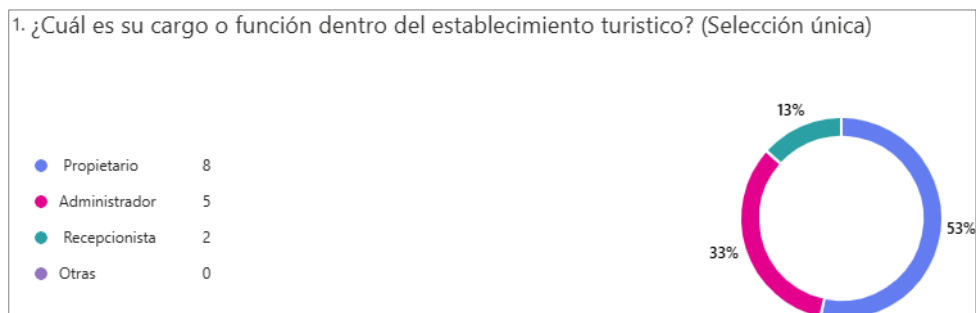
Tras aplicación del instrumento y recopilación de los datos, el paso a seguir es la interpretación de los resultados, esto con el objeto de examinar, interpretar y comprender las características de la población en función al enfoque del proyecto. El análisis cuantitativo es el más apropiado para este caso porque nos ayuda a identificar tendencias o patrones presentes que den como resultados los requerimientos para la elaboración del sistema web gestor de alojamientos. Emplear gráficos como elementos de apoyo permite una mejor visualización e interpretación de los datos, llegando a tomar de mejores decisiones alineadas al objetivo del estudio.

Análisis de las Respuestas

En la figura 4, se observa que un 53% de los participantes ocupan cargo de propietario, mientras que el 33% el de administrador del establecimiento, lo cual indica un alto índice de confiabilidad y credibilidad de los datos reportados.

Figura 4

Resultados - Pregunta 1

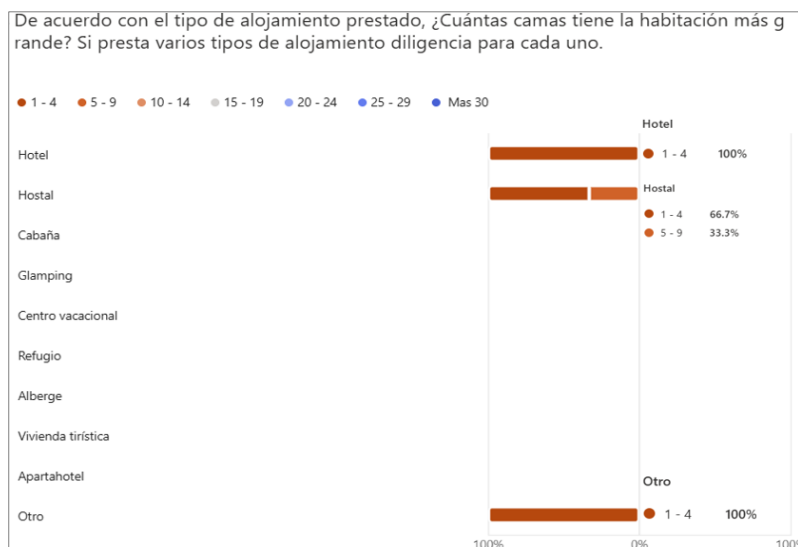


Nota. Autoría propia

En la figura 5 se puede observar que existe una tendencia entre 1 y 4 camas para la habitación más grande, no obstante, la cantidad más alta de camas esta entre 5 y 9 por habitación más grande, indicando que este es el límite máximo que debe gestionar el sistema para evitar desbordamiento de datos, ajustándose a la realidad de los establecimientos.

Figura 5

Resultados - Pregunta 2

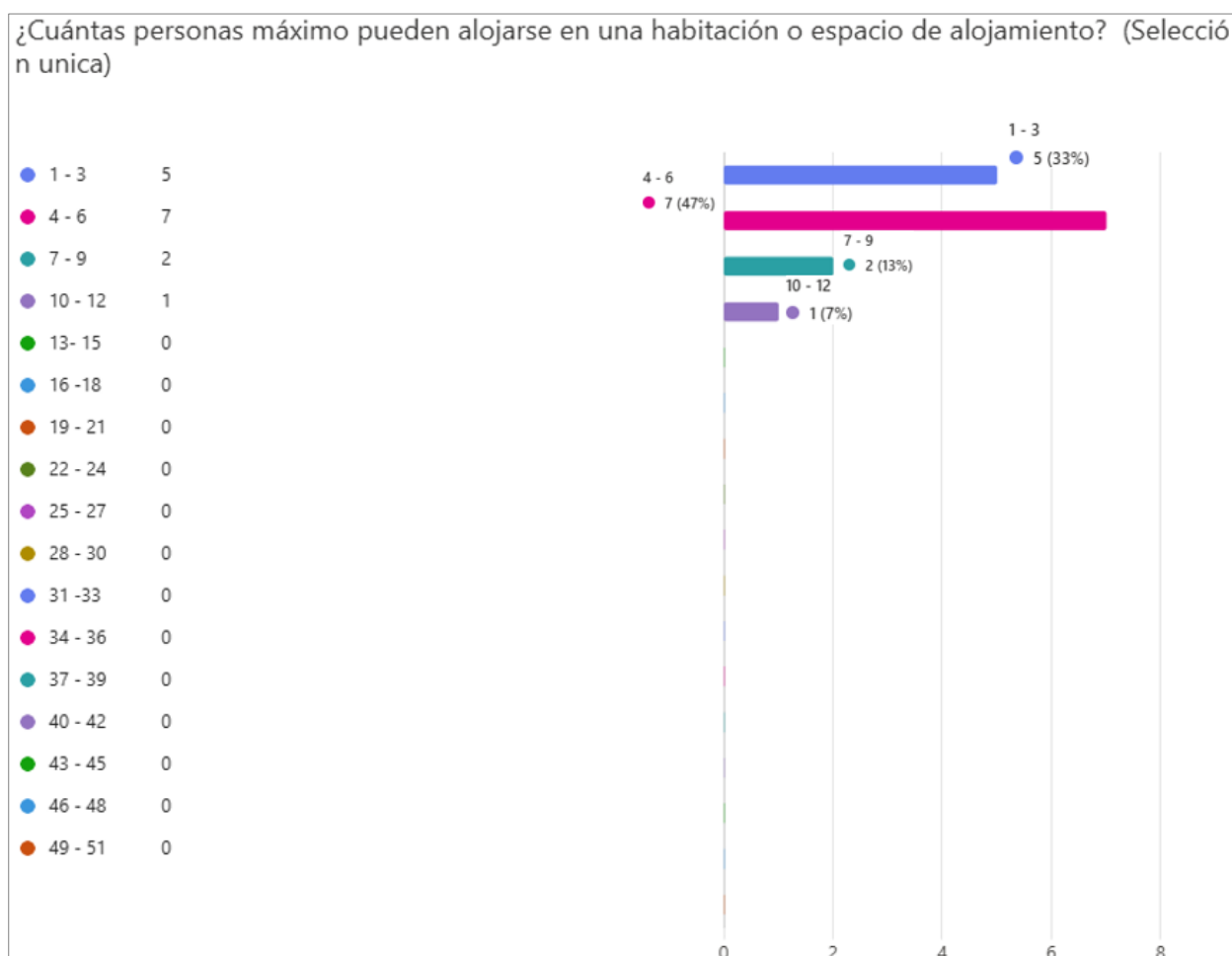


Nota. Autoría propia

En la figura 6 se puede observar que existe una tendencia predominante en capacidad de entre 4 y 6 personas por habitación seguida por la opción 1 y 3 personas, no obstante, la cantidad máxima de ocupación para una habitación esta entre 10 y 12 personas, indicando que este es el límite máximo que debe gestionar el sistema para evitar desbordamiento de datos, ajustándose a la realidad de los establecimientos que ofrecen servicios turísticos de Murillo.

Figura 6

Resultados - Pregunta 3

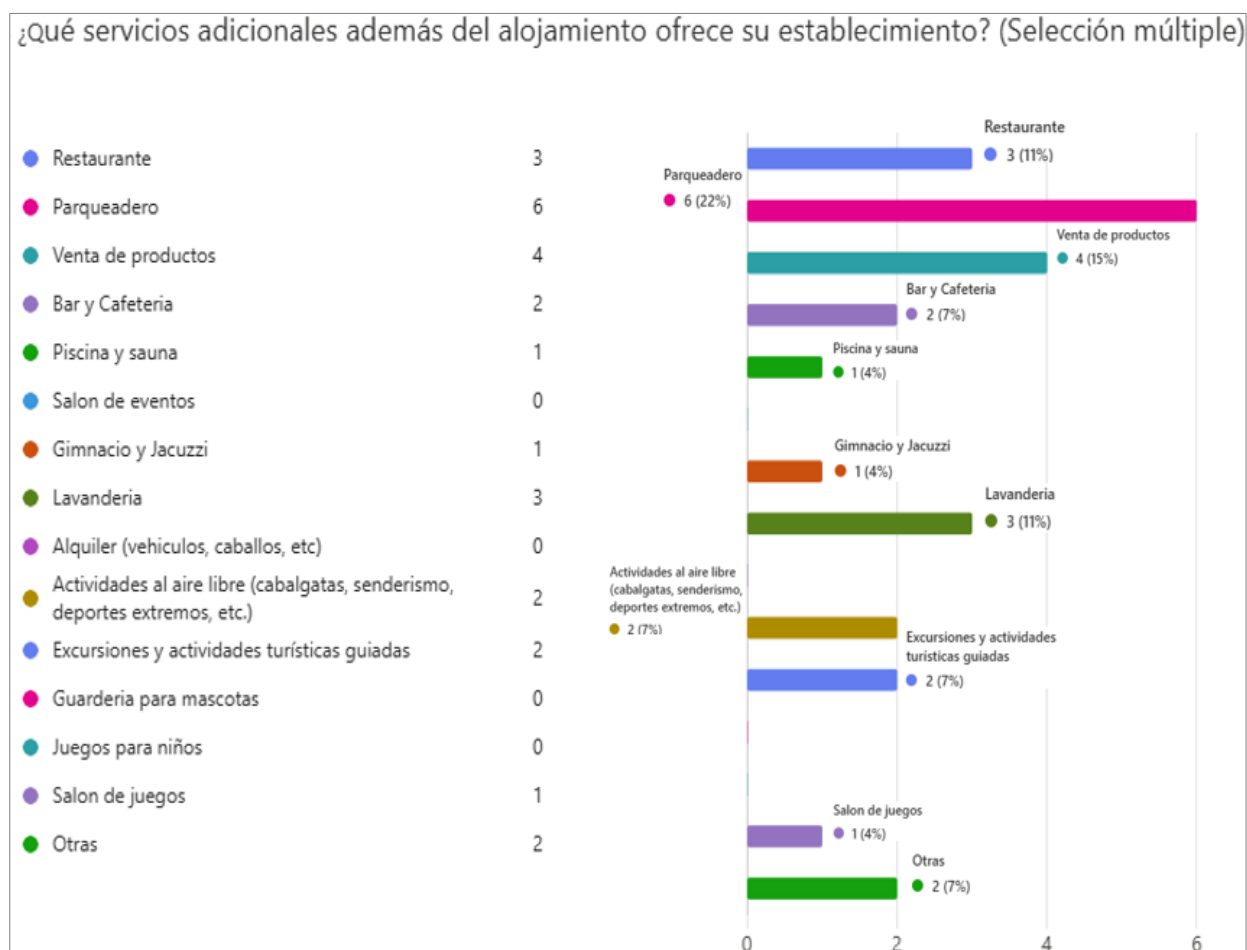


Nota. Autoría propia

En la figura 7 se observa en los establecimientos que señalaron ofrecer servicios adicionales la tendencia más predominante es el parqueadero con un 22%, seguido de venta de productos con un 15% y un 11% para lavandería y restaurante; datos a tener en cuenta para la creación de módulos adicionales en el diseño del software.

Figura 7

Resultados - Pregunta 4

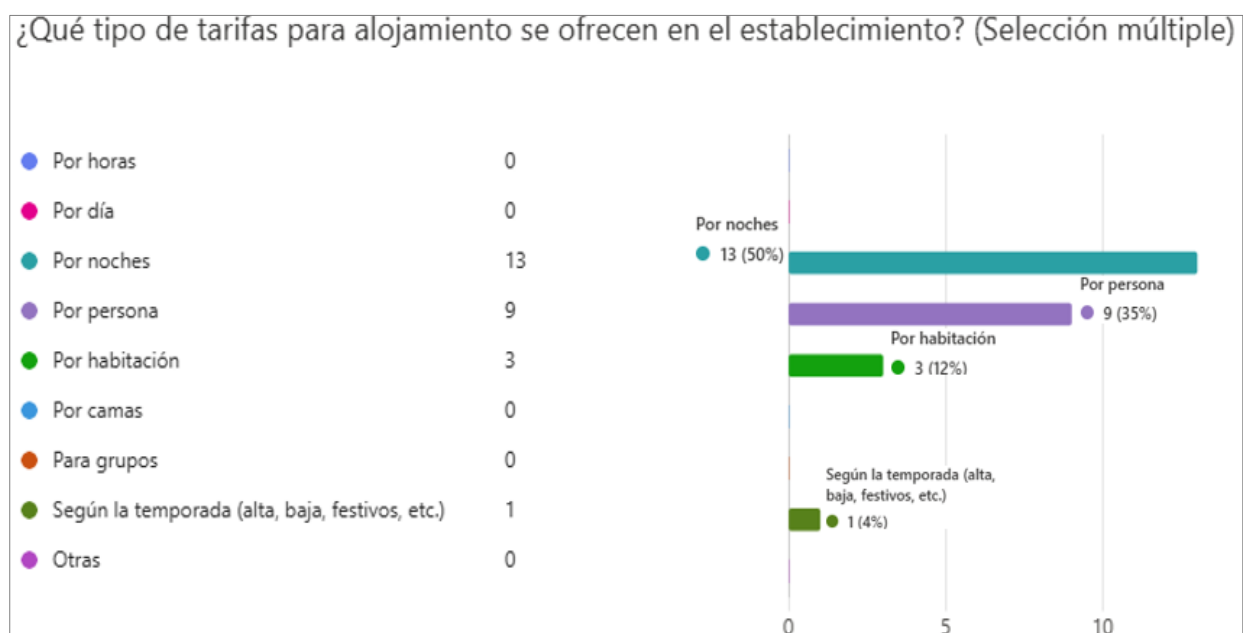


Nota. Autoría propia

En la figura 8, se puede observar que entre los establecimientos existe una tendencia predominante para la tarifa por noches con un 50%, seguida por la opción tarifa por persona 35%, mientras que la tarifa según la temporada apenas se aplica en un 4%, datos a tener en cuenta para el diseño del software, ajustándose a la realidad de los establecimientos.

Figura 8

Resultados - Pregunta 5

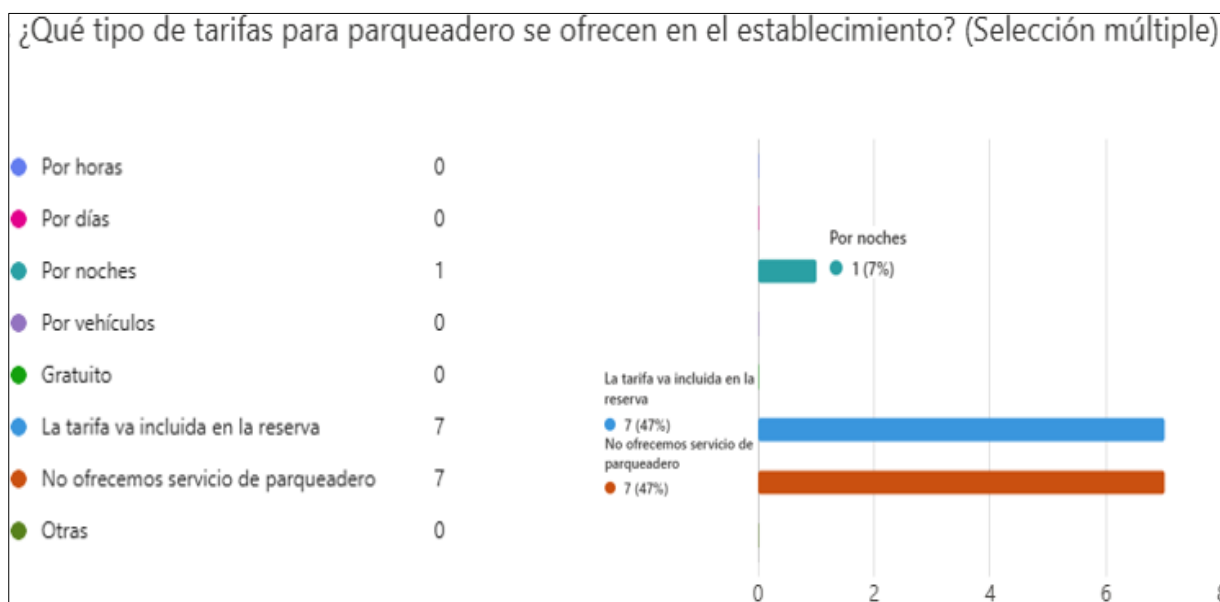


Nota. Autoría propia

En la figura 9, se observa que entre los establecimientos que ofrecen el servicio de parqueadero, un 47% incluye la tarifa dentro del valor de la reserva, mientras que el 7% cobra este servicio adicional por noches. Estos datos son relevantes para el diseño del software, ajustándose a la realidad de los establecimientos.

Figura 9

Resultados - Pregunta 6



Nota. Autoría propia

En la figura 10 se observa que la hora de check-in más común es 3 pm, sin embargo, también se registran otros horarios en menor proporción, indicando que no es viable estandarizar el horario más predominante. Estos datos son relevantes para el diseño del software, ajustándose a la realidad de los establecimientos.

Figura 10

Resultados - Pregunta 7

¿Cuál es la hora de check-in de su establecimiento? (ejm: 6 am. Si no maneja hora de check-in dejar el campo vacío)

6 Respuestas

ID ↑	Nombre	Respuestas
1	anonymous	3:00 pm
2	anonymous	12:00pm
3	anonymous	3
4	anonymous	No
5	anonymous	2pm
6	anonymous	2pm
7	anonymous	2 pm
8	anonymous	3 pm
9	anonymous	3 pm
10	anonymous	3 pm
11	anonymous	3 pm
12	anonymous	3 pm
13	anonymous	12 pm
14	anonymous	3 pm
15	anonymous	3 pm

Nota. Autoría propia

En la figura 11, se observa que la hora de check-in más común es 12 pm, sin embargo, también se registran otros horarios en menor proporción, indicando que no es viable estandarizar el horario más predominante. Estos datos son relevantes para el diseño del software, ajustándose a la realidad de los establecimientos.

Figura 11

Resultados - Pregunta 8

12. ¿Cuál es las hora de check-out de su establecimiento? (ejm: 6 pm: Si no maneja hora de check-in dejar el campo vacio)

6 Respuestas

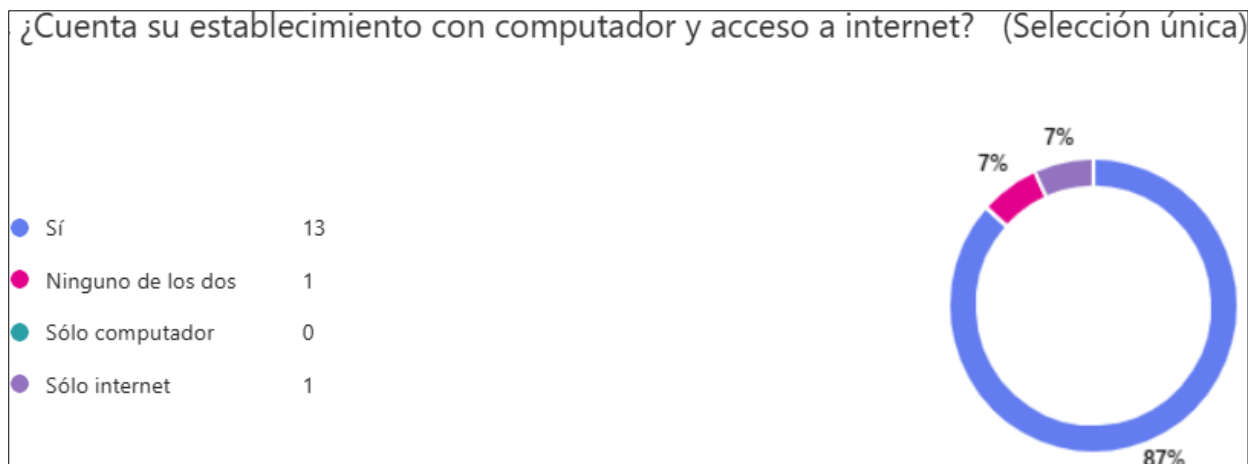
ID ↑	Nombre	Respuestas
1	anonymous	2;00pm
2	anonymous	12:00pm
3	anonymous	12
4	anonymous	No
5	anonymous	11am
6	anonymous	11am
7	anonymous	12 pm
8	anonymous	11 am
9	anonymous	12 pm
10	anonymous	11 am
11	anonymous	12 pm
12	anonymous	12 pm
13	anonymous	3 pm
14	anonymous	12 pm
15	anonymous	1 pm

Nota. Autoría propia

Como se nota en la figura 12, un 87% de los participantes señalan que los establecimientos si cuentan con computador y acceso a internet, lo que sigue una alta capacidad de infraestructura tecnológica y viabilidad para implementar el sistema en el sector.

Figura 12

Resultados - Pregunta 9



Nota. Autoría propia

Como se nota en la figura 13, un 45% de los participantes resaltan que aun utilizan el papel para realizar el registro de huéspedes, lo que indica una resistencia a la adopción de herramientas tecnológicas, requiriéndose acompañamiento para estos establecimientos, no obstante, un 35% que utilizan el computador y en menor medida otras herramientas, que sumadas entre si representan buena base tecnológica y viabilidad para implementar el sistema en el sector.

Figura 13*Resultados - Pregunta 10**Nota. Autoría propia*

En base a la figura 14, se muestra que 11 establecimientos cuentan con tan solo un recepcionista, 2 establecimientos cuentan con 3 recepcionistas y 2 establecimientos con 2 recepcionistas, de los cuales 12 recepcionistas han sido calificados en promedio como “buenos” en su habilidad para manejar el computador, 4 como “excelentes”, 3 como “regulares” y 3 como “deficientes”, evidenciando una alta habilidad de los recepcionistas para manejar el computador, representando la viabilidad suficiente para la implementación del sistema en el área local.

Figura 14

Resultados - Pregunta 11

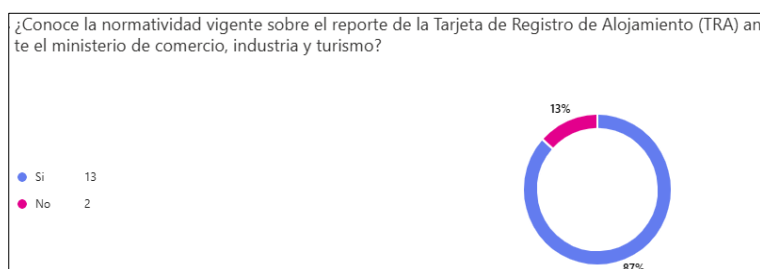


Nota. Autoría propia

Se nota en la figura 15 que, por un total de 15 participantes, más de la mitad (un 87%) señalo que, si conocen la normatividad TRA, dato importante para saber si los prestadores de servicios turísticos comprenden el propósito del sistema.

Figura 15

Resultados - Pregunta 12

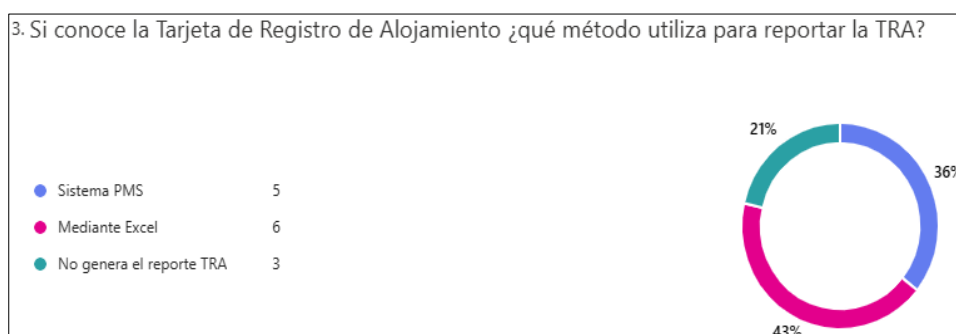


Nota. Autoría propia

Como se nota en la figura 16, un 43% de los participantes señalan que el método para reportar la TRA es mediante Excel, lo que significa que es la herramienta más accesible y común. Mientras tanto el 36% dice que el método que se usa es el sistema PMS, confirmando que gran parte de los prestadores de servicios de alojamiento sí conocen la TRA y el otro 21% que expresa que no genera el reporte TRA por desconocimiento de la normativa.

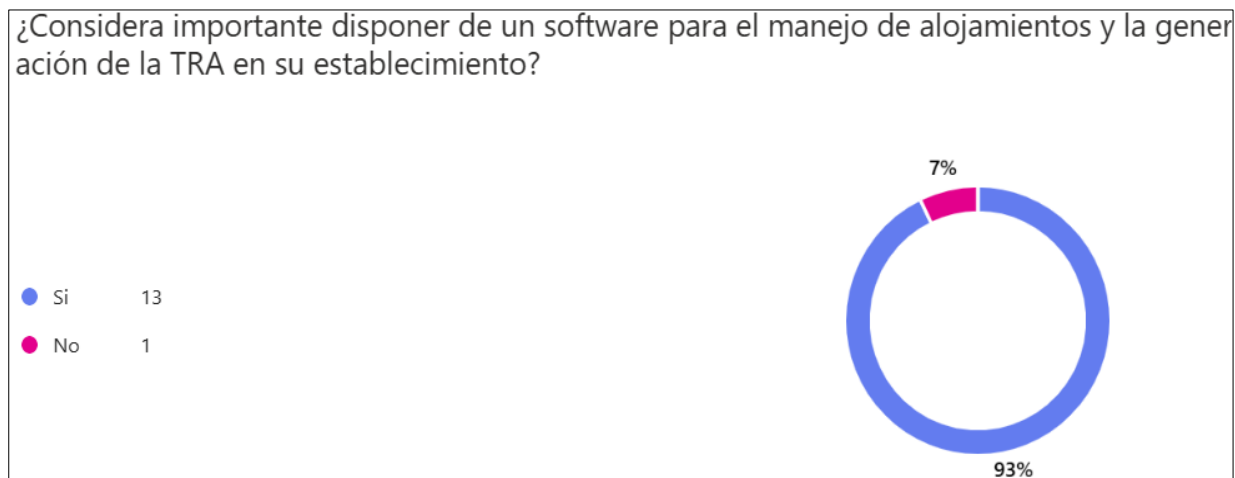
Figura 16

Resultados - Pregunta 13



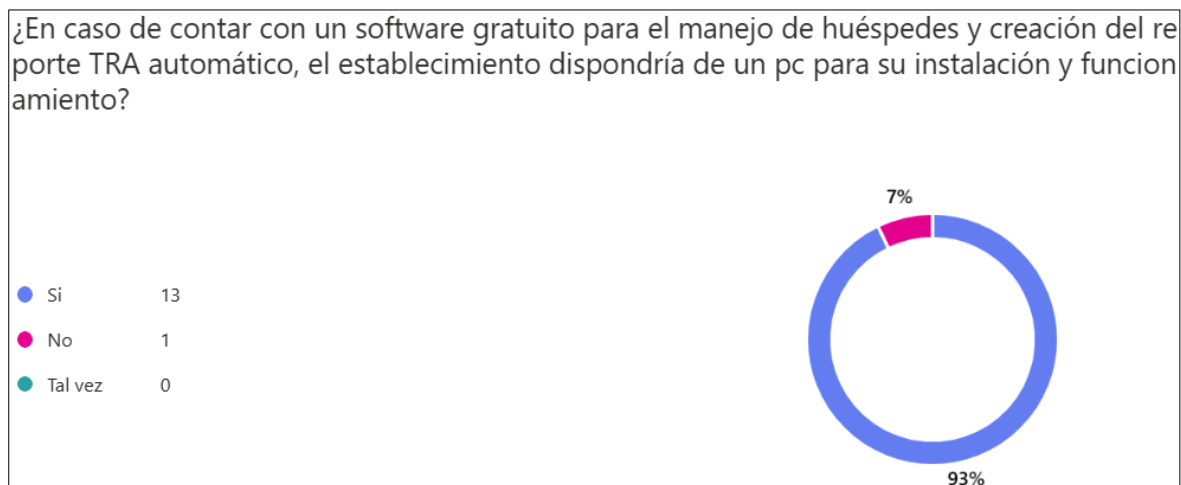
Nota. Autoría propia

En la figura 17 se puede observar que, de un total de 14 participantes, más de la mitad (un 93%) señaló que, si consideran importante contar con software para manejo de alojamientos y reporte TRA, dato importante para saber el interés de los prestadores de servicios turísticos de Murillo en el sistema.

Figura 17*Resultados - Pregunta 14*

Nota. Autoría propia

Como se nota en la figura 18, más de la mitad de los participantes (un 93%) señalan que el establecimiento si dispusiera de un pc para el funcionamiento del sistema, lo que ratifica la aceptación y factibilidad del proyecto desde el punto de vista de infraestructura tecnológica. Mientras tanto el 7% manifiestan lo contrario, lo que representaría una barrera para la adopción del sistema. La categoría de “talvez” no obtuvo respuesta indicando que los participantes tienen claridad de la disponibilidad o no de computadores en sus establecimientos.

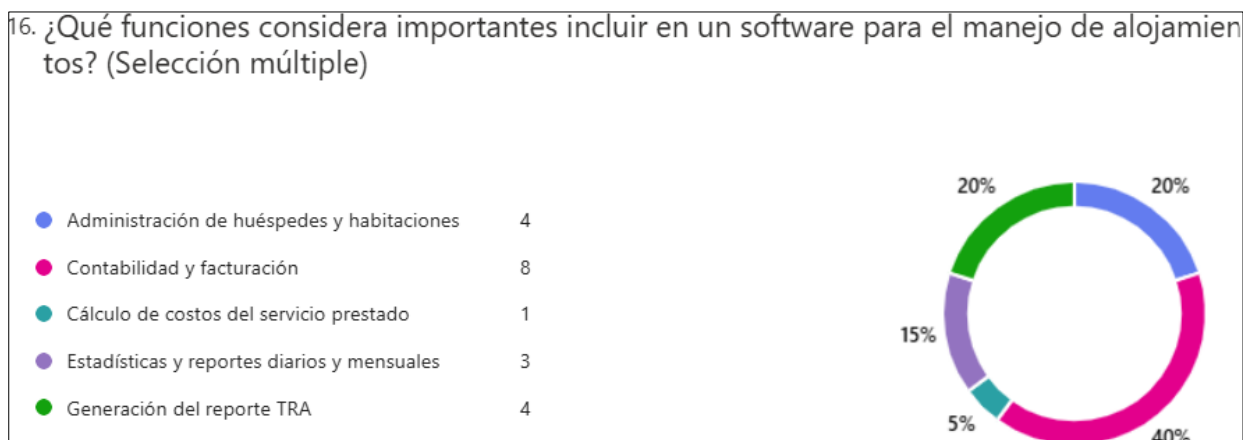
Figura 18*Resultados - Pregunta 15*

Nota. Autoría propia

Como se nota en la figura 19, un 40% de los participantes resaltan que la funcionalidad más importante es la contabilidad y facturación, indicando que los prestadores de servicios de alojamientos buscan un mayor control de su contabilidad y facturación para optimizar la gestión y mejorar la rentabilidad. Mientras tanto la administración de huéspedes y habitaciones, y la generación de la TRA, se registran con una importancia del 20% para cada una, lo que valida la aceptación del sistema en el sector. Adicionalmente las funcionalidades de estadísticas y reportes con el 15% y el cálculo de costos con un 5%, aunque con menos presencia son funcionalidades que se pudieran tener en cuenta para diseño del sistema.

Figura 19

Resultados - Pregunta 16



Nota. Autoría propia

Conclusiones del Análisis de Resultados

Como conclusión del análisis e interpretación de los resultados de la encuesta realizada a los prestadores de servicios turísticos del municipio de Murillo Tolima, se identifica en ellos la necesidad y aceptación de contar con un sistema para la gestión de alojamientos y reporte TRA (93%), además que disponen con la infraestructura tecnológica (87%) y habilidades digitales adecuadas por parte de los recepcionistas (85%), confirmado la factibilidad del proyecto.

En cuanto al diseño funcional y técnico que debe tener el sistema web, se extraen las siguientes características como producto de la encuesta: el mínimo de camas por habitación se encuentra en el rango entre 1 – 4 y el máximo es 5 – 9 y la ocupación máxima de personas alcanza el rango entre 10 - 12 estableciendo los límites que se debe contemplar en el sistema para el diseño de formularios y estructuras de datos sin generar errores de capacidad.

Otros datos importantes que se observaron para agregar dentro de la configuración de reservas del establecimiento dentro sistema web son el manejo de tarifas. Con respecto a las tarifas de alojamiento la modalidad más frecuente se presenta en la noche con un 50% de las

respuestas, seguida por la opción tarifa por persona 35%, mientras que la tarifa según la temporada apenas se aplica en un 4%. Del mismo modo, para la tarifa de parqueadero, un 47% incluye el valor dentro de la reserva, mientras que el 7% cobra este servicio adicional por noches. Por tanto, el sistema debe contemplar múltiples esquemas tarifarios configurables por el usuario.

Las respuestas sobre horarios de check-in y check-out evidencian variabilidad (3 pm y 12 pm como los más comunes), lo cual desaconseja la imposición de horarios fijos. El sistema debe permitir a los establecimientos configurar sus propios horarios de entrada y salida.

