**“Diseño e implementación de una aplicación web para la gestión personalizada de itinerarios turísticos en agencias de viajes del Cusco”**

; &

**1 Escuela Académica Profesional de Ingeniería de Sistemas e Informática, Universidad Continental, Cusco, Perú.**

**RESUMEN EJECUTIVO**

El proyecto “Aplicación Web para la Gestión de Itinerarios Turísticos en la Empresa Panorama en Cusco, 2025” tuvo como objetivo principal desarrollar una solución digital que permita optimizar la administración de itinerarios turísticos, centralizar la información relevante de los visitantes y mejorar la experiencia antes, durante y después del viaje. La empresa Panorama, especializada en ofrecer servicios turísticos tanto a visitantes nacionales como internacionales, enfrentaba varios desafíos. La información estaba registrada de manera dispersa, lo que complicaba la comunicación operativa y requería actualizar los datos de forma manual. Esto causaba retrasos y errores en la gestión de itinerarios.

Para cubrir estas necesidades, se desarrolló una aplicación web que sigue una arquitectura cliente-servidor. En el frontend se usó React.js, mientras que el backend se construyó con Node.js y Express. La base de datos se gestiona con MySQL. La aplicación ofrece varias funcionalidades, como un sistema de administración de usuarios, la gestión de turistas, la creación y edición de itinerarios, notificaciones, un panel con estadísticas y un panel exclusivo para turistas. Este último permite que los turistas vean su itinerario, reciban alertas y descarguen su planificación en formato PDF.

Durante el desarrollo, se realizaron pruebas funcionales, de validación de datos y de experiencia de usuario. Como resultado, se logró un buen rendimiento en la creación, actualización y consulta de itinerarios, además de una mejora en la organización de la información turística. La aplicación contribuye directamente a los Objetivos de Desarrollo Sostenible (ODS), especialmente al ODS 8 (Trabajo decente y crecimiento económico) y al ODS 9 (Industria, innovación e infraestructura), al promover la digitalización responsable en el sector turístico.

Palabras clave: Gestión de itinerarios, Turismo digital, Aplicación web.

**Abstract**

The project “Web Application for Managing Tourist Itineraries at Panorama in Cusco, 2025” aimed to develop a digital solution to optimize itinerary management, centralize relevant visitor information, and improve the experience before, during, and after the trip. Panorama, a company specializing in providing tourist services to both domestic and international visitors, faced several challenges. Information was stored in a fragmented manner, complicating operational communication and requiring manual data updates. This resulted in delays and errors in itinerary management.

To address these needs, a web application was developed using a client-server architecture. React.js was used for the frontend, while the backend was built with Node.js and Express. The database is managed with MySQL. The application includes several functionalities, such as a user management system, tourist management, itinerary creation and editing, notifications, a statistics dashboard, and a dedicated dashboard for tourists. This feature allows tourists to view their itinerary, receive alerts, and download it in PDF format.

During development, functional testing, data validation, and user experience tests were carried out. As a result, the application achieved good performance in creating, updating, and retrieving itineraries, while improving the organization of tourist information. The application directly contributes to the Sustainable Development Goals (SDGs), particularly SDG 8 (Decent Work and Economic Growth) and SDG 9 (Industry, Innovation and Infrastructure), by promoting responsible digitalization in the tourism sector.

Keywords: Itinerary management, Digital tourism, Web application.

**INDICE**

[1. INTRODUCCIÓN 5](#_Toc215652894)

[Antecedentes 5](#_Toc215652895)

[1.1.1. Antecedentes Internacionales 5](#_Toc215652896)

[1.1.2. Antecedentes Nacionales 5](#_Toc215652897)

[1.1.3. Antecedentes Locales (Cusco) 6](#_Toc215652898)

[Identificación y formulación del Problema 6](#_Toc215652899)

[1.1.4. Contexto 6](#_Toc215652900)

[1.1.5. Problema central 7](#_Toc215652901)

[1.1.6. Consecuencia del problema 8](#_Toc215652902)

[1.1.7. Necesidad 9](#_Toc215652903)

[1.1.8. Justificación 9](#_Toc215652904)

[MARCO TEÓRICO 10](#_Toc215652905)

[1.1.9. Sistemas de Información 10](#_Toc215652906)

[1.1.10. Tecnologías Web 10](#_Toc215652907)

[1.1.11. Frameworks y Tecnologías Utilizadas 10](#_Toc215652908)

[1.1.12. Arquitectura Cliente-Servidor 11](#_Toc215652909)

[1.1.13. Ingeniería de Software 11](#_Toc215652910)

[1.1.14. Metodologías Ágiles 11](#_Toc215652911)

[1.1.15. Turismo y Transformación Digital 12](#_Toc215652912)

[1.1.16. Usabilidad y Experiencia de Usuario 12](#_Toc215652913)

[OBJETIVOS DEL PROYECTO 12](#_Toc215652914)

[OBJETIVO GENERAL: 12](#_Toc215652915)

[OBJETIVOS ESPECÍFICOS: 13](#_Toc215652916)

[2. CONOCIMIENTOS DE INGENIERÍA APLICADOS/ RELACIONADOS 13](#_Toc215652917)

[3. INGENIERO Y LA SOCIEDAD 15](#_Toc215652918)

[4. METODOLOGÍA EMPLEADA 16](#_Toc215652919)

[Identificando el Backlog (ordenado por prioridad): 18](#_Toc215652920)

[Llegando a obtener el Sprint Backlog: 22](#_Toc215652921)

[Elaboración de prototipos: 32](#_Toc215652922)

[4.1.1. Pantalla de Inicio de Sesión 32](#_Toc215652923)

[4.1.2. Pantalla de Registro de Correo Electrónico 32](#_Toc215652924)

[4.1.3. Pantalla de Registro de Datos 33](#_Toc215652925)

[4.1.4. Panel Principal Usuario 34](#_Toc215652926)

[4.1.5. Pantalla de Itinerarios 35](#_Toc215652927)

[4.1.6. Pantalla de Ver Documentos 36](#_Toc215652928)

[4.1.7. Pantalla de Ver Notificaciones 36](#_Toc215652929)

[4.1.8. Pantalla de Centro de Soporte 37](#_Toc215652930)

[5. USO DE HERRAMIENTAS MODERNAS 42](#_Toc215652931)

[6. DISEÑO DE INGENIERÍA 44](#_Toc215652932)

[Listado de requerimientos funcionales 44](#_Toc215652933)

[Diseño de base de datos 45](#_Toc215652934)

[Arquitectura de la solución planteada 46](#_Toc215652935)

[Código de la aplicación por capas 47](#_Toc215652936)

[7. GESTIÓN DEL PROYECTO 48](#_Toc215652937)

[Diagrama de Gantt (Línea base) 48](#_Toc215652938)

[Cronograma de actividades – Ejecutado 53](#_Toc215652939)

[Gráfica de trabajo pendiente en el sprint – Jira 53](#_Toc215652940)

[8. PRUEBAS, RESULTADOS Y DISCUSIÓN 55](#_Toc215652941)

[9. LECCIONES APRENDIDAS 62](#_Toc215652942)

[10. Conclusiones 65](#_Toc215652943)

[11. BIBLIOGRAFÍA 66](#_Toc215652944)

[12. Anexos 67](#_Toc215652945)

**Tabla de figuras**

[Figura 4.1 Diagrama de Caso de Uso 18](#_Toc215653019)

[Figura 4.2 Pantalla de Inicio de Sesión 33](#_Toc215653020)

[Figura 4.3 Pantalla de Registro de Correo Electrónico 34](#_Toc215653021)

[Figura 4.4 Pantalla de Registro de Datos 35](#_Toc215653022)

[Figura 4.5 Pantalla del Panel Principal Usuario 36](#_Toc215653023)

[Figura 4.6 Pantalla de Itinerarios 36](#_Toc215653024)

[Figura 4.7 Pantalla de Ver Documentos 37](#_Toc215653025)

[Figura 4.8 Panalla de Ver Notificaciones 38](#_Toc215653026)

[Figura 4.9 Pantalla de Centro de Soporte 38](#_Toc215653027)

[Figura 4.10 Panel Principal del Administrador 40](#_Toc215653028)

[Figura 4.11 Panel Principal Usuario 41](#_Toc215653029)

[Figura 5.1 GitHub de Trabajo 44](#_Toc215653030)

[Figura 6.1 Modelo de Base de Datos 46](#_Toc215653031)

[Figura 6.2 Arquitectura en Capas 48](#_Toc215653032)

[Figura 6.3 Cronograma por Sprint 49](#_Toc215653033)

[Figura 6.4 Sprint 1 50](#_Toc215653034)

[Figura 6.5 Sprint 2 50](#_Toc215653035)

[Figura 6.6 Sprint 3 51](#_Toc215653036)

[Figura 6.7 Sprint 4 51](#_Toc215653037)

[Figura 6.8 Sprint 5 52](#_Toc215653038)

[Figura 6.9 Sprint 6 52](#_Toc215653039)

[Figura 6.10 Sprint 7 53](#_Toc215653040)

[Figura 6.11 Cronograma de Actividades en Jira 54](#_Toc215653041)

[Figura 6.12 Trabajos Pendientes 55](#_Toc215653042)

[Figura 6.13 Inicio de Sesión 56](#_Toc215653043)

[Figura 6.14 Registro de Usuario y Turista 58](#_Toc215653044)

[Figura 6.15 Gestión de Itinerarios 59](#_Toc215653045)

[Figura 6.16 Gestión de Itinerarios 60](#_Toc215653046)

[Figura 6.17 Lista de Turistas 61](#_Toc215653047)

[Figura 6.18 Itinerario en PDF 62](#_Toc215653048)

[Figura 6.19 Panel de Control del Administrador 63](#_Toc215653049)

# INTRODUCCIÓN

## Antecedentes

### Antecedentes Internacionales

Buhalis y Amaranggana (2015) (1) analizaron cómo los destinos turísticos inteligentes pueden mejorar la experiencia del viajero mediante la digitalización y personalización de servicios. Su estudio en destinos europeos demostró que la integración de aplicaciones web y plataformas de información en tiempo real facilita la organización de itinerarios y mejora la satisfacción del turista. Este aporte resulta pertinente para el presente proyecto, dado que evidencia la eficacia de centralizar datos turísticos en sistemas digitales para optimizar procesos operativos.

Buonincontri y Micera (2016) (2) estudiaron la co-creación de experiencias turísticas en contextos digitalizados. A través de un análisis de casos múltiples en Europa, demostraron que las plataformas digitales ayudan a mejorar la interacción entre turistas y prestadores de servicios, lo cual incrementa la calidad y eficiencia del proceso de planificación de viajes. Esto se relaciona con la necesidad de la empresa Panorama de mejorar la comunicación y la gestión de itinerarios mediante herramientas web.

Esper et al. (2025) (3) desarrollaron un sistema para la generación de itinerarios personalizados a través de una aplicación web, probándolo con turistas reales. La investigación concluye que los sistemas digitales contribuyen significativamente a que los visitantes organicen mejor sus recorridos y dispongan de información precisa sobre sus actividades diarias. Este antecedente respalda la idea de que una plataforma web puede mejorar la accesibilidad y comprensión del itinerario por parte del turista.

Solanki (2020) (4) investigó cómo las aplicaciones web transforman la gestión de itinerarios y reservas en empresas turísticas de Asia y Europa. Sus resultados muestran una reducción de errores, mayor eficiencia administrativa y una mejora notable en la comunicación entre turistas, agencias y proveedores.

### Antecedentes Nacionales

Perdiz Reyna (2019) (5) evaluó la experiencia digital en portales turísticos peruanos, concluyendo que un diseño web intuitivo incrementa significativamente la interacción del usuario, el tiempo de navegación y la intención de compra. Esto demuestra que las plataformas web influyen directamente en la decisión del turista, lo cual es un referente para la empresa Panorama, que busca mejorar la experiencia digital de sus visitantes.

Llashag Vega (2020) (6) estudió la relación entre el uso de sistemas web y el incremento del turismo en el Callejón de Huaylas, encontrando una correlación positiva entre el acceso a información confiable en línea y la decisión de viaje. El estudio destaca la importancia de plataformas que centralicen información turística, elemento clave que se integra en la solución propuesta.

Rayco Valles (2020) (7) implementó un sistema web para mejorar la planificación de visitas en la región San Martín, demostrando que la adopción de herramientas digitales incrementa la eficiencia en la organización de recorridos y la satisfacción del usuario. Este caso refuerza la necesidad de digitalizar procesos turísticos en diferentes regiones del país.

### Antecedentes Locales (Cusco)

Huamán (2018) (8) desarrolló un sistema para la administración de reservas turísticas en la agencia RAP Travel, demostrando cómo la digitalización permitió automatizar tareas operativas y fortalecer el control interno. Este caso evidencia que las agencias locales requieren herramientas que centralicen información y reduzcan procesos manuales.

Garrido Cusipaucar (2021) (9) implementó el sistema Turismo Smart v2.5 en la empresa Perú Jamuy, alcanzando mejoras en la gestión de paquetes turísticos, reducción de tiempos de respuesta y aumento del control interno. Sus conclusiones evidencian que un sistema web correctamente diseñado puede adaptarse a las necesidades específicas del sector turístico cusqueño.