Adicionalmente, la mayoría de los encuestados señalaron el interés por servicios adicionales como módulo de contabilidad y facturación, y estadísticas del servicio, considerándolos para versiones futuras del sistema.

Identificación de Requerimientos

El sistema web Gestor de Alojamiento de huéspedes se basará en los elementos que ha de contener la documentación de especificación de requisitos de software como una de las herramientas utilizada en Ingeniería de Requisitos y las definiciones de libro: Compilación de apuntes sobre conceptos fundamentales de la ingeniería de software del Licenciado Cesar Uriel Briano, publicado en el año 2023, quien basa su trabajo en la documentación sobre ingeniería de software del Académico Británico Ian Sommerville, esto con el fin de abarcar los variados tipos de requisitos que surgen en el ámbito del sistema web.

Requerimientos de Descripción General

Incluye todos los elementos que afectan al sistema y a sus requerimientos. Estos factores abarcan el contexto en el que se desarrollará el sistema, como las necesidades de los usuarios y las condiciones técnicas. En otras palabras, son los aspectos generales que ayudan a entender por

qué se requiere el sistema y cómo debe integrarse en su entorno para cumplir con su propósito de manera efectiva.

Usuarios Finales

Lectura y escritura básica para comprender instrucciones, menús y mensajes del sistema.

Capacidad de identificar números y realizar tareas simples como escribir fechas o cantidades.

Saber encender y apagar un dispositivo (computadora o móvil).

Preferencia por interfaces simples e intuitivas.

Manejo básico de un navegador web (abrir, cerrar, navegar entre páginas).

Interacción básica con pantallas táctiles o uso del mouse y teclado.

Habilidad para seguir flujos básicos de navegación, como presionar botones para avanzar o retroceder.

Conocer los datos que se manejan en alojamientos, como nombres de huéspedes, fechas de ingreso y salida, y tarifas.

Conocer la existencia y propósito de la Tarjeta de Registro de Alojamiento (TRA), aunque no sea necesario dominar todos sus detalles.

Restricciones

El sistema debe cumplir con las normativas legales para la generación de la TRA, según lo estipulado por el MINCIT.

Tiempo máximo de desarrollo 4 meses.

Requerimientos de Interfaces

Se detallan todos los elementos que permiten al sistema interactuar de manera eficiente con los usuarios, otros dispositivos o programas, y los canales de comunicación, asegurando su

funcionalidad y compatibilidad en distintos entornos. De esta forma, se garantiza que el sistema sea interoperable y cumpla con las expectativas técnicas y operativas.

Interfaz de Usuario

Las interfaces de usuario deberán ser amigables y fáciles de usar, para que el usuario final tenga una interacción intuitiva y precisa sin realizar el mayor esfuerzo.

La interfaz de usuario debe ser responsiva y diseñada utilizando Bootstrap 5 garantizando compatibilidad con dispositivos de escritorio o móviles de diferentes tamaños.

Interfaz intuitiva con colores suaves, íconos representativos y tipografía legible.

Debe ofrecer un diseño con navegación clara y elementos visuales destacados para acciones críticas (registrar, eliminar o actualizar).

La gestión de reservas se presenta mediante tarjetas visuales que permita identificar rápidamente ocupación y disponibilidad.

Las notificaciones y mensajes del sistema deben ser cortos, claros, personalizados, en formato modal y en tiempo real para el usuario.

Menú principal con accesos rápidos a las funcionalidades clave del sistema web.

Interfaz de Hardware

Procesador de doble núcleo o superior, con una velocidad mínima de 2 GHz.

Mínimo 4 GB de RAM, pero como recomendación 8 GB o más para un óptimo rendimiento del equipo.

20 GB de espacio libre en disco para la instalación de software relacionado (servidores locales, bases de datos, etc.).

Compatibilidad con dispositivos móviles Android (9 o superior) e iOS (13 o superior).

Interfaz de Software

Windows 10 o superior (versión de 64 bits).

Navegadores Google Chrome 109, Microsoft Edge versión 1803 o superiores.

Servidor local XAMPP 8.0.25 o superior, que incluye Apache, MySQL y PHP.

Editor de código IDE Visual Studio Code versión 1.61.2 o superior.

Herramientas de ofimática para manejar informes exportados en formato Excel o PDF.

Interfaces de Comunicaciones

Uso de PDO (PHP Data Objects) para fijar la conexión con la base de datos MySQL, que admite consultas preparadas para prevenir ataques de inyección SQL.

Soporte para sincronizar datos en tiempo real mediante AJAX, garantizando una experiencia fluida para el usuario.

Requerimientos Funcionales

Son la descripción del comportamiento del sistema web de acuerdo con la función que se le agregue. Diversas organizaciones y expertos han desarrollado plantillas y guías sin que alguno se halla atribuido la creación de estas. En la tabla 2 se presentan los diferentes requerimientos funcionales que debe efectuar el sistema para satisfacer las necesidades identificadas.

Tabla 2

Requerimientos Funcionales

ID	Requerimientos	Interesado	Prioridad
R01	El sistema debe solicitar los datos al usuario (nombre y clave) para iniciar sesión.	Administrador y recepcionista	Alta
R01-1	El sistema evaluará los datos del usuario y aprobará o desaprobará los datos.	Administrador y recepcionista	Alta
R-02	El sistema tiene que aceptar la creación de registro de usuarios nuevos	Administrador y recepcionista	Alta

ID	Requerimientos	Interesado	Prioridad
R-03	El sistema tiene que soportar la consulta y actualización de datos de usuarios.	Administrador y recepcionista	Alta
R-05	El sistema debe permitir la eliminación de usuarios.	Administrador	Alta
R-06	El sistema tiene que permitir la asignación de roles a los usuarios del sistema.	Administrador	Alta
R-08	El sistema debe permitir el registro, actualización y eliminación de tipos habitaciones.	Administrador	Alta
R-09	El sistema tiene que soportar el registro, actualización y eliminación de datos de las tarifas para los alojamientos.	Administrador	Alta
R-10	El sistema tiene que aceptar el registro y actualización de información de las reservas.	Administrador y recepcionista	Alta
R-11	El sistema debe permitir la visualización de ocupación actual y disponibilidad de alojamientos.	Administrador y recepcionista	Alta
R-12	El sistema debe gestionar reservas, incluyendo registro, actualización y check-out.	Administrador y recepcionista	Alta
R-13	El sistema debe permitir el cambio de habitación para la reserva.	Administrador y recepcionista	Alta
R-14	El sistema debe mostrar reservas por fecha en el área de trabajo “Gestión de Reservas, con navegación entre días.	Administrador y recepcionista	Alta
R-15	El sistema debe generar, almacenar y exportar el reporte TRA en formato Excel.	Administrador	Alta
R-16	El sistema debe actualizar las interfaces en tiempo real sin recargar la página.	Administrador y recepcionista	Alta
R-17	El sistema debe mostrar alertas para informar errores, éxitos o advertencias al usuario durante su interacción.	Administrador y recepcionista	Alta

Nota. Autoría propia

Requerimientos no Funcionales

Definen las características y restricciones del sistema que no están relacionadas directamente con las funcionalidades específicas, donde se resalten propiedades emergentes del

sistema, respecto a la fiabilidad y rendimiento. En la tabla 3 se presentan los diferentes requerimientos no funcionales que tiene cumplir el sistema para satisfacer las necesidades identificadas.

Tabla 3

Requisitos no Funcionales

ID	Requerimientos	Interesado	Prioridad
R01	El sistema debe generar respuesta ante cualquier solicitud en un tiempo entre 3 y 5 segundos.	Administrador y recepcionista	Alta
R02	El sistema debe ser compatible en sistemas operativos Windows, Linux, IOS, Smartphone y Android.	Administrador y recepcionista	Alta
R-03	Para acceder al sistema será necesario un navegador web.	Administrador y recepcionista	Alta
R-04	El sistema debe ser compatible con el navegador Mozilla Firefox, Google Chrome, Microsoft Edge.	Administrador y recepcionista	Alta
R-05	El sistema debe permitir el acceso controlado basado en roles de usuario.	Administrador y recepcionista	Alta
R-06	El sistema no puede permitir la asignación de rol 'administrador' para más de 1 usuario.	Administrador y recepcionista	Alta
R-07	El sistema tiene que contar con diseño amigable para usuarios con bajo nivel de formación tecnológica.	Administrador y recepcionista	Alta
R-08	El sistema debe hacer uso de encriptación de contraseña.	Administrador y recepcionista	Alta
R-09	El sistema debe estar disponible durante toda la jornada laboral del usuario.	Administrador y recepcionista	Alta
R-10	El sistema debe evitar el ingreso incorrecto de datos mediante mensajes de advertencia.	Administrador y recepcionista	Media
R-11	El sistema debe contar con manual de usuario	Administrador y recepcionista	

Nota. Autoría propia

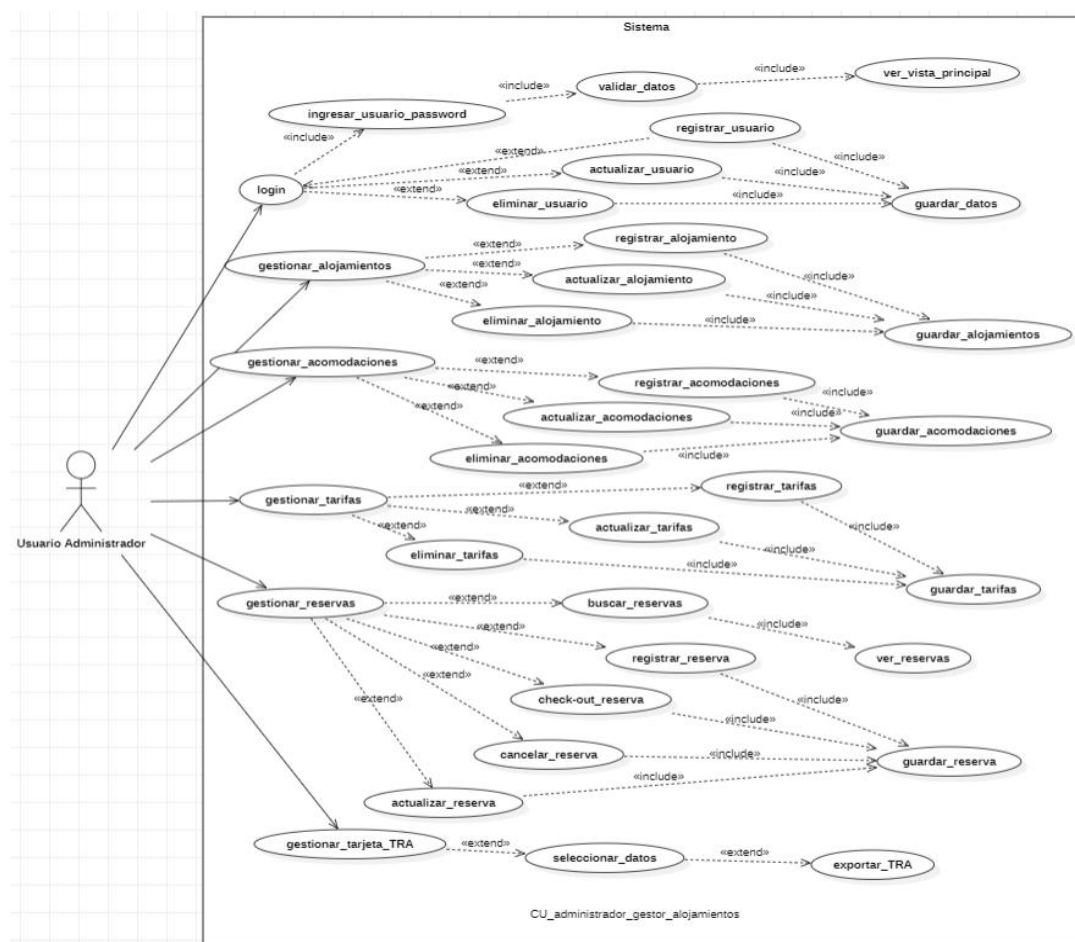
Casos de Uso

Los casos de uso se utilizan para identificar el tipo de interacción y el usuario asociado en el sistema web gráficamente, donde el rectángulo representa el sistema, los óvalos son acciones, la figura humana es el actor y las flechas son las relaciones. La figura 20 ilustra los diferentes escenarios que se manejarán desde el rol “administrador” en el sistema.

CU-01 Usuario Administrador

Figura 20

CU Usuario Administrador



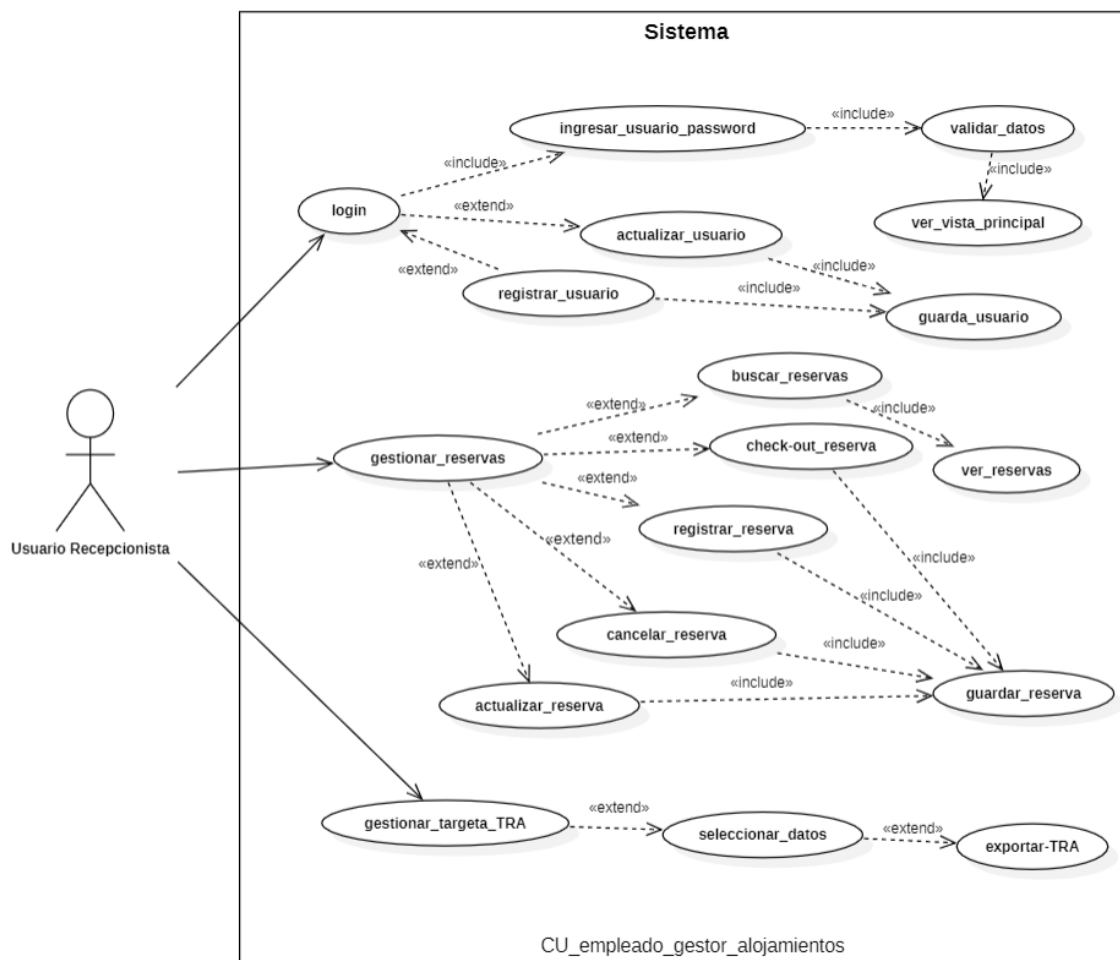
Nota. Autoría propia

La figura 21 ilustra los diferentes escenarios que se manejarán desde el rol “Recepcionista” en el sistema.

CU-02 Usuario Recepcionista

Figura 21

CU Usuario Recepcionista



Nota. Autoría propia

Escenarios de Casos de Uso

Escenario 01 Caso de Uso Actores Administrador y Recepcionista

La tabla 4 describe las condiciones y el paso a paso detallado de los casos de uso.

Tabla 4

Escenario 01 Actor Administrador y Recepcionista

CU-01 Iniciar Sesión															
Versión	01 (02/12/2024)														
Precondición	-El usuario debe tener un registro previo dentro del sistema con identificadores válidos (usuario y clave).														
Descripción	El sistema tendrá que proceder como se puntualiza para el siguiente caso de uso cuando el usuario realice el inicio de sesión.														
Secuencia normal	<table border="0"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario se registra previamente dentro del sistema.</td> </tr> <tr> <td>2</td> <td>El usuario ingresa a la interfaz autenticación a través de la URL designada.</td> </tr> <tr> <td>3</td> <td>Ingresa los datos (usuario y clave) en los campos sugeridos y hace clic al botón 'Entrar'.</td> </tr> <tr> <td>4</td> <td>El sistema recibe las credenciales y las valida con los datos almacenados en la BD.</td> </tr> <tr> <td>5</td> <td>Si las credenciales coinciden, el sistema genera un token de sesión único y lo almacena en el cliente.</td> </tr> <tr> <td>6</td> <td>El usuario es redirigido al área de trabajo "Gestión de reservas", cargando los permisos y recursos disponibles.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario se registra previamente dentro del sistema.	2	El usuario ingresa a la interfaz autenticación a través de la URL designada.	3	Ingresa los datos (usuario y clave) en los campos sugeridos y hace clic al botón 'Entrar'.	4	El sistema recibe las credenciales y las valida con los datos almacenados en la BD.	5	Si las credenciales coinciden, el sistema genera un token de sesión único y lo almacena en el cliente.	6	El usuario es redirigido al área de trabajo "Gestión de reservas", cargando los permisos y recursos disponibles.
Paso	Acción														
1	El usuario se registra previamente dentro del sistema.														
2	El usuario ingresa a la interfaz autenticación a través de la URL designada.														
3	Ingresa los datos (usuario y clave) en los campos sugeridos y hace clic al botón 'Entrar'.														
4	El sistema recibe las credenciales y las valida con los datos almacenados en la BD.														
5	Si las credenciales coinciden, el sistema genera un token de sesión único y lo almacena en el cliente.														
6	El usuario es redirigido al área de trabajo "Gestión de reservas", cargando los permisos y recursos disponibles.														
Postcondición	El usuario tiene una sesión activa con permisos configurados según su rol.														
Excepciones	<table border="0"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Las credenciales ingresadas no coinciden con los registros del sistema. E.1 La interfaz de usuario muestra un mensaje indicando que el nombre de usuario o clave no son válidos.</td> </tr> </tbody> </table>	Paso	Acción	1	Las credenciales ingresadas no coinciden con los registros del sistema. E.1 La interfaz de usuario muestra un mensaje indicando que el nombre de usuario o clave no son válidos.										
Paso	Acción														
1	Las credenciales ingresadas no coinciden con los registros del sistema. E.1 La interfaz de usuario muestra un mensaje indicando que el nombre de usuario o clave no son válidos.														
Comentarios	El sistema deberá restringir la asignación del rol 'administrador' a un único usuario, los demás roles podrán ser asignados a múltiples usuarios según sus funciones definidas en el sistema.														

Nota. Autoría propia

Escenario 02 Caso de Uso Actor Administrador

La tabla 5 describe las condiciones y el paso a paso detallado de los casos de uso.

Tabla 5

Escenario 02 Actor Administrador

CU-01 Gestionar Alojamientos																											
Versión	01 (02/12/2024)																										
Precondición	El usuario debe estar autenticado con un rol “administrador” que permite gestionar alojamientos.																										
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el usuario realice la gestión de servicios de alojamiento (registrar, actualizar, eliminar, ver).																										
Secuencia normal	<table border="0"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario ingresa al área de trabajo "Alojamientos".</td> </tr> <tr> <td>2</td> <td>Completa un formulario con información del alojamiento.</td> </tr> <tr> <td>3</td> <td>Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.</td> </tr> <tr> <td>4</td> <td>El usuario puede ver el nuevo registro en la tabla del área de trabajo.</td> </tr> <tr> <td>5</td> <td>El usuario selecciona un servicio de alojamiento de la lista.</td> </tr> <tr> <td>6</td> <td>El sistema muestra los datos actuales en un formulario editable.</td> </tr> <tr> <td>7</td> <td>El usuario modifica los campos necesarios y registra los cambios.</td> </tr> <tr> <td>8</td> <td>El sistema actualiza los datos y muestra una notificación de confirmación.</td> </tr> <tr> <td>9</td> <td>El usuario puede ver los cambios realizados al registro en la tabla de la vista.</td> </tr> <tr> <td>10</td> <td>El usuario a selecciona un tipo de servicio de alojamiento y confirma su eliminación.</td> </tr> <tr> <td>11</td> <td>El sistema elimina el registro y notifica al usuario con un mensaje de éxito.</td> </tr> <tr> <td>6</td> <td>El usuario puede apreciar en la tabla de la vista el registro eliminado.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario ingresa al área de trabajo "Alojamientos".	2	Completa un formulario con información del alojamiento.	3	Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.	4	El usuario puede ver el nuevo registro en la tabla del área de trabajo.	5	El usuario selecciona un servicio de alojamiento de la lista.	6	El sistema muestra los datos actuales en un formulario editable.	7	El usuario modifica los campos necesarios y registra los cambios.	8	El sistema actualiza los datos y muestra una notificación de confirmación.	9	El usuario puede ver los cambios realizados al registro en la tabla de la vista.	10	El usuario a selecciona un tipo de servicio de alojamiento y confirma su eliminación.	11	El sistema elimina el registro y notifica al usuario con un mensaje de éxito.	6	El usuario puede apreciar en la tabla de la vista el registro eliminado.
Paso	Acción																										
1	El usuario ingresa al área de trabajo "Alojamientos".																										
2	Completa un formulario con información del alojamiento.																										
3	Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.																										
4	El usuario puede ver el nuevo registro en la tabla del área de trabajo.																										
5	El usuario selecciona un servicio de alojamiento de la lista.																										
6	El sistema muestra los datos actuales en un formulario editable.																										
7	El usuario modifica los campos necesarios y registra los cambios.																										
8	El sistema actualiza los datos y muestra una notificación de confirmación.																										
9	El usuario puede ver los cambios realizados al registro en la tabla de la vista.																										
10	El usuario a selecciona un tipo de servicio de alojamiento y confirma su eliminación.																										
11	El sistema elimina el registro y notifica al usuario con un mensaje de éxito.																										
6	El usuario puede apreciar en la tabla de la vista el registro eliminado.																										
Postcondición	Los registros de los tipos de servicios de alojamientos se actualizan y reflejan los cambios en tiempo real en el área de trabajo “Alojamientos”.																										
Excepciones	<table border="0"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Validación de datos fallidos.</td> </tr> </tbody> </table>	Paso	Acción	1	Validación de datos fallidos.																						
Paso	Acción																										
1	Validación de datos fallidos.																										

E.1 El sistema detecta campos vacíos, formato incorrecto, mostrando el mensaje de error.

Comentarios Control de acceso basado en roles para garantizar que solo los administradores puedan realizar estas operaciones.

Nota. Autoría propia

Escenario 03 Caso de Uso Actor Administrador

La tabla 6 describe las condiciones y el paso a paso detallado de los casos de uso.

Tabla 6

Escenario 03 Actor Administrador

CU-01 Gestionar Habitaciones																											
Versión	01 (02/12/2024)																										
Precondición	El usuario debe estar autenticado con un rol “administrador” y registrar previamente los tipos de servicios de alojamientos.																										
Descripción	El sistema tendrá que proceder como se puntualiza para el siguiente caso de uso cuando el usuario realice la gestión de tipos de habitaciones.																										
Secuencia normal	<table border="0"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario ingresa al área de trabajo "Habitaciones".</td> </tr> <tr> <td>2</td> <td>Completa el formulario con información del tipo de habitación.</td> </tr> <tr> <td>3</td> <td>Si se aprueban datos, el sistema los almacena y muestra un mensaje de éxito.</td> </tr> <tr> <td>4</td> <td>El usuario puede ver el nuevo registro en la tabla del área de trabajo.</td> </tr> <tr> <td>5</td> <td>El usuario selecciona un tipo de acomodación de la lista.</td> </tr> <tr> <td>6</td> <td>El sistema muestra los datos actuales en un formulario editable.</td> </tr> <tr> <td>7</td> <td>El usuario modifica los campos necesarios y registra los cambios.</td> </tr> <tr> <td>8</td> <td>El sistema actualiza los datos y muestra una notificación de confirmación.</td> </tr> <tr> <td>9</td> <td>El usuario puede ver los cambios realizados al registro en la tabla de la vista.</td> </tr> <tr> <td>10</td> <td>El usuario selecciona un tipo de habitación y confirma su eliminación.</td> </tr> <tr> <td>11</td> <td>El sistema elimina el registro y notifica al usuario con un mensaje de éxito.</td> </tr> <tr> <td>6</td> <td>El usuario puede apreciar en la tabla el registro eliminado.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario ingresa al área de trabajo "Habitaciones".	2	Completa el formulario con información del tipo de habitación.	3	Si se aprueban datos, el sistema los almacena y muestra un mensaje de éxito.	4	El usuario puede ver el nuevo registro en la tabla del área de trabajo.	5	El usuario selecciona un tipo de acomodación de la lista.	6	El sistema muestra los datos actuales en un formulario editable.	7	El usuario modifica los campos necesarios y registra los cambios.	8	El sistema actualiza los datos y muestra una notificación de confirmación.	9	El usuario puede ver los cambios realizados al registro en la tabla de la vista.	10	El usuario selecciona un tipo de habitación y confirma su eliminación.	11	El sistema elimina el registro y notifica al usuario con un mensaje de éxito.	6	El usuario puede apreciar en la tabla el registro eliminado.
Paso	Acción																										
1	El usuario ingresa al área de trabajo "Habitaciones".																										
2	Completa el formulario con información del tipo de habitación.																										
3	Si se aprueban datos, el sistema los almacena y muestra un mensaje de éxito.																										
4	El usuario puede ver el nuevo registro en la tabla del área de trabajo.																										
5	El usuario selecciona un tipo de acomodación de la lista.																										
6	El sistema muestra los datos actuales en un formulario editable.																										
7	El usuario modifica los campos necesarios y registra los cambios.																										
8	El sistema actualiza los datos y muestra una notificación de confirmación.																										
9	El usuario puede ver los cambios realizados al registro en la tabla de la vista.																										
10	El usuario selecciona un tipo de habitación y confirma su eliminación.																										
11	El sistema elimina el registro y notifica al usuario con un mensaje de éxito.																										
6	El usuario puede apreciar en la tabla el registro eliminado.																										
Postcondición	Los registros tipos de habitación se actualizan y reflejan los cambios en tiempo real en el área de trabajo “Habitaciones”.																										
Excepciones	<table border="0"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> </table>	Paso	Acción																								
Paso	Acción																										

	1	Validación de datos fallidos. E.1 El sistema detecta campos vacíos, formato incorrecto, mostrando el mensaje de error.
Comentarios		Control de acceso basado en roles para garantizar que solo los administradores puedan realizar estas operaciones.

Nota. Autoría propia

Escenario 04 Caso de Uso Actor administrador

La tabla 7 describe las condiciones y el paso a paso detallado de los casos de uso para el actor administrador en la gestión de “Tarifas”.

Tabla 7

Escenario 04 Actor Administrador

CU-01 Gestionar Tarifas																									
Versión	01 (02/12/2024)																								
Precondición	El usuario debe estar autenticado con un rol “administrador” que permite gestionar tarifas.																								
Descripción	El sistema tendrá que proceder como se puntualiza para el siguiente caso de uso cuando el usuario realice la gestión de tarifas.																								
Secuencia	<table border="1"> <thead> <tr> <th>Paso</th> <th>Acción</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>El usuario ingresa al área de trabajo "Tarifas".</td> </tr> <tr> <td>2</td> <td>Completa el formulario con información las tarifas.</td> </tr> <tr> <td>3</td> <td>Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.</td> </tr> <tr> <td>4</td> <td>El usuario puede ver el nuevo registro en la tabla del área de trabajo.</td> </tr> <tr> <td>5</td> <td>El usuario selecciona una tarifa de la lista.</td> </tr> <tr> <td>6</td> <td>El sistema muestra los datos actuales en un formulario editable.</td> </tr> <tr> <td>7</td> <td>El usuario modifica los campos necesarios y almacena los cambios.</td> </tr> <tr> <td>8</td> <td>El sistema actualiza los datos y muestra una notificación de confirmación.</td> </tr> <tr> <td>9</td> <td>El usuario puede ver los cambios realizados al registro en la tabla de la vista.</td> </tr> <tr> <td>10</td> <td>El usuario selecciona una tarifa y confirma su eliminación.</td> </tr> <tr> <td>11</td> <td>El sistema elimina el registro y notifica al usuario con un mensaje de éxito.</td> </tr> </tbody> </table>	Paso	Acción	1	El usuario ingresa al área de trabajo "Tarifas".	2	Completa el formulario con información las tarifas.	3	Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.	4	El usuario puede ver el nuevo registro en la tabla del área de trabajo.	5	El usuario selecciona una tarifa de la lista.	6	El sistema muestra los datos actuales en un formulario editable.	7	El usuario modifica los campos necesarios y almacena los cambios.	8	El sistema actualiza los datos y muestra una notificación de confirmación.	9	El usuario puede ver los cambios realizados al registro en la tabla de la vista.	10	El usuario selecciona una tarifa y confirma su eliminación.	11	El sistema elimina el registro y notifica al usuario con un mensaje de éxito.
Paso	Acción																								
1	El usuario ingresa al área de trabajo "Tarifas".																								
2	Completa el formulario con información las tarifas.																								
3	Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.																								
4	El usuario puede ver el nuevo registro en la tabla del área de trabajo.																								
5	El usuario selecciona una tarifa de la lista.																								
6	El sistema muestra los datos actuales en un formulario editable.																								
7	El usuario modifica los campos necesarios y almacena los cambios.																								
8	El sistema actualiza los datos y muestra una notificación de confirmación.																								
9	El usuario puede ver los cambios realizados al registro en la tabla de la vista.																								
10	El usuario selecciona una tarifa y confirma su eliminación.																								
11	El sistema elimina el registro y notifica al usuario con un mensaje de éxito.																								
normal																									

	6	El usuario puede apreciar en la tabla de la vista el registro eliminado.
Postcondición		Los registros de las tarifas se actualizan y reflejan los cambios en tiempo real en la interfaz de usuario.
Excepciones	Paso	Acción
	1	Validación de datos fallidos.
	E.1	El sistema detecta campos vacíos, formato incorrecto, mostrando el mensaje de error.
Comentarios		Control de acceso basado en roles para garantizar que solo los administradores puedan realizar estas operaciones.

Nota. Autoría propia

Escenario 05 Caso de Uso Actores administrador y Recepcionista

La tabla 8 describe las condiciones y el paso a paso detallado de los casos de uso.

Tabla 8

Escenario 05 Actor Administrador Recepcionista

CU-01 Gestionar Reservas		
Versión		01 (02/12/2024)
Precondición		El usuario debe estar autenticado con un rol “administrador” y registrar previamente los tipos de alojamiento, tarifas y habitaciones.
Descripción		El sistema tendrá que proceder como se puntualiza para el siguiente caso de uso cuando el usuario realice la gestión de reservas (registrar, actualizar, ver, etc.).
Secuencia	Paso	Acción
normal	1	El usuario administrador ingresa al área de trabajo "Gestión de Reservas".
	2	Completa el formulario con información las reservas.
	3	El sistema valida si el tipo de habitación está disponible en las fechas seccionadas.
	4	Si se aprueban los datos, el sistema los almacena en la base de datos y muestra un mensaje de éxito.
	5	El usuario puede ver el nuevo registro en color rojo en la tarjeta de la vista.
	6	El usuario selecciona la reserva de la lista.
	7	El sistema muestra los datos actuales en un formulario editable.
	8	El usuario modifica los campos necesarios y almacena los cambios.

	9	El sistema actualiza los datos y muestra una notificación de confirmación.
	10	El usuario administrador puede ver los cambios realizados al registro en la tarjeta que del área de trabajo.
	11	El usuario selecciona la reserva y confirma su salida.
	12	El sistema da salida a la reserva y notifica al usuario con un mensaje de éxito.
	13	El usuario puede apreciar en color verde la liberación del tipo habitación con el mensaje “sin ocupación”.
	14	El usuario selecciona la reserva, se abre un formulario y elige el tipo de habitación al cual va a enviar la reserva.
	15	Al validar que el tipo de habitación está disponible, se muestra un mensaje de éxito.
	16	El usuario puede apreciar en color verde la liberación del tipo habitación con el mensaje “sin ocupación” y en color rojo el tipo habitación con la información de la reserva.
	17	El usuario selecciona la reserva y confirma su cancelación.
	18	El sistema cancela el registro y notifica al usuario con un mensaje de éxito.
	19	El usuario puede apreciar en color verde la liberación del tipo habitación con el mensaje “sin ocupación”.
	20	El usuario selecciona la fecha a consultar o navegar entre fecha anterior o posterior.
	21	El sistema muestra las reservas activas para las fechas consultadas y su estado.
Postcondición		Los registros de las reservas se actualizan y reflejan los cambios en tiempo real en la interfaz de usuario.
Excepciones	Paso	Acción
	1	Validación de datos fallidos.
	E.1	El sistema detecta campos vacíos, formato incorrecto, mostrando el mensaje de error.
	2	Validación de disponibilidad
	E.2-1	Si se intenta registrar una reserva en fechas ya ocupadas, el sistema muestra un mensaje advirtiendo que la habitación está reservada.
	E.2-2	Si se intenta cambiar a un tipo de alojamiento no disponible, el sistema muestra un mensaje advirtiendo que está ocupada.
Comentarios		Las validaciones de datos deben garantizar que no existan conflictos en las fechas, habitaciones ocupadas o dependencias antes de completar las acciones.

Los cambios en las tarifas deben reflejarse automáticamente en los costos totales de la reserva.