## Identificación y formulación del Problema

### Contexto

La empresa Panorama, ubicada en la ciudad del Cusco, enfrenta dificultades en la gestión y organización de los itinerarios turísticos que ofrece a sus visitantes. La dispersión de información, el uso de procesos manuales y la falta de un sistema centralizado limitan la eficiencia operativa y generan errores en la actualización de actividades. Esta situación afecta la calidad del servicio, dificulta la comunicación con los turistas y reduce la capacidad de la empresa para ofrecer una experiencia personalizada y moderna acorde a las exigencias del sector turístico actual.

### Problema central

La empresa Panorama enfrenta serias limitaciones en la gestión de sus itinerarios turísticos debido a la ausencia de un sistema web centralizado que permita organizar y actualizar la información de manera eficiente. Actualmente, gran parte del proceso se realiza mediante hojas de cálculo, documentos independientes y comunicación manual entre las áreas, lo cual ocasiona dispersión de datos, duplicidad de información y una alta probabilidad de errores operativos.

Estas deficiencias se vuelven aún más críticas en un lugar como Cusco, uno de los destinos turísticos más visitados del país, donde los viajeros requieren información precisa y actualizada de manera constante. La ausencia de una herramienta digital integrada provoca demoras al modificar actividades, inconsistencias en la planificación y dificultades para ofrecer a los turistas información clara y a tiempo sobre sus recorridos. Todo esto termina afectando la experiencia del visitante y perjudicando la imagen profesional de la agencia.

Además, la gestión manual reduce la capacidad de la empresa para hacer un seguimiento adecuado de los cambios, coordinar al personal y mantener un registro ordenado de los itinerarios. Esto genera una operación reactiva, dependiente de la comunicación informal y propensa a errores, lo que se traduce en menor eficiencia interna y en la insatisfacción de los usuarios.

En conjunto, la falta de un sistema web para la administración de itinerarios impide que Panorama optimice sus procesos, mejore su organización y responda a las exigencias actuales del turismo digital, afectando su competitividad y su capacidad de brindar un servicio moderno, confiable y alineado con las necesidades reales del turista.

### Consecuencia del problema

La ausencia de un sistema web centralizado para la gestión de itinerarios turísticos en la empresa Panorama genera diversas consecuencias que afectan tanto la operación interna como la experiencia del visitante. Entre las principales se encuentran:

* **Desorganización operativa**:  
  La información dispersa y los procesos manuales dificultan la coordinación entre las áreas de la empresa, provocando inconsistencias en los itinerarios, pérdida de datos y retrasos en la asignación de actividades.
* **Errores en la información brindada al turista**:  
  Debido a la actualización no uniforme de los itinerarios, es frecuente que los turistas reciban horarios, rutas o actividades desactualizadas, lo que afecta directamente su experiencia y la percepción de calidad del servicio.
* **Demoras en la atención y respuesta**:  
  La falta de una plataforma centralizada obliga al personal a buscar información en múltiples fuentes, lo que incrementa los tiempos de respuesta y dificulta resolver consultas en el momento.
* **Baja satisfacción del turista**:  
  La información incorrecta, los cambios no comunicados y la falta de claridad en el itinerario generan confusión, molestias y disminuyen la confianza del visitante en la agencia.
* **Pérdida de eficiencia y mayor carga laboral**:  
  Los procesos manuales obligan al personal a realizar tareas repetitivas y duplicadas, reduciendo su productividad y aumentando el riesgo de cometer errores.
* **Limitación en la competitividad de la empresa**:  
  En un mercado cada vez más digitalizado, la falta de herramientas tecnológicas modernas coloca a la empresa Panorama en desventaja frente a otras agencias que sí ofrecen sistemas automatizados y plataformas digitales más completas.
* **Dificultad para tomar decisiones basadas en datos**:  
  La ausencia de un registro estructurado impide generar reportes confiables, analizar tendencias y optimizar los servicios ofrecidos, limitando la capacidad de planificación estratégica.

### Necesidad

Es fundamental desarrollar una aplicación web que centralice y organice toda la información relacionada con los itinerarios turísticos de la empresa Panorama. Esta plataforma permitirá gestionar de manera eficiente las actividades, horarios y datos de los visitantes, facilitando la creación, actualización y consulta de itinerarios en tiempo real. Con ello, se reducirá significativamente los errores causados por los procesos manuales y la dispersión de la información.

Además, el sistema debe proporcionar al personal un acceso rápido a los datos necesarios para coordinar los recorridos, comunicar cambios y ofrecer información precisa a los turistas. Contar con una herramienta digital moderna es clave para mejorar la calidad del servicio, agilizar la toma de decisiones y cumplir con las demandas del turismo actual, donde tener información clara y actualizada es crucial.

La implementación de esta solución permitirá a Panorama ser más proactiva, optimizando sus procesos internos y asegurando que los turistas reciban un servicio organizado, confiable y alineado con las nuevas tendencias de digitalización en el sector turístico.

### Justificación

La implementación de TravelDesk responde a la necesidad de modernizar y optimizar la gestión de itinerarios turísticos en la empresa Panorama. Antes, la administración se realizaba a través de hojas de Excel, lo que causaba desorden, duplicación de datos, confusiones y dificultades para coordinar actividades entre el equipo de turismo y los usuarios. Esto impactaba directamente la calidad del servicio y aumentaba el riesgo de cometer errores durante los viajes.

Con TravelDesk, toda la información se centraliza en un sistema web, lo que permite a los administradores crear, modificar y organizar itinerarios de forma rápida y estructurada. Además, integra servicios clave como hoteles, transporte, tours y actividades contratadas. El sistema también envía notificaciones automáticas cuando hay cambios en los itinerarios, mejorando la comunicación con los turistas y asegurando que siempre estén al tanto.

Además, ofrecer a los usuarios un espacio donde puedan ver sus horarios, destinos y servicios, y la opción de descargar esta información en formato PDF para tenerla sin conexión, incrementa la confianza y satisfacción del cliente. Así, los turistas siempre cuentan con información clara y accesible en todo momento.

## MARCO TEÓRICO

### Sistemas de Información

Un Sistema de Información (SI) es un conjunto de elementos interrelacionados que recopilan, procesan, almacenan y distribuyen información con el fin de apoyar las actividades de una organización (Laudon & Laudon, 2020). En el sector turismo, los SI permiten gestionar datos de viajeros, itinerarios, actividades, reservas y comunicaciones operativas.

### Tecnologías Web

Una aplicación web es un software accesible desde un navegador y ejecutado mediante tecnologías web, lo cual facilita su uso en múltiples dispositivos sin necesidad de instalaciones locales (Pressman, 2019). Son ideales para agencias turísticas ya que los turistas acceden a su información de forma remota y en tiempo real.

HTML, CSS y JavaScript

Son las tecnologías base para el desarrollo frrontend:

* HTML5 estructura los contenidos.
* CSS3 define estilos visuales.
* JavaScript implementa interacción dinámica.

### Frameworks y Tecnologías Utilizadas

**Node.js**

Node.js es un entorno de ejecución de JavaScript del lado del servidor basado en el motor V8 de Google. Permite construir aplicaciones escalables y de alto rendimiento gracias a su modelo asíncrono y orientado a eventos (Tilkov & Vinoski, 2010). Fue utilizado para manejar la lógica backend de la aplicación.

**Express.js**

Express es un framework minimalista que facilita la creación de API REST, manejo de rutas, middleware y servicios web (Brown, 2022). En este proyecto gestiona las operaciones CRUD de itinerarios, usuarios y turistas.

**React.js**

React es una librería de JavaScript para construir interfaces basadas en componentes reutilizables (Facebook, 2021). Permite desarrollar frontends interactivos, rápidos y modulares, adecuados para paneles administrativos y vistas de turistas.

**MySQL**

MySQL es un sistema de gestión de base de datos relacional ampliamente utilizado por su eficiencia, seguridad y compatibilidad con aplicaciones web (Silberschatz, 2019). Su estructura tabular permite organizar turistas, itinerarios y usuarios manteniendo la integridad de datos.

**Git y GitHub**

Git es un sistema de control de versiones distribuido que permite gestionar cambios de código de forma segura. GitHub es la plataforma donde se aloja el repositorio del proyecto, facilitando colaboración, control de versiones y despliegue continuo.

**Visual Studio Code**

Visual Studio Code es un IDE ligero y extensible utilizado para escribir, depurar y administrar el código del proyecto. Facilita la integración con Git y múltiples herramientas de desarrollo web.

**Figma**

Figma es una herramienta para el diseño colaborativo de interfaces. Se utilizó para crear prototipos de alta fidelidad y definir la experiencia de usuario (UX) antes de la implementación.

### Arquitectura Cliente-Servidor

La arquitectura cliente-servidor separa las funciones de presentación (cliente) y procesamiento (servidor), permitiendo escalabilidad y separación lógica de responsabilidades (Sommerville, 2016). En el proyecto:

* Cliente: React.js.
* Servidor: Node.js + Express.
* Base de datos: MySQL.

### Ingeniería de Software

La ingeniería de software proporciona principios y prácticas para desarrollar software de calidad. Pressman y Maxim (2021) destacan actividades esenciales como:

* Análisis de requisitos
* Diseño
* Codificación
* Pruebas
* Mantenimiento

Todas fueron aplicadas en la construcción de la aplicación web.

### Metodologías Ágiles

**Scrum**

Scrum es un marco ágil enfocado en entregas incrementales y colaborativas. Define roles, eventos y artefactos como:

* Product Backlog
* Sprint Backlog
* Incremento
* Reuniones de planificación y revisión

**PMI / PMBOK**

El PMBOK proporciona lineamientos para la gestión profesional de proyectos (Project Management Institute, 2021). Se aplicó principalmente para:

* Gestión del alcance
* Gestión del tiempo
* Gestión de riesgos
* Gestión de la calidad

Este enfoque permitió organizar las fases del proyecto de manera formal.

### Turismo y Transformación Digital

La digitalización del turismo implica incorporar tecnologías que optimicen la información, la comunicación y los servicios ofrecidos a los viajeros. Según Buhalis (2020), los turistas demandan acceso a itinerarios, datos actualizados y personalización en tiempo real, lo cual coincide con los objetivos del presente proyecto.

Itinerarios turísticos

Los itinerarios representan la secuencia de actividades programadas para el turista, incluyendo horarios, visitas, transporte y recomendaciones (OMT, 2023). Su digitalización facilita:

* Accesibilidad desde cualquier dispositivo
* Reducción de errores
* Actualizaciones instantáneas
* Mejor experiencia del usuario

### Usabilidad y Experiencia de Usuario

Nielsen (2012) indica que un sistema usable debe ser fácil de aprender, eficiente, libre de errores y satisfactorio. Esto fundamentó el diseño de interfaces limpias e intuitivas mediante Figma y React.

## OBJETIVOS DEL PROYECTO

### OBJETIVO GENERAL:

Determinar en qué medida la implementación de una aplicación web influye en la gestión de itinerarios turísticos en la empresa Panorama en Cusco, 2025.

### OBJETIVOS ESPECÍFICOS:

* **Gestión del alcance en la organización de itinerarios turísticos – Control y delimitación del trabajo**

**Objetivo:** Determinar cómo la implementación de una aplicación web influye en la gestión del alcance, asegurando que todas las actividades relacionadas con la creación, edición y seguimiento de itinerarios turísticos se encuentren claramente definidas y controladas dentro de la plataforma. Esto permitirá evitar tareas fuera del alcance, optimizar recursos y garantizar que los itinerarios cumplan con los requerimientos establecidos por la empresa Panorama.

**Medición del Éxito:** El éxito se medirá mediante la reducción del 85% en desviaciones del alcance (actividades no planificadas), la correcta definición del 100% de los itinerarios en la plataforma y la disminución del tiempo de planificación en un 40% gracias a la automatización y centralización de procesos.

* **Gestión de las comunicaciones en el desarrollo de itinerarios turísticos – Información clara y oportuna**

**Objetivo:** Analizar cómo la implementación de una aplicación web impacta en la gestión de las comunicaciones, garantizando que la información sobre itinerarios, cambios y notificaciones fluya de manera eficiente entre administradores y turistas. El sistema debe permitir alertas en tiempo real y un canal centralizado para reducir errores y mejorar la experiencia del usuario.

**Medición del Éxito:** El éxito se medirá mediante la entrega del 100% de notificaciones en tiempo real, una reducción del 90% en errores de comunicación y una tasa de satisfacción superior al 85% en la claridad y oportunidad de la información proporcionada a los turistas.

# CONOCIMIENTOS DE INGENIERÍA APLICADOS/ RELACIONADOS

**● Conocimiento en Matemáticas**

**Área empleada:**  
 Estadística básica, análisis de datos, cálculo de porcentajes y representación numérica.

**Aplicación en el proyecto:**  
 Se emplearon para procesar la información almacenada en la base de datos, calcular métricas como promedios, porcentajes y totales, y estructurar los valores necesarios para generar gráficos estadísticos dentro de la aplicación. Estos cálculos permitieron la correcta visualización de datos utilizando la librería RChart.js (Chart.js).