Las reservas finalizadas o canceladas no deben aparecer en la lista de reservas activas.

Nota. Autoría propia

Fase 2 - Diseño

Teniendo definidos los requerimientos de la fase 1 para el sistema web, principalmente los de descripción general: disposición de infraestructura tecnología y habilidades digitales adecuadas de los recepcionistas, y requisitos funcionales: límites mínimos y máximos de camas y personas alojadas por habitación, horarios de check-in y check-out, tarifas, módulos de usuario, establecimiento, tarifas, tipos de alojamientos y habitaciones, reservas y generación de la TRA y sus respectivas funcionalidades: listar, registrar, actualizar, buscar y eliminar, se procede al diseño, considerando también restricciones como las normativas legales para la generación de la TRA estipuladas por el Ministerio de Comercio, Industria y Turismo, todo ello como resultado de la investigación y análisis previo de datos otorgados por parte del usuario final, en este caso los prestadores de servicios de alojamiento turístico.

Base Estructural

Dentro del diseño y desarrollo del sistema web de alojamientos se aplicará la programación orientada a objetos (POO), como base estructural, permitiendo modularidad y reutilización del código. Este paradigma se integrará con la técnica estándar de modelado de sistemas UML (Lenguaje de Modelado Unificado) empleada en ingeniería de software como herramienta para documentar y diseñar sistemas, basándose en representaciones graficas que ayudan a entender las acciones del sistema, diagramas de entidad-relación para estructurar los datos, patrón de arquitectura es decir la configuración de los componentes del sistema y prototipos de interfaz de usuario para interactuar con las funcionalidades y el diseño.

Dicho modelo fue probado y validado por varios autores e ingenieros de software como Booch 1998 con su libro “El Lenguaje de Modelado Unificado: Manual de Referencia”, Pressman 2010 con su libro “Ingeniería de software un enfoque practico” y Sommerville 2011

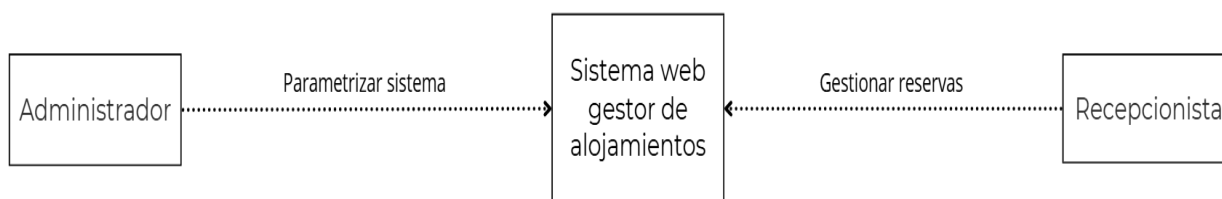
con su libro “Ingeniería de software”, considerando la documentación de este último como más práctica, detallada, aplicable y vigente para el diseño de sistema web gestor de alojamientos, ya que se basa en prácticas reales e investigaciones científicas.

Modelado de Contexto

El gráfico UML, diagrama de contexto de la figura 22, hace referencia el autor a que son “las fronteras del sistema” (Sommerville, 2011, p. 121), en otras palabras, expone los componentes externos que interactúan con el sistema.

Figura 22

Contexto del Sistema Web



Nota. Autoría propia

Modelado Dirigido por Datos

Una vez definido el contexto del sistema web se procede a representar el flujo de acciones en el diagrama de actividades UML, este permite “mostrar las actividades incluidas en un proceso o en el procesamiento de datos” (Sommerville, 2011, p. 134), el cual es indicado para visualizar la secuencia lógica cada funcionalidad del sistema web de alojamientos. Estos gráficos representan las acciones del usuario en nodos rectangulares, los hexágonos indican decisiones y las flechas conectan las secuencias para guiar al usuario.

La figura 23 ilustra el flujo de la actividad del sistema web al iniciar sesión:

Se accede a la página de inicio del sistema web.

Digita el nombre usuario y la clave en el formulario.

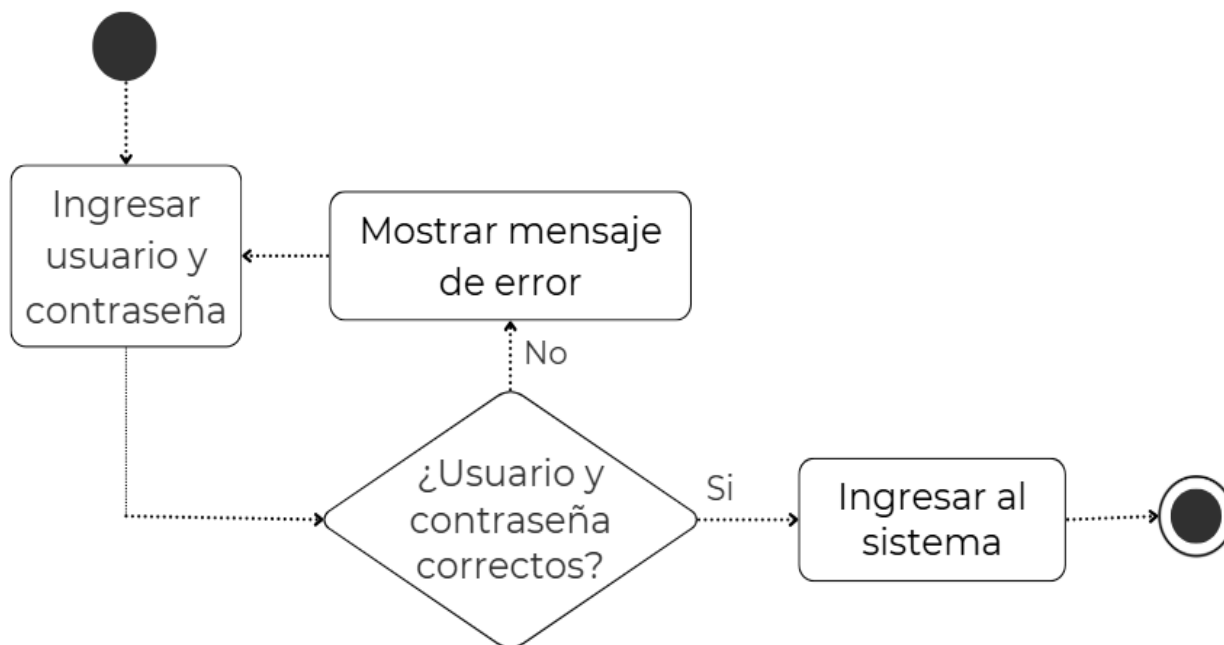
El sistema web valida los datos.

Si los datos son correctos, el sistema web lo redireccionará al área de trabajo “Gestor”,

En caso de que los datos no sean correctos se muestra un mensaje de error y le permitirá intentarlo de nuevo.

Figura 23

Diagrama Flujo de Actividad Inicio de Sesión



Nota. Autoría propia

La figura 24 ilustra el flujo de la actividad del sistema web al registrar un empleado:

Se ingresa al sistema web. Luego se dirige al área de trabajo “Empleados”.

Ver si el empleado ya existe. Si es así, se puede acceder a la información desde el formulario de edición y actualizar la información si es necesario.

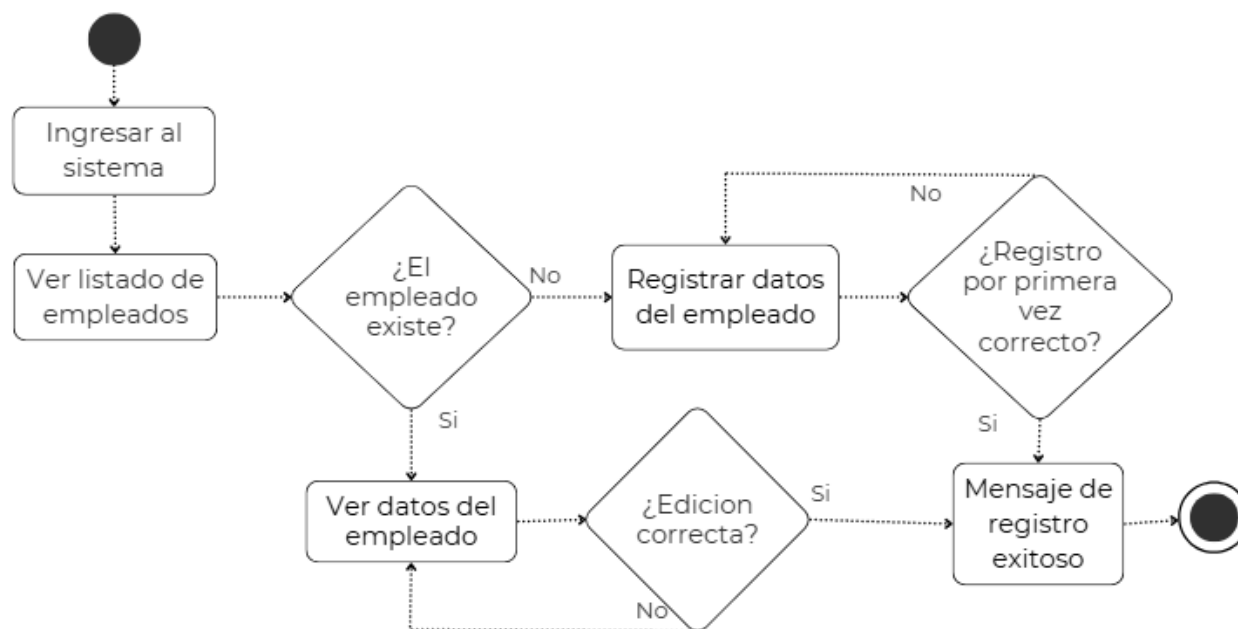
En caso de que el empleado no este registrado en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validara que los datos contengan el formato correcto para todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 24

Diagrama Flujo de Actividad Registrar Usuario Empleado



Nota. Autoría propia

La figura 25 ilustra el flujo de la actividad del sistema web al registrar un nuevo tipo de alojamiento:

Se ingresa al sistema web. Luego se dirige al área de trabajo “Alojamientos”.

Ver si el alojamiento ya existe. Si es así, se puede acceder a la información del tipo de alojamiento desde el formulario de edición y actualizar la información si es necesario.

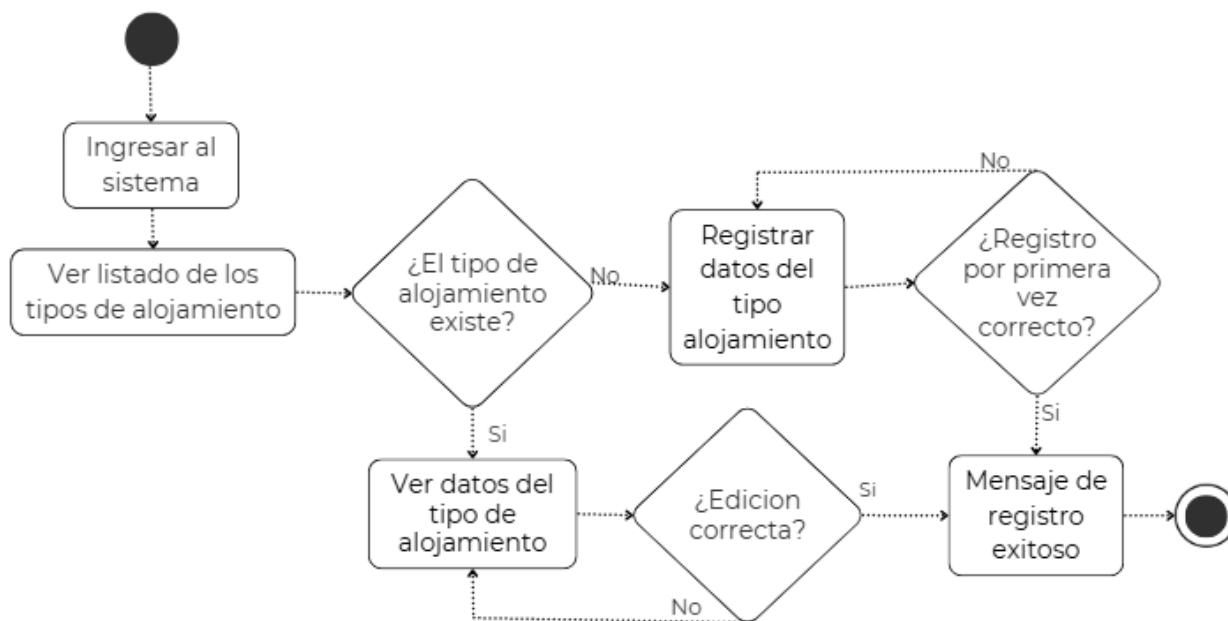
En caso de que el tipo de alojamiento no este registrado en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validara que los datos contengan el formato correcto para todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 25

Diagrama Flujo de Actividad Registrar Tipos de Alojamiento



Nota. Autoría propia

La figura 26 ilustra el flujo de la actividad del sistema web al registrar un nuevo tipo de habitación:

Se ingresa al sistema web. Luego se dirige al área de trabajo “Habitaciones”.

Ver si la habitación ya existe. Si es así, se puede acceder a la información desde el formulario de edición y actualizar la información si es necesario.

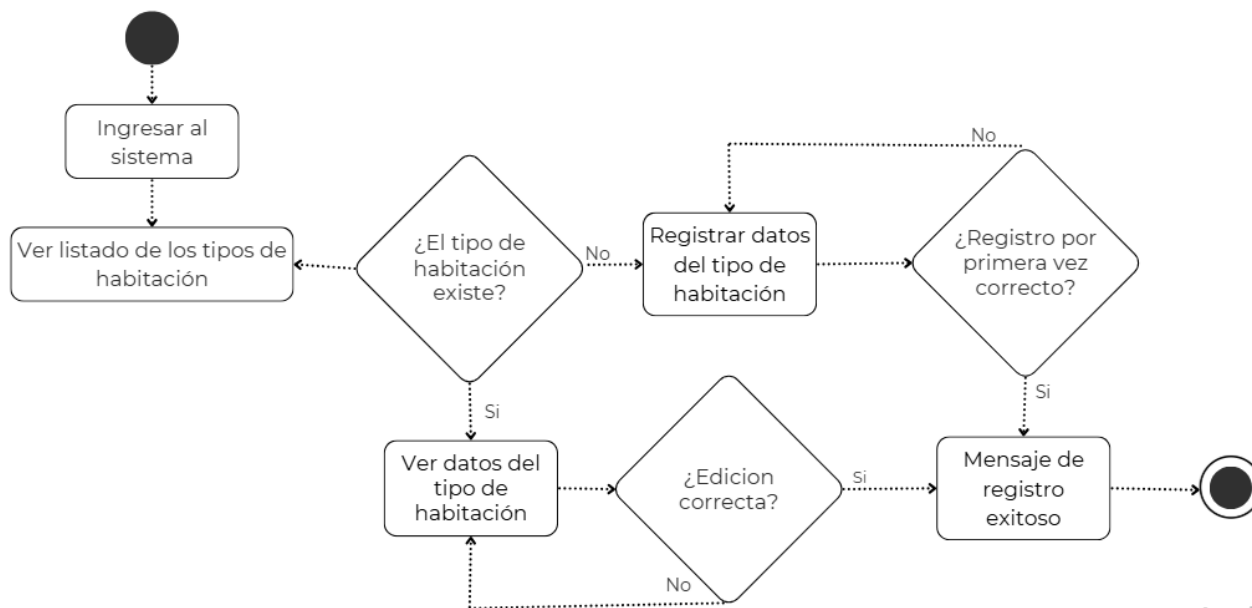
En caso de que la habitación no este registrada en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validara que los datos contengan el formato correcto en todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 26

Diagrama Flujo de Actividad Registrar Tipos de Habitación



Nota. Autoría propia

La figura 27 ilustra el flujo de la actividad del sistema web al registrar una nueva tarifa de alojamiento:

Se ingresa al sistema web. Luego se dirige al área de trabajo de “Tarifas”.

Ver si la tarifa de alojamiento ya existe. Si es así, se puede acceder a la información desde el formulario de edición y actualizar la información si es necesario.

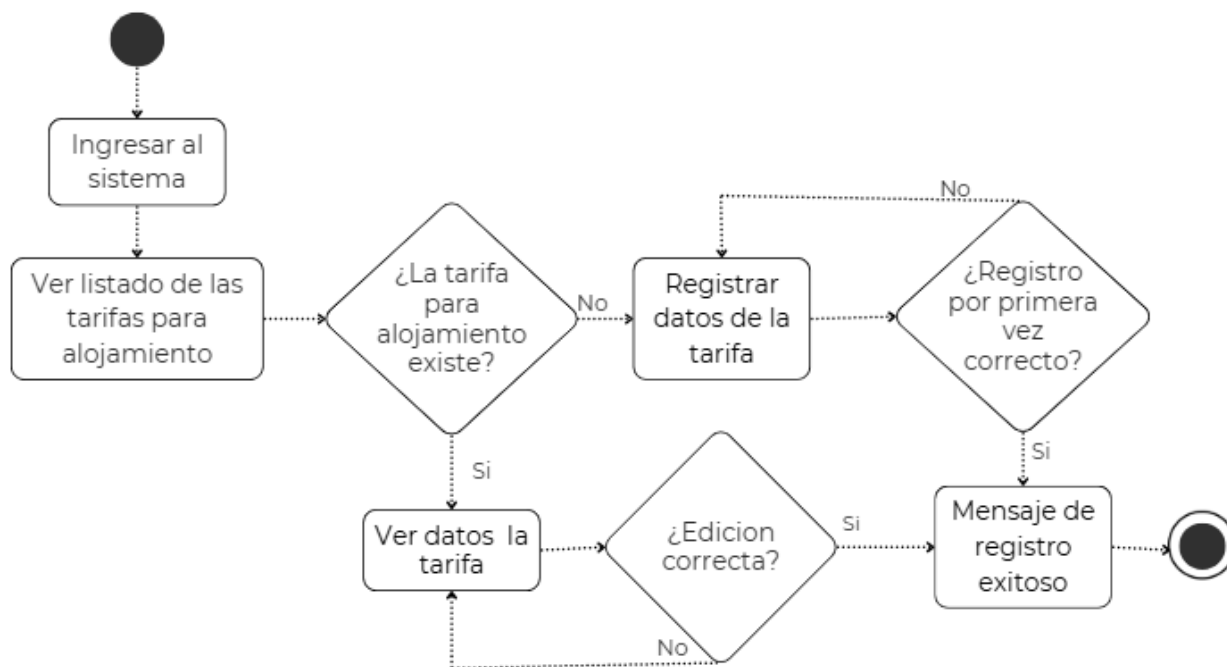
En caso de que la tarifa de alojamiento no este registrada en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validara que los datos contengan el formato correcto en todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 27

Diagrama Flujo de Actividad Registrar Tarifas de Alojamiento



Nota. Autoría propia

La figura 28 ilustra el flujo de la actividad del sistema web al registrar una nueva tarifa del parqueadero:

Se ingresa al sistema web. Luego se dirige al área de trabajo de “Tarifas”.

Ver si la tarifa del parqueadero ya existe. Si es así, se puede acceder a la información desde el formulario de edición y actualizar la información si es necesario.

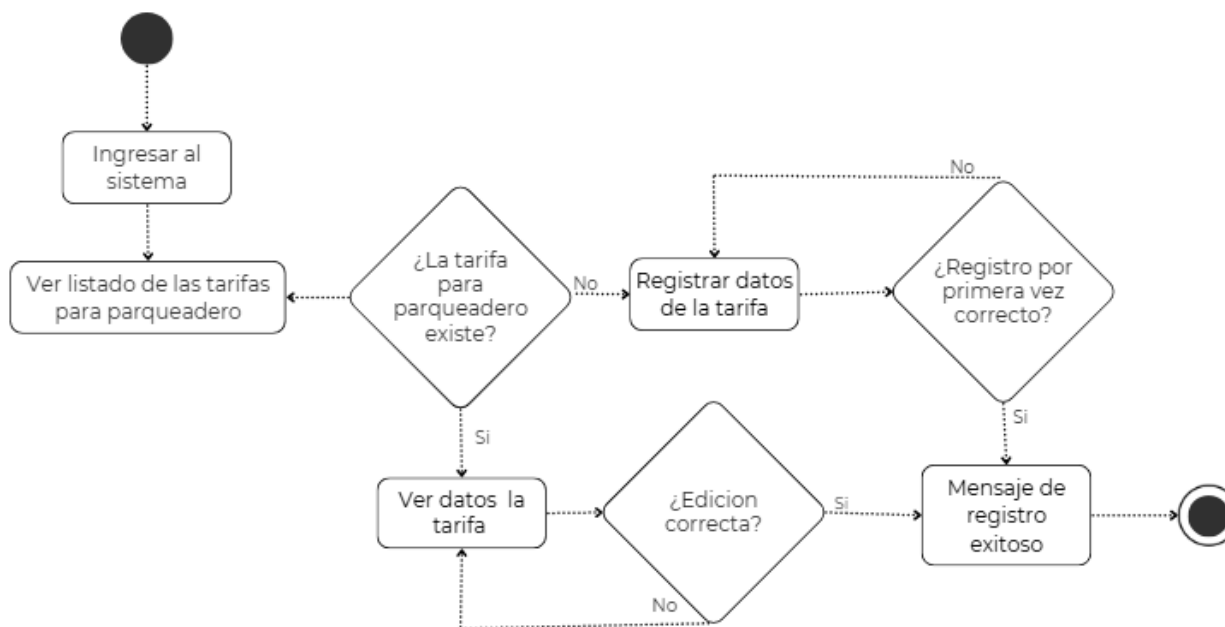
En caso de que la tarifa del parqueadero no este registrada en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validara que los datos contengan el formato correcto en todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 28

Diagrama Flujo de Actividad Registrar Tarifas del Parqueadero



Nota. Autoría propia

La figura 29 ilustra el flujo de la actividad del sistema web al registrar una nueva reserva para huésped principal:

Se ingresa al sistema web. Luego se dirige al área de trabajo “Gestión de reservas”.

Ver si la reserva para el huésped principal ya existe. Si es así, se puede acceder a la información desde el formulario de edición y actualizar la información si es necesario.

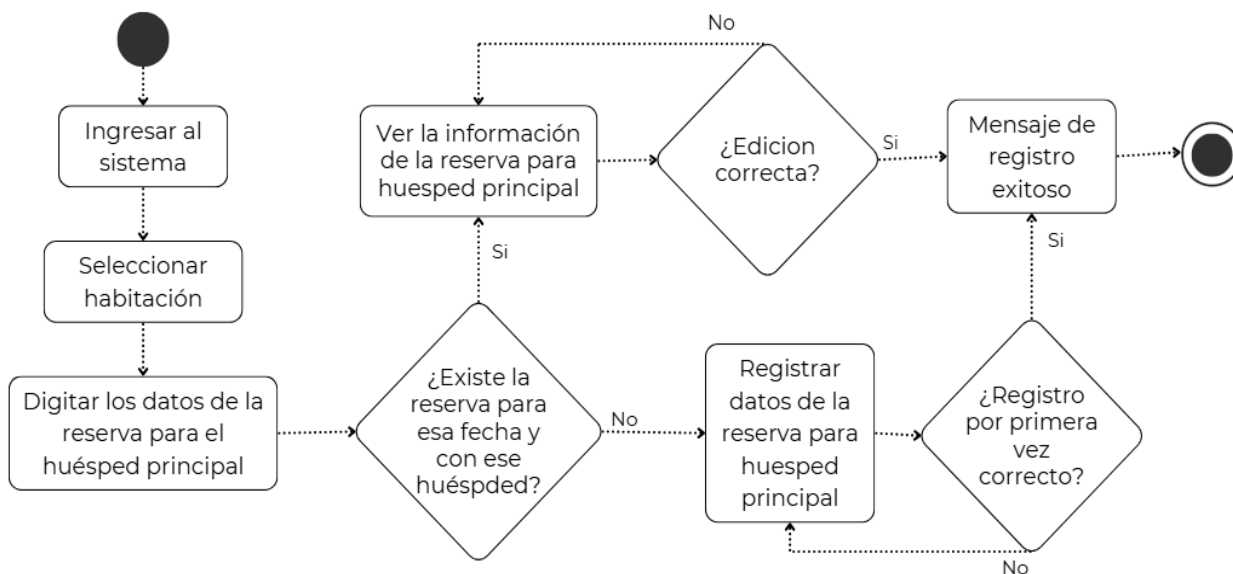
En caso de que la reserva para el huésped principal no este registrada en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validara que los datos contengan el formato correcto en todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 29

Diagrama Flujo de Actividad Registrar Reserva para Huésped Principal



Nota. Autoría propia

La figura 30 ilustra el flujo de la actividad del sistema web al registrar un nuevo huésped acompañante en la reserva:

Se ingresa al sistema web. Luego se dirige al área de trabajo “Gestión de reservas”.

Ver si la reserva para el huésped principal ya existe. Si es así, verificar si existen acompañantes.

Si el acompañante está registrado, puede acceder a la información desde el formulario de edición y actualizar la información si es necesario.

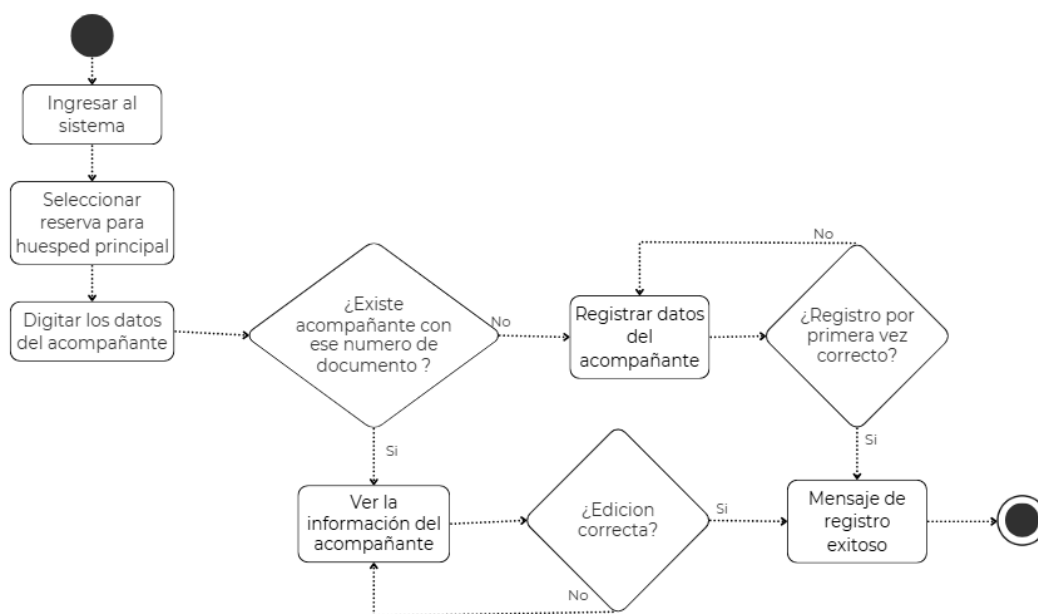
En caso de que el acompañante no esté registrado en el sistema web, dirigirse al formulario de registro para digitar los datos.

El sistema web validará que los datos contengan el formato correcto en todos los campos.

Si los campos del formulario de registro y edición se validan correctamente se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 30

Diagrama Flujo de Actividad Registrar Huésped Acompañante



Nota. Autoría propia

La figura 31 ilustra el flujo de la actividad del sistema web al realizar traslado de huéspedes a nueva habitación.

Se ingresa al sistema web. Luego se dirige al área de trabajo “Gestión de reservas”.

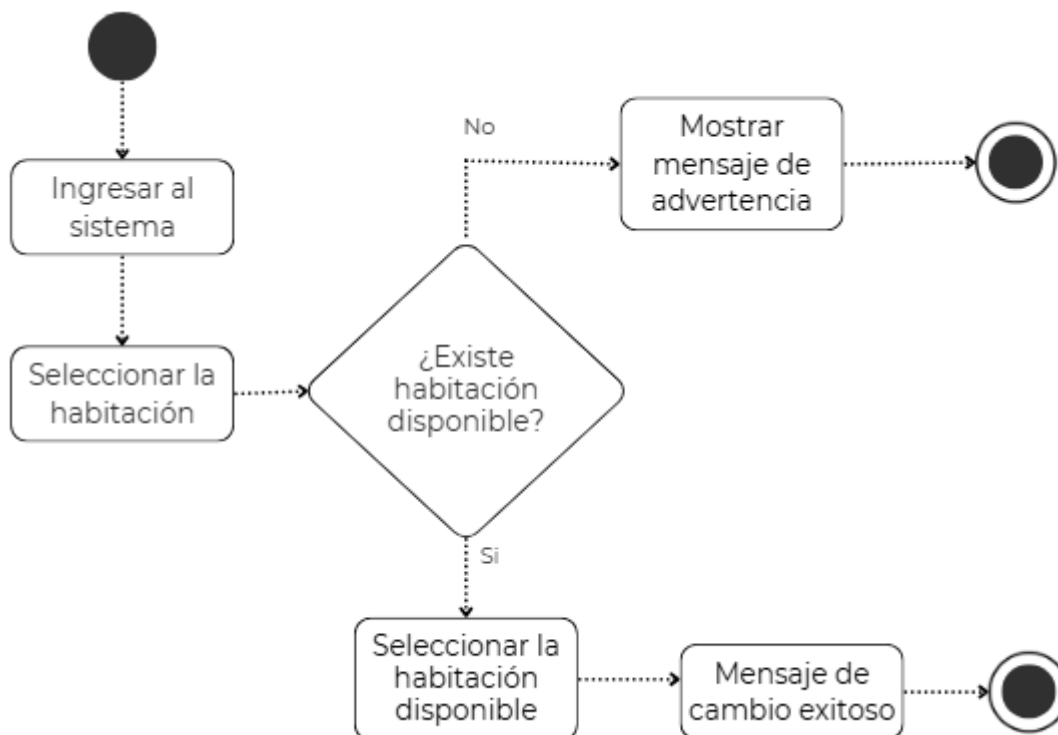
Ver si la reserva existe. Si es así, puede hacer traslado de huéspedes a otra habitación desde el formulario.

En caso de que la habitación no esté disponible por ocupación, el sistema web le mostrar un mensaje advertencia finalizando el proceso.

Una vez confirmada la acción, en ambos casos se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 31

Diagrama Flujo de Actividad Traslado de Habitación



Nota. Autoría propia

La figura 32 ilustra el flujo de la actividad del sistema web al realizar check-out de los huéspedes de la reserva.

Se ingresa al sistema web. Luego se dirige al área de trabajo “Gestión de reservas”.

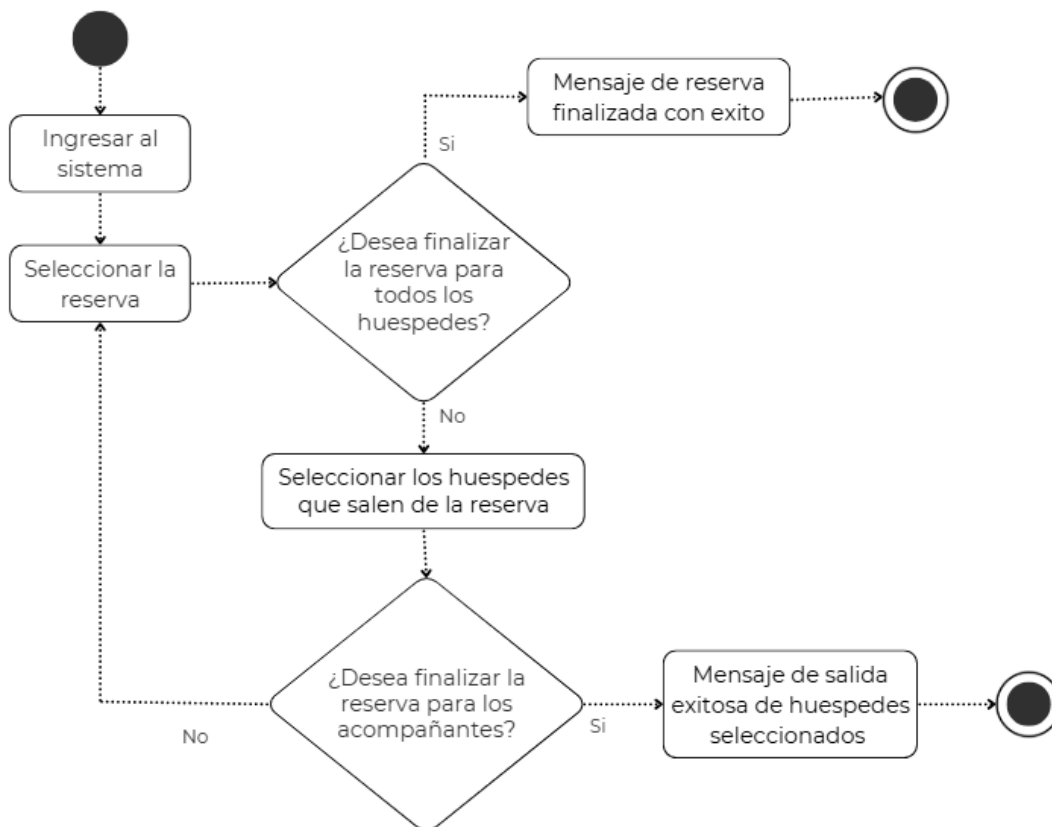
Ver si la reserva existe. Si es así, puede hacer check-out para todos los huéspedes de la reserva desde el formulario.

De lo contrario puede seleccionar los huéspedes que deseen hacer check-out y la reserva permanecerá activa con demás los huéspedes sin ningún problema.

Una vez confirmada la acción, en ambos casos se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 32

Diagrama Flujo de Actividad Check-out para los Huespedes



Nota. Autoría propia

La figura 33 ilustra el flujo de la actividad del sistema web al realizar generación y descargue de la TRA en formato Excel.

Se ingresa al sistema web. Luego se dirige al área de trabajo de “Reporte TRA”.