**Módulo donde se aplica:**  
 Módulo de reportes y gráficos estadísticos del sistema, así como en la capa lógica donde se procesan los datos antes de ser enviados al frontend.

**● Conocimiento en Ingeniería**

**Área empleada:**  
 Ingeniería de Software, arquitectura cliente–servidor, diseño de bases de datos relacionales, desarrollo frontend y backend, buenas prácticas de codificación.

**Aplicación en el proyecto:**  
 Se aplicaron principios de ingeniería de software para el diseño y desarrollo de la aplicación web TravelDesk, implementando una arquitectura basada en React.js para el frontend y Node.js con Express para el backend. Asimismo, se utilizó MySQL para la gestión eficiente de datos, aplicando normalización, relaciones y consultas optimizadas. También se emplearon patrones de diseño, control de versiones y técnicas de validación para garantizar la calidad y consistencia del sistema.

**Módulo donde se aplica:**  
 En todo el sistema: módulo de autenticación, gestión de usuarios, creación y administración de itinerarios, manejo de paquetes turísticos, reportes y comunicación entre frontend–backend.

# INGENIERO Y LA SOCIEDAD

**Justificación social:**

El desarrollo del proyecto responde a la necesidad de modernizar los procesos de atención al cliente en la agencia de viajes Panorama Travel. La implementación de una plataforma web permitirá ofrecer un servicio más eficiente, accesible y personalizado, contribuyendo al fortalecimiento del turismo local y nacional. Esta solución tecnológica busca mejorar la experiencia del turista, incrementando su satisfacción y confianza en los servicios ofrecidos por la empresa.

**Justificación económica:**

Desde el punto de vista económico, la aplicación web optimiza los recursos operativos al reducir tiempos de respuesta, minimizar errores humanos y disminuir la dependencia de procesos manuales. Esto incrementa la competitividad de la agencia frente al mercado nacional e internacional. Además, fomenta la innovación tecnológica en el sector turístico, impulsando la inversión en herramientas digitales y la generación de empleo calificado en áreas como desarrollo web y soporte técnico.

**Justificación ambiental:**

Aunque el impacto ambiental directo es mínimo, la solución contribuye indirectamente a la sostenibilidad al reducir el uso de materiales impresos (itinerarios, mapas, folletos) mediante la digitalización de la información turística. Asimismo, la plataforma puede incluir recomendaciones que promuevan prácticas responsables, como el uso eficiente del transporte y el respeto por áreas naturales y culturales, alineándose con los principios del turismo sostenible.

**Acontecimientos tecnológicos y científicos:**

El desarrollo del proyecto se apoya en avances recientes en el ámbito tecnológico y científico que han transformado la forma en que se gestionan los servicios turísticos. Entre los principales acontecimientos destacan:

**Evolución de las tecnologías web**: El uso de frameworks modernos como React.js y Node.js ha permitido crear aplicaciones más dinámicas, escalables y seguras, facilitando la interacción en tiempo real entre usuarios y sistemas. Estos avances han acelerado la digitalización en sectores como el turismo, donde tener información actualizada es crucial.

**Arquitecturas cliente servidor optimizadas**: La adopción de arquitecturas distribuidas y el patrón MVC han mejorado la separación de responsabilidades, la escalabilidad y la mantenibilidad de las aplicaciones. Esto facilita que las plataformas turísticas gestionen grandes volúmenes de datos de manera más eficiente.

**Transformación digital en turismo**: La tendencia global hacia los destinos turísticos inteligentes y la digitalización de itinerarios responde a la creciente demanda de experiencias personalizadas y acceso inmediato a información confiable. Este cambio está respaldado por estudios que muestran cómo la tecnología mejora la satisfacción del turista.

**Herramientas de diseño y colaboración**: Plataformas como Figma y sistemas de control de versiones (Git/GitHub) han transformado la forma en que se diseñan y desarrollan software, permitiendo un trabajo colaborativo más eficiente, prototipado rápido y un control de cambios efectivo en proyectos tecnológicos. Funcional Gestionar itinerarios turísticos

# METODOLOGÍA EMPLEADA

En este proyecto, se empleará una metodología que combine elementos de la metodología Ágil con el marco de trabajo Scrum, adaptándolos a las restricciones específicas del proyecto. Si bien se tomarán varios principios de Scrum, la existencia de una fecha límite fija impedirá la implementación de Sprints en su forma pura.

Para la gestión general del proyecto, se adoptará el enfoque del PMI (Project Management Institute), basado en la guía PMBOK (Project Management Body of Knowledge). Esto proporcionará una estructura para la planificación, ejecución, seguimiento y control de las diversas actividades del proyecto.

En cuanto al desarrollo del software, se utilizará el marco de trabajo ágil Scrum. Scrum facilitará un proceso de desarrollo iterativo e incremental, permitiendo la entrega de funcionalidades en etapas y la adaptación a los cambios.

El trabajo se realizó mediante:

Un análisis de requerimientos a través de Historias de usuario

|  |  |  |
| --- | --- | --- |
| **Tipo de Caso de Uso** | **Nombre del Caso de Uso** | **Descripción** |
| **Funcional** | **Consultar itinerarios establecidos** | Permite visualizar los itinerarios turísticos creados por el administrador, incluyendo actividades, horarios y duración. |
| **Funcional** | **Gestión de itinerarios asignados** | El usuario (turista) puede visualizar los itinerarios que el administrador le ha asignado desde el sistema. |
| **Funcional** | **Consultar turistas registrados** | El administrador puede visualizar la lista completa de turistas registrados, incluyendo filtros y opciones de búsqueda. |
| **Funcional** | **Gestionar itinerarios turísticos** | Permite crear, editar o eliminar itinerarios turísticos predefinidos disponibles para asignación. |
| **Funcional** | **Generar PDF del itinerario** | El sistema genera un documento descargable con el itinerario asignado al turista, incluyendo detalles de días y actividades. |
| **No funcional** | **Tiempo de respuesta inferior a 2 segundos** | Las consultas del sistema deben procesarse rápidamente para ofrecer una experiencia fluida. |
| **No funcional** | **Alta disponibilidad (99.9%)** | El sistema debe mantenerse operativo la mayoría del tiempo para no afectar las operaciones del personal administrativo. |
| **No funcional** | **Seguridad en el manejo de datos personales** | El sistema debe proteger la información de los turistas siguiendo estándares básicos de protección de datos. |

Generando el diagrama de casos de uso: Diagrama

El contenido generado por IA puede ser incorrecto.

Figura 4.1 Diagrama de Caso de Uso

## Identificando el Backlog (ordenado por prioridad):

|  |  |  |  |
| --- | --- | --- | --- |
| Prioridad | Historia de Usuario | Estado | Estimación (Fibonacci) |
| Alta | HU 1.1: Como administrador, quiero crear la base de datos relacional para organizar turistas e itinerarios. | Pendiente | 3 |
| Alta | HU 1.2: Como desarrollador, quiero definir relaciones y claves foráneas para garantizar integridad. | Pendiente | 3 |
| Alta | HU 1.3: Como administrador, quiero disponer de scripts de creación para replicar la BD en diferentes entornos. | Pendiente | 2 |
| Alta | HU 1.4: Como administrador, quiero configurar respaldos automáticos para prevenir pérdida de datos. | Pendiente | 3 |
| Media | HU 1.5: Como administrador, quiero restaurar respaldos cuando ocurra un error para recuperar el sistema. | Pendiente | 2 |
| Media | HU 1.6: Como administrador, quiero recibir notificaciones de fallos en los respaldos para tomar acciones. | Pendiente | 2 |
| Alta | HU 1.7: Como gerente, quiero realizar una prueba piloto con un grupo reducido de usuarios para validar el sistema. | Pendiente | 3 |
| Media | HU 1.8: Como turista de prueba, quiero interactuar con la aplicación para dar retroalimentación. | Pendiente | 2 |
| Media | HU 1.9: Como personal de soporte, quiero registrar incidencias de la prueba para mejoras futuras. | Pendiente | 2 |
| Alta | HU 2.1: Como administrador, quiero registrar turistas con sus datos personales para mantener actualizada la información. | Pendiente | 3 |
| Alta | HU 2.2: Como turista, quiero que mis datos sean ingresados de manera segura para proteger mi privacidad. | Pendiente | 3 |
| Alta | HU 2.3: Como personal, quiero validar campos obligatorios al registrar turistas para evitar errores. | Pendiente | 2 |
| Alta | HU 2.4: Como personal autorizado, quiero editar datos de turistas para mantener información actualizada. | Pendiente | 3 |
| Media | HU 2.5: Como administrador, quiero registrar el historial de ediciones para asegurar trazabilidad. | Pendiente | 2 |
| Media | HU 2.6: Como turista, quiero que mis datos se editen bajo autorización para proteger mi información. | Pendiente | 2 |
| Alta | HU 2.7: Como administrador, quiero eliminar turistas del sistema para mantener la base limpia. | Pendiente | 3 |
| Alta | HU 2.8: Como administrador, quiero que exista confirmación antes de eliminar para evitar errores. | Pendiente | 2 |
| Media | HU 2.9: Como personal, quiero que las eliminaciones generen registros para auditoría. | Pendiente | 2 |
| Alta | HU 2.10: Como personal, quiero listar turistas registrados para gestionar mejor la información. | Pendiente | 3 |
| Alta | HU 2.11: Como administrador, quiero consultar fichas detalladas de turistas para ver toda su información. | Pendiente | 3 |
| Media | HU 2.12: Como personal, quiero exportar listados de consultas para análisis externo. | Pendiente | 2 |
| Alta | HU 2.13: Como personal, quiero buscar turistas por nombre para ubicarlos rápidamente. | Pendiente | 3 |
| Media | HU 2.14: Como administrador, quiero filtrar turistas por nacionalidad para reportes. | Pendiente | 2 |
| Media | HU 2.15: Como personal, quiero buscar por fecha de registro para análisis temporal. | Pendiente | 2 |
| Alta | HU 2.16: Como administrador, quiero validar que se llenen campos obligatorios para registros correctos. | Pendiente | 3 |
| Media | HU 2.17: Como turista, quiero recibir mensajes claros de error para completar mis datos correctamente. | Pendiente | 2 |
| Media | HU 2.18: Como desarrollador, quiero definir reglas de validación para estandarizar los datos. | Pendiente | 2 |
| Alta | HU 3.1: Como usuario, quiero iniciar sesión con usuario y contraseña para acceder al sistema. | Pendiente | 3 |
| Alta | HU 3.2: Como administrador, quiero validar credenciales en el backend para seguridad. | Pendiente | 3 |
| Media | HU 3.3: Como usuario, quiero mensajes de error claros para entender fallos de autenticación. | Pendiente | 2 |
| Alta | HU 3.4: Como usuario, quiero recuperar mi contraseña por correo para acceder al sistema. | Pendiente | 3 |
| Alta | HU 3.5: Como administrador, quiero que el sistema valide identidad antes de recuperar para proteger cuentas. | Pendiente | 3 |
| Media | HU 3.6: Como usuario, quiero un enlace seguro de restablecimiento para proteger mi acceso. | Pendiente | 2 |
| Alta | HU 3.7: Como administrador, quiero asignar roles a usuarios para controlar accesos. | Pendiente | 3 |
| Media | HU 3.8: Como administrador, quiero modificar permisos para ajustar niveles de acceso. | Pendiente | 2 |
| Media | HU 3.9: Como personal, quiero tener acceso solo a funciones autorizadas para evitar errores. | Pendiente | 2 |
| Alta | HU 3.10: Como usuario, quiero que mi sesión se cierre por inactividad para proteger mis datos. | Pendiente | 3 |
| Media | HU 3.11: Como administrador, quiero configurar tiempos de sesión para mayor seguridad. | Pendiente | 2 |
| Media | HU 3.12: Como usuario, quiero recibir alertas antes de cierre automático para no perder mi trabajo. | Pendiente | 2 |
| Alta | HU 3.13: Como administrador, quiero registrar accesos de usuarios para tener trazabilidad. | Pendiente | 3 |
| Media | HU 3.14: Como administrador, quiero auditar cambios realizados para identificar responsables. | Pendiente | 2 |
| Media | HU 3.15: Como gerente, quiero consultar reportes de auditoría para garantizar transparencia. | Pendiente | 2 |
| Alta | HU 4.1: Como personal, quiero crear itinerarios para turistas para organizar viajes. | Pendiente | 3 |
| Alta | HU 4.2: Como personal, quiero editar itinerarios ya creados para corregir datos. | Pendiente | 3 |
| Media | HU 4.3: Como administrador, quiero eliminar itinerarios innecesarios para mantener la base ordenada. | Pendiente | 2 |
| Alta | HU 4.4: Como turista autenticado, quiero ver mi itinerario completo para conocer mi viaje. | Pendiente | 3 |
| Media | HU 4.5: Como turista, quiero ver detalles de actividades para planificar mejor. | Pendiente | 2 |
| Media | HU 4.6: Como turista, quiero acceder desde móvil para consultar fácilmente. | Pendiente | 2 |
| Alta | HU 4.7: Como turista, quiero descargar mi itinerario en PDF para tenerlo sin conexión. | Pendiente | 3 |
| Media | HU 4.8: Como turista, quiero que el PDF tenga un diseño claro para entender mi viaje. | Pendiente | 2 |
| Media | HU 4.9: Como personal, quiero incluir el logo de la empresa en el PDF para personalizarlo. | Pendiente | 2 |
| Alta | HU 4.10: Como administrador, quiero exportar itinerarios a Excel para análisis. | Pendiente | 3 |
| Media | HU 4.11: Como personal, quiero exportar reportes de turistas a Excel para compartirlos. | Pendiente | 2 |
| Media | HU 4.12: Como gerente, quiero consolidar datos en Excel para informes. | Pendiente | 2 |
| Media | HU 4.13: Como personal, quiero recibir notificaciones de cambios en itinerarios para estar informado. | Pendiente | 2 |
| Media | HU 4.14: Como administrador, quiero configurar alertas internas para coordinar al equipo. | Pendiente | 2 |
| Media | HU 4.15: Como gerente, quiero consultar historial de notificaciones para control. | Pendiente | 2 |
| Media | HU 4.16: Como administrador, quiero registrar cambios en itinerarios para tener trazabilidad. | Pendiente | 2 |
| Media | HU 4.17: Como personal, quiero consultar el historial de cambios para conocer ajustes. | Pendiente | 2 |
| Media | HU 4.18: Como gerente, quiero reportes de modificaciones para auditoría. | Pendiente | 2 |
| Alta | HU 5.1: Como administrador, quiero generar reportes de turistas para análisis interno. | Pendiente | 3 |
| Alta | HU 5.2: Como gerente, quiero generar reportes de itinerarios para seguimiento. | Pendiente | 3 |
| Media | HU 5.3: Como administrador, quiero guardar reportes generados para consultas futuras. | Pendiente | 2 |
| Media | HU 5.4: Como administrador, quiero ver cantidad de turistas por nacionalidad para estadísticas. | Pendiente | 2 |
| Media | HU 5.5: Como gerente, quiero reportes gráficos de turistas por fecha para analizar tendencias. | Pendiente | 2 |
| Media | HU 5.6: Como administrador, quiero estadísticas de itinerarios activos para medir actividad. | Pendiente | 2 |
| Media | HU 5.7: Como turista extranjero, quiero ver la aplicación en inglés para entender mi itinerario. | Pendiente | 2 |
| Media | HU 5.8: Como administrador, quiero configurar idiomas para atender diferentes públicos. | Pendiente | 2 |
| Baja | HU 5.9: Como turista, quiero cambiar idioma en tiempo real para comodidad. | Pendiente | 1 |
| Media | HU 5.11: Como administrador, quiero configurar tasas de cambio para precisión en costos. | Pendiente | 2 |