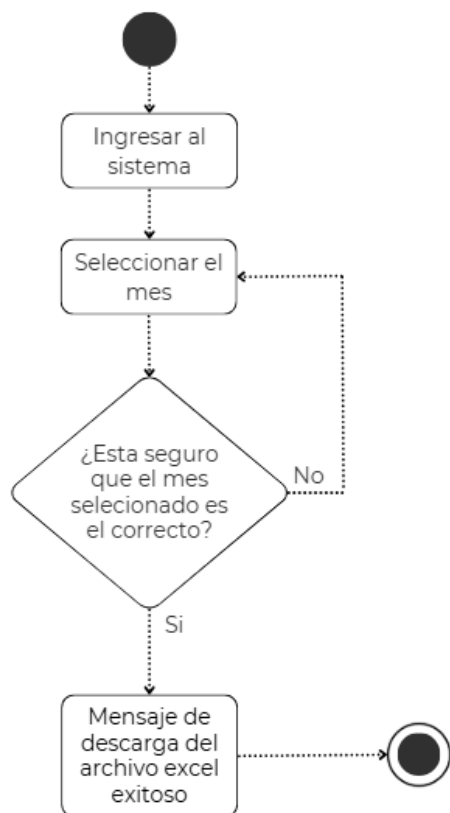
Seleccionar mes.

En caso de que el mes elegido no tenga información, se mostrara un mensaje de advertencia y podrá elegir nuevamente.

De lo contrario se creará el archivo Excel con la información y se mostrará un mensaje notificando el éxito del proceso y este finalizará.

Figura 33

Diagrama Flujo de Actividad Generación de la TRA



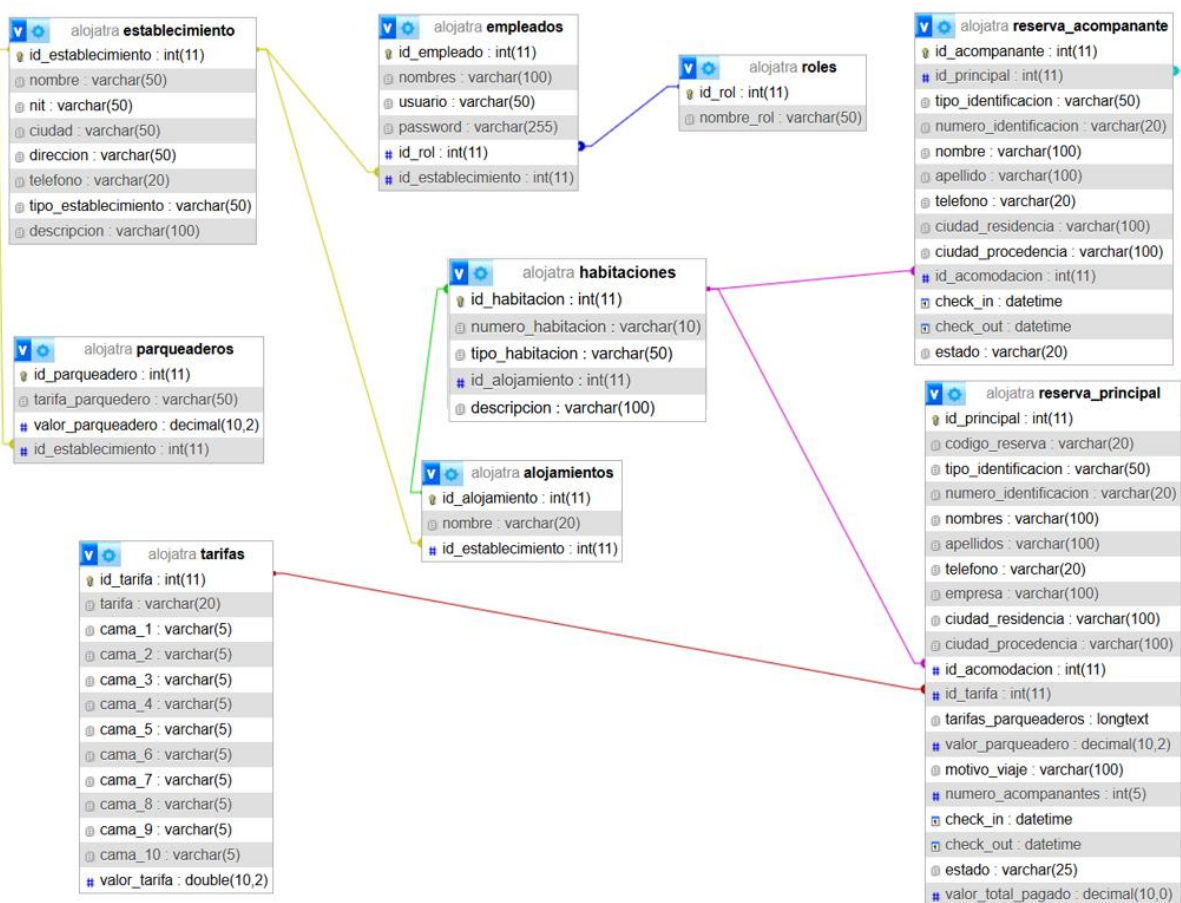
Nota. Autoría propia

Modelo de la Estructura de Datos

Para representar gráficamente los datos del sistema web de alojamientos, se aplicará el modelo entidad – relación que consiste en el diseño de entidades (personas, lugares, cosas), atributos (características) y sus conexiones que ayudan relacionar los datos para búsquedas precisas. En la figura 34, se describe la estructura de la base de datos para el sistema alojamiento con las siguientes entidades de acuerdo con los requerimientos identificados en los prestadores de servicios turísticos y en el manual TRA.

Figura 34

Diseño Entidad - Relación



Nota. El grafico presenta como se ven los datos del sistema estructurados y almacenados en tablas con sus respectivas relaciones para realizar consultas.

La tabla “establecimiento” contiene los atributos necesarios para guardar la información del establecimiento.

La tabla “empleados” contiene los atributos necesarios para guardar la información del usuario empleado y a su vez está ligada a la tabla “roles” para obtener y asignar los diferentes roles a los usuarios empelados con el fin controlar el acceso a funcionalidades en el sistema.

La tabla “rol” contiene los atributos necesarios para guardar la información del rol que el empleado que se maneja en el establecimiento.

La tabla “alojamientos” contiene los atributos necesarios para guardar la información del tipo de servicio de alojamiento que maneja en el establecimiento (cabañas, hotel, hostales, etc.).

La tabla “habitaciones” contiene los atributos necesarios para guardar la información del tipo de acomodación que se maneja en los alojamientos (cama doble, sencilla, cabaña, apartamento, casa, finca, etc.) Además, está ligada a la tabla “alojamientos” para saber a cuál servicio de alojamientos pertenece la habitación en caso de que el establecimiento cuente con más de uno.

La tabla “tarifas” contiene los atributos necesarios para guardar la información de las diferentes tarifas que se maneja en los tipos de alojamiento.

La tabla “parqueaderos” contiene los atributos necesarios para guardar la información de las tarifas para parqueadero.

La tabla “reserva_principal” contiene los atributos necesarios para guardar la información de la reserva para el huésped principal según la información solicitada en el manual TRA. Se encuentra ligada a las tablas “habitaciones” y “tarifas” para obtener el valor y el tipo de acomodación al que se va a realizar la reserva.

La tabla “reserva_acompañante” contiene los atributos necesarios para guardar la información del huésped acompañante el cual está vinculada a la tabla “habitaciones” y reserva del huésped principal según el manual TRA.

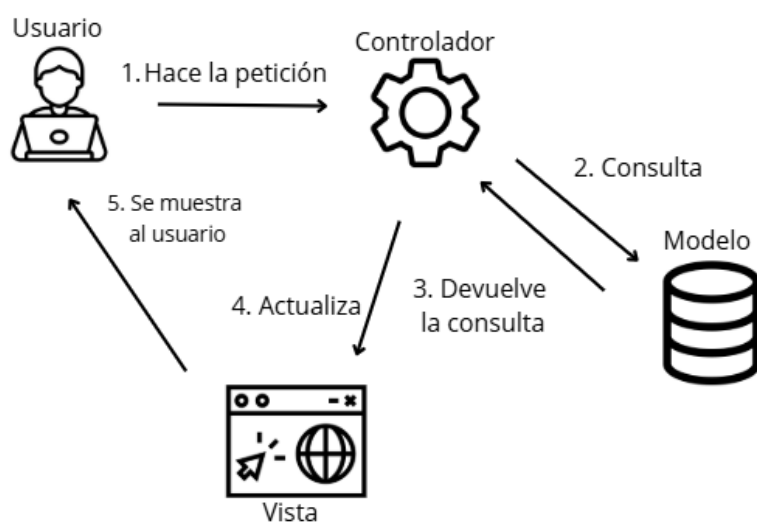
Arquitectura del Sistema Web

El diseño de la arquitectura es “cómo tiene que diseñarse la estructura global de ese sistema” (Sommerville, 2011, p.148) en otras palabras es la organización e interacción de los componentes o archivos que contienen el código de los lenguajes de programación del sistema.

Por su modularidad, escalabilidad, seguridad, organización, rendimiento y reutilización de código se eligió el patrón de arquitectura Modelo – Vista – Controlador, porque como se aprecia en la figura 35, el componente modelo gestiona los datos, la componente vista muestra la información y el componente controlador interactúa con el usuario actualizando el modelo o la vista, además que es patrón de arquitectura que más se alinea con la programación orientada a objetos.

Figura 35

Modelo - Vista - Controlador



Nota. Patrón de arquitectura MVC.

El modo de operación de la arquitectura M – V – C es: el usuario realiza la petición, el controlador la recibe y consulta en la base de datos a través del modelo, una vez los tenga, este devuelve la consulta con los datos al controlador para que actualice la vista con la información solicitada.

Consideraciones de Seguridad y Calidad del Software

En este contexto, es importante reconocer que el desarrollo software, en general, se encuentra guiado por marcos normativos y estándares internacionales que promueven buenas prácticas orientadas a garantizar la seguridad y la calidad de los sistemas, especialmente en aquellos que manejan información sensible o están dirigidos a sectores con alta responsabilidad operativa como la salud, la banca o los entornos gubernamentales. Entre los estándares más reconocidos se encuentran la ISO/IEC 27001 que se enfoca en la gestión de la seguridad de la información, mientras que la ISO/IEC 27002 actúa como marco complementario, al proporcionar directrices concretas sobre controles técnicos y organizativos. Por su parte, la ISO/IEC 25010 establece criterios de evaluación de calidad, permitiendo que el sistema sea confiable, usable, eficiente y mantenible.

Dicho lo anterior, para el sistema web se implementara como gestión de seguridad y control: cifrado de contraseñas para evitar robo de contraseñas, evitar exposición innecesaria de datos sensibles en las interfases de usuario (número de documento, contraseñas, correos, etc.), roles de acceso: administrador y recepcionista, validación de permisos (administrador y recepcionista: acceso total a gestión de usuarios, reservas, reportes, y recepcionista: no puede eliminar datos) procurando la protección de datos de los usuarios empleados y de huéspedes.

Respecto a los criterios de calidad se aplicará las características:

Funcionalidad: cumplir con los requisitos funcionales.

Usabilidad: interfaces claras, formularios intuitivos, botones bien ubicados, colores consistentes (verde para confirmar, rojo para eliminar), tipografía legible (system-ui y negrita en campos importantes).

Rendimiento y eficiencia: carga dinámica con AJAX, uso de DataTables para listar registros con paginación sin recargar toda la vista, consultas optimizadas.

Mantenibilidad: patrón MVC, comentarios claros, uso de clases y funciones reutilizables.

Seguridad: Uso de “password_hash” para contraseñas para evitar almacenamiento en texto plano, validación de tipos de datos en formularios con el fin de evitar el registro de datos incompletos, consultas preparadas para proteger los datos de inyecciones SQL.

Directrices de Accesibilidad Web WCAG

Además de los estándares mencionados, también existen normativas orientadas a mejorar la experiencia y el acceso equitativo a las plataformas digitales, así como a optimizar los procesos organizacionales en el desarrollo de software. La normativa WCAG en su versión actual 2.1 establece directrices de accesibilidad que buscan garantizar que las aplicaciones web sean usables por personas con o sin discapacidad, promoviendo un diseño más inclusivo. Tenerlas en cuenta favorece la equidad en el acceso al sistema web gestor de alojamientos.

Las directrices que se adoptaran son: contraste de color (texto oscuro sobre fondo blanco y viceversa) para las áreas de trabajo y formularios, identificación por propósito (los campos de los formularios deben tener etiquetas claras “label” y “placeholders” significativos), mensajes claros de error y uso de encabezados jerárquicos. Otras directrices como navegación por teclado, zoom y lectores de pantalla, cambio de idioma y atajos contextuales ya se encuentran de forma nativa, incorporada en los navegadores web y funcionan en cualquier página web.

Prototipos de Interfaz de usuario

Partiendo de las consideraciones sobre normativa WCAG y criterios de calidad y seguridad, se procede a la creación de los prototipos del sistema, los cuales son versiones visuales de interfaz de usuario que sirven para ilustrar la apariencia del sistema web, de acuerdo con los diagramas modelados con los requerimientos, generalmente realizando bosquejos en papel que se refinan en herramientas de diseño hasta llegar al prototipo ideal. No obstante, otra alternativa válida es el diseño del prototipo realizado directamente con las tecnologías que se emplearan en el desarrollo del sistema: HTML5 la cual da la estructura, CSS diseña y personaliza, y Bootstrap 5 proporciona componentes y planillas permitiendo la modificación rápida y sobre la misma sin tener crear nuevos prototipos.

Así pues, para crear una experiencia de usuario más accesible e intuitiva se agregarán iconos para mayor comprensión de las acciones. Los botones contarán con colores primarios los cuales representan cada acción: verde: registrar, azul: editar y rojo: eliminar. Para evitar la fatiga visual y dar un aspecto limpio, se utilizará el tipo de letra system-ui, con un tamaño adecuado y legible de 16px para texto y párrafos, y de 24px para títulos.

Todos los formularios usaran la misma plantilla generando coherencia y facilitando su uso. Las secciones contarán con una tabla para listar la información registrada desde los formularios. El diseño interactivo podrá simular en primera medida como se vería y como interactuaría el usuario con el sistema web. Hay que mencionar que no es funcional, solo será el prototipo sobre el cual se procederá a codificar en la etapa de implementación. Se agregan datos ficticios a la estructura HTML5 para verificar el comportamiento responsive de las tablas y de los demás contenedores de texto.

El prototipo de la figura 36 representa la interfaz con el formulario para iniciar sesión en el sistema web, que cuenta con los siguientes elementos: Encabezado con nombre del sistema web, campos para ingresar las credenciales de autenticación (nombre de usuario y clave) y botón de acceso, identificación de propósito con etiquetas “label” claras y “placeholders” significativos y contraste de color según directrices de accesibilidad WCAG.

Figura 36

Prototipo del Formulario Inicio de Sesión

El prototipo muestra un formulario de inicio de sesión con el siguiente diseño:

- Encabezado:** Logo "DigiHostal" y subtítulo "Gestor de alojamientos y reservas".
- Título de sección:** "Inicio de sesión" en verde.
- Etiqueta de usuario:** "Usuario" con una flecha que apunta a la etiqueta "Label" claro.
- Campo de usuario:** Un campo de texto con el placeholder "Escriba el usuario" y una flecha que apunta a la etiqueta "placeholder" significativo. Incluye un ícono de usuario.
- Etiqueta de clave:** "Clave".
- Campo de clave:** Un campo de texto con el placeholder "Escriba la clave" y un ícono de candado.
- Botón de acceso:** Un botón verde con el texto "Entrar".
- Enlaces:** "Recuparar mi contraseña" y "Regístra el establecimiento aquí" en azul.

Nota. Prototipo del formulario para inicias sesión.

Al ingresar al sistema web de gestión de alojamientos, se podrá apreciar el menú principal al lado izquierdo. En el menú aparece en todas las opciones para navegar por las

diferentes áreas de trabajo, al igual que el nombre de usuario y rol de quien ingresa, con el fin de asegurar la correcta identificación y habilitar accesos autorizados, reforzando la seguridad.

Como se aprecia en la figura 37, el prototipo HTML5 del área de trabajo cuenta para la esquina superior derecha con un botón de acción que puede expandir la sección a pantalla completa, seguidamente para mostrar el contenido se hace dentro de una card o tarjeta, con título de la sección y el botón para abrir el formulario de registro al lado superior izquierdo.

Figura 37

Prototipo del Área de Trabajo “Empleado”

The screenshot displays a web application interface for managing employees. The interface includes a sidebar with navigation options, a main header, and a table of employees with search and pagination controls.

Navigation Sidebar:

- Gestor
- Reporte TRA
- Configuraciones
- Configurar Establecimiento
- Configurar Empleados**
- Configurar Alojamientos
- Configurar Habitaciones
- Configurar Tarifas
- Salir

Main Header: patriciaPopayan (Admin) | Gestor de Alojamientos

Employee Management Card:

- Empleados** (+)
- Show 10 entries
- Search:

Nombre completo	Nombre de Usuario	Rol	Establecimiento	Acciones
Martin Jimenez	Martin1234	Recepcionista	Hostal Principal	Edit Delete
Patricia Popayan	patriciaPopayan	Admin	Hostal Principal	Edit Delete

Showing 1 to 2 of 2 entries

Previous 1 Next

Nota. Prototipo del área de trabajo empleados.

Debajo se encuentra un selector para listar la cantidad de registros en las tablas, al lado superior derecho se encuentra un campo para búsqueda, debajo una tabla con la información y los botones para abrir los formularios de edición y eliminación de registros, y finalmente en la parte inferior derecha la opción de paginación. Esta misma estructura se utiliza para las áreas de trabajo de alojamientos, habitación y tarifas, donde se aplicando los criterios de seguridad no se mostrará datos sensibles como la contraseña y no se permite la eliminación de datos desde el rol “Recepcionista”, también se evidencia el contraste de color texto oscuro sobre fondo blanco y viceversa, y encabezado jerárquico para comprensión de lectura, según la directriz WCAG.

Para el registro de un nuevo empleado, se hará clic en el botón verde en parte superior izquierda del área de trabajo que desplegará el formulario de la figura 38, del mismo modo, dentro de la tabla el botón azul abre el formulario editar. Una vez realizado el registro o modificación de los datos se guardará la información desde el botón de registro o actualización según sea el caso. El botón rojo muestra el modal para eliminar registro.

Figura 38

Prototipo del Formulario Registro de Usuario Empleado

Registro de empleados

Nombre completo **Nombre de Usuario**

Escriba el nombre completo Escriba el usuario

Clave **Confirmar Clave**

Escriba una clave Escriba otra vez la clave

Rol **Establecimiento**

Recepcionista Hostal Principal

Registrar

Nota. Prototipo del formulario para registrar empleados

Este mismo estilo y estructura donde se observa la aplicación de directrices de accesibilidad WCAG con navegación por teclado (campo resaltado en color azul), identificación por propósito etiquetas “label” claras y “placeholders” significativos, mensajes claros y como criterios de calidad se utiliza botones bien ubicados, formularios intuitivos con texto comprensible, validación de formularios (alertas de campos vacíos o tipos de datos incorrectos),

se aplicará a los demás formularios de registro y edición de establecimiento, alojamientos, habitaciones, tarifas y reservas de huésped principal y acompañante.

La figura 39 ilustra. el prototipo HTML5 del área de trabajo “Habitaciones”.

Figura 39

Prototipo del Área de Trabajo “Habitaciones”



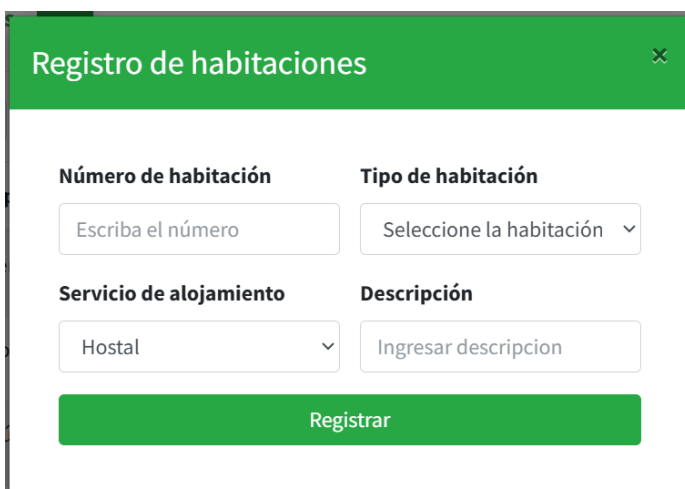
No.	Tipo de habitación	Servicio de alojamiento	Descripción	Acciones
100	Sencilla	Hostal	TV, baño privado, vista	[Editar] [Eliminar]
101	Doble	Hostal	TV, baño privado, vista	[Editar] [Eliminar]
102	Múltiple	Hostal	TV, cap. 5 personas	[Editar] [Eliminar]

Nota. Prototipo del área de trabajo habitación

La figura 40 ilustra, el prototipo HTML5 el formulario de registro de habitaciones.

Figura 40

Prototipo del Formulario Registro de Tipo de Habitaciones



Registro de habitaciones

Número de habitación
Escriba el número

Tipo de habitación
Seleccione la habitación

Servicio de alojamiento
Hostal

Descripción
Ingresar descripción

Registrar

Nota. Prototipo del formulario para registrar habitación.

La figura 41 ilustra el prototipo HTML5 del área de trabajo “Alojamientos”, aplicando el modo contraste según la directrices WCAG.

Figura 41

Prototipo de Área de Trabajo “Alojamientos”



Nota. Prototipo del área de trabajo alojamientos usando el cambio de contraste

La figura 42 ilustra, el prototipo HTML5 del formulario de registro de alojamientos y su versión modo contraste.

Figura 42

Prototipo del Formulario Registro de Servicios de Alojamiento

Registro de servicios de alojamientos ✕

Servicio de alojamiento

Establecimiento

Registrar

Registro de servicios de alojamientos ✕

Servicio de alojamiento

Establecimiento

Registrar

Nota. Prototipo del formulario para registrar los alojamientos.

La figura 43, ilustra el prototipo HTML5 del área de trabajo “Establecimiento”.

Figura 43

Prototipo del Área de Trabajo “Establecimiento”

patriciaPopayan (Admin) | Gestor de Alojamientos

Datos del establecimiento

Nombre: NIT:

Ciudad: Dirección:

Telefono: Tipo de establecimiento:

Descripción:

[Registrar](#)

Nota. Prototipo del área de trabajo datos del establecimiento

La figura 44 ilustra, el prototipo HTML5 del área de trabajo “Tarifas”

Figura 44

Prototipo del Área de Trabajo “Tarifas”

patriciaPopayan (Admin) | Gestor de Alojamientos

Tarifas de habitaciones

Tarifa	Cama1	Cama2	Cama3	Cama4	Cama5	Cama6	Cama7	Cama8	Cama9	Cama10	\$ Valor	Acciones
I	1	0	0	0	0	0	0	0	0	0	\$50000	✎ ✖
II	2	0	0	0	0	0	0	0	0	0	\$90000	✎ ✖
III	1	1	1	0	0	0	0	0	0	0	\$140000	✎ ✖
4 personas	1	2	1	0	0	0	0	0	0	0	\$200000	✎ ✖
Familiar	2	2	1	0	0	0	0	0	0	0	\$240000	✎ ✖
IIII												

Tarifas para parqueaderos

Tarifa	\$ Valor	Acciones
Noche	\$10000.00	✎ ✖
Por dias	\$10000.00	✎ ✖
Por horas	\$2000.00	✎ ✖
Motos	\$10000.00	✎ ✖
Carros	\$11000.00	✎ ✖

Nota. Prototipo del área de trabajo tarifas.

La figura 45 ilustra, el prototipo HTML5 del formulario de registro de tarifas de habitaciones.

Figura 45

Prototipo del Formulario Registro de Tarifas de Habitaciones

Nombre tarifa	Cama 1	Cama 2	Cama 3	Cama 4	Cama 5
<input type="text" value="Nombre"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>
Cama 6	Cama 7	Cama 8	Cama 8	Cama 10	Total Tarifa
<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Cantidad"/>	<input type="text" value="Total"/>

Nota. Prototipo del formulario para registrar las tarifas

La figura 46 ilustra el prototipo HTML5 del área de trabajo “Reporte TRA”

Figura 46

Prototipo del Área de Trabajo “Reporte TRA”

patriciaPopayan (Admin) Gestor de Alojamientos

Generar Reporte TRA Seleccionar Mes: ----- de ---

Show 10 entries Search:

Reserva ↕	Acomodación ↕	Tarifa ↕	Huésped Principal ↕	Acompañantes ↕	Check_in ↕	Check_out ↕	Estado ↕	Valor Total ↕

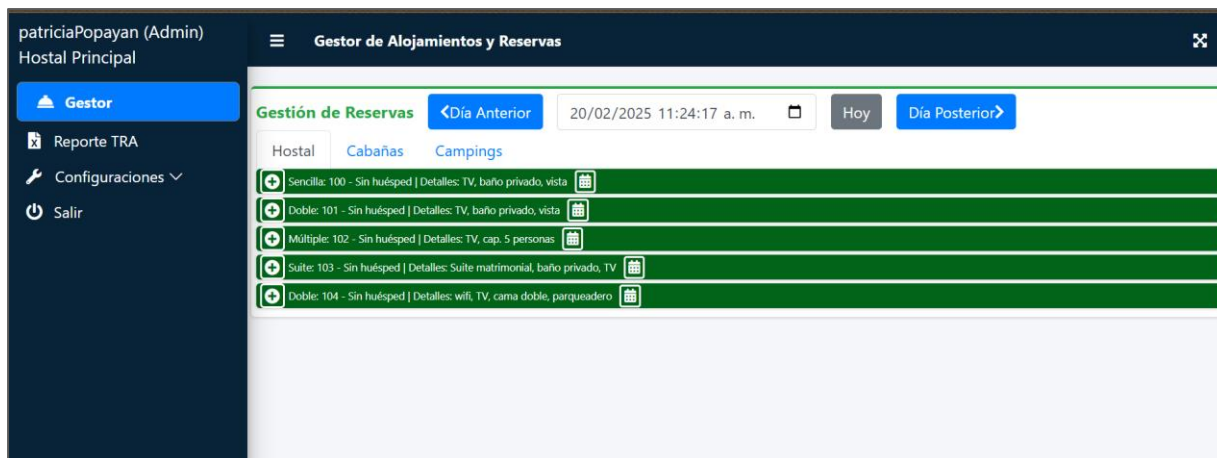
Nota. Prototipo del área de trabajo reporte tra.

La figura 47 ilustra el prototipo HTML5 del área de trabajo “Gestión de Reservas”, desde la cual se podrá navegar por reservas futuras y por las pestañas de los tipos de alojamiento y sus habitaciones correspondientes.

Las tarjetas de color verde identifican las habitaciones o que no tienen reserva activa y en color rojo las que tienen reserva con su respectiva información, evitando mostrar datos sensibles de los huéspedes como números de documento, como lo determina el criterio de seguridad. En la parte izquierda se encuentra el botón para abrir el formulario de registro reserva huésped principal, en el centro el botón para el formulario de registro de acompañantes, al lado derecho los botones bien ubicados según la WCAG para editar reserva, finalizar reserva, cambio de habitación y disponibilidad de habitaciones por mes, además, contraste de color texto oscuro sobre fondo blanco y viceversa, y colores consistentes.

Figura 47

Prototipo del Área de Trabajo “Gestión de Reservas”



Nota. Prototipo del área de trabajo gestión de reservas.

La figura 48 ilustra, el prototipo HTML5 del formulario para registro de reserva.

Figura 48

Prototipo del Formulario Registro de Reservas

Registrar reserva para huésped principal

Reserva: 0037 | Habitación: 100 - Sencilla | Tarifa: I - \$50000 | A/hantes: 0 | Tipo de Identificación: Tipo de identificación | No. Identificación: Escriba el número

Nombres: Escriba el nombre | Apellidos: Escriba los apellidos | Teléfono: Escriba el teléfono | Empresa: Escriba el empresa | Ciudad de Residencia: Escriba la ciudad | Ciudad de Procedencia: Escriba la ciudad

Tarifa de parqueadero

Noche	Por días	Por horas	Motos	Carros	Valor Parqueadero
0	0	0	0	0	\$ 0

Motivo del viaje: Motivo del viaje | Check-in: dd/mm/aaaa --:-- ----- | Check-out: dd/mm/aaaa --:-- ----- | Estado: Estado de la reserva | Valor Total: \$

Registrar

Activar Windows
Ve a Configuración para activar Windo

Nota. Prototipo del formulario para registrar reserva.

La figura 49 ilustra, el prototipo en HTML5 del formulario para registro de reserva y sus respectivos campos.

Figura 49

Prototipo del Formulario de Registro Acompañante

Registro de acompañante

Tipo de Acomodación: 100 - Sencilla | Huésped Principal: Juan Carlos Aristizabal Marquez | Tipo de Identificación: Tipo de identificación | Número de identificación: Número

Nombres: Ingresar nombres | Apellidos: Ingresar apellidos | Telefono: Telefono | Ciudad de residencia: Ingresar ciudad de residencia

Ciudad de procedencia: Ingresar ciudad de procedencia | Check_in: dd/mm/aaaa --:-- ----- | Check_out: dd/mm/aaaa --:-- ----- | Estado: Estado de la reserva

Registrar

Nota. Prototipo del formulario para registrar acompañante.

La figura 50 ilustra el prototipo HTML5 del formulario para finalizar reserva, donde se podrá ver y seleccionar los huéspedes que harán check-out a la reserva haciendo clic al botón finalizar reserva.

Figura 50

Prototipo del Formulario Finalizar Reserva



Finalizar Reserva

Huésped Principal

Juan Carlos Aristizabal Marquez

Acompañantes

Lucia Daniela Pardo Flores

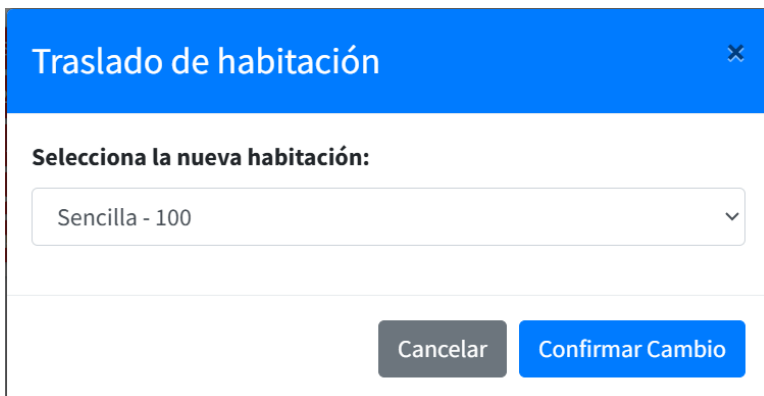
Cancelar Finalizar Reserva

Nota. Prototipo del formulario para finalizar reserva

La figura 51 ilustra, el prototipo HTML5 del formulario para realizar cambio de habitación.

Figura 51

Prototipo del Formulario Cambio de Habitación



Traslado de habitación

Selecciona la nueva habitación:

Sencilla - 100

Cancelar Confirmar Cambio

Nota. Prototipo del formulario para cambiar habitación.

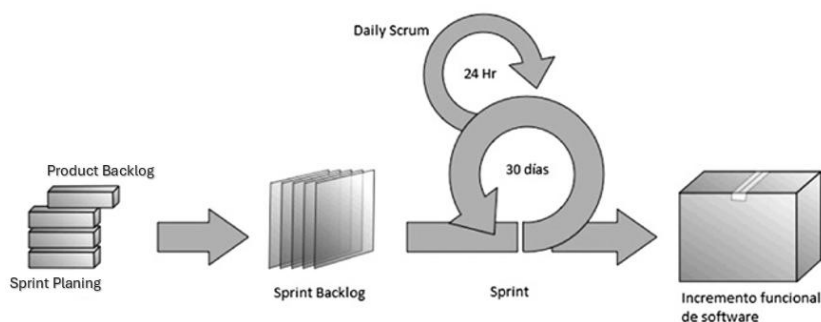
Fase 3 - Implementación

La implementación o construcción del sistema web se refiere a la escritura de código utilizando lenguajes de programación, en este caso PHP y JavaScript, para convertir los requerimientos de descripción general y funcionales, y el prototipo previamente diseñado en la fase 2 en un sistema web funcional. Aplicando el marco Scrum de la metodología Agile, con el fin de aprovechar al máximo el tiempo con que se cuenta para desarrollar el código, se buscó la documentación con este enfoque, que ha sido probado y aplicado por diversos autores, como lo son Juan Palacio en el año 2008 con su publicación “Flexibilidad con Scrum” y Manuel Trigas en el año 2012 y su publicación “Metodología de Scrum”, Ken Schwaber y Jeff Sutherland con su publicación “La guía definitiva de Scrum: Las Reglas del Juego” del año 2020, tomando como referente la documentación de este último autor para ser aplicada a la construcción del sistema, por la completa y detallada explicación del proceso.

Los elementos ilustrados en la figura 52, son la información para detallar las acciones y tareas necesarias del método scrum a fin de lograr el objetivo o requerimientos del usuario final.

Figura 52

Elementos de Scrum



Nota: Adaptado de Metodología Scrum: Fases del Sprint de Andrés Navarro, 2013,

https://www.researchgate.net/figure/Figura-1-Metodologia-Scrum-Fases-de-un-Sprint-27-Figure-1-Scrum-Methodology-Sprint_fig1_273302003, CC BY 2.0

Estos son: Product Backlog contiene las tareas y funcionalidades a desarrollar, Sprint Backlog detalla las tareas que se desarrollaran en el sprint en curso, incremento es la funcionalidad finalizada y totalmente operativa.

Implementación del Código para el Sistema

Teniendo el prototipo diseñado del sistema web gestor de alojamientos en la etapa anterior, se procede a definir la revisión de planes de versión o planificación del sprint:

Planificación de los Sprints

Agrupando los requerimientos funcionales: listar, registrar, actualizar, eliminar, finalizar y buscar para las secciones usuarios, tipos de alojamientos y acomodaciones, tarifas, reservas, establecimiento y generar la TRA, se obtienen las 5 historias de usuario a realizar en la tabla 9 con su ID, requerimiento o historia, prioridad, tiempo estimado y fecha de inicio:

Tabla 9

Product Backlog

Id	Historia	Prioridad	Tiempo	Fecha de inicio
1	Desarrollar la arquitectura base, funcionalidades de autenticación y gestión del sistema	Alta	5 días	25/02/2025
2	Desarrollo de módulos críticos para gestión del sistema	Alta	5 días	04/03/2025
3	Desarrollo de la gestión, validación y carga dinámica de reservas	Alta	5 días	11/03/2025
4	Desarrollo de funcionalidades para gestionar check-out y generación de la TRA	Alta	5 días	18/03/2025
5	Desarrollo de funcionalidades para la optimización del sistema	Alta	5 días	25/03/2025

Nota. Historias de usuario para realizar divididas en 5 sprint.

Primer Sprint: Arquitectura y Autenticación

Para empezar con la implementación del código de la primera historia del product backlog “Desarrollar la arquitectura base, funcionalidades de autenticación y gestión del sistema”, la tabla 10 representa la división de esta, en tareas más puntuales para realizar cada día de la semana:

Tabla 10

Spring Backlog – Primer Sprint

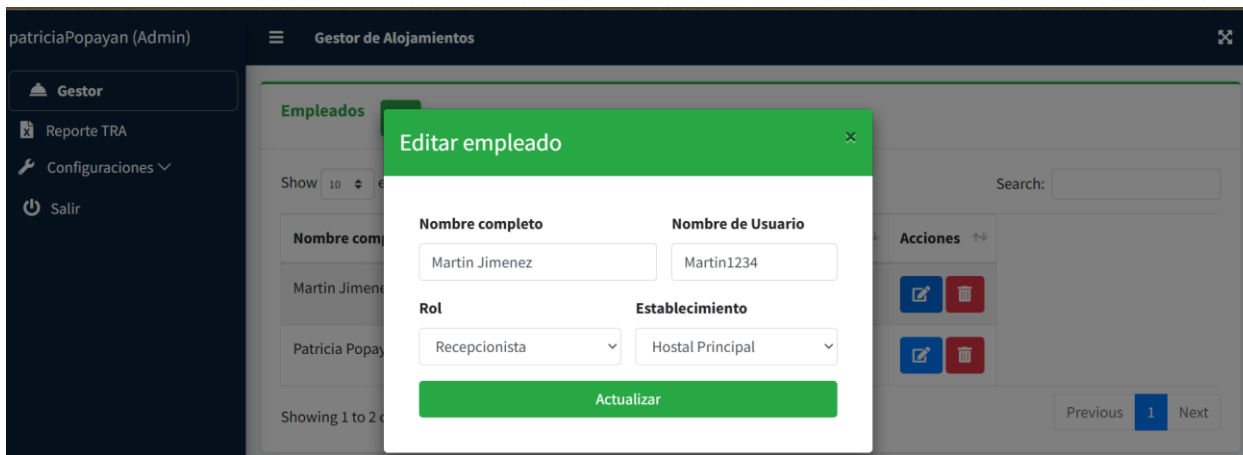
Id	Tarea	Descripción	Responsable	Prioridad	Tiempo	Estado
1.1	Crear la estructura base del proyecto MVC	Configuración de carpetas y archivos base en PHP	Patricia	Alta	5 horas	Terminado
1.2	Implementar CRUD establecimiento	Registrar, listar, actualizar el establecimiento en BD.	Patricia	Alta	4 horas	Terminado
1.3	Implementar CRUD usuarios	Registrar, listar, actualizar y eliminar los usuarios de la BD.	Patricia	Alta	6 horas	Terminado
1.4	Crear inicio de sesión	Mantener inicio de sesión activo y cierre de sesión	Patricia	Alta	5 horas	Terminado
1.5	Implementar CRUD Alojamientos	Registrar, listar, actualizar y eliminar alojamientos de la base de datos	Patricia	Alta	5 horas	Terminado

Nota. Tareas para realizar en el primer sprint.

La figura 53 presenta en acción el formulario de edición de usuario empleado.

Figura 53

Edición de Formulario Usuario Empleado

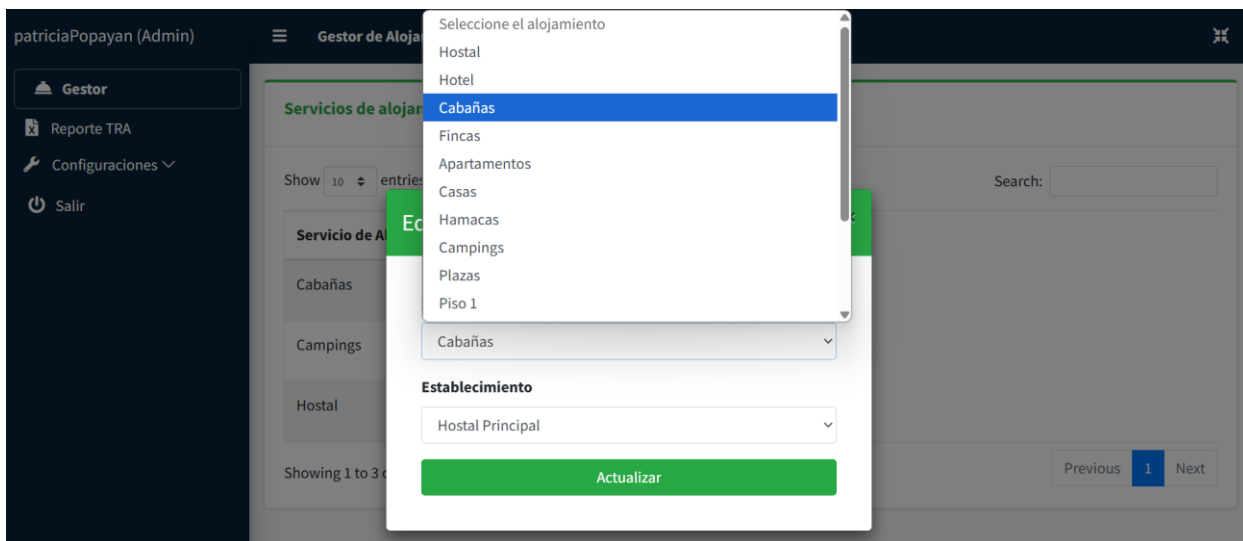


Nota. Actualizar los empleados.

La figura 54 ilustra en acción el formulario de edición de tipos de alojamientos.

Figura 54

Edición de Formulario Tipos de Alojamientos



Nota. Actualizar alojamientos.

Sprint Review - Primer Sprint

Durante el Sprint se logra completar todas las tareas acordadas:

La configuración completa de la arquitectura MVC en el sistema.

El sistema permite el inicio y cierre de sesión correctamente, con sus respectivas validaciones, mensajes de alerta o confirmación para los usuarios.

Los módulos creados están completamente funcionales.

Los datos registrados se guardan perfectamente en la base de datos y se muestran en las diferentes interfaces.

Las actualizaciones y eliminaciones se realizan sin errores.

Tiempo estimado: 1 semana (5 días, 25 horas).

Estado del Sprint: Finalizado.

Segundo Sprint: Módulos Críticos del Sistema

Para empezar con la codificación de la segunda historia del product backlog “Desarrollo de módulos críticos para gestión del sistema”, la tabla 11 representa la división de esta, en tareas más puntuales para realizar cada día de la semana:

Tabla 11

Spring Backlog – Segundo Sprint

Id	Tarea	Descripción	Responsable	Prioridad	Tiempo	Estado
2.1	Implementar CRUD acomodaciones	Registrar, listar, actualizar y eliminar acomodaciones de la base de datos	Patricia	Alta	5 horas	Terminado
2.2	Implementar CRUD tarifas de alojamiento	Registrar, listar, actualizar y eliminar tarifas de alojamiento la base de datos	Patricia	Alta	5 horas	Terminado

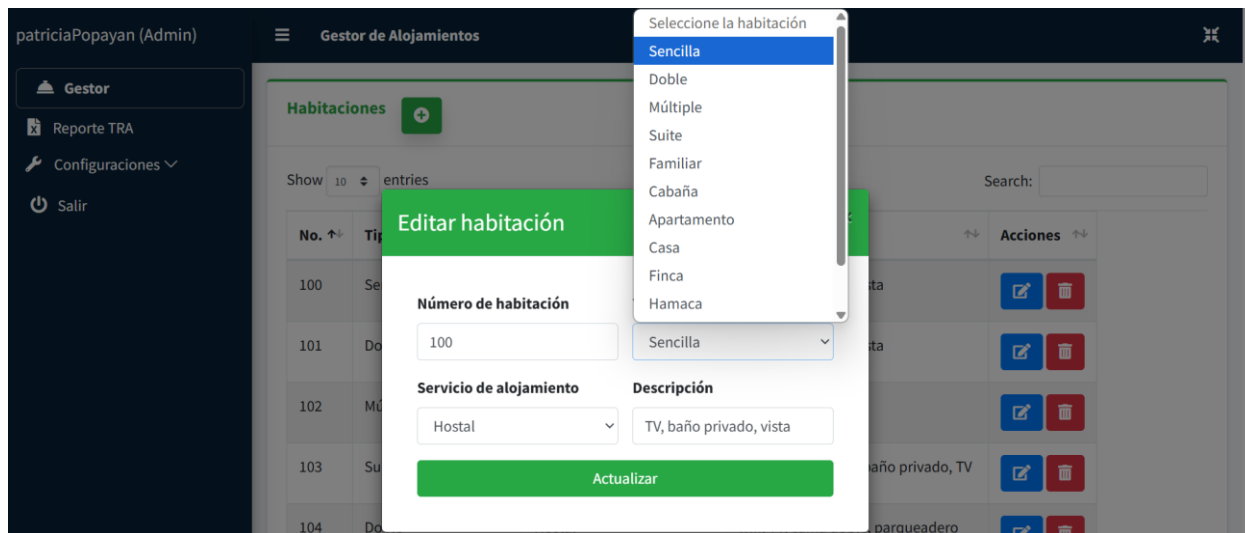
2.3	Implementar CRUD tarifas de parqueadero	Registrar, listar, actualizar y eliminar tarifas de parqueaderos la base de datos	Patricia	Media	5 horas	Terminado
2.4	Implementar carga dinámica de datos en la vista recepción	Codificar la carga dinámica de pestañas de navegación para obtener alojamientos, acomodaciones	Patricia	Media	10 horas	Terminado

Nota. Tareas para realizar en el segundo sprint.

La figura 55 se ilustra en acción el formulario de edición de tipos de alojamientos con los datos correspondientes.

Figura 55

Edición de Formulario Tipo de Habitación para Alojamientos



Nota. Editar habitación.

La figura 56 ilustra en acción el formulario para editar tarifas.

Figura 56

Edición de Formulario Tarifas

patriciaPopayan (Admin) | Gestor de Alojamientos

Gestor

- Reporte TRA
- Configuraciones
- Salir

Tarifas de habitaciones

Tarifas para parqueaderos

Editar tarifa para alojamiento

En los campos "cama 1, cama 2, etc" se escribe el número personas que ocupan la cama. Ejm: Cama 1= 2 personas

Tipo de tarifa	Cama 1	Cama 2	Cama 3	Cama 4	Cama 5
4 personas	1	2	1	0	0
Cama 6	Cama 7	Cama 8	Cama 9	Cama 10	Valor tarifa
0	0	0	0	0	200000

Actualizar

Tarifa	\$ Valor	Acciones
tarifa	\$10000.00	[icon]
tarifa	\$10000.00	[icon]
tarifa	\$2000.00	[icon]
tarifa	\$10000.00	[icon]
tarifa	\$11000.00	[icon]

Nota. Edición de las tarifas.

Sprint Review - Segundo Sprint

Durante el Sprint se logra completar todas las tareas acordadas:

Los módulos creados están completamente funcionales.

Los datos registrados se guardan perfectamente en la base de datos y se muestran en las diferentes interfaces.

Las actualizaciones y eliminaciones se realizan sin errores.

La carga dinámica muestra correctamente los alojamientos y las habitaciones en las pestañas de navegación.

Tiempo estimado: 1 semana (5 días, 25 horas).

Estado del Sprint: Finalizado.

Tercer Sprint: Carga Dinámica de Reservas

Para empezar con la codificación de la tercera historia del product backlog “Desarrollo de la gestión y carga dinámica de reservas”, la tabla 12 representa la división de esta, en tareas más puntuales para realizar cada día de la semana:

Tabla 12

Sprint Backlog – Tercer Sprint

Id	Tarea	Descripción	Responsable	Prioridad	Tiempo	Estado
3.1	Implementar carga dinámica en el formulario de registro de reserva	Codificar la carga dinámica con AJAX de datos estándar hacia el formulario de registro de reserva.	Patricia	Alta	5 horas	Terminado
3.2	Implementar CRUD reserva huésped principal	Codificar las funcionalidades registrar, listar y actualizar huésped principal.	Patricia	Alta	10 horas	Terminado
3.3	Implementar el valor de la reserva	Codificar el cálculo entre tarifas de alojamiento y parqueadero para el total de la reserva.	Patricia	Alta	5 horas	Terminado
3.4	Implementar CRUD huésped acompañante	Codificar las funcionalidades registrar, listar y actualizar huésped acompañante.	Patricia	Alta	5 horas	Terminado

Nota. Tareas de tercer sprint.

La figura 57 ilustra en acción el formulario de edición reservas huésped principal.

Figura 57

Edición de la Reserva para Huésped Principal

Nota. Edición de la reserva.

La figura 58 ilustra el formulario de edición reservas huésped acompañante.

Figura 58

Edición del Huésped Acompañante

Nota. Formulario para registrar acompañantes.

Sprint Review - Tercer Sprint

Durante el Sprint se logra completar todas las tareas acordadas:

El formulario de registro de reservas obtiene automáticamente el código de la reserva, el número y la acomodación, y las tarifas del parqueadero.

El sistema permite la gestión de la reserva de huésped principal y acompañantes.

El cálculo del valor total de la reserva se ejecuta sin errores.

Los datos registrados se guardan perfectamente en la base de datos y se muestran en la interfaz.

Tiempo estimado: 1 semana (5 días, 25 horas).

Estado del Sprint: Finalizado.

Cuarto Sprint: Check-out y TRA

Para empezar con la codificación de la cuarta historia del product backlog “Desarrollo de funcionalidades para gestionar check-out y generación de la TRA”, la tabla 13 representa la división de esta, en tareas más puntuales para realizar cada día de la semana:

Tabla 13

Sprint Backlog – Cuarto Sprint

Id	Tarea	Descripción	Responsable	Prioridad	Tiempo	Estado
4.1	Validar número máximo de acompañantes según la tarifa	Implementar validación para restringir la cantidad de registro de acompañantes según la tarifa seleccionada.	Patricia	Media	5 horas	Terminado
4.2	Implementar check-out de huéspedes	Codificar funcionalidad de check-out con opciones según el	Patricia	Alta	10 horas	Terminado

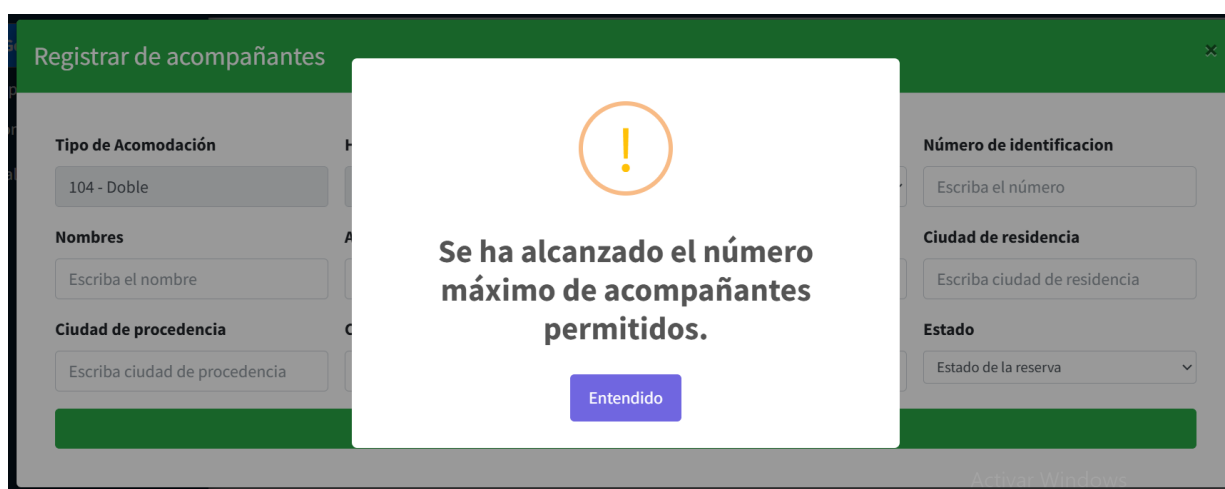
	escenario: todos los huéspedes, todos los acompañantes, algunos acompañantes.				
4.3	Implementar el check-out para huésped principal y su reemplazo	Codificar funcionalidad para check-out para huésped principal y selección de su reemplazo si la reserva continua activa	Patricia	Alta	5 horas Terminado
4.4	Implementar la generación de TRA	Codificar funcionalidad para generación y registro de la TRA	Patricia	Alta	5 horas Terminado

Nota. Tareas del cuarto sprint

La figura 59 ilustra el mensaje de advertencia cuando se llega al límite de acompañantes a registrar.

Figura 59

Validación Registro Limite de Acompañantes

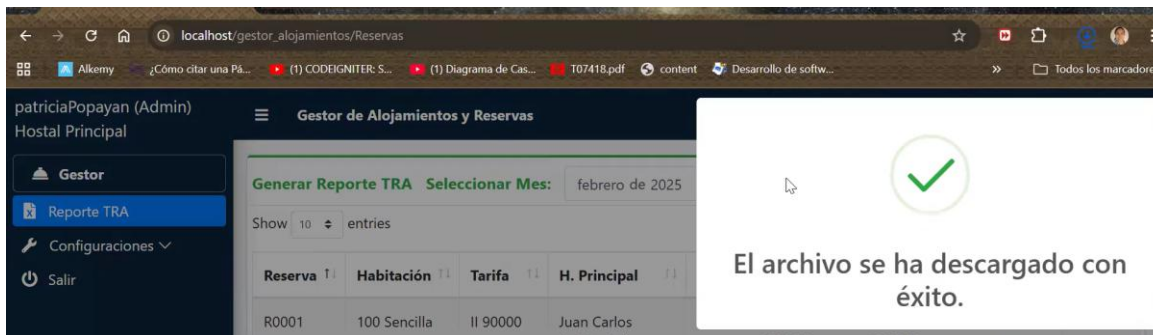


Nota. Alerta para el usuario cuando intenta registrar mas acompañantes de los permitidos.

La figura 60 ilustra la selección del mes y descarga del reporte TRA.

Figura 60

Descarga del Reporte TRA



Nota. Mensaje de éxito cuando se descarga el reporte TRA.

La figura 61 ilustra el reporte TRA generado desde el sistema web por el usuario en formato Excel.

Figura 61

Reporte TRA

The screenshot shows an Excel spreadsheet with the following data:

Tipo de identificación	Número de identificación	Nombres	Apellidos	Ciudad de residencia	Ciudad de procedencia	Número de la habitación	Motivo de viaje	Número de acompañantes
C.C	1104706956	Ariel Enrique	Ceballos Pedraza	Bogotá	Ibagué	100	Salud y atención médica	1
C.C	1105409685	Fernando Lorenzo	Hernandez Fernandez	Bogota	Murillo	100	Tránsito	0
C.C	1104578694	Francisca Lucia	Rendon Bayona	Murillo	Libano	100	Visitas a familiares y a amigos	0
C.C	45856942	Francisco Alberto	Hernandez Herrera	Popayan	Popayan	100	Salud y atención médica	0
C.C	1103450694	Juan Manuel	Medrano Tique	Murillo	Murillo	101	Compras	0
C.C	54586990	Hector daniel	lopez Iizarazo	medellin	medellin	101	Visitas a familiares y a amigos	0
C.C	1104706589	Ana Berta	Restrepo Sanchez	Mariquita	Mariquita	102	Vacaciones, recreo y ocio	1
C.C	1103459384	Josefina	Flores Valbuena	Cali	Cali	103	Visitas a familiares y a amigos	2
Pasaporte	22949382	Francisco Javier	Bedoya De la Torre	Mexico D. F	Mexico D. F	103	Negocios y motivos profesionales	1
C.C	93457732	Deyanira	Lopez Camargo	Ibagué	Medellin	201	Vacaciones, recreo y ocio	2
C.C	1103457894	Yadira Andrea	Camargo Gil	Bogota	Neiva	104	Compras	1
C.C	1104394856	Cristian Danilo	Romero Farfan	Bogotá	Bogotá	104	Negocios y motivos profesionales	1
C.C	64738923	Maria Magdalena	Fernandez Hernandez	Medellin	Medellin	202	Vacaciones, recreo y ocio	4

Nota. Reporte TRA en formato Excel.

Sprint Review - Cuarto Sprint

Durante el Sprint se logra completar todas las tareas acordadas:

Se validan correctamente la limitación de numero de acompañantes.

El check-out a los huéspedes se aplica sin errores para sus respectivos escenarios si afectar otras reservas.

El check-out y reemplazo de huésped principal se ejecuta correctamente;

La generación y descarga de la TRA se realiza correctamente.

Tiempo estimado: 1 semana (5 días, 25 horas).

Estado del Sprint: Finalizado.

Quinto Sprint: Optimización del Sistema

Para empezar con la codificación de la quinta tarea del product backlog “Desarrollo de funcionalidades para la optimización del sistema”, la tabla 14 representa la división de esta, en tareas más puntuales para realizar cada día de la semana:

Tabla 14

Sprint Backlog – Quinto Sprint

Id	Tarea	Descripción	Responsable	Prioridad	Tiempo	Estado
5.1	Implementar el autocompletado del formulario reservas huésped principal y acompañantes.	Codificar autocompletado para los formularios de registro de huéspedes buscando por número de documento.	Patricia	Media	5 horas	Terminado
5.2	Implementar filtro de búsqueda de reservas y	Codificar la búsqueda de reservas por fecha: día anterior, posterior, actual y por mes	Patricia	Alta	5 horas	Terminado

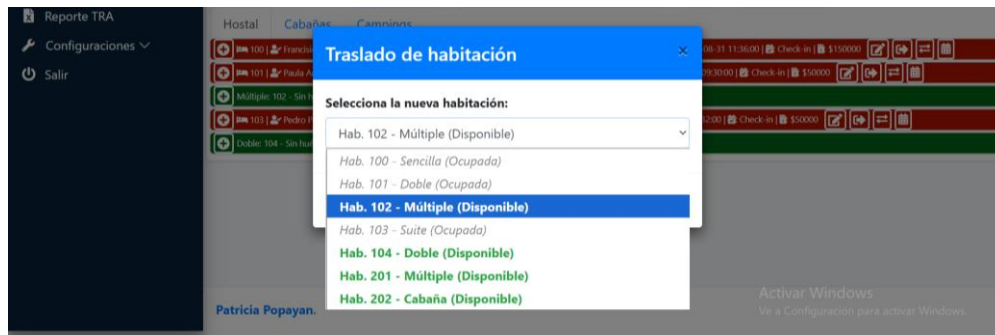
	disponibilidad de habitaciones.					
5.3	Implementar función de cambio de alojamiento.	Desarrollar la funcionalidad para cambio de habitación de reserva activa, manteniendo la información de los huéspedes.	Patricia	Media	5 horas	Terminado
5.4	Implementar restricción de disponibilidad de fechas en reservas.	Codificar la funcionalidad para evitar el registro de reservas en fechas ocupadas.	Patricia	Alta	5 horas	Terminado
5.5	Revisión de código	Verificación de la integridad del sistema.	Patricia	Alta	5 horas	Terminado

Nota. Autoría propia

La figura 62 ilustra el formulario en acción con opciones de disponibilidad para cambiar la reserva de habitación.

Figura 62

Cambiar Reservas de Habitación

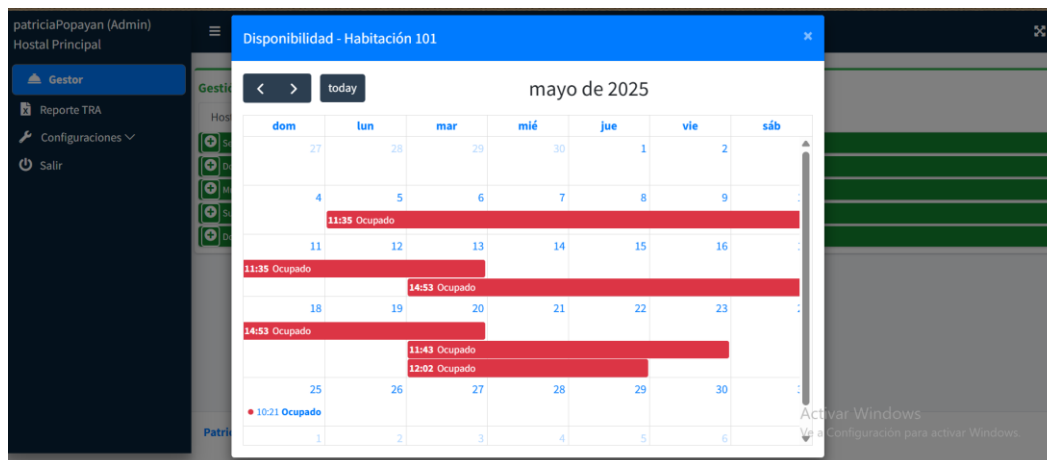


Nota. Formulario con las opciones de habitación disponible para cambio de habitación.

La figura 63 ilustra el formulario en acción para ver la disponibilidad de habitación por fechas.

Figura 63

Disponibilidad de Habitaciones



Nota. Formulario que muestra la ocupación de las habitaciones por mes

Sprint Review - Quinto Sprint

Terminada la semana del quinto Sprint se evalúa el cumplimiento de cada tarea.

Durante el Sprint se logra completar todas las tareas acordadas:

El autocompletado de los formularios de registro de huéspedes se funciona correctamente.

Los filtros de búsqueda permiten encontrar reservas fácilmente.

El cambio de alojamiento se realiza sin perder los datos de la reserva.

Los filtros de búsqueda permiten encontrar reservas fácilmente.

La validación para evitar reservas en fechas ocupadas funciona sin errores.

Tiempo estimado: 1 semana (5 días, 25 horas).

Estado del Sprint: Finalizado.

Fase 4 - Pruebas

La fase de pruebas representa un componente esencial en el ciclo de desarrollo de software. Según Sommerville (2011) “las pruebas intentan demostrar que un programa hace lo que se intenta que haga, así como descubrir defectos en el programa antes de usarlo” (p. 206) permitiendo confirmar que el sistema opera de forma adecuada, respondiendo a las especificaciones planteadas y que no presenta fallos que afecten su funcionamiento. Las pruebas que se implementarán al sistema web gestor de alojamientos se definen y comparan en la tabla 15.

Tabla 15

Tipos de Pruebas para Software

Características	Pruebas unitarias	Pruebas de integración	Pruebas piloto
Enfoque	Funciones, métodos	Varios módulos o componentes	Sistema completo
Propósito	Validar lógica interna	Verificar interacción entre módulos	Simular uso real del sistema
Alcance	Una unidad de código	Varias unidades colaborando	Todo el sistema (frontend, backend, base de datos)
Uso de base de datos	No	Si (real o simulada)	Si (entorno real)
Interfaz de usuario	No	No	Si
Velocidad de ejecución	Alta	Media	Baja
Ejemplo típico	Probar que sumar (2,3) devuelve 5	Probar que UsuarioModel::crear() guarda datos con Query	Simular que el usuario se registra en la web y recibe confirmación

Nota. Tipos de pruebas de software que se realizarán al sistema web.

Para fines del presente proyecto en su proceso de pruebas unitarias y de integración se hace uso del framework PHPUnit, herramienta ideal y popular en sistemas creados en lenguaje PHP y arquitectura MVC. Funciona creando una clase de prueba donde se crean métodos, usando aserciones (assertEqual, assertTrue, assertCount) que sirven para comparar, se ejecutan las pruebas desde la terminal desde donde también se ve el resultado.

Las pruebas se crearán para: archivos de estructura del proyecto (conexión, Query, Views, Controller), controlador y modelo “Recepciones” (métodos de obtención de datos, CRUD y check-out de reservas) que es donde se maneja la lógica más compleja y crítica del sistema web.

Por consecuencia, se omite las pruebas para los módulos (usuarios, alojamientos, acomodaciones y tarifas) porque las funciones de obtención de datos y CRUD reutilizan la misma estructura del modelo y controlador “Recepciones”, lo que significa que su comportamiento se valida indirectamente al probar exhaustivamente dicho módulo. Esta práctica está alineada con el principio de cobertura eficaz en pruebas de software, priorizando el código más crítico y propenso a errores.

Pruebas Unitarias

Las pruebas unitarias “son el proceso de probar componentes del programa, como métodos o clases de objetos” (Sommerville, 2011, p. 211). Estas las puede realizar el desarrollador de forma manual creando el código y revisado el resultado por sí mismo, proceso óptimo para proyectos pequeños. Otra manera es automatizando el proceso por framework (PHPUnit) donde se simulan los datos, clases, métodos, objetos, etc., permitiendo resultados en menor tiempo comparado con las pruebas manuales. En consecuencia, se aplicará un test por método público con más de un escenario en un solo archivo.

Prueba Unitaria Estructura del Proyecto

En la tabla 16 se detalla el conjunto de pasos que se ejecutan para probar los métodos en la prueba unitaria para los archivos que corresponden a la estructura del proyecto.

Tabla 16

Resultados de Pruebas Unitarias Archivos de Estructura del Proyecto

Id	Descripción	Entrada	Resultado Obtenido	Estado
CP1	Probar que select() retorna un único registro correctamente	SQL: SELECT * FROM tabla. Parámetros: []	Se ejecutó la consulta con éxito, retornando un array asociativo con una sola fila: ['id' => 1, 'nombre' => Luis']	Aprobado
CP2	Probar que selectAll() retorna múltiples registros	SQL: SELECT * FROM tabla. Parámetros: []	El método devolvió una lista de arrays con registros múltiples; verificado count(\$resultado) >= 2	Aprobado
CP3	Probar que save() inserta datos correctamente	SQL: INSERT INTO tabla (...). Parámetros: ['Juan', 'Correo']	Se insertó el registro correctamente. El método retornó 1, indicando que una fila fue afectada en la base de datos.	Aprobado
CP4	Probar que delete() elimina el registro correctamente	SQL: DELETE FROM tabla WHERE id = ?. Parámetros: [1]	El método ejecutó correctamente la sentencia y retornó 1, indicando que una fila fue eliminada	Aprobado
CP5	Probar que conect() retorna un objeto PDO	Ninguna	Se verificó que el método retorna una instancia válida de PDO mediante instanceof	Aprobado
CP6	Probar que conect() no retorna un valor nulo	Ninguna	El método no retornó null. Se obtuvo una conexión activa con la base de datos	Aprobado
CP7	Probar que la vista del controlador Home se carga correctamente	Vista: "inicio"	Se cargó correctamente la vista inicio.php, mostrando el contenido esperado dentro de la estructura HTML del sistema	Aprobado
CP8	Probar carga de la vista listado desde	Vista: "listado"	El controlador Usuario invocó la carga de la vista listado.php y el contenido	Aprobado

Id	Descripción	Entrada	Resultado Obtenido	Estado
	el controlador Usuario		renderizado contiene "Vista de Usuario"	
CP9	Probar que model() carga correctamente el modelo	Modelo: "FakeControllerModel"	Se instanció correctamente la clase del modelo solicitado sin errores. Confirmado por el retorno del mensaje "modelo cargado"	Aprobado
CP10	Probar que verificarRol() permite acceso con rol autorizado	Rol: 'admin'	La sesión activa pertenece al rol autorizado. No se produjo redirección ni errores. Acceso permitido	Aprobado
CP11	Probar que verificarRol() niega acceso con rol no autorizado	Rol: 'admin' con sesión 'user'	El método arrojó correctamente un mensaje de error "No tienes permiso para acceder a esta sección.", impidiendo acceso no autorizado	Aprobado

Nota. Resultado y prueba unitaria para los archivos de la estructura del proyecto.

Prueba Unitaria RecepcionesModel

En la tabla 17 se detalla el conjunto de pasos que se ejecutan para probar los métodos en las pruebas unitarias que corresponden al Modelo Recepción.