## Llegando a obtener el Sprint Backlog:

**Sprint 1: Configuración Inicial**

Historia de Usuario 1.1: Crear Base de Datos Inicial

**Descripción:** Como administrador, quiero tener una base de datos relacional para organizar turistas, usuarios e itinerarios.

Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 1.1.1 | Diseñar modelo entidad-relación | Analista BD | 6 | Pendiente |
| 1.1.2 | Crear tablas en MySQL con claves primarias | DBA | 8 | Pendiente |
| 1.1.3 | Definir claves foráneas y relaciones | DBA | 6 | Pendiente |
| 1.1.4 | Verificar integridad con datos de prueba | QA Engineer | 4 | Pendiente |

Historia de Usuario 1.2: Definición de relaciones

Descripción: Como desarrollador, necesito definir relaciones y claves foráneas para garantizar la integridad de los datos.Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 1.2.1 | Documentar relaciones y reglas de integridad | Analista BD | 4 | Pendiente |
| 1.2.2 | Implementar ON DELETE y ON UPDATE | DBA | 5 | Pendiente |
| 1.2.3 | Validar integridad referencial | QA Engineer | 3 | Pendiente |

Historia de Usuario 1.3: Scripts de creación

Descripción: Como administrador, necesito disponer de scripts de creación para replicar la BD en distintos entornos.Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 1.3.1 | Escribir script de creación completo | DBA | 5 | Pendiente |
| 1.3.2 | Probar script en entorno de desarrollo | QA Engineer | 4 | Pendiente |
| 1.3.3 | Probar script en entorno de pruebas | QA Engineer | 4 | Pendiente |
| 1.3.4 | Documentar uso del script | Analista BD | 3 | Pendiente |

**Sprint 2: Seguridad y Validación de la Base de Datos**

HU 19.1: Configuración de respaldos automáticos

**Descripción:** Como administrador, quiero configurar respaldos automáticos para garantizar la seguridad de la información de la base de datos.

Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 19.1.1 | Definir política de respaldo (frecuencia, retención) | Analista BD | 4 | Pendiente |
| 19.1.2 | Configurar tarea automática de respaldo en MySQL | DBA | 6 | Pendiente |
| 19.1.3 | Probar recuperación desde respaldo | QA Engineer | 5 | Pendiente |
| 19.1.4 | Documentar proceso de respaldo y restauración | Analista BD | 3 | Pendiente |

HU 24.1: Ejecución de prueba piloto

**Descripción:** Como administrador, necesito ejecutar una prueba piloto para validar la integridad y funcionamiento de la base de datos antes de pasar a producción.

Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 24.1.1 | Cargar datos de prueba en la base de datos | Analista BD | 5 | Pendiente |
| 24.1.2 | Ejecutar consultas de validación | QA Engineer | 6 | Pendiente |
| 24.1.3 | Evaluar rendimiento inicial | DBA | 5 | Pendiente |
| 24.1.4 | Documentar resultados de la prueba piloto | QA Engineer | 4 | Pendiente |

**Sprint 3: Gestión de Usuarios y Autenticación**

HU 2.1: Registro de usuarios

**Descripción:** Como usuario, quiero poder registrarme en el sistema proporcionando mis datos básicos para acceder a las funcionalidades.

Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 2.1.1 | Diseñar formulario de registro | Frontend Dev | 5 | Pendiente |
| 2.1.2 | Implementar validaciones de campos | Frontend Dev | 6 | Pendiente |
| 2.1.3 | Conectar formulario con base de datos | Backend Dev | 8 | Pendiente |
| 2.1.4 | Pruebas de creación de usuario | QA Engineer | 4 | Pendiente |

HU 2.2: Inicio de sesión de usuarios

**Descripción:** Como usuario, quiero iniciar sesión con mis credenciales para acceder a mi cuenta.

Tareas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 2.2.1 | Diseñar formulario de inicio de sesión | Frontend Dev | 4 | Pendiente |
| 2.2.2 | Implementar validación de credenciales | Backend Dev | 7 | Pendiente |
| 2.2.3 | Manejo de errores de autenticación | Backend Dev | 4 | Pendiente |
| 2.2.4 | Pruebas de acceso correcto e incorrecto | QA Engineer | 5 | Pendiente |

HU 2.3: Gestión de roles y permisos

**Descripción:** Como administrador, quiero asignar roles y permisos a los usuarios para controlar el acceso a las funcionalidades del sistema.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 2.3.1 | Definir estructura de roles en la BD | Analista BD | 4 | Pendiente |
| 2.3.2 | Implementar asignación de roles en el backend | Backend Dev | 7 | Pendiente |
| 2.3.3 | Crear interfaz de administración de roles | Frontend Dev | 6 | Pendiente |
| 2.3.4 | Pruebas de accesos según permisos | QA Engineer | 5 | Pendiente |

**Sprint 4: Gestión de Turistas**

HU 3.1: Registro de turistas

**Descripción:** Como administrador, quiero registrar turistas en el sistema para llevar un control de sus datos personales y de viaje.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 3.1.1 | Diseñar formulario de registro de turistas | Frontend Dev | 6 | Pendiente |
| 3.1.2 | Validar campos obligatorios (nombre, pasaporte, contacto) | Frontend Dev | 5 | Pendiente |
| 3.1.3 | Conectar con base de datos para almacenar turistas | Backend Dev | 7 | Pendiente |
| 3.1.4 | Pruebas de registro de turistas | QA Engineer | 4 | Pendiente |

HU 3.2: Edición de información de turistas

**Descripción:** Como administrador, quiero editar los datos de un turista registrado para mantener la información actualizada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 3.2.1 | Crear interfaz de edición de turistas | Frontend Dev | 5 | Pendiente |
| 3.2.2 | Implementar validaciones de actualización | Backend Dev | 6 | Pendiente |
| 3.2.3 | Guardar cambios en la base de datos | Backend Dev | 5 | Pendiente |
| 3.2.4 | Pruebas de edición de turistas | QA Engineer | 4 | Pendiente |

HU 3.3: Eliminación de turistas

**Descripción:** Como administrador, quiero eliminar turistas del sistema para mantener solo información relevante.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 3.3.1 | Agregar opción de eliminar en listado | Frontend Dev | 4 | Pendiente |
| 3.3.2 | Implementar borrado lógico en la BD | Backend Dev | 6 | Pendiente |
| 3.3.3 | Verificar integridad de datos relacionados | Analista BD | 5 | Pendiente |
| 3.3.4 | Pruebas de eliminación de turistas | QA Engineer | 4 | Pendiente |

HU 3.4: Listado de turistas

**Descripción:** Como administrador, quiero visualizar un listado de los turistas registrados para acceder fácilmente a su información.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 3.4.1 | Diseñar tabla de turistas en interfaz | Frontend Dev | 6 | Pendiente |
| 3.4.2 | Implementar paginación y filtros básicos | Frontend Dev | 5 | Pendiente |
| 3.4.3 | Consultar turistas desde la base de datos | Backend Dev | 6 | Pendiente |
| 3.4.4 | Pruebas de visualización de turistas | QA Engineer | 4 | Pendiente |

HU 3.5: EXTRA S4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 3.5.1 | Convertir la vista de editar turistas en modal en la vista de listar turistas | Frontend Dev | 3 | Pendiente |

**Sprint 5: Gestión de Itinerarios**

HU 4.1: Creación de itinerarios

**Descripción:** Como administrador, quiero crear itinerarios personalizados para asignarlos a los turistas según su paquete de viaje.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 4.1.1 | Diseñar formulario de creación de itinerarios | Frontend Dev | 6 | Pendiente |
| 4.1.2 | Implementar validaciones de campos obligatorios | Frontend Dev | 5 | Pendiente |
| 4.1.3 | Guardar itinerario en la base de datos | Backend Dev | 7 | Pendiente |
| 4.1.4 | Pruebas de creación de itinerarios | QA Engineer | 4 | Pendiente |

HU 4.2: Edición de itinerarios

**Descripción:** Como administrador, quiero editar los itinerarios ya creados para ajustar horarios o actividades según necesidad.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 4.2.1 | Crear interfaz de edición de itinerarios | Frontend Dev | 5 | Pendiente |
| 4.2.2 | Implementar actualización en la base de datos | Backend Dev | 6 | Pendiente |
| 4.2.3 | Validar cambios de itinerarios | QA Engineer | 4 | Pendiente |
| 4.2.4 | Documentar casos de edición | Analista BD | 3 | Pendiente |
| S5.2 | Concretar nuevas funciones de edición de itinerarios del Frontend (Observaciones DOCX S5-HU4.2.3 y HU-4.2.4) | Frontend Dev | 3 | Pendiente |
| S5.3 | Concretar nuevas funciones de edición de itinerarios del Backend (Observaciones DOCX S5-HU4.2.3 y HU-4.2.4) | Backend Dev | 3 | Pendiente |

HU 4.3: Eliminación de itinerarios

**Descripción:** Como administrador, quiero eliminar itinerarios para mantener la base de datos organizada.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 4.3.1 | Agregar opción de eliminar en listado | Frontend Dev | 4 | Pendiente |
| 4.3.2 | Implementar borrado lógico en la base de datos | Backend Dev | 6 | Pendiente |
| 4.3.3 | Validar que turistas no pierdan historial | Analista BD | 5 | Pendiente |
| 4.3.4 | Pruebas de eliminación de itinerarios | QA Engineer | 4 | Pendiente |

HU 4.4: Visualización de itinerarios

**Descripción:** Como administrador, quiero visualizar un listado de itinerarios creados para consultarlos fácilmente.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 4.4.1 | Diseñar tabla de itinerarios en interfaz | Frontend Dev | 6 | Pendiente |
| 4.4.2 | Implementar filtros por fecha y turista | Frontend Dev | 5 | Pendiente |
| 4.4.3 | Consultar itinerarios desde la base de datos | Backend Dev | 6 | Pendiente |
| 4.4.4 | Pruebas de visualización de itinerarios | QA Engineer | 4 | Pendiente |

HU EXTRA agregados: S5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| S5.1 | Mejora de diseño y experiencia de usuario en Creación de itinerarios(Observaciones DOCX S5-HU4.1.4) | Frontend Dev | 3 | Pendiente |
| S5.2 | Concretar las nuevas funciones de edición de itinerarios respecto al Frontend (Observaciones DOCX S5-HU4.2.3 y HU-4.2.4) | Frontend Dev | 3 | Pendiente |
| S5.3 | Concretar las nuevas funciones de edición de itinerarios respecto al Backend (Observaciones DOCX S5-HU4.2.3 y HU-4.2.4) | Backend Dev | 3 | Pendiente |

SPRINT 6 – Panel del Turista: Visualización de Itinerario y Notificaciones

HU 6.1: Panel del Turista (Itinerario, Notificaciones y PDF)

**Descripción:** Como turista, quiero ingresar a mi panel personal donde pueda visualizar mi itinerario completo y diario, recibir notificaciones sobre mis actividades, y descargar mi plan de viaje en formato PDF, para mantenerme informado y organizado durante mi viaje.

Fase 1: Diseño y Frontend (UI/UX + Integración visual)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 6.1.1 | Diseñar la maqueta del Dashboard del Turista (header con saludo, sidebar, área de contenido) | UX/UI Designer | 5 | Pendiente |
| 6.1.2 | Implementar el layout principal (header + sidebar + rutas internas) | Frontend Dev | 7 | Pendiente |
| 6.1.3 | Diseñar e implementar la vista “Mi Itinerario” con cards por día y actividad | Frontend Dev | 8 | Pendiente |
| 6.1.4 | Crear el modal “Ver más detalles” con descripción, lugar, horario y duración | Frontend Dev | 5 | Pendiente |
| 6.1.5 | Diseñar e implementar la vista de Notificaciones del Turista conectada a backend | Frontend Dev | 6 | Pendiente |
| 6.1.6 | Crear la vista Descargar PDF con botón para generar el archivo | Frontend Dev | 5 | Pendiente |
| 6.1.7 | Implementar funcionalidad responsive (grid adaptable, vista móvil optimizada) | Frontend Dev | 4 | Pendiente |

Fase 2: Backend y Conexión (API + Datos reales + Validaciones)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 6.1.8 | Crear endpoint para obtener itinerarios del turista autenticado | Backend Dev | 6 | Pendiente |
| 6.1.9 | Implementar consulta SQL con unión de tablas (turistas, itinerarios, actividades) | Backend Dev | 5 | Pendiente |
| 6.1.10 | Crear endpoint para recuperar notificaciones del turista | Backend Dev | 6 | Pendiente |
| 6.1.11 | Crear endpoint que genere PDF con datos del turista | Backend Dev | 4 | Pendiente |
| 6.1.12 | Validar resultados de API (QA y Backend) | QA Engineer | 3 | Pendiente |

SPRINT 7 – Panel de Estadísticas y Análisis del Sistema

HU 7.1: Generación de reportes de turistas

**Descripción:** Como administrador, quiero visualizar estadísticas gráficas interactivas sobre turistas, itinerarios y notificaciones, para analizar el rendimiento del sistema y la actividad general de la empresa Panorama.

Fase 1: Diseño y Frontend (UI/UX + Integración visual)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 7.1.1 | Diseñar la interfaz del Panel de Estadísticas dentro del dashboard del administrador | UX/UI Designer | 5 | Pendiente |
| 7.1.2 | Implementar componentes gráficos (usando librería como Recharts o Chart.js) | Frontend Dev | 8 | Pendiente |
| 7.1.3 | Crear tarjetas resumen de métricas principales (total turistas, itinerarios, notificaciones) | Frontend Dev | 6 | Pendiente |
| 7.1.4 | Implementar filtros por rango de fechas o periodo de análisis | Frontend Dev | 5 | Pendiente |
| 7.1.5 | Ajustar diseño responsive (desktop y móvil) | Frontend Dev | 3 | Pendiente |
| 7.1.6 | Validar visualización correcta de los gráficos con datos reales | QA Engineer | 4 | Pendiente |

Fase 2: Backend y Conexión (API + Datos dinámicos)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| 7.1.7 | Crear endpoint que retorne métricas principales | Backend Dev | 6 | Pendiente |
| 7.1.8 | Implementar consultas SQL agregadas (COUNT, GROUP BY) para generar datos estadísticos | Backend Dev | 5 | Pendiente |
| 7.1.9 | Optimizar respuesta para filtros de rango de fechas | Backend Dev | 4 | Pendiente |
| 7.1.10 | Validar precisión de datos y coherencia entre módulos | QA Engineer | 3 | Pendiente |

HU EXTRA agregados: S7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Tarea | Responsable | Estimación (Horas) | Estado |
| S7-8.1.1 | Auditar diseño actual: detectar inconsistencias visuales y de usabilidad | UX/UI Designer | 4 | Pendiente |

## Elaboración de prototipos:

Se diseñaron y desarrollaron prototipos de la interfaz de usuario y de las funcionalidades principales del sistema, utilizando herramientas como Canva. Estos prototipos incluyen pantallas clave, como la visualización de itinerarios y el formulario de preferencias turísticas.

El objetivo principal es validar los requerimientos funcionales propuestos, evaluando su viabilidad técnica y su usabilidad a través de retroalimentación directa de usuarios simulados y expertos.

### Pantalla de Inicio de Sesión

* Campos: Email, Password, Código/ID
* Botón: Sign up
* Enlace: Crear Cuenta

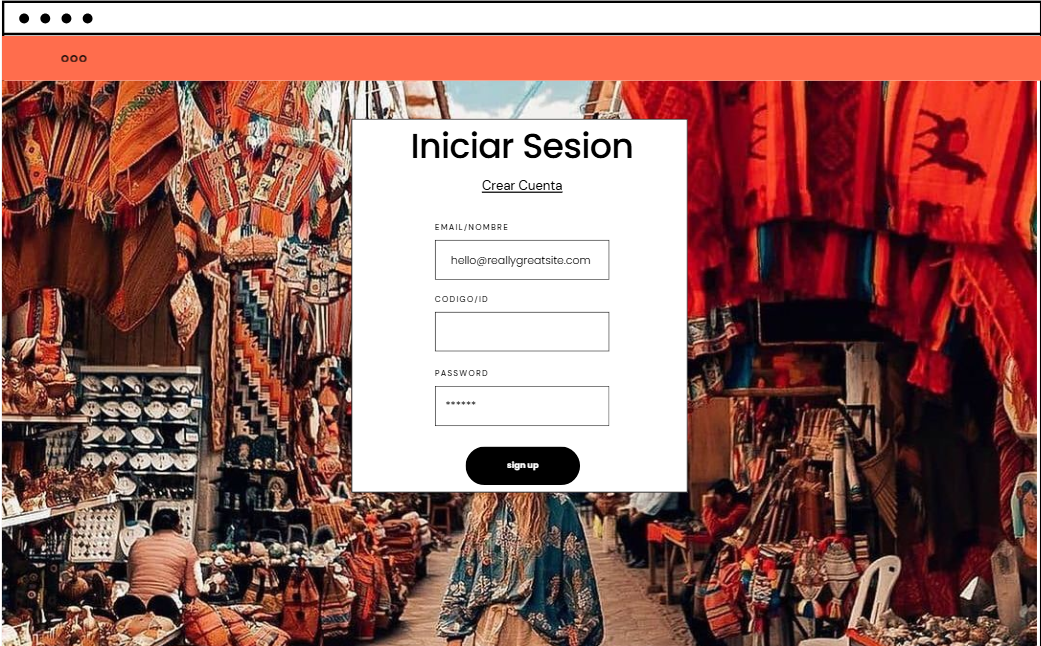


Figura 4.2 Pantalla de Inicio de Sesión

### Pantalla de Registro de Correo Electrónico

* Campos: Correo, Código de Verificación.
* Botón: Siguiente

Diagrama

El contenido generado por IA puede ser incorrecto.

Figura 4.3 Pantalla de Registro de Correo Electrónico

### Pantalla de Registro de Datos

* Campos: Nombre, Apellidos, DNI, Pasaporte, Nacionalidad, Fecha de Nacimiento, Género.
* Botón: Registrar

Gráfico, Diagrama

El contenido generado por IA puede ser incorrecto.

Figura 4.4 Pantalla de Registro de Datos

### Panel Principal Usuario

* Sidebar: Navegación (Ver Itinerarios, Documentos, Notificaciones, Soporte.
* Header: Titulo, nombre de Usuario
* Sección principal: Tarjetas de sección para el acceso rápido (Ver Itinerarios, Documentos, Notificaciones, Soporte).

Interfaz de usuario gráfica, Diagrama

El contenido generado por IA puede ser incorrecto.

Figura 4.5 Pantalla del Panel Principal Usuario

### Pantalla de Itinerarios

* Sección Principal: Calendario con las actividades programas del itinerario, Campos de información acerca del itinerario.
* Sidebar: Navegación (Ver Itinerarios, Documentos, Notificaciones, Soporte.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Figura 4.6 Pantalla de Itinerarios

### Pantalla de Ver Documentos

* Sidebar: Navegación (Ver Itinerarios, Documentos, Notificaciones, Soporte.
* Sección Principal (Campos con información acerca de Documentos Personales., Transporte, Alojamiento, Actividades y tours.)

Imagen que contiene Tabla

El contenido generado por IA puede ser incorrecto.

Figura 4.7 Pantalla de Ver Documentos

### Pantalla de Ver Notificaciones

* Sidebar: Navegación (Ver Itinerarios, Documentos, Notificaciones, Soporte.
* Sección Principal (Notificaciones acerca de los Documentos, Alojamiento, Tours, Pago)

Imagen que contiene Texto

El contenido generado por IA puede ser incorrecto.

Figura 4.8 Panalla de Ver Notificaciones

### Pantalla de Centro de Soporte

* Sidebar: Navegación (Ver Itinerarios, Documentos, Notificaciones, Soporte.
* Sección Principal (Campos de Historial de consultas junto con un Número de soporte de la empresa)

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Figura 4.9 Pantalla de Centro de Soporte

**Implementación de la solución:**

La implementación de la solución se realizó en el hosting de la plataforma **Hostinger**, utilizando un entorno optimizado para aplicaciones web modernas. Para el despliegue del frontend, se generó la versión compilada del proyecto React.js mediante *npm run build*, la cual fue subida al servidor y configurada dentro del administrador de archivos de Hostinger o mediante FTP. El sistema se ejecuta a través de un dominio o subdominio asignado, asegurando una interfaz accesible para los usuarios finales.

En cuanto al backend, Hostinger permite la ejecución del servidor Node.js utilizando su panel de “**Aplicaciones Node.js**”. Desde este módulo se cargó los archivos del servidor desarrollados con Express, se instalaron las dependencias mediante *npm install* y se configuraron las variables de entorno necesarias (.env) para la conexión con la base de datos, autenticación JWT y puertos de servicio. El backend se mantiene activo mediante el gestor interno que ofrece Hostinger para procesos Node.js, permitiendo reinicios automáticos y control del servicio.

La base de datos MySQL fue creada directamente desde el panel de control de Hostinger, configurando un usuario exclusivo para producción y restringiendo accesos externos. El esquema del sistema TravelDesk se importó mediante phpMyAdmin, garantizando la integridad de la información. La comunicación entre backend y base de datos será interna, aumentando la seguridad del sistema.

Para asegurar el despliegue, se configuro un certificado **SSL** gratuito mediante la herramienta integrada de Hostinger, habilitando el acceso HTTPS en toda la plataforma. Asimismo, se implementó medidas de seguridad como cifrado de contraseñas, validaciones del lado del servidor y manejo de roles para administradores y turistas.

Finalmente, se realizaron pruebas funcionales y de rendimiento utilizando el dominio publicado en Hostinger, verificando el correcto funcionamiento de la API, la estabilidad del frontend y la comunicación con la base de datos antes de la puesta en funcionamiento definitiva del sistema.

Interfaz de usuario gráfica, Gráfico

El contenido generado por IA puede ser incorrecto.

Figura 4.10 Panel Principal del Administrador

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Figura 4.11 Panel Principal Usuario

**Pruebas y calidad de software:**

Se realizaron pruebas exhaustivas para garantizar la estabilidad, funcionalidad y seguridad del sistema. Estas pruebas incluyeron:

Pruebas funcionales: para asegurar que el sistema cumpla los requerimientos especificados.

Pruebas de rendimiento: para medir el tiempo de respuesta del asistente y del sistema de itinerarios.

Pruebas multiplataforma: para garantizar que el sistema sea compatible con distintos dispositivos y navegadores.

**Aportes / Descubrimientos**

PMV 1: Módulo de Autenticación y Control de Acceso

Este producto mínimo viable corresponde al desarrollo del módulo que administra el acceso seguro al sistema. Incluye el inicio de sesión, validación de credenciales, recuperación de contraseña y asignación de permisos según el rol del usuario (administrador o agente de viajes). Su finalidad es garantizar la protección de la información y asegurar que cada usuario interactúe únicamente con las funciones que le corresponden. Este PMV establece la base de seguridad necesaria para el uso adecuado del sistema.