Tabla 17

Resultados de Pruebas Unitarias RecepcionesModel

Id	Descripción	Entrada	Resultado Obtenido	Estado
CP1	Probar obtención de tarifa parqueadero	ID parqueadero: 1	5000	Aprobado
CP2	Probar obtención de tarifa de alojamientos	ID tarifa: 1	100000	Aprobado
CP3	Probar obtención de tarifas múltiples de parqueadero	IDs: [1,2]	4000 y 6000	Aprobado
CP4	Calcular número de acompañantes por reserva	ID reserva: 1	2	Aprobado

CP5	Calcular total de camas activadas	ID tarifa: 1	3	Aprobado
CP6	Calcular ocupación total de camas	ID tarifa: 1	3	Aprobado
CP7	Calcular disponibilidad de camas	ID tarifa: 1	0	Aprobado
CP8	Obtener establecimientos disponibles	-	[Hotel Murillo, Cabañas Nevado]	Aprobado
CP9	Obtener alojamientos por establecimiento	-	[Alojamiento 1 - Hotel Murillo, Alojamiento 2 - Cabañas]	Aprobado
CP10	Obtener habitaciones registradas	-	[101 - Alojamiento 1, 102 - Alojamiento 2]	Aprobado
CP11	Obtener parqueaderos disponibles	-	[Moto - 2000, Carro - 5000]	Aprobado
CP12	Obtener reservas principales registradas	-	[{codigo_reserva: RVA001, estado: activo}]	Aprobado
CP13	Registrar una nueva reserva principal	Datos simulados	OK	Aprobado
CP14	Modificar una reserva principal existente	Datos actualizados	"modificado"	Aprobado
CP15	Editar datos de una reserva por ID	ID principal: 1	{nombres: "Juan", codigo_reserva: "RVA001"}	Aprobado
CP16	Obtener huéspedes activos	ID principal: 1	[{nombre: "Carlos", estado: "Check-in"}]	Aprobado
CP17	Obtener acompañantes por ID	ID acompañante: 1	{nombre: "Carlos"}	Aprobado
CP18	Obtener acompañantes activos por ID	ID principal: 1	[{nombre: "Carlos", estado: "Check-in"}]	Aprobado
CP19	Finalizar reserva principal	ID principal: 1	true	Aprobado
CP20	Actualizar estado de principal y acompañantes	ID: 1, estado: Check-out	true	Aprobado
CP21	Finalizar reserva estableciendo check-out	ID principal: 1, fecha: 2025-04-23	1	Aprobado
CP22	Obtener reserva principal por ID	ID: 1	150000	Aprobado
CP23	Obtener último ID por código reserva	Código: RVA001	1	Aprobado

CP24	Duplicar acompañantes a nueva reserva	ID nueva reserva y datos acompañantes	true	Aprobado
------	---------------------------------------	---------------------------------------	------	----------

Nota. Resultado de la prueba unitaria para archivo RecepcionesModel.

Pruebas Unitaria Controlador Recepción

En la tabla 18 se detalla el conjunto de pasos que se ejecutan para probar los métodos en las pruebas unitarias que corresponden al Controlador Recepciones.

Tabla 18

Resultados de Pruebas Unitarias Controlador Recepción

Id	Descripción	Entrada	Resultado Obtenido	Estado
CP1	Probar obtención de tarifa de parqueadero	ID parqueadero: 1	Método retornó 5000 correctamente	Aprobado
CP2	Probar obtención de tarifa de alojamiento	ID tarifa: 1	Método retornó 100000 correctamente	Aprobado
CP3	Validar que `registrar()` retorne error si falta campo obligatorio	JSON sin campo `nombre`	Mensaje de error recibido: "El campo nombre es obligatorio"	Aprobado
CP4	Validar que `registrar()` registre correctamente una reserva	JSON completo con campos válidos	Salida JSON: {"success": "Si", ...}	Aprobado
CP5	Validar que `registrar()` asigne 0 parqueaderos si `tarifas_parqueadero` es nulo	JSON sin campo `tarifas_parqueadero`	Campo cantidad_vehiculos registrado como {}	Aprobado
CP6	Validar que `listar()` devuelva un array con reservas activas	Ninguna	Método devolvió un array con 2 reservas activas simuladas	Aprobado
CP7	Validar que `obtenerCodigoReserva()` retorne código correctamente	Parámetros: ID cliente, fecha	Código retornado: "RVA-20250506"	Aprobado
CP8	Validar que `calcularNumeroAcompañantes()` cuente acompañantes activos	ID principal: 10	Método retornó: 3	Aprobado
CP9	Validar que `obtenerAcompañante()` devuelva los datos del acompañante	ID acompañante: 5	Método retornó datos simulados del acompañante (nombre, apellido, estado...)	Aprobado

CP10	Validar que `editar()` actualice correctamente la información	JSON con datos modificados e ID existente	Método devolvió los datos modificados correctamente en salida JSON	Aprobado
------	---	---	--	----------

Nota. Resultado de la prueba unitaria para archivo Controlador Recepciones.

Pruebas de Integración

Estas pruebas buscan verificar la correcta interacción y comunicación entre los módulos o componentes del sistema con servicios externos (bases de datos, APIS de terceros, etc). Para evitar alteraciones en los registros reales, asegurar rapidez, simplicidad e independencia se utilizará la simulación de base de datos SQLite (en memoria) permitiendo crear datos de cero para cada prueba detectando errores reales y no errores falsos causados por datos residuales o inconsistentes. Los datos son temporales y solo existen mientras se ejecuta la prueba.

Pruebas de Integración Estructura del Proyecto

En la tabla 19 se detalla las pruebas de integración que corresponden al archivo Query.

Tabla 19

Resultados de Pruebas de Integración Archivo Query

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
CP1	Insertar un usuario en la base de datos	SQL: INSERT INTO usuarios (nombre, apellido) VALUES (?, ?)Parámetros: ['Juan', 'Méndez']	SQLite (PDO)	Retorna 1. Usuario 'Juan Méndez' insertado correctamente.	Aprobado
CP2	Seleccionar un usuario por nombre	SQL: SELECT * FROM usuarios WHERE nombre = ?Parámetros: ['María']	SQLite (PDO)	Se retorna arreglo correcto con datos de María Torres.	Aprobado
CP3	Seleccionar todos los usuarios	SQL: SELECT * FROM usuarios	SQLite (PDO)	Se retornaron los registros de Ana y Luis.	Aprobado
CP4	Eliminar un usuario por nombre	SQL: DELETE FROM usuarios WHERE nombre = ?Parámetros: ['Pedro']	SQLite (PDO)	Retorna 1. Verificado que Pedro fue eliminado.	Aprobado

Nota. Resultado de la prueba de integración para el archivo Query de la estructura del proyecto.

Pruebas de Integración RecepcionesModel

En la tabla 20 se detalla el conjunto de pasos que se ejecutan para probar los métodos en las pruebas de integración que corresponden al Modelo Recepción.

Tabla 20

Resultados de Prueba de Integración RecepcionesModel

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
CP1	Insertar una reserva en la tabla recepciones	Método: save('recepciones', [...]) Datos: reserva con campos codigo_reserva, fecha_ingreso, estado, etc.	SQLite (PDO)	Se insertó una fila en recepciones y el método devolvió 1	Aprobado
CP2	Finalizar una reserva actualizando el estado a FINALIZADO en recepciones y acompañantes	Método: actualizarEstadoReservaYacompanantes(\$id_principal) Dato: ID de reserva principal activa	SQLite (PDO)	Estados de la reserva y acompañantes actualizados a FINALIZADO correctamente	Aprobado
CP3	Obtener la reserva principal activa por ID	Método: getRvaPrincipalById(\$id) Dato: ID existente de reserva principal	SQLite (PDO)	Se obtuvo un array con los datos completos del registro según el ID	Aprobado
CP4	Obtener el último ID principal por código de reserva	Método: obtenerUltimoIdPrincipalPorCodigoReserva(\$codigo) Dato: código de reserva existente	SQLite (PDO)	Se retornó el ID esperado con valor máximo asociado al código	Aprobado
CP5	Finalizar la reserva principal con fecha y hora	Método: finalizarReservaPrincipal(\$id_principal) Dato:	SQLite (PDO)	Campos de salida actualizados correctamente con fecha y hora actuales	Aprobado

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
		ID de reserva principal activa			
CP6	Verificar disponibilidad cuando no hay conflictos	Método: verificarDisponibilidad (\$id_habitacion, \$fecha_ingreso, \$fecha_salida) Datos: habitación libre en ese rango	SQLite (PDO)	Método devolvió true, sin registros conflictivos en ese periodo	Aprobado
CP7	Verificar disponibilidad al actualizar una reserva	Método: verificarDisponibilidad AlActualizar(\$id_habitacion, \$fecha_ingreso, \$fecha_salida, \$id_excluir) Datos: rango válido	SQLite (PDO)	Se verificó correctamente, sin interferencias por otras reservas	Aprobado
CP8	Duplicar acompañantes activos en una nueva reserva	Método: duplicarAcompañantes EnNuevaReserva(\$id_origen, \$id_destino) IDs válidos de reservas origen y destino	SQLite (PDO)	Nuevos registros insertados correctamente en destino	Aprobado
CP9	Obtener acompañantes activos de una reserva	Método: obtenerAcompañantes Activos(\$id_principal) Dato: ID de reserva principal	SQLite (PDO)	Se obtuvo un array con acompañantes activos correctos	Aprobado
CP10	Obtener huéspedes activos de una reserva	Método: obtenerHuespedesActivos(\$id_principal) Dato: ID de reserva principal	SQLite (PDO)	Lista devuelta contiene los huéspedes esperados con estado activo	Aprobado

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
CP11	Obtener acompañantes por ID	Método: obtenerAcompañantesP (PDO) orIds(\$ids) Dato: lista de IDs válidos	SQLite	Datos de acompañantes consultados correctamente por IDs	Aprobado
CP12	Finalizar reserva completa (acompañantes + principal)	Método: finalizarReserva(\$id_pr (PDO) principal) Dato: ID de reserva principal	SQLite	Todos los registros actualizados correctamente con estado FINALIZADO	Aprobado

Nota. Resultado de las pruebas de integración para los métodos del archivo RecepcionesModel.

Pruebas de Integración Controlador Recepción

En la tabla 21 se detallan las pruebas de integración al Controlador Recepción.

Tabla 21

Resultados de Pruebas de Integración Controlador Recepción

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
CP1	Registrar una nueva reserva sin id_principal	Método: registrar()JSON: {"fecha_ingreso": "2024-01-01", "fecha_salida": "2024-01-02", "numero_personas": 1, "codigo_reserva": "R001", "cliente": {...}, "parqueaderos": [], "tarifas_parqueaderos": "{}"}	SQLite (PDO) y RecepcionesModel	Datos insertados correctamente en clientes, reserva_principal, detalle_reserva. Se mostró el mensaje "reserva registrada..."	Aprobado
CP2	Registrar una nueva reserva con id_principal y	Método: registrar()JSON: {"fecha_ingreso": ..., "id_principal": 1,	SQLite (PDO) y RecepcionesModel	Se insertó reserva, cliente, y se duplicaron los acompañantes activos correctamente	Aprobado

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
	duplicación de acompañantes	"codigo_reserva": "R002", "cliente": {...}, "parqueaderos": [], "cantidad_vehiculos": "{}"			
CP3	Validar registro con campos vacíos obligatorios	Método: registrar(JSON: {})	SQLite (PDO) y RecepcionesModel	Mensajes de validación mostrados ("fecha_ingreso es obligatorio", etc.). No se realizaron inserciones	Aprobado
CP4	Registrar y verificar inserción correcta de huésped, reserva y detalle	Método: registrar(JSON con huésped y datos válidos)	SQLite (PDO) y RecepcionesModel	Verificación en base de datos: registros válidos en huésped, reserva_principal, detalle_reserva	Aprobado
CP5	Finalizar una reserva con detalle activo y parqueaderos	Método: finalizarReserva(1) Datos preexistentes: detalle activo y parqueaderos asociados	SQLite (PDO) y RecepcionesModel	Se actualizaron correctamente los estados y fechas en tablas correspondientes	Aprobado
CP6	Finalizar una reserva principal que aún tiene detalles activos	Método: finalizarReserva(1) Reserva principal con otros detalles activos	SQLite (PDO) y RecepcionesModel	Solo el detalle actual fue finalizado. La reserva principal permanece activa	Aprobado
CP7	Finalizar reserva principal automáticamente al no haber más detalles activos	Método: finalizarReserva(1) Último detalle activo	SQLite (PDO) y RecepcionesModel	Estado de la reserva principal cambió a 0, se registró fecha_salida y observación final	Aprobado
CP8	Verificar disponibilidad cuando no hay cruce de fechas	Método: verificarDisponibilida d("2024-05-01", "2024-05-03")	SQLite (PDO) y RecepcionesModel	Se retornó true, no hubo conflicto de fechas	Aprobado

Id	Descripción	Entrada	Integración	Resultado Obtenido	Estado
CP9	Verificar disponibilidad con cruce de fechas ya registradas	Método: verificarDisponibilida d("2024-01-01", "2024-01-02")(Fecha de una reserva ya existente)	SQLite (PDO) y RecepcionesModel	Se retornó false al detectar una reserva existente en ese rango	Aprobado
CP10	Verificar disponibilidad al actualizar una reserva con fechas que no generan conflicto con otras reservas	Método: verificarDisponibilida dAlActualizar("2024- 05-01", "2024-05-02", 1) Reserva existente con id_detalle = 1 y sin otras reservas conflictivas	SQLite (PDO) y RecepcionesModel	Método devolvió true, sin cruce con otras reservas activas	Aprobado
CP11	Verificar disponibilidad al actualizar con fechas que se cruzan con otras reservas activas (diferente id)	Método: verificarDisponibilida dAlActualizar("2024- 01-01", "2024-01-02", 1) Hay otra reserva activa diferente con cruce	SQLite (PDO) y RecepcionesModel	Se detectó cruce, método retornó false correctamente	Aprobado

Nota. Resultado de las pruebas de integración para los métodos del controlador Recepciones.

Conclusión General de las Pruebas Unitarias y de Integración

Como resultado de la ejecución de las pruebas unitarias e integración realizadas al sistema web Gestor de Alojamientos mediante PHPUnit, se obtiene que las unidades de código individuales y en conjunto funcionan correctamente. Las pruebas unitarias aplicadas validaron que los métodos de la estructura del proyecto, modelo y controlador se comportan correctamente ante entradas válidas y generan respuestas esperadas, sin arrojar errores ni comportamientos inesperados. Todas las pruebas fueron aprobadas y no se evidenciaron fallos en la lógica interna del sistema.

En cuanto a las pruebas de integración, estas confirman la adecuada comunicación entre los distintos módulos del sistema (controladores, modelos y vistas), así como la interacción correcta con la base de datos a través de SQLite en memoria, permitiendo simular múltiples escenarios sin afectar datos reales. Se verificó el correcto funcionamiento de procesos complejos como el registro de reservas, finalización de reservas, gestión de acompañantes y validación de disponibilidad. Todos los casos de prueba definidos fueron exitosos, reflejando consistencia, precisión y estabilidad en la ejecución del sistema.

Prueba Piloto

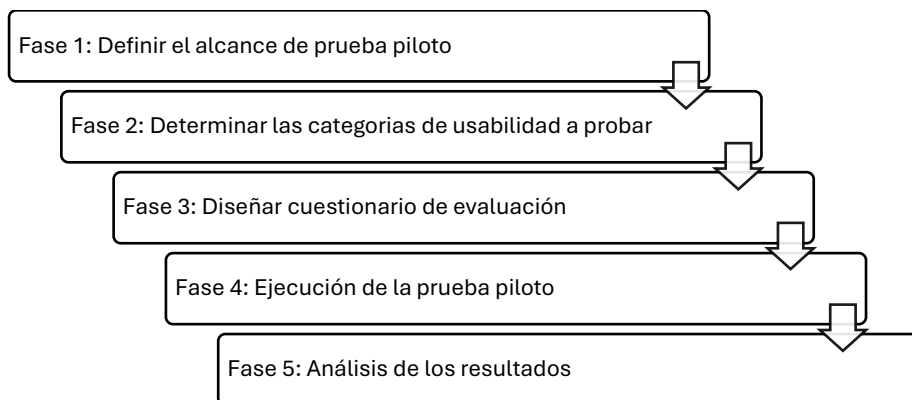
La prueba piloto es “un proceso informal donde los usuarios experimentan con un nuevo producto de software, para ver si les gusta y si hace lo que necesitan” (Sommerville, 2011, p. 228) se utiliza para probar que el sistema web cumple con las condiciones y requerimientos establecidos por el usuario dentro de un entorno real o simulado, analizando los resultados en busca de posibles errores o anomalías, permitiendo la oportuna corrección antes de su despliegue a producción.

La prueba piloto está planteada para recoger la percepción del usuario bajo el concepto de usabilidad propuesta en la publicación “Metodologías de testing de software y su aplicación en el Centro de informática de la universidad EAFIT” propuesto por Alejandro Cálad Álvarez y Juan David Ruíz Calle en el año 2009, donde se establece el marco referencial vigente para pruebas de software bajo el enfoque de R. Pressman.

En este contexto, la figura 64 se presentan las fases a seguir para realizar la prueba piloto:

Figura 64

Fases de la Prueba Piloto



Nota. Paso a paso para aplicar la prueba piloto.

Fase 1: Definir el Alcance de Prueba Piloto

Objetivo: identificar la percepción del usuario final, validando la usabilidad y funcionalidad del sistema.

Limites: la prueba piloto se enmarca en la metodología mencionada anteriormente para tal propósito, enfocándose únicamente en aplicar la encuesta como instrumento recolector de datos cuantificables, evaluando de manera conjunta la interactividad, disposición y personalización del sistema web para obtener un promedio de usabilidad y percepción por parte de los usuarios, sin incorporar métricas como por ejemplo: tareas completadas, tiempos de ejecución, nivel de satisfacción o indicadores basados en estándares internacionales, ya que exceden de los alcances metodológicos establecidos por el autor.

Criterios de aceptación: El sistema web será aprobado con un promedio de satisfacción de 4.5 sobre 5 (puntuación máxima) por pregunta.

Ambiente o escenario de la prueba: demo del sistema web alojando en servidor local XAMPP y línea <https://digihostal.infinityfreeapp.com/>.

Tiempo estimado de la prueba: 1 hora x 3 días.

Áreas de trabajo para evaluar: Inicio de sesión, empleados, establecimiento, alojamientos, habitaciones, tarifas, Gestión de reservas y Reporte de la TRA.

Roles de usuarios involucrados: “Recepcionista” encargado de gestionar las reservas, “Administrador” Usuario encargado de parametrizar el sistema, y “Supervisor” encargado de observar, aplicar evaluación y validar resultados de la prueba.

Tareas para realizar en la prueba: la tabla 22 presentan las tareas a realizar por cada usuario administrador y recepcionista:

Tabla 22

Tareas de la Prueba Piloto

Rol	Tarea
Administrador - Recepcionista	Iniciar sesión en el sistema
Administrador - Recepcionista	Registrar, actualizar, trasladar y finalizar una reserva
Administrador - Recepcionista	Generar y visualizar la TRA
Administrador	Crear, editar y eliminar usuarios
Administrador	Modificar datos del establecimiento
Administrador	Configurar tarifas, parqueaderos, tipos de alojamientos y habitaciones

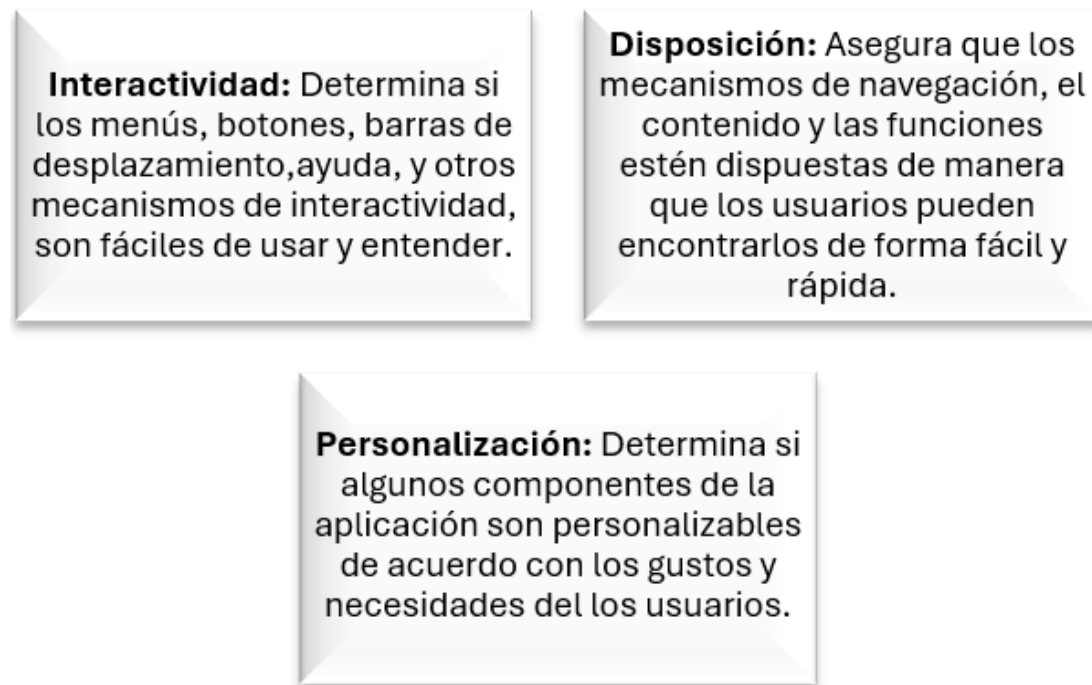
Nota. Tareas que deben realizar los usuarios administrador y recepcionista en el sistema web.

Fase 2: Determinar las Categorías de Usabilidad a Probar

Se presenta en la figura 65, las categorías seleccionadas para probar de manera práctica. De las mencionadas por la metodología del autor, se tomarán las que se consideran importantes para identificar la usabilidad y percepción general del usuario alineándose a objetivo de probar el sistema web, así bien “la selección de cada categoría depende de la aplicación a probar ya que no todas las categorías son evaluadas en las diferentes aplicaciones” (Cálad y Ruiz, 2009, p. 88).

Figura 65

Categorías de Usabilidad para la Prueba Piloto



Nota. Categorías de usabilidad que se evaluarán en la prueba piloto.

La prueba de interactividad verificará la fluidez de interacción frecuente en la manipulación de datos por parte de los usuarios, por el lado de la disposición servirá para probar que no haya confusiones en la organización visual de las interfaces y evaluando la personalización se identificará la percepción y aceptación del sistema web al lograr acomodarse a gestión particular de cada establecimiento.

Fase 3: Diseñar Cuestionario de Evaluación

Para diseñar el cuestionario de evaluación de la prueba piloto de la figura 66, se crearán preguntas que abarquen las 3 categorías de usabilidad determinadas para que sean evaluadas en conjunto y así obtener una percepción integral del sistema.

Figura 66

Formato de la Encuesta para la Evaluar la Prueba Piloto

Evaluación de la prueba piloto						
Estimado/a participante						
Agradecemos su colaboración en la evaluación de la prueba piloto utilizado en nuestra investigación. Sus respuestas nos ayudarán a mejorar la calidad y efectividad del sistema web gestor de alojamientos. La información proporcionada será utilizada exclusivamente con fines de validación y mejora de este.						
Maque con una X la opción de su preferencia.						
Datos Generales						
1. Nombre (opcional): _____						
2. Cargo o Función: _____						
Preguntas para evaluar la prueba piloto	Totalmente en desacuerdo	Algo en desacuerdo	Ni de acuerdo, ni en desacuerdo	Algo de acuerdo	Totalmente de acuerdo	Total
	1	2	3	4	5	
Logra iniciar sesión en el sistema web sin dificultad.						
Administrador						
Logra gestionar (crear, editar) dentro del módulo de establecimiento sin dificultad.						
Logra gestionar (crear, editar, eliminar) dentro del módulo de usuario sin dificultad.						
Logra gestionar (crear, editar, eliminar) dentro del módulo de alojamientos sin dificultad.						
Logra gestionar (crear, editar, eliminar) dentro del módulo de habitaciones sin dificultad.						
Logra gestionar (crear, editar, eliminar) dentro del módulo de tarifas sin dificultad.						
Logra gestionar (crear, editar, eliminar) parqueaderos dentro del módulo de tarifas sin dificultad.						
Recepcionista y Administrador						
Logra gestionar (crear, editar, trasladar, finalizar) dentro del módulo de reservas sin dificultad.						
Logra descargar el reporte TRA dentro del módulo de generar la TRA sin dificultad						
Percepción general						
El software responde de forma rápida y fluida a la realización de las acciones.						
El sistema web se acomoda a la gestión de alojamientos y reservas de su establecimiento.						
Estoy satisfecho con el uso general del sistema web de gestión de alojamiento.						
Recomendaría el uso de este sistema a otros prestadores de servicios turísticos.						

Nota. Formato de la encuesta para evaluar la prueba piloto.

Se retoma la metodología para encuestas aplicada en la “fase 1 identificación de requerimientos y necesidades” como modelo para diseñar el cuestionario, con opciones de respuesta la escala Likert de 1 a 5 para captar distintos niveles de acuerdo o satisfacción de los usuarios, facilitando el análisis cuantitativo de los resultados.

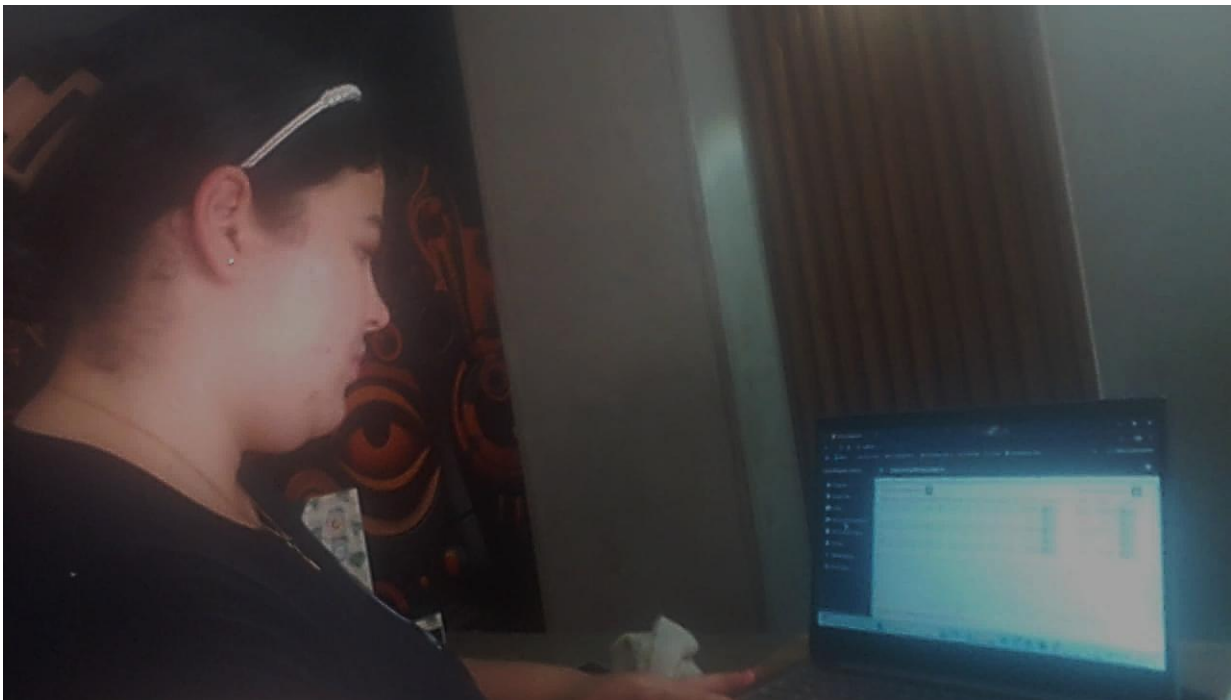
Fase 4: Ejecución de la Prueba Piloto

La ejecución de la prueba piloto se realiza visitando 3 establecimientos que ofrecen servicios de alojamiento, donde 4 usuarios reales interactúan con el sistema web, realizando tareas como la gestión de reservas, registro de empleados, tarifas, alojamiento, habitaciones y generación del reporte TRA.

La figura 67 ilustra el usuario recepcionista del establecimiento Hotel Suarez, interactuando con el sistema.

Figura 67

Prueba Piloto Establecimiento 1 Usuario Recepcionista

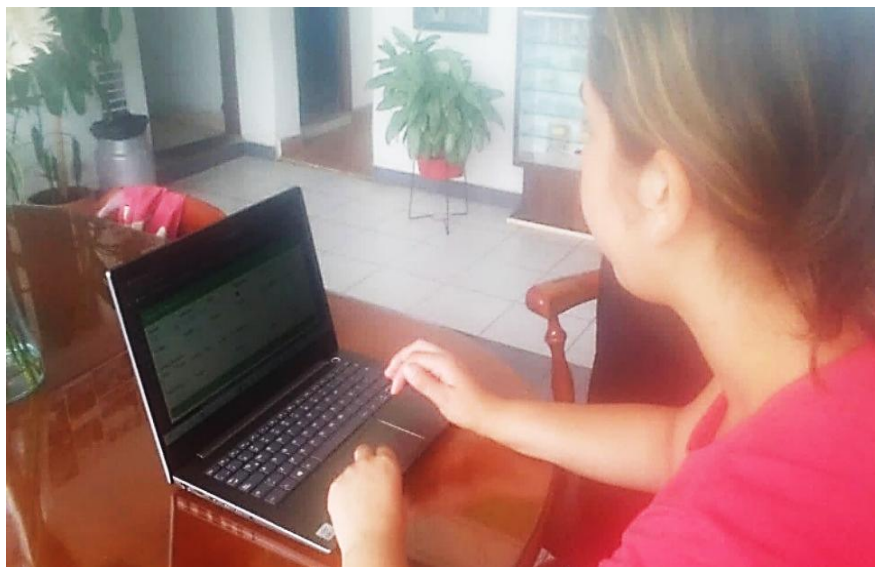


Nota. Usuario Recepcionista interactuando con el sistema web gestor de reservas.

La figura 68 ilustra el usuario Admin - Recepcionista del establecimiento Hotel Chalet.

Figura 68

Prueba Piloto Establecimiento 2 Usuario Admin-Recepcionista

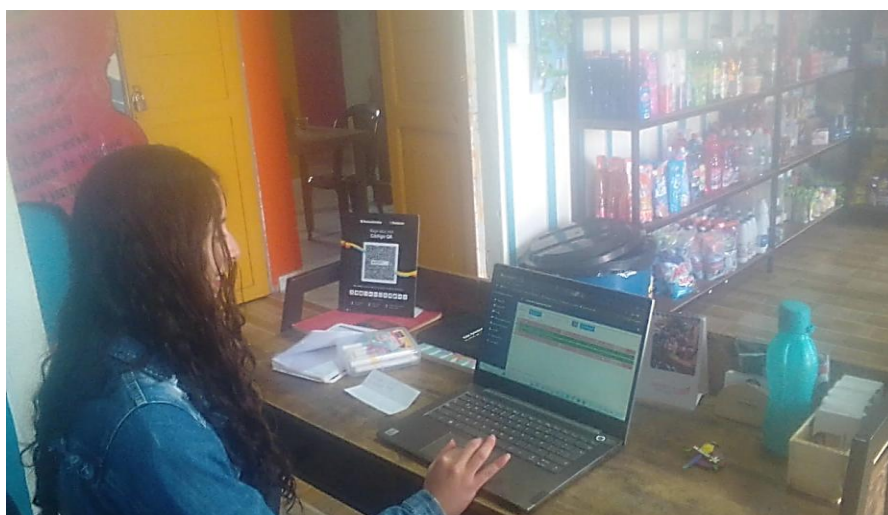


Nota. Usuario Admin - Recepcionista interactuando con el sistema web gestor de reservas.

La figura 69 ilustra el usuario Recepcionista del establecimiento Hostal Casa Celeste.

Figura 69

Prueba Piloto Establecimiento 3 Usuario Recepcionista

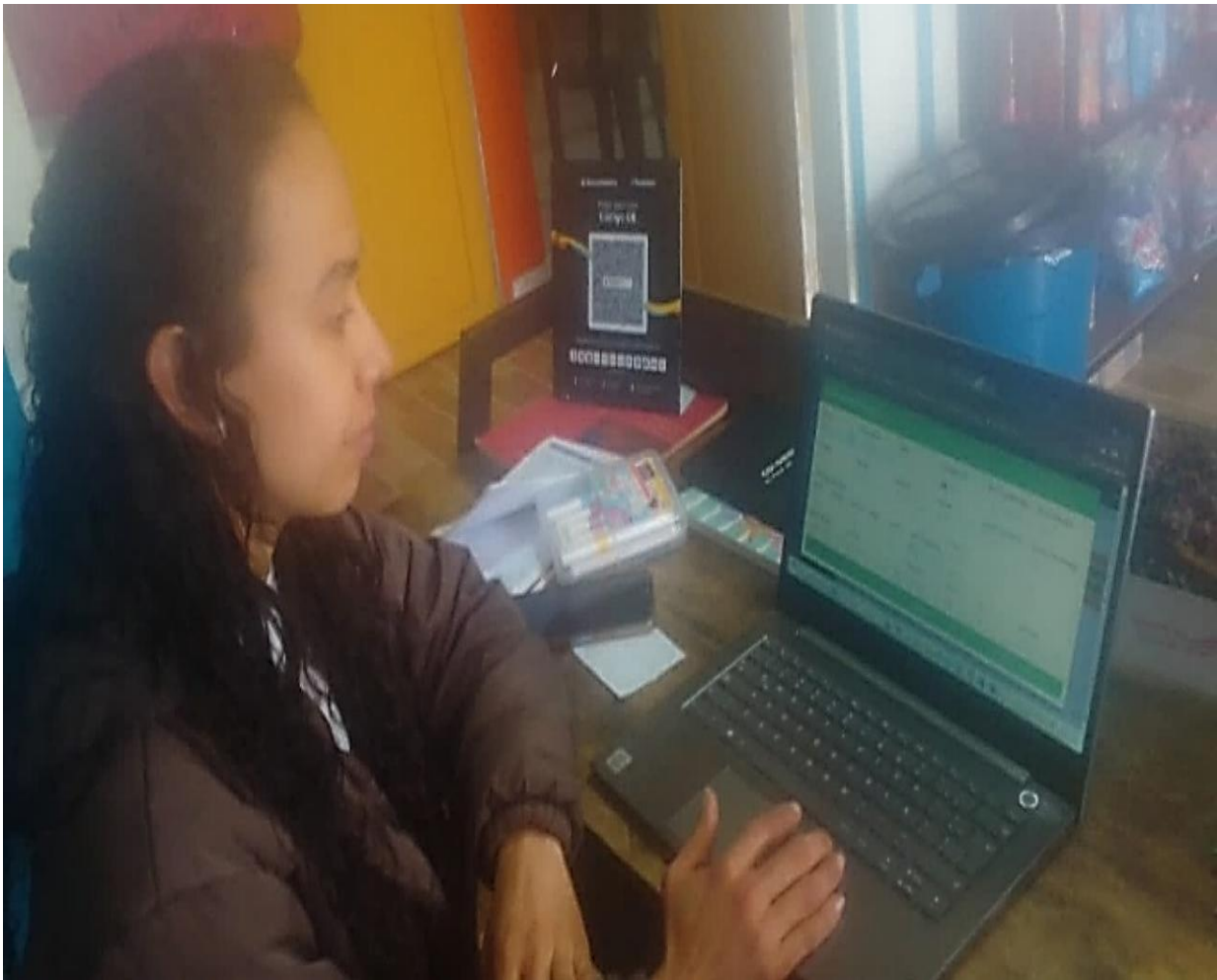


Nota. Usuario Recepcionista interactuando con el sistema web gestor de reservas.