PMV 2: Módulo de Gestión de Turistas

Este PMV se refiere a la implementación de un módulo que permite registrar, consultar, actualizar y mantener el historial de turistas dentro del sistema. Proporciona un control ordenado y centralizado de los datos personales de los clientes, permitiendo a la agencia gestionar de manera eficiente la información necesaria para la asignación posterior de itinerarios. Es un componente clave para garantizar un servicio personalizado y facilitar la trazabilidad de los turistas atendidos.

PMV 3: Gestión de Itinerarios Turísticos Predefinidos

Este producto mínimo viable abarca el desarrollo del módulo que permite crear, editar, eliminar y consultar itinerarios turísticos previamente definidos por la agencia. Cada itinerario contiene actividades, lugares, duraciones y costos aproximados. Este PMV constituye el núcleo operativo del sistema, ya que centraliza la oferta turística que posteriormente será asignada a los turistas.

PMV 4: Asignación Personalizada de Itinerarios a Turistas

Este PMV se refiere al proceso mediante el cual un agente selecciona a un turista y asigna uno o varios itinerarios que coincidan con sus preferencias, duración del viaje o presupuesto. Este PMV mejora significativamente la experiencia del usuario y agiliza la atención dentro de la agencia.

PMV 5: Módulo de Gestión de Actividades Turísticas

Este producto consiste en el registro y administración de actividades turísticas (horarios, descripción, ubicación, restricciones, etc.) que posteriormente se asocian a los itinerarios. Su objetivo es disponer de una base de datos completa y actualizada que permita mantener consistencia y precisión en la información presentada al turista. Este PMV es esencial para asegurar que los itinerarios mantengan datos coherentes y organizados.

PMV 6: Generación de Itinerarios en Formato PDF

Este PMV comprende la capacidad del sistema para generar un documento PDF descargable que incluya los datos del turista, el itinerario asignado, fechas, actividades y detalles relevantes del viaje. Su finalidad es proporcionar a la agencia y al turista un documento formal, presentable y fácil de entregar o archivar, mejorando la profesionalidad del servicio y facilitando el uso administrativo.

PMV 7: Dashboard de Indicadores del Sistema

Este producto mínimo viable corresponde al desarrollo de un panel de control que muestra métricas clave del sistema, como número de turistas registrados, itinerarios más asignados, cantidad de itinerarios creados y próximos viajes programados. Su propósito es ofrecer una visión global del funcionamiento del sistema, permitiendo al administrador tomar decisiones informadas y monitorear el rendimiento del servicio de manera continua.

# USO DE HERRAMIENTAS MODERNAS

Durante el desarrollo del proyecto se emplearon diversas herramientas tecnológicas modernas que facilitaron la planificación, diseño, desarrollo, gestión y validación del sistema propuesto. Cada herramienta se seleccionó estratégicamente por su compatibilidad con entornos ágiles, facilidad de colaboración en equipo y aporte técnico al flujo de trabajo.

Herramientas para gestión del proyecto

**Jira**: Plataforma para la gestión ágil del proyecto, utilizada para crear y administrar historias de usuario, definir tareas por sprint y hacer seguimiento detallado del avance mediante tableros Scrum/Kanban.

**Excel**: Utilizado para la representación formal del cronograma y diagramas de Gantt.

**OneDrive**: Plataforma de almacenamiento y colaboración en la nube para compartir documentos y versiones del informe entre los miembros del equipo.

Herramientas para diseño y prototipado

**ERDPlus**: Aplicada para el modelado entidad-relación (MER) de la base de datos, identificando entidades, relaciones y cardinalidades

Herramientas para desarrollo frontend:

**Angular 16:** Es un framework de desarrollo web basado en TypeScript, usado para construir una interfaz modular, responsiva y fácil de mantener.

Angular Material: Una biblioteca de componentes visuales de Google que asegura consistencia en el diseño, accesibilidad y sigue buenas prácticas de UI/UX.

**Chart.js:** Librería utilizada para mostrar de manera visual los resultados de encuestas o métricas del sistema a través de gráficos de barras y líneas.

Herramientas para desarrollo backend:

**Spring Boot (Java):** Framework backend elegido por su eficiencia y su estructura en capas, lo que facilita exponer endpoints REST y mantener el principio de separación de responsabilidades.

**MySQL 8.0 (XAMPP):** Sistema de gestión de bases de datos relacional empleado para almacenar la información estructurada, como usuarios, itinerarios y reservas.

**Postman:** Herramienta para probar servicios REST, validando respuestas del servidor, cabeceras (headers) y datos enviados (payload) de los endpoints desarrollados.

Herramientas complementarias:

**GitHub:** Plataforma de control de versiones que se usa para alojar el código fuente, gestionar ramas, pull requests y la documentación técnica.

**Google Forms:** Herramienta para crear y difundir encuestas de satisfacción. El formulario se compartió automáticamente al finalizar el viaje del turista.

Generador de Códigos QR: Se utilizó para crear códigos QR que redirigen a los formularios de encuestas o videos demostrativos de los productos mínimos viables (PMV).

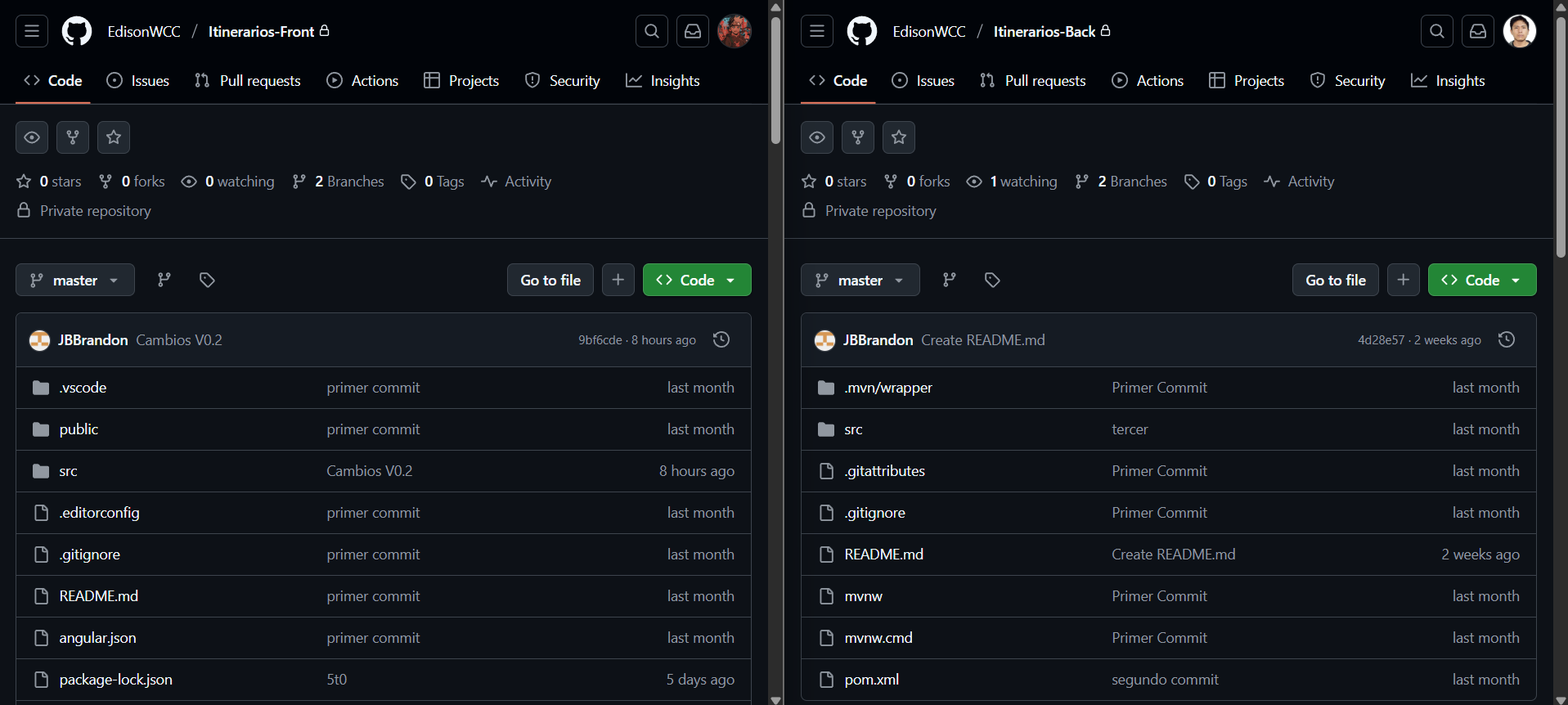


Figura 5.1 GitHub de Trabajo

Link(Requiere acceso):  
<https://github.com/EdisonWCC/Taller-de-Proyectos-2-Itinerarios-turisticos>

# DISEÑO DE INGENIERÍA

## Listado de requerimientos funcionales

|  |  |  |
| --- | --- | --- |
| Nombre del Caso de Uso | Descripción | Interfaz |
| Consultar itinerarios establecidos | Permite visualizar los itinerarios turísticos registrados en el sistema, incluyendo actividades, horarios y descripciones. | Dashboard – Admin / Dashboard – Turista |
| Gestionar itinerarios turísticos | Permite crear, editar y eliminar itinerarios turísticos desde el panel administrativo. | Dashboard – Admin |
| Consultar turistas registrados | El administrador puede visualizar, filtrar y gestionar la lista de turistas en la base de datos. | Dashboard – Admin |
| Asignar itinerarios a turistas | El administrador puede seleccionar un itinerario existente y asignarlo a un turista específico. | Dashboard – Admin |
| Generar PDF del itinerario | El usuario puede descargar su itinerario asignado en formato PDF con detalle de actividades por día. | Dashboard – Turista |

## Diseño de base de datos

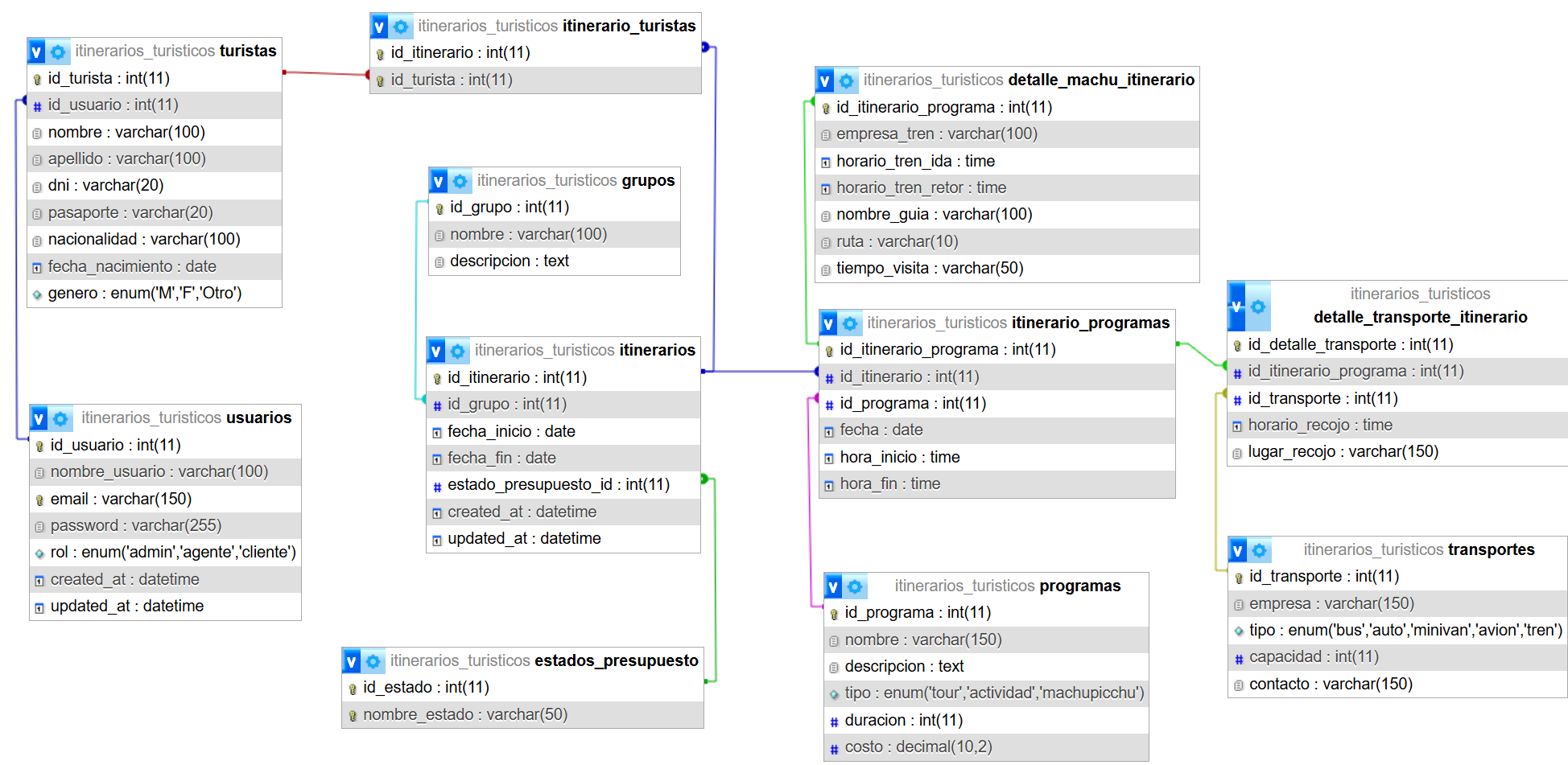


Figura 6.1 Modelo de Base de Datos

## Arquitectura de la solución planteada

La aplicación está diseñada bajo una arquitectura multicapa cliente-servidor, siguiendo el patrón Modelo-Vista-Controlador (MVC) y una lógica desacoplada mediante servicios REST. Este enfoque garantiza escalabilidad, mantenibilidad y separación clara de responsabilidades. Los requerimientos funcionales definidos para el sistema —consultar itinerarios establecidos, gestionar reservas automatizadas, brindar asesoramiento turístico en tiempo real, automatizar respuestas de preguntas frecuentes y permitir la elección de preferencias— se integran dentro de esta estructura, asegurando que cada caso de uso se soporte adecuadamente desde el diseño técnico.