La figura 70 ilustra el usuario Admin - Recepcionista del establecimiento Hostal Casa Celeste.

Figura 70

Prueba Piloto Establecimiento 3 Usuario Recepcionista



Nota. Usuario Admin - Recepcionista interactuando con el sistema web gestor de reservas.

Al finalizar la interacción de los usuarios con el sistema, se procede a responder la encuesta para recolectar los resultados que evaluarán la prueba piloto.

La figura 71 ilustra las respuestas de la encuesta del usuario recepcionista del Hotel Suarez.

Figura 71

Respuesta Evaluación de la Prueba Piloto Usuario 1 Hotel Suarez

Evaluación de la prueba piloto						
Estimado/a participante						
Agradecemos su colaboración en la evaluación de la prueba piloto utilizado en nuestra investigación. Sus respuestas nos ayudarán a mejorar la calidad y efectividad del sistema web gestor de alojamientos. La información proporcionada será utilizada exclusivamente con fines de validación y mejora de este.						
Maque con una X la opción de su preferencia.						
Datos Generales						
1. Nombre (opcional): Jennifer milena torres flores						
2. Cargo o Función: Recepcionista						
Pregunta	Totalmente en desacuerdo	Algo en desacuerdo	Ni de acuerdo, ni en desacuerdo	Algo de acuerdo	Totalmente de acuerdo	Total
	1	2	3	4	5	
Logra iniciar sesión en el sistema web sin dificultad.					X	5
Administrador						
Logra gestionar (crear, editar) dentro del módulo de establecimiento sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de usuario sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de alojamientos sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de habitaciones sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de tarifas sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) parqueaderos dentro del módulo de tarifas sin dificultad.					X	5
Recepcionista y Administrador						
Logra gestionar (crear, editar, trasladar, finalizar) dentro del módulo de reservas sin dificultad.					X	5
Logra descargar el reporte TRA dentro del módulo de generar la TRA sin dificultad					X	5
Percepción general						
El software responde de forma rápida y fluida a la realización de las acciones.					X	5
El sistema web se acomoda a la gestión de alojamientos y reservas de su establecimiento.					X	5
Estoy satisfecho con el uso general del sistema web de gestión de alojamiento.					X	5
Recomendaría el uso de este sistema a otros prestadores de servicios turísticos.					X	5

Nota. Respuesta de la encuesta en la prueba piloto del usuario recepcionista.

La figura 72 ilustra las respuestas de la encuesta del usuario recepcionista del establecimiento Hotel Chalet.

Figura 72

Respuesta Evaluación de la Prueba Piloto Usuario 2 Hotel Chalet

Evaluación de la prueba piloto						
Estimado/a participante						
Agradecemos su colaboración en la evaluación de la prueba piloto utilizado en nuestra investigación. Sus respuestas nos ayudarán a mejorar la calidad y efectividad del sistema web gestor de alojamientos. La información proporcionada será utilizada exclusivamente con fines de validación y mejora de este.						
Maque con una X la opción de su preferencia.						
Datos Generales						
1. Nombre (opcional): Yuri morales						
2. Cargo o Función: Administrador - Recepcionista						
Pregunta	Totalmente en desacuerdo	Algo en desacuerdo	Ni de acuerdo, ni en desacuerdo	Algo de acuerdo	Totalmente de acuerdo	Total
	1	2	3	4	5	
Logra iniciar sesión en el sistema web sin dificultad.					X	5
Administrador						
Logra gestionar (crear, editar) dentro del módulo de establecimiento sin dificultad.				X	X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de usuario sin dificultad.				X		4
Logra gestionar (crear, editar, eliminar) dentro del módulo de alojamientos sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de habitaciones sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de tarifas sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) parqueaderos dentro del módulo de tarifas sin dificultad.					X	5
Recepcionista y Administrador						
Logra gestionar (crear, editar, trasladar, finalizar) dentro del módulo de reservas sin dificultad.					X	5
Logra descargar el reporte TRA dentro del módulo de generar la TRA sin dificultad					X	5
Percepción general						
El software responde de forma rápida y fluida a la realización de las acciones.					X	5
El sistema web se acomoda a la gestión de alojamientos y reservas de su establecimiento.					X	5
Estoy satisfecho con el uso general del sistema web de gestión de alojamiento.					X	5
Recomendaría el uso de este sistema a otros prestadores de servicios turísticos.					X	5

Nota. Respuesta de la encuesta en la prueba piloto del usuario Admin -Recepcionista.

La Figura 73 ilustra las respuestas de la encuesta del usuario recepcionista del establecimiento Hostal Casa Celeste.

Figura 73

Respuesta Evaluación de la Prueba Piloto Usuario 3 Hostal Casa Celeste

Evaluación de la prueba piloto						
Estimado/a participante						
Agradecemos su colaboración en la evaluación de la prueba piloto utilizado en nuestra investigación. Sus respuestas nos ayudarán a mejorar la calidad y efectividad del sistema web gestor de alojamientos. La información proporcionada será utilizada exclusivamente con fines de validación y mejora de este.						
Maque con una X la opción de su preferencia.						
Datos Generales						
1. Nombre (opcional): Dana Caleño						
2. Cargo o Función: Recepcionista						
Pregunta	Totalmente en desacuerdo	Algo en desacuerdo	Ni de acuerdo, ni en desacuerdo	Algo de acuerdo	Totalmente de acuerdo	Total
	1	2	3	4	5	
Logra iniciar sesión en el sistema web sin dificultad.					X	5
Administrador						
Logra gestionar (crear, editar) dentro del módulo de establecimiento sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de usuario sin dificultad.				X		5
Logra gestionar (crear, editar, eliminar) dentro del módulo de alojamientos sin dificultad.				X		5
Logra gestionar (crear, editar, eliminar) dentro del módulo de habitaciones sin dificultad.				X		5
Logra gestionar (crear, editar, eliminar) dentro del módulo de tarifas sin dificultad.				X		5
Logra gestionar (crear, editar, eliminar) parqueaderos dentro del módulo de tarifas sin dificultad.				X		5
Recepcionista y Administrador						
Logra gestionar (crear, editar, trasladar, finalizar) dentro del módulo de reservas sin dificultad.				X		5
Logra descargar el reporte TRA dentro del módulo de generar la TRA sin dificultad					X	5
Percepción general						
El software responde de forma rápida y fluida a la realización de las acciones.					X	5
El sistema web se acomoda a la gestión de alojamientos y reservas de su establecimiento.				X		5
Estoy satisfecho con el uso general del sistema web de gestión de alojamiento.				X		5
Recomendaría el uso de este sistema a otros prestadores de servicios turísticos.				X		5

Nota. Respuesta de la encuesta en la prueba piloto del usuario recepcionista.

La Figura 74 ilustra la encuesta respondida por el usuario administrador - recepcionista del establecimiento Hostal Casa Celeste.

Figura 74

Respuesta Evaluación de la Prueba Piloto Usuario 4 Hostal Casa Celeste

Evaluación de la prueba piloto						
Estimado/a participante						
Agradecemos su colaboración en la evaluación de la prueba piloto utilizado en nuestra investigación. Sus respuestas nos ayudarán a mejorar la calidad y efectividad del sistema web gestor de alojamientos. La información proporcionada será utilizada exclusivamente con fines de validación y mejora de este.						
Maque con una X la opción de su preferencia.						
Datos Generales						
1. Nombre (opcional): Marcela Jiménez						
2. Cargo o Función: Administrador - Recepcionista						
Pregunta	Totamente en desacuerdo	Algo en desacuerdo	Ni de acuerdo, ni en desacuerdo	Algo de acuerdo	Totamente de acuerdo	Total
	1	2	3	4	5	
Logra iniciar sesión en el sistema web sin dificultad.					X	5
Administrador						
Logra gestionar (crear, editar) dentro del módulo de establecimiento sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de usuario sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de alojamientos sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de habitaciones sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) dentro del módulo de tarifas sin dificultad.					X	5
Logra gestionar (crear, editar, eliminar) parqueaderos dentro del módulo de tarifas sin dificultad.					X	5
Recepcionista y Administrador						
Logra gestionar (crear, editar, trasladar, finalizar) dentro del módulo de reservas sin dificultad.				X		4
Logra descargar el reporte TRA dentro del módulo de generar la TRA sin dificultad					X	5
Percepción general						
El software responde de forma rápida y fluida a la realización de las acciones.					X	5
El sistema web se acomoda a la gestión de alojamientos y reservas de su establecimiento.				X		4
Estoy satisfecho con el uso general del sistema web de gestión de alojamiento.				X		4
Recomendaría el uso de este sistema a otros prestadores de servicios turísticos.				X		4

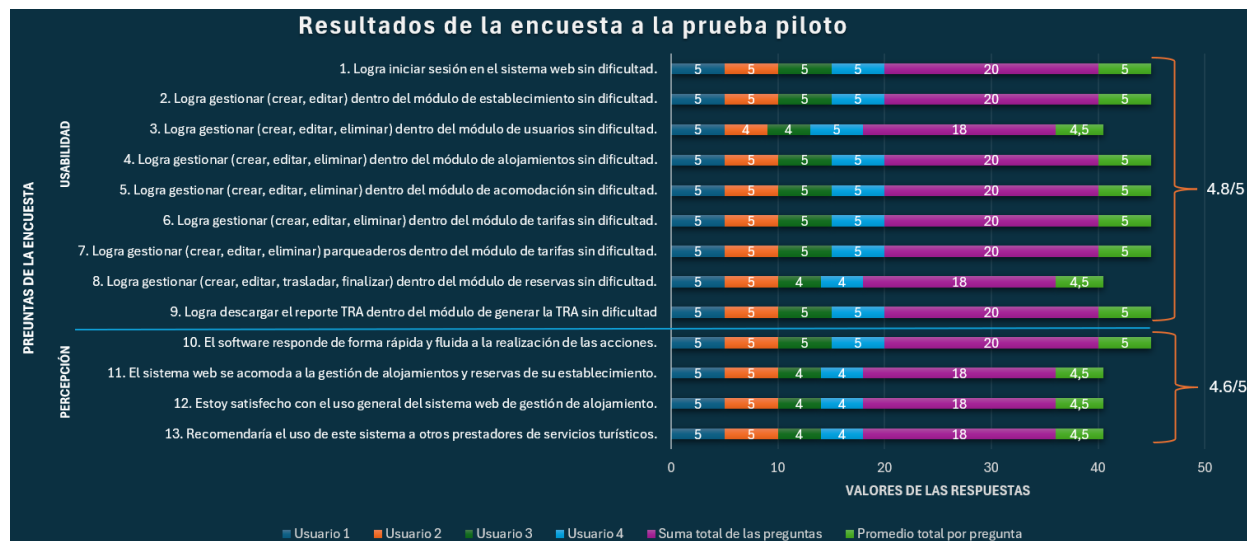
Nota. Respuesta de la encuesta en la prueba piloto del usuario Admin - Recepcionista.

Fase 5: Análisis de los Resultados

Aplicada la prueba piloto y la encuesta, se presentan los resultados resumidos en la figura 75 abarcando las respuestas de los usuarios y el promedio total por ítem.

Figura 75

Resultados de la Prueba Piloto



Nota. Resultado total de la encuesta para evaluar el sistema web en la prueba piloto.

Vale resaltar que la encuesta contó con 13 preguntas, 9 enfocadas en evaluar las categorías de usabilidad (interactividad y disposición) de las diferentes funciones del sistema y 4 enfocadas en evaluar la categoría de personalización, también identificar la percepción general del sistema desarrollado, en resumen, de la evaluación de la usabilidad se obtuvo un promedio de 4.8/5.0 y en percepción general un promedio de 4.6/5.0, en usabilidad se identifica la necesidad de mejorar en términos de interactividad, disposición y simplificación de procesos en las áreas específicas de gestión de reservas y usuarios.

En cuanto a la percepción general se observa que, si bien el sistema responde de forma dinámica y estable a la realización de acciones, se sigue una oportunidad de mejora en la

adaptación del sistema a las necesidades específicas de algunos usuarios o establecimientos, para aumentar la satisfacción de los usuarios.

Finalmente, los resultados del gráfico respaldan la conclusión de que el sistema presenta un alto grado de usabilidad percibida, alineándose con los principios establecidos por metodologías de testing de Cálad y Ruiz (2009), desarrollándose una prueba piloto pertinente y suficiente, validando positivamente la experiencia de usuario, sin incorporar métricas adicionales que excedan los alcances y metodología definidas para cumplir con el objetivo. La ligera variación en algunos promedios invita a considerar ajustes menores en las secciones de mayor complejidad operativa, para garantizar una experiencia aún más intuitiva y consistente.

Conclusiones

En cumplimiento del primer objetivo para identificar los requisitos y necesidades, se diseñó y aplicó con éxito el instrumento de encuesta que permitió identificar la viabilidad y necesidad en los prestadores de servicios turísticos de contar un sistema para gestionar los alojamientos.

Los resultados de los datos recolectados permitieron obtener un panorama detallado de las capacidades tecnológicas y habilidades digitales suficientes para implementar un sistema web de este tipo. Además, se logró identificar los principales elementos que deben contemplarse en el diseño del sistema, tales como los tipos de alojamiento más comunes, la capacidad de camas y personas por habitación, horarios flexibles, tarifas de reserva, y servicios complementarios todo ello detallado en el apartado “conclusiones del análisis de resultados”.

Estos hallazgos permitieron estructurar los requerimientos clave de descripción general, legales, de interfaces, funcionales, no funcionales que deben ser aplicados dentro del diseño y desarrollo del sistema, que cumpla con las condiciones detectadas, ajustándose al contexto regional y a la normatividad vigente.

Finalizada la fase 2, relacionada con el segundo objetivo: diseño del sistema web gestor de alojamientos, se obtiene como resultado un prototipo estructurado no funcional donde se organizó la navegación, textos, tablas, botones, formularios, etc., todo alineado con los requerimientos funcionales identificados para la gestión y control de visitantes, y la generación del reporte TRA conforme la normatividad vigente, priorizando la usabilidad, accesibilidad y eficiencia, junto a su respectiva base de datos. Gracias al modelado UML se logra definir las funcionalidades y acciones que realizará el usuario final dentro del sistema y los componentes

para la arquitectura modelo vista controlador, visualizando toda la operatividad reduciendo posibles errores antes de la implementación.

Como resultado de la fase 3, correspondiente al tercer objetivo, se logra implementar el código del sistema web para la gestión de visitantes, materializando en funcionalidades los requerimientos y diseño del sistema, aplicando como estrategia cortos periodos de tiempo con marco Scrum de la metodología ágil. Se logra iterar sobre cada historia y tareas específicas, facilitando semana a semana la entrega de módulos operativos: alojamientos, habitaciones, gestión de reservas, tarifas, establecimiento, usuario empleado, generación de la TRA y demás acciones necesarias. Los resultados obtenidos hasta el momento evidencian la coherencia y el cumplimiento del sistema con los objetivos planteados.

Dando cumplimiento al cuarto objetivo, los resultados obtenidos en la fase 4 pruebas, permiten afirmar que el desarrollo del sistema alcanzó un nivel óptimo de funcionamiento tanto a nivel técnico como de experiencia de usuario. Por un lado, las pruebas unitarias y de integración realizadas con PHPUnit evidenciaron que los métodos desarrollados responden correctamente ante diferentes entradas y escenarios, sin generar errores ni comportamientos inesperados; además, se comprobó una adecuada interacción entre los componentes del sistema, lo que refleja una arquitectura bien estructurada. Por otro lado, los resultados de la prueba piloto aplicada en un entorno real en la prestación del servicio de alojamiento turístico mostraron una percepción mayoritariamente positiva, destacando que las 9 funcionalidades clave obtuvieron una calificación promedio de 4.8/5.0, lo que indica una alta usabilidad, sugiriendo áreas específicas donde se pueden aplicar mejoras en la experiencia de interacción. De igual manera, en cuanto a la percepción general del sistema con un 4.6/5.0, se resaltó la estabilidad en la respuesta del

software, mientras que aspectos como la adaptación a las necesidades del usuario y la recomendación del sistema también obtuvieron una valoración destacada.

En conclusión, se confirma el cumplimiento del objetivo general de diseñar e implementar un sistema web para la gestión del alojamiento turístico en Murillo Tolima, conforme a la normatividad vigente, aportando una solución pertinente para la modernización del sector.

Enfoque organizacional: el desarrollo de este proyecto demostró que una adecuada planeación combinada con disciplina y perseverancia permitió el desarrollo de los objetivos trazados en un tiempo relativamente corto, lo cual crea una visión sistemática orientada para enfrentar grandes retos, superar limitaciones y materializar metas.

Enfoque social: el desarrollo de proyectos sociales permite crear empatía con las comunidades donde se necesitan soluciones que mejoren la calidad de vida de las comunidades, despertando sensibilidad para entender que la creación de software no sólo es un ejercicio técnico, también genera lazos humanos dando oportunidad al servicio social y comunitario.

Enfoque empresarial: el proyecto demostró que es posible transformar necesidades sociales en oportunidades de emprendimiento tecnológico, sentando las bases para impulsar iniciativas que contribuyan a la generación de empresa para el desarrollo local y regional.

Enfoque profesional: este proyecto fortaleció habilidades técnicas y de solución de problemas, aportando una visión más consciente sobre el rol del desarrollador de software como agente de cambio en entornos sociales, utilizando la tecnología digital para mejorar el día a día de las personas, respondiendo a necesidades específicas.

Recomendaciones

A partir de los resultados del proyecto, se proponen las siguientes recomendaciones:

Tenido en cuenta los resultados de prueba piloto, se identificaron oportunidades de mejora para los módulos de “Usuarios” y “Reservas” donde los usuarios manifestaron, en menor medida, cierta dificultad, con el fin de aumentar la experiencia de uso y adaptación del sistema web a los establecimientos. Se recomienda ajustar la interfaz, los flujos de navegación, simplificación de formularios y aplicar pruebas de aceptación para verificar que los cambios incorporados satisfacen el usuario final.

Para futuras versiones del sistema web, es preciso agregar los módulos de facturación, contabilidad, reportes estadísticos, ya que los prestadores de servicios turísticos manifestaron su interés por estas funcionalidades complementarias. Para ello es necesario la integración del sistema web mediante APIs, con software que ofrezca estos servicios y trabajar de la mano con contadores, los usuarios finales y expertos en visualización de datos, para asegurar que las funcionalidades sean pertinentes, útiles y fáciles de usar.

En cuanto a la escalabilidad del sistema, se recomienda el diseño de una estrategia que contemple el uso de lenguajes o frameworks que garanticen un desarrollo modular, rendimiento y estabilidad del sistema. Como el sistema se implementó combinando PHP y JavaScript, se puede migrar gradualmente a Laravel y Vue.js organizando el sistema en módulos independientes y creando pruebas para validar las nuevas tecnologías antes de su adopción definitiva.

Referencias Bibliograficas

Apachefriends. (2024). *Acerca de XAMPP*. Apachefriends.org.

<https://www.apachefriends.org/es/about.html#:~:text=El%20objetivo%20de%20XAMPP%20es,usos%20comerciales%20como%20no%20comerciales.>

Blanco, C. (2011). Encuesta y estadística. Métodos de Investigación Cuantitativa en Ciencias Sociales y Comunicación. Universidad San Gregorio de Portoviejo.

<http://librodigital.sangregorio.edu.ec/librosusgp/B0055.pdf>

Bravo, T. (2019). *Desarrollo de instrumentos para la evaluación: Cuestionarios*. Instituto

Nacional para la Evaluación de la Educación México. <https://www.inee.edu.mx/wp-content/uploads/2019/08/P2A355.pdf>

Briano, C. (2023). *Compilación de apuntes sobre conceptos fundamentales de la Ingeniería de Software*. 2° Edición.

http://bibliotecadigital.econ.uba.ar/download/libros/Briano_compilacion_apuntes.pdf

Calad, A, y Ruiz, J. (2009). *Metodologías de testing de software y su aplicación en el centro de informática de la universidad EAFIT*. Universidad EAFIT.

<https://repository.eafit.edu.co/server/api/core/bitstreams/7c41761e-d229-47b9-8855-a8d2650d62d9/content>

Chekin. (2025). *Chekin como software de gestión*. <https://chekin.com/blog/automatiza-la-tarjeta-de-registro-de-alojamiento-tra-y-sire-con->

[chekin/#:~:text=Chekin%20como%20Software%20de%20Gesti%C3%B3n,in%20hasta%20el%20check%20Dout.](https://chekin.com/blog/automatiza-la-tarjeta-de-registro-de-alojamiento-tra-y-sire-con-chekin/#:~:text=Chekin%20como%20Software%20de%20Gesti%C3%B3n,in%20hasta%20el%20check%20Dout.)

- Chiner, E. (2011). *Tema 8 investigación descriptiva mediante encuestas*. Universidad de Alicante. <https://rua.ua.es/dspace/bitstream/10045/19380/34/Tema%208-Encuestas.pdf>
- Díaz, V. (2002). *Tipos de encuesta y diseños de investigación*. Universidad Pública de Navarra UPNA. https://www.unavarra.es/personal/vidaldiaz/pdf/tipos_encuestas.PDF
- DigiTRA. (2025). *¿Qué es DIGITRA?* <https://digitra.rent/>
- Developer.mozilla. (2024). *¿Qué es JavaScript?* Developer.mozilla.org. Recuperado 08/09/2024. https://developer.mozilla.org/es/docs/Learn/JavaScript/First_steps/What_is_JavaScript
- EasyTRA. (2025). *¿Qué es exactamente lo que hace easytra.app?* <https://easytra.app/>
- Escofet, A., Folgueiras, P., Luna, E., y Palou, B. (2016). *Elaboración y validación de un cuestionario para la valoración de proyectos de aprendizaje-servicio*. Revista Mexicana de Investigación Educativa, 21(70), 929-949. <https://www.redalyc.org/pdf/140/14046162013.pdf>
- Fernández, T. (2014). *La Encuesta noción general, tipos de encuestas y sugerencias para mejorar su calidad*. Universidad de la República - Facultad de Ciencias. https://eva.interior.udelar.edu.uy/pluginfile.php/20337/mod_resource/content/1/2014%20ficha11encuesta.wpd.pdf
- ISO – International Organization for Estandardization. (2022). *ISO – IEC 27001: 2022 Sistema de gestión de la seguridad de la información*. <https://www.iso.org/standard/27001>
- ISO – International Organization for Estandardization. (2022). *ISO – IEC 27002: 2022 Information security controls*. <https://www.iso.org/es/contents/data/standard/07/56/75652.html>

- ISO – International Organization for Estandardization. (2011). ISO – IEC 25010: 2011 *Ingeniería de sistemas de software*. <https://www.iso.org/standard/35733.html>
- Krasner, G. E., y Pope, S. T. (1988). *A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System*. *Journal of Object-Oriented Programming*, 1(3), 26-49.
https://www.researchgate.net/publication/239452280_A_Description_of_the_Model-View-Controller_User_Interface_Paradigm_in_the_Smalltalk80_System
- Lopez Romo, H. (1998). *La metodología de encuesta*. biblioteca.marco.edu.mx.
https://biblioteca.marco.edu.mx/files/metodologia_encuestas.pdf
- MINTIC. (2023). *Prestadores de servicios turísticos*. Ministerio de Comercio, Industria y Turismo. <https://www.mincit.gov.co/getattachment/f7c53ff0-36c2-435b-884c-bb5e847f8866/Prestadores-de-servicios-turisticos-%E2%80%93-Definiciones.aspx>
- Nielsen, J. (2020). *Usability Engineering*. Morgan Kaufmann.
<https://dl.acm.org/doi/pdf/10.5555/2821575>
- Nixon. (2009). *Learning PHP, MySQL & JavaScript*.
<https://books.google.hn/books?id=Zwzc4qUZnqMC&printsec=copyright&hl=es#v=onepage&q&f=false>
- Oracle. (2024). *¿Qué es MySQL?* Oracle.com.co. <https://www.oracle.com/co/mysql/what-is-mysql/>
- Palacio, J. (2008). *Flexibilidad con Scrum*. Scrum Manager.
https://www.scrummanager.com/files/flexibilidad_con_scrum.pdf
- Pérez, D. (2019). *Qué es CSS3 y sus fundamentos*. Openwebinars.net

<https://openwebinars.net/blog/que-es-css3/>.

PHP. (2021). *¿Qué es PHP?* PHP.net. <https://www.php.net/manual/es/intro-what-is.php>

PHPUnit. (2025). Manual de PHPUnit. <https://docs.phpunit.de/en/8.5/>

Practum Management Group. (2023). *¿Cuándo conviene combinar metodología ágil y de cascada?* <https://www.linkedin.com/pulse/cu%C3%A1ndo-conviene-combinar-metodolog%C3%ADa-%C3%A1gil-y-de-cascada-practum/>

Pressman, R. (2010). *Ingeniería del Software. Un Enfoque Práctico. 7 edición.* Universidad de Cauca. http://artemisa.unicauca.edu.co/~cardila/IS_Libro_Pressman_7.pdf

Rumbaugh, J., Jacobson, I. y Booch, G. (2000). *El Lenguaje de Modelado Unificado: Manual de Referencia.* Rational Software Corporation. <https://altior-edunube.com/wp-content/uploads/2022/04/Lect-8-Ing-Soft.pdf>

Saavedra. (2023). *¿Qué es HTML5 y para qué sirve?* ebac.mx. <https://ebac.mx/blog/que-es-html5>

Schwaber, K. y Sutherland, J. (2020). *La Guía Scrum. La Guía Definitiva de Scrum: Las Reglas del Juego.* Scrum Guides. <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf>

Siigo. (2018). *¿Qué es un software de gestión?* Software de gestión Siigo. <https://www.siigo.com/blog/empresario/software-de-gestion/>

Sommerville, I. (2011). *Ingeniería de Software. 9 edición.* Pearson educación, México, 2011. https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf

Tramincit. (2025). *Tarjeta de registro de alojamiento - TRA*. Tramincit. Comercio, Industria y Turismo.

<https://tramincit.gov.co/ayuda#:~:text=El%20no%20reportar%20la%20informaci%C3%B3n,la%20Ley%202068%20de%202022.>

Tramincit. (2024). *Tarjeta de registro de alojamiento -TRA. Grupo 3: Prestadores usuarios no conectados*. Ministerio de Comercio, Industria y Turismo.

<https://tramincit.gov.co/servicios/alojamientos-sin-conexion-a-internet>

Trigas Gallego, M. (2012). *Metodología Scrum*. Repositorio Institucional Universidad Abierta de Cataluña.

<https://openaccess.uoc.edu/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf>

W3C. (2024). Sumario de WCAG 2. Web accessibility initiative (wai).

<https://www.w3.org/WAI/standardsguidelines/wcag/es#:~:text=WCAG%202.2%20tiene%2013%20pautas,:%20A%2C%20AA%20y%20AAA.>

W3Schools. (2024). *Bootstrap 5: Introducción*. W3Schools.com.

https://www.w3schools.com/bootstrap5/bootstrap_get_started.php

Yahaya N. y Maidin, S. (2022). *El modelo en cascada con Agile Scrum como modelo ágil híbrido para el equipo de ingeniería de software*. Conferencia: 10.^a Conferencia

Internacional sobre Gestión de Servicios Cibernéticos y de TI (CITSM) 2022.

https://www.researchgate.net/publication/365336966_The_Waterfall_Model_with_Agile_Scrum_as_the_Hybrid_Agile_Model_for_the_Software_Engineering_Team

Apéndices

Apéndice A

Manual de Usuario

Manual de usuario

Sistema gestor de alojamientos, reservas y generador de reporte TRA.

Versión 1.0

2025

Jeimy Patricia Popayan Valencia

Introducción

El presente manual de usuario tiene como objetivo guiar a los prestadores de servicios turísticos en el uso del sistema web gestor de alojamientos, una herramienta desarrollada para optimizar el proceso de registro de huéspedes, gestión de reservas, tarifas, alojamientos y generación de reporte TRA.


El sistema gestor de alojamientos ha sido creado para ser fácil de usar y adaptable a las necesidades del establecimiento, permitiendo una mejor organización de la información y el cumplimiento de normativas legales del reporte de la Tarjeta de Registro de Alojamiento (TRA).

Este documento describe paso a paso cada módulo del sistema, facilitando su uso incluso a personas con conocimientos básicos en informática. A lo largo del manual se encontrarán capturas de pantalla, explicaciones detalladas y recomendaciones que permitirán sacar el máximo provecho al sistema.

Requerimientos técnicos:

Computador: Sistema operativo: Windows 7 o superior, Navegador: Google, Chrome, Microsoft Edge o Firefox, paquete de Office, memoria RAM mínimo 4 GB.

¿Cómo ingresar al sistema gestor de alojamientos?

En la pantalla del computador haga clic en el icono  o ingresando al enlace <https://digihostal.infinityfreeapp.com/>, se abrirá una nueva pestaña en el navegador que muestra el formulario para iniciar sesión dentro del “sistema gestor de alojamientos”.

¿Como registrar los datos del establecimiento y empleado con rol “Admin”?

Para registrar el establecimiento y el usuario empleado con rol “Admin”, en el formulario de inicio de sesión, haga clic sobre [Regístra el establecimiento aquí](#), ubicado en la parte inferior, que abre el formulario, completar los campos y hacer clic el botón “registrar” para guardar la información como se ve en la figura 1. No dejar campos vacíos.

Figura 1

Formulario de Registro de Datos del Establecimiento y Usuario Rol "Admin"

Registrar establecimiento y usuario empleado con rol "Admin"

Datos del establecimiento

Nombre del establecimiento **NIT**

Ej: Hotel Sol Ej: 123456789

Ciudad **Dirección**

Ej: Medellín Ej: Calle 10 #20-30

Teléfono **Tipo de establecimiento**

Ej: 3101234567 Ej: Hotel, Hostal...

Descripción

Ej: Alojamiento familiar cerca al centro...

Datos del primer usuario empleado "Admin"

Nombre completo

Ej: Ana Gómez

Nombre de usuario

Ej: anagomez

Clave

Mínimo 6 caracteres

Confirmar Clave

Vuelva a escribir la clave

Registrar

Activar Windows
Ve a Configuración para activar Windows

Nota. Formulario para registrar de primer vez el establecimiento y usuario Admin, como primer paso para obtener las credenciales para ingresar al sistema web gestor de reservas y alojamientos.

¿Cómo iniciar sesión en el sistema?

Si completo con éxito el registro del establecimiento y del usuario empleado con rol "Admin", proceda a iniciar sesión en el "sistema gestor de alojamientos". Escriba los datos **nombre de usuario** y **clave** en los campos del formulario y hacer clic en el botón "Entrar". Si los datos son correctos se entra al gestor de alojamientos, pero si uno o ambos datos son incorrectos se mostrará un mensaje de error como se ve en la figura 2.

Figura 2*Formulario Inicio de Sesión*

DigiHostal
Gestor de alojamientos y reservas

Inicio de sesión

Campos del formulario

Usuario
Escriba el usuario

Clave
Escriba la clave

Entrar

[Recuperar mi contraseña](#)

[Regístra el establecimiento aquí](#)

DigiHostal
Gestor de alojamientos y reservas

Inicio de sesión

Usuario o password incorrectos

Usuario
patriciaPopayan

Clave
.....

Entrar

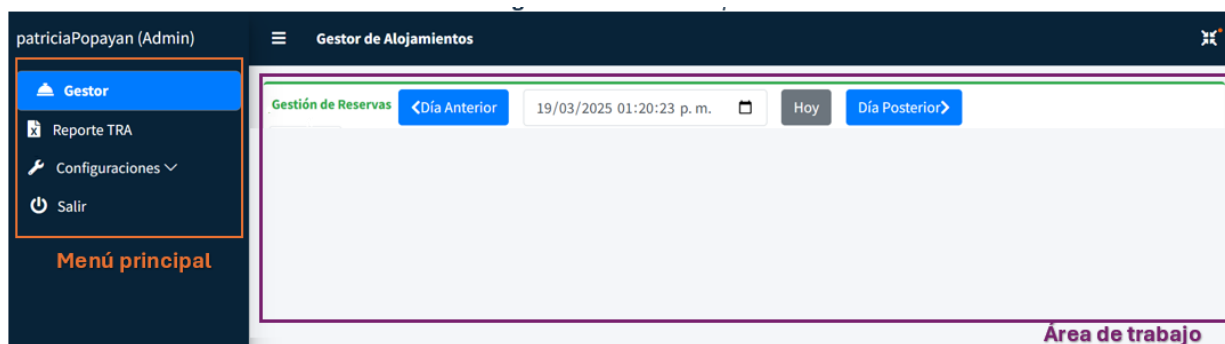
[Recuperar mi contraseña](#)

[Regístra el establecimiento aquí](#)

Nota. Formulario de inicio de sesión.

Pantalla principal del sistema:

Al entrar al “sistema gestor de alojamientos” con usuario y clave, se debe observar la pantalla principal del sistema, donde se encuentra el “Área de trabajo” y el “Menú principal” con las opciones Gestor, Reporte TRA y Configuración como se muestra en la figura 3.

Figura 3*Área de Trabajo “Gestor de Reservas” y Menú Principal*

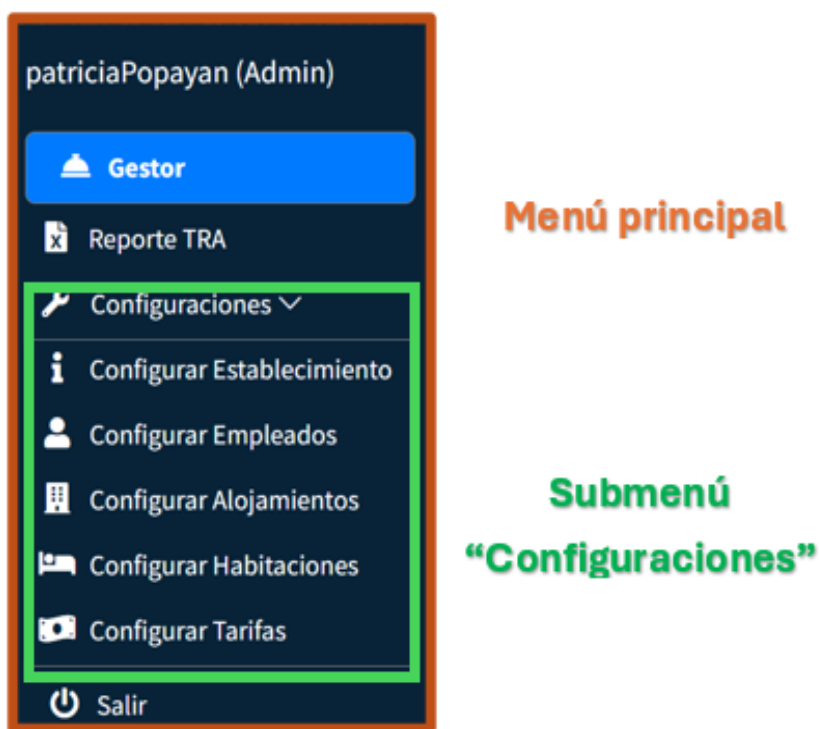
Nota. Área de trabajo “Gestor de Reservas” listo para configurar.

¿Cómo configurar el “Área de trabajo” del sistema?

La primera vez que se ingresa al sistema, es necesario configurar el "Área de Trabajo" (figura 3). Esto se hace registrando los datos del establecimiento, empleados, alojamientos, habitaciones y tarifas. Esta información es la base para gestionar las reservas, y se puede registrar al hacer clic en las opciones del submenú "Configuración" del “Menú Principal”.

Figura 4

Submenú desplegable “Configuraciones”, ubicado dentro del “Menú Principal”



Nota. Menú para configurar los datos del sistema.

¿Cómo editar los datos del establecimiento en el sistema?

En el “Menú Principal” (figura 4) haga clic en **Configurar Establecimiento** para entrar al área de trabajo “Establecimiento”, que muestra el formulario donde se muestran los datos del establecimiento previamente registrados. Para cambiar algún dato puede borrar la información de cualquier campo, haga clic en el botón “Registrar” para guardar como se muestra en la figura 5.

Figura 5

Área de trabajo “Establecimiento”

patriciaPopayan (Admin)
Hostal Principal

Gestor de Alojamientos y Reservas

Datos del establecimiento

Nombre	Hostal Principal	NIT	00000001
Ciudad	Murillo - Tolima	Dirección	Calle 09 #01-01 Barrio Centro
Telefono	3100000011	Tipo de establecimiento	Hostal

Descripción
Confortantes habitaciones multiples con baño privado, agua cliente, Wifi

Actualizar

Nota. Área de trabajo “Establecimiento” con los datos previamente registrados.

¿Cómo registrar los datos de los empleados con rol “Recepcionista” en el sistema?

Si completó con éxito el registro de datos del establecimiento y usuario con rol “Admin”, proceda a registrar los datos de los empleados con rol “Recepcionista”. Haga clic en **Configurar Empleados** en el submenú “Configuraciones” para entrar al área de trabajo “Empleados”, que muestra el listado de empleados registrados y los botones que abren los formularios de registro, actualización y eliminación de información como se muestra en la figura 6.

Figura 6

Área de trabajo “Empleados” Datos de Ejemplo

patriciaPopayan (Admin)

Gestor de Alojamientos

Empleados + *Abrir formulario para registrar empleado*

Show 10 entries Buscar empleado Search:

Lista de empleados registrados

Nombre completo	Nombre de Usuario	Rol	Establecimiento	Acciones
Martin Jimenez	Martin1234	Recepcionista	Hostal Principal	+ ✎ ✖ <i>Abrir formulario para eliminar empleado</i>
Patricia Popayan	patriciaPopayan	Admin	Hostal Principal	+ ✎ ✖ <i>Abrir formulario para actualizar datos empleado</i>

Showing 1 to 2 of 2 entries Previous 1 Next

Nota. Área de trabajo “Empleados” con los botones de acciones.



Para **registrar** un empleado, haga clic sobre  ubicado en la parte superior del área de trabajo “Empleados”, que abre el formulario, completar los campos y hacer clic el botón “registrar” para guardar la información como se ve en la figura 7 lado izquierdo. No dejar campos vacíos. Para **actualizar** datos, haga clic sobre  ubicado frente al empleado de la lista que abre el formulario de la figura al lado derecho, realice los cambios y haga clic en el botón “Actualizar” para guardar la información.


Figura 7

Formulario de Registro y Edición de Datos Usuario Empleado



La imagen muestra dos formularios de usuario. El formulario de la izquierda, titulado "Registro de empleados", contiene campos para "Nombre completo", "Nombre de Usuario", "Clave", "Confirmar Clave", "Rol" (con un menú desplegable que muestra "Recepcionista") y "Establecimiento" (con un menú desplegable que muestra "Hostal Principal"). Un botón verde "Registrar" está ubicado al final. El formulario de la derecha, titulado "Editar empleado", muestra los datos de un empleado existente: "Nombre completo" (Martin Veloza), "Nombre de Usuario" (Martin123), "Rol" (Recepcionista) y "Establecimiento" (Hostal Torres Blancas). Un botón verde "Actualizar" está ubicado al final.

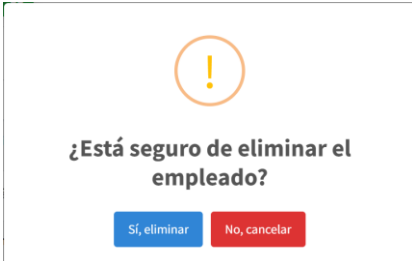
Nota. Formulario para registrar y editar los empleados.

Para eliminar un empleado, haga clic en  ubicado frente al empleado de la lista, que abre un mensaje de confirmación con las opciones: “Si, eliminar”, o “No, cancelar” (figura 8).

Nota. Solo empleado con rol “Admin” tiene permiso para realizar la eliminación.

Figura 8


Mensaje de Confirmación para Eliminar Usuario Empleado




El mensaje de confirmación muestra un ícono de advertencia (un signo de exclamación dentro de un círculo naranja) en la parte superior. Debajo del ícono, el texto pregunta: "¿Está seguro de eliminar el empleado?". Al final del mensaje, hay dos botones: uno azul con el texto "Sí, eliminar" y uno rojo con el texto "No, cancelar".

Nota. Mensaje de alerta cuando se desea eliminar un empleado.


¿Cómo registrar los datos del alojamiento en el sistema?

Si completó con éxito el registro de datos del establecimiento y empleados, proceda a registrar los tipos de alojamientos. Haga clic en el botón  **Configurar Alojamientos** del submenú “Configuración” para entrar al área de trabajo “Alojamientos”. Siga las instrucciones que se explican en el ítem “¿Cómo registrar los datos de los empleados con rol “Recepcionista” en el sistema?”.


¿Cómo registrar los datos de las habitaciones en el sistema?

Si completó con éxito el registro de datos del establecimiento, empleados y alojamientos, proceda a registrar las habitaciones. Haga clic en  **Registrar Habitaciones** del submenú “Configuración” para entrar al área de trabajo “Habitaciones”. Siga las instrucciones que se explican en el ítem “¿Cómo registrar los datos de los empleados con rol “Recepcionista” en el sistema?”.

¿Cómo registrar los datos de las habitaciones en el sistema?

Si completó con éxito el registro de los datos del establecimiento, empleados, alojamientos y habitaciones, proceda a registrar las tarifas. Haga clic  **Registrar Tarifas** del submenú “Configuración” para entrar al área de trabajo “Tarifas”. Siga las instrucciones que se explican en el ítem “¿Cómo registrar los datos de los empleados con rol “Recepcionista” en el sistema?”.

¿Cómo registrar los datos de las reservas en el sistema?

Si completó con éxito el registro de los datos del establecimiento, empleados, alojamientos, habitaciones y tarifas, proceda a registrar las reservas. Haga clic en  **Gestor** del “Menú Principal” para entrar al área de trabajo “Gestión de Reservas”, que muestra los tipos

alojamientos, las habitaciones y los botones que abren los formularios de registro, actualización, traslado y finalización de la reserva (figura 9).

Figura 9

Área de Trabajo “Gestión de Reservas”



Nota. Área de trabajo “Gestión de reservas”.

Las habitaciones libres se muestran en color verde y las ocupadas en color rojo con los datos de la reserva.



Para **registrar reserva para huésped principal**, haga clic en  de la habitación que está libre, o en  de la habitación que está ocupada si va a reservar en otra fecha disponible, que abre el formulario (figura 10), completar los campos. Si el huésped ya ha hecho una reserva antes, puede escribir el “número de identificación” y se llenarán los campos automáticamente, haga clic el botón “Registrar” para guardar la información.

Figura 10*Formulario de Registro de Reserva Huésped Principal*

Registrar reserva para huésped principal

Reserva R0016	Acomodación 101 - Doble	Tarifa III - \$140000	Acompañantes : 2	Tipo de Identificación Tipo de identificación	No. Identificación Número
Nombres Nombres	Apellidos Apellidos	Teléfono Teléfono	Empresa Empresa	Ciudad de Residencia Ciudad	Ciudad de Procedencia Ciudad
Tarifa de parqueadero					
Noche 0	Por días 0	Por horas 0	Motos 0	Carros 0	Valor Parqueadero \$ 0
Motivo del viaje Motivo del viaje	Check-in dd/mm/aaaa --:-- -----	Check-out dd/mm/aaaa --:-- -----	Estado Estado de la reserva	Valor Total \$	

Registrar

Nota. Formulario para registrar los datos del huésped principal y reserva.


Si completo con éxito el registro de la reserva para huésped principal, siga con el **registro de huéspedes acompañantes**. Haga clic en  que abre el formulario (figura 11) y siga la instrucción que se explica anteriormente en “**registrar reserva para huésped principal**”.



Figura 11*Formulario de Registro Huésped Acompañante*

Registro de acompañante

Tipo de Acomodación 100 - Sencilla	Huésped Principal Juan Carlos Aristizabal Marquez	Tipo de Identificación Tipo de identificación	Número de identificación Número
Nombres Ingresar nombres	Apellidos Ingresar apellidos	Telefono Telefono	Ciudad de residencia Ingresar ciudad de residencia
Ciudad de procedencia Ingresar ciudad de procedencia	Check_in dd/mm/aaaa --:-- -----	Check_out dd/mm/aaaa --:-- -----	Estado Estado de la reserva

Registrar

Nota. Formulario para registro de huésped acompañante.

Para **actualizar** los datos del huésped principal haga clic en  y para huésped acompañante en , que abre el formulario de actualización, realice los cambios y haga clic en el botón “Actualizar” para guardar la información. Si existe alguna duda, siga las instrucciones se explican en el ítem “¿Cómo registrar los datos de los empleados con rol “Recepcionista” en el sistema?”.

¿Cómo trasladar los huéspedes de habitación?


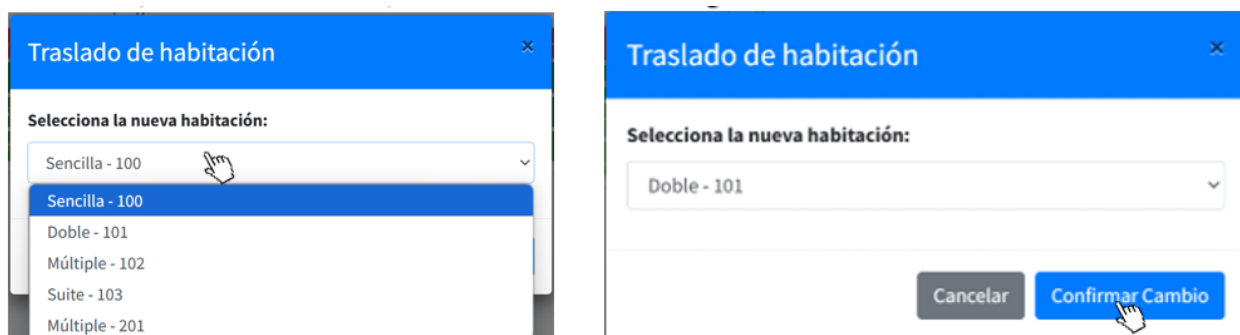
Haga clic en , que abre el formulario de la figura 12, desplegar las opciones del campo “Seleccionar nueva habitación”, elija una y haga clic en el botón “Confirmar cambio”.


Figura 12

Formulario de Registro Huésped Acompañante



Nota. Formulario de traslado de habitación.

¿Cómo finalizar la reserva?

Para finalizar (check-out) la reserva de todos los huéspedes haga clic en , que abre un formulario con los huéspedes marcados para salir (recuadro azul). Luego, presione el botón “Finalizar Reserva” como se muestra en la figura 13.

Si algún huésped se retira antes de la fecha de check-out, quite la marca del recuadro azul en los huéspedes que van a continuar en la reserva. Así, solo el huésped que queda marcado será el que salga de la reserva. Después, haga clic en “Finalizar Reserva”.

Figura 13

Formulario de Finalizar (Check-out) de la Reserva

Nota. Formulario para finalizar reservas.

¿Como ver la disponibilidad de habitaciones en el sistema?



Haga clic en  de la habitación ocupada o en  de la habitación sin ocupación, que abre el formulario donde se puede ver por mes cual es la ocupación y disponibilidad de la habitación.

Figura 14

Formulario de Disponibilidad por Habitación

Nota. Formulario para ver ocupación por mes de cada habitación.

Otra forma de ver la disponibilidad de habitaciones es haga clic en [< Día Anterior](#) 01/05/2025 02:23:21 p. m. [Hoy](#) [Día Posterior >](#) ubicados en la parte superior del área de trabajo “Gestión de Reservas”, se verá las reservas del día posterior en todas las habitaciones o día anterior. Con cada clic avanza 24 horas. Para volver al día presente haga clic en el botón [Hoy](#) .

¿Cómo generar el reporte TRA en el sistema?

Haga clic en [Reporte TRA](#) del “Menú Principal” para entrar al área de trabajo “Generar Reporte TRA”, seleccionar el mes desde el campo “Seleccionar el mes” y haga clic sobre [📄](#), que inicia la descarga del archivo EXCEL en su computador como se muestra en la figura 15.

Figura 15

Generar Descarga del Reporte TRA



Nota. Generar reporte TRA.

Apéndice B

Resumen Analítico Especializado RAE

RESUMEN ANALITICO ESPECIALIZADO RAE	
1. Información General	
Tema	Sistemas de gestión. Desarrollo de sistema web
Título	Diseño e implementación de sistema web gestor de alojamiento con reporte TRA para prestadores de servicios turísticos en Murillo Tolima.
Autor	Jeimy Patricia Popayan Valencia
Asesor	Omar Leonardo Leyton Mendieta
Fuentes Bibliograficas	<p>Se utilizaron 39 referencias bibliográficas, entre las principales utilizadas en el desarrollo del proyecto son:</p> <p>Briano, C. (2023). Compilación de apuntes sobre conceptos fundamentales de la Ingeniería de Software. 2° Edición. http://bibliotecadigital.econ.uba.ar/download/libros/Briano_compilacion_apuntes.pdf</p> <p>Cálad, A. y Ruíz, J. (2009). Metodologías de testing de software y su aplicación en el centro de informática de la universidad EAFIT. Universidad EAFIT. Medellín - Colombia. https://repository.eafit.edu.co/server/api/core/bitstreams/7c41761e-d229-47b9-8855-a8d2650d62d9/content</p>

	<p>Lopez Romo, H. (1998). La metodología de encuesta. biblioteca.marco.edu.mx. https://biblioteca.marco.edu.mx/files/metodologia_encuestas.pdf</p> <p>Sommerville, I. (2011). Ingeniería de Software. 9 edición. Pearson educación, México, 2011. https://gc.scalahed.com/recursos/files/r161r/w25469w/ingdelsoftwarelibro9_compressed.pdf</p> <p>Tramincit (2024). Manual para prestadores de servicios de alojamiento turístico usuarios no conectados. https://tramincit.gov.co/tarjeta/media/docs/Manual-TRA-Usuarios-NO-CONECTADOS-23-04-2023_1.pdf</p> <p>Trigas Gallego, M. (2012). Metodología Scrum. Repositorio Institucional Universidad Abierta de Cataluña. https://openaccess.uoc.edu/bitstream/10609/17885/1/mtrigasTFC0612memoria.pdf</p> <p>Yahaya N. y Maidin, S. (2022). El modelo en cascada con Agile Scrum como modelo ágil híbrido para el equipo de ingeniería de software. Conferencia: 10.^a Conferencia Internacional sobre Gestión de Servicios Cibernéticos y de TI (CITSM) 2022. https://www.researchgate.net/publication/365336966_The_Waterfall_Model_with_Agile_Scrum_as_the_Hybrid_Agile_Model_for_the_Software_Engineering_Team</p>
Año	2025

Resumen	<p>Este proyecto propone diseñar e implementar un sistema web para la gestión del servicio de alojamiento turístico como herramienta indispensable para los pequeños y nuevos empresarios en el municipio de Murillo Tolima, zona que se destaca por su creciente desarrollo y que requiere por parte de los comerciantes cumplir con la normatividad vigente en cuanto al registro y reporte de los visitantes a través el sistema TRA (Tarjeta de Registro de Alojamiento) del Ministerio de Comercio, Industria y Turismo.</p> <p>El soporte teórico y de diseño se basa en la aplicación de tecnologías de software como PHP puro, JavaScript, HTML5, Bootstrap 5 y bases de datos relacionales MySQL, utilizando un proceso de desarrollo secuencial que garantiza un progreso organizado y controlado. Los propósitos incluyen mejorar la automatización de tareas, optimizar la gestión de información y desarrollar un sistema accesible y efectivo.</p> <p>El método propuesto abarca etapas detalladas de recopilación y análisis de información, diseño, desarrollo, pruebas, implementación y mantenimiento. Se espera como resultado la entrega de un sistema funcional que mejore la gestión de registros y la creación del reporte TRA, beneficiando tanto a los prestadores de servicios turísticos como al Ministerio, al facilitar la supervisión del sector en la región.</p>
Palabras Clave	Sistema web, software, TRA (Tarjeta de Registro de Alojamiento), prestadores de servicios turísticos.
Contenido	<p>Introducción..... 17</p> <p>Planteamiento del Problema 18</p>

Objetivos.....	19
Objetivo General	19
Objetivos específicos.....	19
Justificación.....	20
Alcance	22
Límites del Proyecto.....	22
Marco Referencial	23
Antecedentes.....	23
Marco Conceptual	24
Ingeniería de Requerimientos.....	24
Establecimientos de Alojamientos Turísticos	25
Tarjeta de Registro de Alojamiento -TRA.....	25
Sistemas de Gestión.....	25
Marco Teórico.....	26
Metodología Hibrida Cascada y Agile Scrum	26
Diseño del Sistema	27
Lenguajes de Programación y Herramientas de Desarrollo	28
Bases de Datos Relacionales	29
Servidor Local y Entorno de Desarrollo.....	29
Impacto y Pertinencia	30
Bases Legales	30
Metodología.....	32
Fase 1: identificación de los requerimientos	32

	Fase 2: Diseño	33
	Fase 3: Implementación.....	34
	Fase 4: Pruebas	34
	Fase 1: Identificación de Requerimientos y necesidades	35
	Etapas de Construcción del Instrumento: Encuesta.....	36
	Identificación del Objetivo del Estudio	37
	Desglose del Objetivo en Dimensiones e Indicadores	38
	Elaboración de Secuencia.....	39
	Introducción y Solicitud de Colaboración	39
	Selección o Filtro del Respondente	39
	Preguntas Relacionadas con Objetivos.....	40
	Preguntas Demográficas	41
	Agradecimiento e Informe sobre Posible Supervisión	41
	Datos de Identificación, como Nombre y Dirección, para Posterior Supervisión.....	42
	Redacción de Preguntas.....	42
	Identificación y Redacción de Alternativas de Respuesta	45
	Identificación de Pases y Dependencias.....	52
	Redacción de Apoyos y Tarjetas.....	52
	Redacción de Instrucciones	53
	Instrucciones Generales para la Encuesta.....	53
	Instrucciones Específicas.....	54
	Evaluación del Instrumento	54

Diseño de la Muestra	56
Análisis y Lectura de Datos.....	56
Análisis de las Respuestas	56
Conclusiones del Análisis de Resultados	70
Identificación de Requerimientos.....	71
Requerimientos de Descripción General	71
Usuarios Finales	72
Restricciones.....	72
Requerimientos de Interfaces	72
Interfaz de Usuario	73
Interfaz de Hardware	73
Interfaz de Software	74
Interfaces de Comunicaciones.....	74
Requerimientos Funcionales	74
Requerimientos no Funcionales	75
Casos de uso	77
CU-01 Usuario Administrador	77
CU-02 Usuario Recepcionista	78
Escenarios de Casos de Uso	79
Escenario 01 Caso de Uso Actores Administrador y Recepcionista.....	79
Escenario 02 Caso de Uso Actor Administrador	80
Escenario 03 Caso de Uso Actor Administrador	81

	Escenario 04 Caso de Uso Actor administrador	82
	Escenario 05 Caso de Uso Actores administrador y Recepcionista.....	83
	Fase 2 - Diseño	86
	Base Estructural.....	86
	Modelado de Contexto	87
	Modelado Dirigido por Datos.....	87
	Modelo de la Estructura de Datos	99
	Arquitectura del Sistema Web	101
	Consideraciones de Seguridad y Calidad del Software	102
	Directrices de Accesibilidad Web WCAG.....	103
	Prototipos de Interfaz de usuario	104
	Fase 3 - Implementación	115
	Implementación del Código para el Sistema	116
	Planificación de los Sprints	116
	Primer Sprint: Arquitectura y Autenticación	117
	Sprint Review - Primer Sprint	119
	Segundo Sprint: Módulos Críticos del Sistema.....	119
	Sprint Review - Segundo Sprint	121
	Tercer Sprint: Carga Dinámica de Reservas	122
	Sprint Review - Tercer Sprint	124
	Cuarto Sprint: Check-out y TRA.....	124
	Sprint Review - Cuarto Sprint	127

	Quinto Sprint: Optimización del Sistema.....	127
	Sprint Review - Quinto Sprint.....	129
	Fase 4 - Pruebas.....	130
	Pruebas Unitarias.....	131
	Prueba Unitaria Estructura del Proyecto	132
	Prueba Unitaria RecepcionesModel	133
	Pruebas Unitaria Controlador Recepción	135
	Pruebas de Integración	136
	Pruebas de Integración Estructura del Proyecto	136
	Pruebas de Integración RecepcionesModel.....	137
	Pruebas de Integración Controlador Recepción	139
	Conclusión General de las Pruebas Unitarias y de Integración.....	141
	Prueba Piloto	142
	Fase 1: Definir el Alcance de Prueba Piloto	143
	Fase 2: Determinar las Categorías de Usabilidad a Probar	144
	Fase 3: Diseñar Cuestionario de Evaluación	145
	Fase 4: Ejecución de la Prueba Piloto	147
	Fase 5: Análisis de los Resultados.....	154
	Conclusiones.....	156
	Recomendaciones	159
	Referencias Bibliograficas.....	160
	Apéndices	165
	Manual de Usuario	165

2. Planteamiento Del Problema

En Colombia, el Ministerio de Comercio, Industria y Turismo exige a todos los establecimientos como hoteles, apartahoteles, hostales, centros vacacionales, campamentos, glampings, refugios, albergues, viviendas turísticas, etc., la emisión de la Tarjeta de Registro de Alojamiento (TRA) de cada uno de sus huéspedes. Este proceso es fundamental para el control y la legalidad del sector turístico en el país. Sin embargo, en muchas zonas turísticas, especialmente en áreas rurales, de difícil acceso o en proceso de desarrollo turístico se presentan dificultades de conectividad, acceso tecnológico, capacitación y formación en servicios turísticos, representando obstáculos significativos por superar para cumplir con las normativas y exigencias del servicio.

En Murillo Tolima, los establecimientos turísticos carecen de herramientas tecnológicas que les permitan gestionar de manera eficiente el registro de sus huéspedes y la generación de la TRA debido a su limitada infraestructura tecnológica. Esto no solo dificulta la operatividad diaria, sino que también los expone a sanciones legales por parte del Estado al no cumplir con las regulaciones vigentes.

El problema radica en la necesidad de desarrollar un sistema amigable que permita a los establecimientos turísticos tener una herramienta para gestionar y registrar a sus huéspedes, y generar la TRA, asegurando la sincronización y reporte de los datos ante el Ministerio de Comercio, Industria y Turismo de manera fácil y oportuna.

En conclusión, la pregunta central que motiva esta propuesta es: ¿Cómo lograr construir un sistema web que permita la gestión del alojamiento y la generación del reporte TRA como

herramienta para los prestadores de servicios turísticos en Murillo Tolima, facilitando el cumplimiento de las normas legales y mejorando la eficiencia operativa del sector?

3. Objetivos

Objetivo General

Diseñar e implementar un sistema web para la gestión del alojamiento como herramienta para los prestadores de servicios turísticos cumpliendo la normatividad vigente TRA en el municipio de Murillo Tolima.

Objetivos específicos

- Identificar los requerimientos y necesidades de los prestadores de servicios de alojamiento turístico conforme al contexto regional y a la normatividad vigente;
- Diseñar el sistema web para la gestión y control de visitantes, y la generación del reporte TRA conforme la normatividad vigente;
- Implementar el sistema web para la gestión de visitantes;
- Probar el sistema en un entorno real en la prestación del servicio de alojamiento turístico.

4. Metodología

La construcción del sistema web gestor de alojamiento y huéspedes empleará el enfoque metodológico híbrido en desarrollo de software combinando el modelo en cascada con el modelo ágil: Scrum, organizando el proceso en fases secuenciales, asegurando un avance controlado y sistemático en el ciclo de vida del proyecto, pero integrando conceptos ágiles en las fases finales. Según Praxium Management Group “Una forma de combinar estas metodologías es utilizar la de cascada en las fases iniciales del proyecto, como la planificación,

y luego cambiar a la ágil en las fases posteriores, como la implementación y la prueba (PMG, 2023)” priorizando el modelo en cascada. Este enfoque híbrido permite una planificación detallada y un control riguroso de cada etapa, garantizando la calidad y la coherencia del producto final.

5. Principales referentes conceptuales

Ingeniería de Requerimientos

La ingeniería de requerimientos es una etapa fundamental dentro del desarrollo de software ya que “es el proceso de interactuar con los participantes del sistema para descubrir sus requerimientos” (Briano, 2023, p. 81), con el fin de identificar, analizar, documentar y gestionar las necesidades y expectativas de los usuarios respecto al sistema que se va a construir. Esta fase ayuda a tener una visión clara de lo que el usuario final necesita, permitiendo diseñar soluciones que realmente resuelvan sus problemas y cumplan con los objetivos del proyecto. Además, una buena gestión de requerimientos reduce errores y mejora la calidad final del software.

Sistemas de Gestión

Son plataformas de software utilizados para la administración y automatización de los procesos de negocios de las empresas. El objetivo es facilitar el control de actividades, como la administración de inventarios, ventas, reservas, clientes, personal, contabilidad, entre otras, dependiendo del tipo de organización, “simplifican los procesos operativos, productivos y burocráticos de una organización” (Siigo, 2018). Entre las características más importantes de estos softwares se incluyen la integración con otros sistemas, escalabilidad, usabilidad, el soporte técnico, flexibilidad y personalización, así también como el análisis y generación de reportes.

6. Principales Referentes Teóricos

Arquitectura del Sistema: El diseño del sistema se basará en el patrón de arquitectura Modelo-Vista-Controlador (MVC) "favorece la separación de responsabilidades y la escalabilidad del sistema" (Sommerville, 2016) facilitando su mantenimiento y escalabilidad. Se elaborarán diagramas de flujo, diagramas de entidad-relación (ERD), y maquetas de la interfaz de usuario utilizando herramientas como UML (Unified Modeling Language).

Lenguajes de Programación y Herramientas de Desarrollo: Los lenguajes a utilizar para la implementación del sistema son: PHP y JAVASCRIPT, y como herramientas de desarrollo se utilizarán HTML5, CSS, BOOTSTRAP 5. Esta combinación permite construir una aplicación web dinámica y responsiva, donde PHP se encargará de la lógica del lado del servidor y la interacción con la base de datos, mientras que JavaScript facilitará la interactividad en el lado del cliente. HTML5 y CSS proporcionarán la estructura y el estilo visual del sistema, y Bootstrap 5 contribuirá con componentes reutilizables y un diseño adaptable para mejorar la experiencia del usuario en distintos dispositivos.

7. Resultados

Como resultado de este proyecto se destaca lo siguiente:

- Informe de tesis de grado para la modalidad “proyecto aplicado”, con el contenido referente al desarrollo del proyecto de investigación.
- Desarrollo del sistema web gestor de alojamientos.

8. Conclusiones

En cumplimiento del primer objetivo para identificar los requisitos y necesidades, se diseñó y aplico con éxito el instrumento de encuesta que permitió identificar la viabilidad y necesidad en los prestadores de servicios turísticos de contar un sistema para gestionar los alojamientos.

Los resultados de los datos recolectados permitieron obtener un panorama detallado de las capacidades tecnologías y habilidades digitales suficientes para implementar un sistema web de este tipo. Además, se logró identificar los principales elementos que deben contemplarse en el diseño del sistema, tales como los tipos de alojamiento más comunes, la capacidad de camas y personas por habitación, horarios flexibles, tarifas de reserva, y servicios complementarios todo ello detallado en el apartado “conclusiones del análisis de resultados”.

Estos hallazgos permitieron estructurar los requerimientos clave de descripción general, legales, de interfaces, funcionales, no funcionales que deben ser aplicado dentro del diseño y desarrollo del sistema, que cumpla con las condiciones detectadas, ajustándose al contexto regional y a la normatividad vigente.

Finalizada la fase 2, relacionada con el segundo objetivo: diseño del sistema web gestor de alojamientos, se obtiene como resultado un prototipo estructurado no funcional donde se organizó la navegación, textos, tablas, botones, formularios, etc., todo alineado con los requerimientos funcionales identificados para la gestión y control de visitantes, y la generación del reporte TRA conforme la normatividad vigente, priorizando la usabilidad, accesibilidad y eficiencia, junto a su respectiva base de datos. Gracias al modelado UML se logra definir las funcionalidades y acciones que realizara el usuario final dentro del sistema y los componentes para la arquitectura modelo vista controlador, visualizando toda la operatividad reduciendo posibles errores antes de la implementación.

Como resultado de la fase 3, correspondiente al tercer objetivo, se logra implementar el código del sistema web para la gestión de visitantes, materializando en funcionalidades los requerimientos y diseño del sistema, aplicando como estrategia cortos periodos de tiempo con marco Scrum de la metodología ágil. Se logra iterar sobre cada historia y tareas específicas, facilitando semana a semana la entrega de módulos operativos: alojamientos, habitaciones, gestión de reservas, tarifas, establecimiento, usuario empleado, generación de la TRA y demás acciones necesarias. Los resultados obtenidos hasta el momento evidencian la coherencia y el cumplimiento del sistema con los objetivos planteados.

Dando cumplimiento al cuarto objetivo, los resultados obtenidos en la fase 4 pruebas, permiten afirmar que el desarrollo del sistema alcanzó un nivel óptimo de funcionamiento tanto a nivel técnico como de experiencia de usuario. Por un lado, las pruebas unitarias y de integración realizadas con PHPUnit evidenciaron que los métodos desarrollados responden correctamente ante diferentes entradas y escenarios, sin generar errores ni comportamientos inesperados; además, se comprobó una adecuada interacción entre los componentes del sistema, lo que refleja una arquitectura bien estructurada. Por otro lado, los resultados de la prueba piloto aplicada en un entorno real en la prestación del servicio de alojamiento turístico mostraron una percepción mayoritariamente positiva, destacando que las 9 funcionalidades clave obtuvieron una calificación promedio de 4.8/5.0, lo que indica una alta usabilidad, sugiriendo áreas específicas donde se pueden aplicar mejoras en la experiencia de interacción. De igual manera, en cuanto a la percepción general del sistema con un 4.6/5.0, se resaltó la estabilidad en la respuesta del software, mientras que aspectos como la adaptación a las necesidades del usuario y la recomendación del sistema también obtuvieron una valoración destacada.

En conclusión, se confirma el cumplimiento del objetivo general de diseñar e implementar un sistema web para la gestión del alojamiento turístico en Murillo Tolima, conforme a la normatividad vigente, aportando una solución pertinente para la modernización del sector.

Enfoque organizacional: el desarrollo de este proyecto demostró que una adecuada planeación combinada con disciplina y perseverancia permitió el desarrollo de los objetivos trazados en un tiempo relativamente corto, lo cual crea una visión sistemática orientada para enfrentar grandes retos, superar limitaciones y materializar metas.

Enfoque social: el desarrollo de proyectos sociales permite crear empatía con las comunidades donde se necesitan soluciones que mejoren la calidad de vida de las comunidades, despertando sensibilidad para entender que la creación de software no sólo es un ejercicio técnico, también genera lazos humanos dando oportunidad al servicio social y comunitario.

Enfoque empresarial: el proyecto demostró que es posible transformar necesidades sociales en oportunidades de emprendimiento tecnológico, sentando las bases para impulsar iniciativas que contribuyan a la generación de empresa para el desarrollo local y regional.

Enfoque profesional: este proyecto fortaleció habilidades técnicas y de solución de problemas, aportando una visión más consciente sobre el rol del desarrollador de software como agente de cambio en entornos sociales, utilizando la tecnología digital para mejorar el día a día de las personas, respondiendo a necesidades específicas.