**1. Capa de Presentación (Frontend)**  
Desarrollada con React.js, permite la interacción directa del turista y del administrador mediante interfaces gráficas modernas y responsivas. En esta capa se implementan las vistas necesarias para que el usuario consulte los itinerarios sugeridos, acceda a las excursiones disponibles, mantenga comunicación en tiempo real con el asistente. Incluye formularios para autenticación y registro, paneles de visualización de itinerarios, interfaz del chatbot y componentes dinámicos que facilitan la experiencia del usuario, especialmente para los casos de uso relacionados con la consulta de itinerarios, la elección de preferencias y el asesoramiento turístico inmediato.

**2. Capa Lógica (Backend)**  
Implementada con Node.js y el framework Express, expone una API RESTful que procesa solicitudes del cliente, aplica reglas de negocio y gestiona la autenticación y autorización mediante JWT. En esta capa reside la lógica que permite automatizar la gestión de reservas, generar respuestas a preguntas frecuentes y procesar consultas turísticas en tiempo real. Asimismo, organiza el flujo de información entre la capa de presentación y la base de datos, asegurando coherencia, validación de datos y la ejecución de operaciones asociadas a cada uno de los casos de uso funcionales establecidos.

**3. Capa de Datos**  
La base de datos, construida sobre MySQL, almacena información estructurada en tablas relacionadas que incluyen datos sobre usuarios y roles, itinerarios turísticos, preferencias, reservas y actividades. Este modelo relacional asegura la integridad referencial y permite realizar consultas eficientes. De esta manera, se pueden recuperar itinerarios establecidos, registrar las acciones de los usuarios, almacenar sus preferencias. Su diseño está optimizado para soportar de manera confiable tanto las operaciones de lectura como de escritura necesarias para cumplir con los requerimientos funcionales del sistema.

**Comunicación entre capas**  
La interacción entre las distintas capas del sistema se realiza a través de peticiones HTTP seguras, lo que garantiza la confidencialidad y protección de los datos. La arquitectura fue diseñada de manera modular, escalable y fácilmente integrable con otros servicios turísticos en el futuro. Cada capa contribuye al cumplimiento de los cinco requerimientos funcionales definidos, asegurando que la solución responda de forma eficiente a las necesidades tanto de la agencia de viajes como de los usuarios finales.

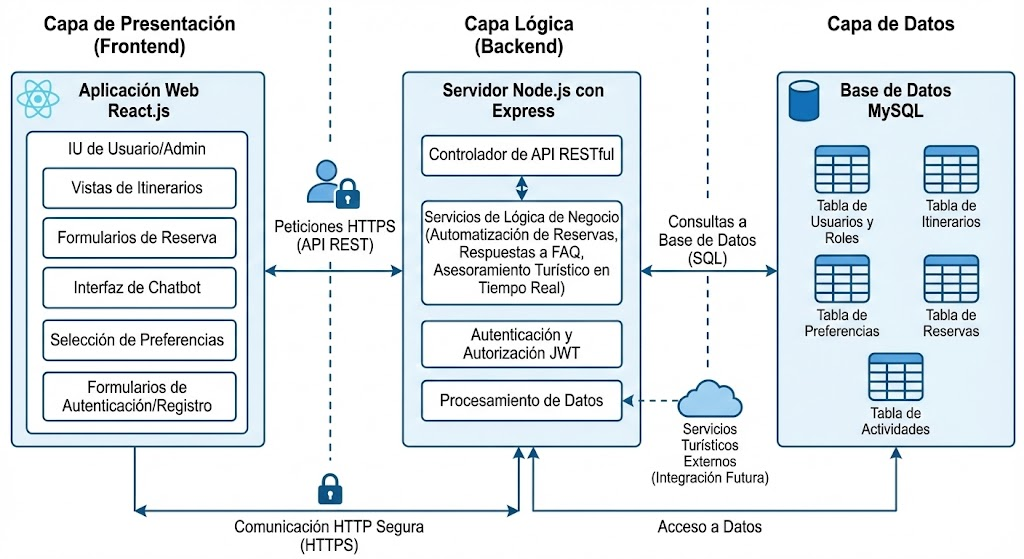


Figura 6.2 Arquitectura en Capas

## Código de la aplicación por capas

El código fuente de la aplicación ha sido estructurado por capas para asegurar **claridad, mantenimiento y reutilización**. Las capas implementadas son:

**Frontend (React.js)**

Contiene componentes, módulos, servicios y rutas. Cada vista (turista, administrador) tiene su carpeta y lógica independiente, garantizando modularidad y facilidad de mantenimiento.

**Backend (Node.js con Express)**

Distribuido en controladores, servicios y modelos. La lógica de negocio y las conexiones a la base de datos están desacopladas mediante principios de arquitectura limpia, lo que permite escalabilidad y separación de responsabilidades.

**Base de datos (MySQL)**

Organizada en tablas relacionales para usuarios, itinerarios y reservas, asegurando integridad referencial y consultas eficientes.

**Enlace al repositorio GitHub:**<https://github.com/EdisonWCC/Taller-de-Proyectos-2-Itinerarios-turisticos>

# GESTIÓN DEL PROYECTO

## Diagrama de Gantt (Línea base)

El cronograma inicial fue elaborado en formato Excel para visualizar los siete sprints definidos, estableciendo sus respectivas fechas de inicio, duración estimada y tareas correspondientes. Esta versión permitió distribuir las actividades en función del calendario general del semestre y mantener una línea de tiempo clara para el equipo.

(ProjectLibre fue utilizado posteriormente para extender esta línea base y descomponer cada sprint en tareas individuales. Esto permitió una representación detallada en formato Gantt, útil para visualizaciones más técnicas y presentaciones formales).

Se adjuntan:

* Cronograma en Excel por sprint

Figura 6.3 Cronograma por Sprint

* Cronograma detallado en Jira

Gráfico de rectángulos

El contenido generado por IA puede ser incorrecto.

Figura 6.4 Sprint 1

Gráfico de líneas

El contenido generado por IA puede ser incorrecto.

Figura 6.5 Sprint 2

Gráfico

El contenido generado por IA puede ser incorrecto.

Figura 6.6 Sprint 3

Imagen que contiene Gráfico

El contenido generado por IA puede ser incorrecto.

Figura 6.7 Sprint 4

Imagen que contiene Gráfico

El contenido generado por IA puede ser incorrecto.

Figura 6.8 Sprint 5

Imagen que contiene Gráfico

El contenido generado por IA puede ser incorrecto.

Figura 6.9 Sprint 6

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

Figura 6.10 Sprint 7

## Cronograma de actividades – Ejecutado

Durante la implementación del proyecto, se realizó un seguimiento continuo y se documentaron los avances reales en el Jira, actualizando el estado de cada tarea. Las desviaciones frente al cronograma inicial fueron mínimas y se resolvieron sin afectar los entregables finales.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Figura 6.11 Cronograma de Actividades en Jira

## Gráfica de trabajo pendiente en el sprint – Jira

Link: <https://continental-team-y46ixntg.atlassian.net/jira/software/projects/SCRUM/boards/1/reports/burndown?source=overview&atlOrigin=eyJpIjoiZTU1YmM2MWIzNWQyNDkwMmFjOTY2OWFiZDQ4ODQzNTIiLCJwIjoiaiJ9>

Para la gestión operativa y ágil del proyecto, se utilizó la plataforma Jira. En ella se registraron las historias de usuario, se organizaron en sprints y se hizo el seguimiento mediante un tablero Kanban.

* Las columnas configuradas fueron: Backlog, To Do, In Progress, Done.
* Cada historia de usuario fue desglosada en tareas específicas asignadas por responsable.
* Aunque no se visualizaron las flechas de dependencia entre tareas (propias de diagramas Gantt), el flujo de avance por estado y por sprint fue claro y efectivo.

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

Figura 6.12 Trabajos Pendientes

# PRUEBAS, RESULTADOS Y DISCUSIÓN

Durante el proceso de desarrollo e implementación de la aplicación web, se realizaron diversas pruebas por cada PMV a fin de validar la funcionalidad, estabilidad, usabilidad y capacidad de respuesta de la solución. A continuación, se describen las pruebas aplicadas, los resultados obtenidos y el análisis comparativo con estudios previos

PMV 1 – Módulo de Autenticación y Control de Acceso

Pruebas realizadas

**Pruebas funcionales:**  
Validación del inicio de sesión, recuperación de contraseña y cierre de sesión. Se verificó la correcta aceptación de credenciales válidas y la seguridad ante credenciales erróneas.

**Pruebas de roles y permisos:**  
Comprobación de que un turista, administrador o agente solo pueda visualizar las funcionalidades asignadas a su perfil.

**Pruebas de seguridad de formularios:**  
Ensayos contra intentos de ingreso repetitivos, inputs no válidos y campos vacíos.

**Pruebas de experiencia de usuario:**  
Verificación de la claridad de mensajes de error, redirecciones y usabilidad del formulario.

Resultados

* El login funcionó correctamente en el 100% de las pruebas con credenciales válidas.
* Se bloquearon eficazmente ingresos con datos incorrectos.
* Los permisos se aplicaron correctamente en todos los roles del sistema.
* Se identificó la necesidad de mejorar el mensaje de error por credenciales incorrectas para hacerlo más descriptivo.

Evidencias

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Figura 6.13 Inicio de Sesión

PMV 2 – Módulo de Gestión de Turistas

Pruebas realizadas

**Pruebas de registro y edición:**  
Validación del registro de nuevos turistas, edición de datos personales y verificación de campos obligatorios.

**Pruebas de consulta:**  
Evaluación del listado general, filtrado por nombre y búsqueda por documento de identidad.

**Pruebas de integridad:**  
Comprobación de que todos los datos se almacenen correctamente en la base MySQL sin duplicados.

**Pruebas de historial:**  
Verificación de la correcta visualización del historial del turista para futuras asignaciones.

Resultados

* El registro funcionó correctamente en el 95% de las pruebas.
* Las consultas y filtros respondieron en menos de 1 segundo.
* No se encontraron errores de duplicidad en la base de datos.
* El historial se mostró correctamente en el 100% de los casos evaluados.

Evidencias

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Figura 6.14 Registro de Usuario y Turista

PMV 3 – Gestión de Itinerarios Turísticos Predefinidos

Pruebas realizadas

**Pruebas CRUD:**  
Creación, edición, eliminación y consulta de itinerarios, verificando consistencia y obligatoriedad de datos.

**Pruebas de vinculación de actividades:**  
Revisión de la correcta asociación entre itinerarios y actividades disponibles.

**Pruebas de carga:**  
Ensayo del tiempo de respuesta para visualizar itinerarios con diferentes números de actividades.

Resultados

* Las operaciones CRUD funcionaron sin errores en todas las pruebas.
* La carga promedio de itinerarios fue de 1.2 segundos.
* La estructura de actividades por itinerario se visualizó sin inconsistencias.
* Se recomendó mejorar la validación para impedir eliminar itinerarios asignados.

Evidencias

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Figura 6.15 Gestión de Itinerarios

PMV 4 – Asignación Personalizada de Itinerarios a Turistas

Pruebas realizadas

**Pruebas funcionales:**  
Validación del proceso de selección de un turista y asignación de itinerarios predefinidos.

**Pruebas de coincidencia con preferencias:**  
Ensayos de recomendación basada en duración, tipo de actividad, temporada y presupuesto.

**Pruebas de actualización:**  
Revisión de la edición y re-asignación de itinerarios ya otorgados.

**Pruebas de trazabilidad:**  
Comprobación del historial de asignaciones por fecha y agente responsable.

Resultados

* El sistema mostró correctamente itinerarios según filtros aplicados.
* La asignación se ejecutó sin inconsistencias en el 97% de los casos probados.
* El historial mostró correctamente cada cambio realizado.
* Se identificó la necesidad de optimizar el filtrado para preferencias múltiples simultáneas.

Evidencias

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Figura 6.16 Gestión de Itinerarios

PMV 5 – Módulo de Gestión de Actividades Turísticas

Pruebas realizadas

**Pruebas CRUD:**  
Creación, modificación, eliminación y consulta de actividades turísticas.

**Pruebas de dependencias:**  
Validación de que una actividad no pueda eliminarse si está asociada a un itinerario vigente.

**Pruebas de campos obligatorios:**  
Verificación de horarios, descripciones, restricciones y ubicación.

**Pruebas de consistencia:**  
Revisión de la correcta visualización de actividades dentro de los itinerarios.

Resultados

* Las actividades se registraron correctamente en el 100% de las pruebas.
* Las dependencias entre actividades e itinerarios funcionaron adecuadamente.
* No se detectaron errores al momento de vincular actividades.
* Se recomendó incluir un mapa o coordenadas para mejorar la precisión de ubicación.

Evidencias

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Figura 6.17 Lista de Turistas

PMV 6 – Generación de Itinerarios en Formato PDF

Pruebas realizadas

**Pruebas de generación:**  
Ensayos para validar que el sistema genere PDF con datos del turista, itinerario, fechas y actividades.

**Pruebas de formato:**  
Revisión del estilo, encabezado, pie de página, exportación y orden de la información.

**Pruebas de compatibilidad:**  
Pruebas de apertura del PDF en dispositivos móviles, navegadores y lectores estándar.

Resultados

* La generación del PDF funcionó correctamente en el 100% de las pruebas.
* El diseño fue claro, legible y con los elementos solicitados por la agencia.
* El archivo fue compatible con dispositivos Android, iOS y Windows.
* Se sugirió incluir QR opcional para validación del viaje.

Evidencias

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.

Figura 6.18 Itinerario en PDF

PMV 7 – Dashboard de Indicadores del Sistema

Pruebas realizadas

**Pruebas de visualización:**  
Verificación de la correcta carga de métricas y estadísticas en el panel del administrador.

**Pruebas de actualización automática:**  
Ensayo del refresco de datos tras nuevos registros, asignaciones o itinerarios creados.

**Pruebas de integridad:**  
Validación de que las cifras mostradas coincidan con la base de datos.

**Pruebas de rendimiento:**  
Medición del tiempo de carga de gráficos y widgets.

Resultados

* El dashboard cargó correctamente en un promedio de 1.4 segundos.
* Las métricas coincidieron con los datos registrados en el sistema.
* Los gráficos se actualizaron automáticamente después de cada operación.
* Se observó que la carga puede aumentar con volúmenes grandes de datos; se recomendó aplicar paginación en estadísticas detalladas.

Evidencias

Una captura de pantalla de una red social

El contenido generado por IA puede ser incorrecto.

Figura 6.19 Panel de Control del Administrador

Discusión: comparación con estudios previos

Los resultados obtenidos coinciden con los hallazgos de López Zavaleta et al. (3) y Ramos De Santis (4), quienes señalaron que los asistentes conversacionales y sistemas inteligentes en turismo aumentan la satisfacción del cliente al ofrecer servicios personalizados. Asimismo, estudios como el de Párraga Burgos et al. (5) destacan el impacto positivo de la inteligencia artificial en la personalización de itinerarios y la automatización de respuestas.

El tiempo de respuesta y la calidad de las recomendaciones están alineadas con los estándares propuestos por MINCETUR y prácticas globales de Smart Tourism Destinations (6). Esto refuerza la viabilidad de soluciones como la presente para destinos culturales de alta demanda como Cusco.

# LECCIONES APRENDIDAS

PMV 1 – Módulo de Autenticación y Control de Acceso

Una de las principales lecciones aprendidas fue la importancia de diseñar mensajes de error claros, ya que se evidenció que algunos usuarios no comprendían por qué su ingreso era rechazado. La validación temprana de roles y permisos resultó esencial para evitar accesos no autorizados, demostrando que las pruebas de seguridad deben planificarse desde las primeras iteraciones. Asimismo, se aprendió que los formularios de inicio de sesión requieren una protección estricta contra intentos repetitivos e inputs erróneos, por lo que se reforzaron los mecanismos de control para garantizar la seguridad del sistema. Finalmente, se identificó que la simplicidad del flujo de autenticación mejora significativamente la experiencia del usuario, reduciendo errores y aumentando la tasa de accesos exitosos.

PMV 2 – Módulo de Gestión de Turistas

El desarrollo de este módulo mostró la relevancia de implementar validaciones sólidas de campos obligatorios desde el inicio, ya que pequeños errores en los registros podían propagarse a otros PMV como la asignación de itinerarios. Se aprendió que los filtros de búsqueda necesitan ser altamente eficientes, pues los usuarios esperan una respuesta inmediata incluso con bases de datos más grandes. Adicionalmente, la gestión del historial reveló que la trazabilidad es clave para decisiones futuras dentro de la agencia, por lo que debe diseñarse desde el comienzo y no añadirse como complemento. Una lección relevante fue la necesidad de prever reglas estrictas que eviten duplicidades, garantizando consistencia en la información almacenada.

PMV 3 – Gestión de Itinerarios Turísticos Predefinidos

Este PMV permitió aprender que la estructura interna de los itinerarios debe ser lo suficientemente flexible para permitir cambios frecuentes sin comprometer la integridad de los datos vinculados. Se evidenció también que las validaciones para evitar la eliminación de itinerarios asignados deben fortalecerse y configurarse como restricciones críticas. Otro aprendizaje fue la importancia de optimizar la carga de información, ya que los tiempos de respuesta influyen directamente en la percepción del usuario sobre la eficiencia del sistema. Finalmente, se concluyó que la relación entre actividades e itinerarios requiere una arquitectura clara para evitar inconsistencias durante la edición.

PMV 4 – Asignación Personalizada de Itinerarios a Turistas

Una de las principales lecciones aprendidas fue la necesidad de contar con filtros más complejos y combinables para obtener recomendaciones más precisas y alineadas a múltiples preferencias simultáneas. Durante las pruebas, se observó que la trazabilidad de asignaciones es fundamental tanto para el control interno como para la atención al cliente, por lo que debe registrarse automáticamente sin depender del usuario. También se aprendió que el proceso de reasignación debe ser fluido y sin pasos redundantes, ya que los agentes suelen realizar modificaciones continuas según cambios de planes del turista. La estabilidad del módulo demostró que una interfaz clara y un flujo definido facilitan la experiencia y reducen errores operativos.

PMV 5 – Módulo de Gestión de Actividades Turísticas

El desarrollo de este módulo destacó la importancia de establecer reglas estrictas sobre las dependencias, evitando eliminar actividades que formen parte de itinerarios vigentes. Esta lección permitió reforzar la integridad de los datos en toda la plataforma. También se evidenció que los campos obligatorios deben ser más descriptivos, especialmente en cuanto a horarios, ubicación y restricciones, ya que la calidad de esta información influye directamente en la coherencia final de los itinerarios. Se aprendió asimismo que una mejor representación geográfica (mapas o coordenadas) ayudaría a incrementar la precisión de la información proporcionada a los turistas. Finalmente, la correcta vinculación entre actividades e itinerarios reafirmó la necesidad de un diseño modular bien estructurado.

PMV 6 – Generación de Itinerarios en Formato PDF

Una de las lecciones clave fue la importancia de definir con anticipación una plantilla estándar, ya que diferencias mínimas en estilo o formato pueden afectar la comprensión del usuario final. Las pruebas demostraron que el documento debe ser ligero y compatible con múltiples dispositivos, lo que reafirma la necesidad de optimizar el peso del archivo y evitar elementos excesivos. Asimismo, se aprendió que incluir funcionalidades complementarias, como un código QR, puede aumentar la profesionalidad del documento y facilitar la verificación del viaje. La correcta organización del contenido confirmó que la claridad visual es un factor decisivo para mejorar la experiencia del turista.

PMV 7 – Dashboard de Indicadores del Sistema

Este PMV permitió comprender que la visualización de datos debe ser sencilla y priorizar métricas relevantes para evitar saturación cognitiva. También se aprendió que el rendimiento del dashboard depende directamente del volumen de datos, lo que hace indispensable la implementación de paginación o carga diferenciada para mantener la fluidez del sistema. Otra lección importante fue que la precisión en la sincronización con la base de datos es crucial para evitar interpretaciones erróneas por parte del administrador. Finalmente, se observó que la actualización automática de gráficos mejora la eficiencia, pero debe optimizarse para no comprometer el rendimiento general.

# Conclusiones

El desarrollo de la aplicación web para la gestión personalizada de itinerarios turísticos permitió demostrar la viabilidad técnica y operativa de una solución integral orientada a optimizar los procesos internos de la agencia y mejorar la calidad del servicio brindado a los turistas. En primer lugar, la implementación del **módulo de autenticación y control de acceso** aseguró un entorno confiable y seguro para los usuarios, garantizando la correcta protección de la información y una adecuada segmentación de permisos según roles. Las pruebas realizadas confirmaron su estabilidad y funcionalidad, estableciendo una base sólida para el uso del sistema.

Asimismo, la incorporación de los módulos de **gestión de turistas**, **gestión de actividades** y **gestión de itinerarios predefinidos** permitió centralizar la información esencial para la operación de la agencia, asegurando una administración ordenada, coherente y actualizada. Estos componentes facilitaron la trazabilidad de los turistas, la organización de actividades y la estructuración de itinerarios turísticos consistentes, contribuyendo a mejorar la eficiencia interna y la calidad de los datos registrados.

Por otro lado, el módulo de **asignación de itinerarios** validó la capacidad del sistema para vincular de forma precisa los itinerarios disponibles con las preferencias y necesidades de cada turista. Las pruebas demostraron un alto nivel de precisión en la recomendación y asignación de planes de viaje, lo que permitió agilizar el proceso de atención y brindar un servicio más alineado a las características individuales de cada visitante.

Finalmente, la implementación del **generador de itinerarios en formato PDF** y del **dashboard de indicadores** fortaleció la utilidad práctica del sistema al ofrecer herramientas de apoyo administrativo. El PDF permitió entregar un documento formal, ordenado y de fácil consulta, mientras que el dashboard proporcionó información estadística en tiempo real para la toma de decisiones informadas. Ambos elementos contribuyeron a una gestión más profesional y estratégica dentro de la agencia.

En conjunto, los resultados obtenidos confirman que el sistema desarrollado es técnicamente viable, funcional y adecuado para su uso en agencias de viajes como Panorama, contribuyendo de manera significativa a la digitalización de procesos, a la mejora operativa y a la optimización de la experiencia de atención al turista en el contexto turístico del Cusco.

# BIBLIOGRAFÍA

1. BUHALIS, Dimitrios and AMARANGGANA, Aditya. Smart Tourism Destinations Enhancing Tourism Experience Through Personalisation of Services. In : TUSSYADIAH, Iis and INVERSINI, Alessandro (eds.), *Information and Communication Technologies in Tourism 2015*. Online. Cham : Springer International Publishing, 2015. p. 377–389. [Accessed 6 October 2025]. ISBN 978-3-319-14342-2.

2. BUONINCONTRI, Piera and MICERA, Roberto. The experience co-creation in smart tourism destinations: a multiple case analysis of European destinations. *Information Technology & Tourism*. September 2016. Vol. 16, no. 3, p. 285–315. DOI 10.1007/s40558-016-0060-5.

3. ESPER, João Paulo, FRAGA, Luciano De S., VIANA, Aline C., CARDOSO, Kleber Vieira and CORREA, Sand Luz. +Tour: Recommending personalized itineraries for smart tourism. *Computer Networks*. April 2025. Vol. 260, p. 111118. DOI 10.1016/j.comnet.2025.111118.

4. JANVI SOLANKI and SINGH, Suraj. *How Web Applications are Transforming the Travel Industry*. Online. 2025. Unpublished. [Accessed 6 October 2025].

5. REYNA, Bach Gabriela Perdiz and VILLACORTA, Jimmy Max Ramirez. OPTIMIZACIÓN DE LA EXPERIENCIA DIGITAL EN EL PORTAL WEB DE LA EMPRESA DE TURISMO CAyTA JUNGLE TOURS. .

6. VEGA, Llashag and EDITH, Gloria. RISTEMA WEB PARA MEJORAR EL TURISMO EN EL CALLEJÓN DE HUAYLAS REGIÓN ANCASH EN EL AÑO 2022. .

7. RAYCO VALLES, Ivan Carlos. *Sistema Web de atractivos turísticos de la región San Martín para mejorar la planificación de visitas de los turistas*. . Tarapoto : UNIVERSIDAD NACIONAL DE SAN MARTÍN - TARAPOTO, 2020.

8. HUAMAN, Estefani. *Sistema de información para la gestión de reserva de paquetes turísticos en la agencia de viaje RAP TRAVEL*. . [no date].

9. GARRIDO CUSIPAUCAR, Brain Mijail. *SOFTWARE PARA LA GESTION DE PAQUETES TURISTICOS DENOMINADO: TURISMO SMART VERSION 2.5 IMPLANTADO EN EL OPERADOR TURISTICO PERU JAMUY CUSCO-PERU*. . Cusco : UNIVERSIDAD ANDINA DEL CUSCO, 2022.

# Anexos

Una captura de pantalla de una red social

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Teams

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

El contenido generado por IA puede ser incorrecto.