**REQUISITOS VIATGE**

**REQUISITOS FUNCIONALES**

REQUISITOS DE USUARIO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Entradas** | **Salidas** |
| RF001 | Registro en la plataforma | El sistema debe permitir al usuario el registro en la plataforma, para acceder a todas sus características. | Información de registro | Confirmación de registro |
| RF002 | Inicio de sesión en la plataforma | El sistema debe permitir al usuario iniciar sesión para que este pueda ingresar a la plataforma y hacer uso de ella. | Información de inicio de sesión | Acceso a la plataforma |
| RF003 | Creación de usuario | El sistema debe permitir la creación de un nuevo perfil de usuario. | Información del usuario | Confirmación de creación de usuario |
| RF004 | Actualización de usuario | El sistema debe permitir la actualización de la información del perfil del usuario. | Información actualizada del usuario | Confirmación de actualización del usuario |
| RF005 | Eliminación de usuario | El sistema debe permitir la eliminación de un perfil de usuario. | Información del usuario | Confirmación de eliminación de usuario |
| RF006 | Visualización de usuarios | El sistema debe permitir la visualización de la información del perfil de usuario. | Ninguna | Información del usuario |

REQUISITOS DE HOTEL

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Entradas** | **Salidas** |
| RF007 | Búsqueda de destinos | El sistema debe permitir al usuario realizar la búsqueda de diferentes destinos, lugares u hoteles. | Preferencias del usuario | Lista de destinos, lugares u hoteles |
| RF008 | Filtro por hoteles | El sistema debe permitir al usuario realizar un filtro por hoteles. | Criterios de filtro | Lista filtrada de hoteles |
| RF009 | Creación de hotel | El sistema debe permitir la creación de un nuevo registro de hotel. | Información del hotel | Confirmación de creación de hotel |
| RF010 | Actualización de hotel | El sistema debe permitir la actualización de la información del hotel. | Información actualizada del hotel | Confirmación de actualización del hotel |
| RF011 | Eliminación de hotel | El sistema debe permitir la eliminación de un registro de hotel. | Información del hotel | Confirmación de eliminación de hotel |
| RF012 | Visualización de hoteles | El sistema debe permitir la visualización de la información del hotel. | Ninguna | Información del hotel |

REQUISITOS DE RESERVA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Entradas** | **Salidas** |
| RF013 | Ingreso de fechas de reserva | El sistema debe permitir al usuario ingresar la fecha de entrada y salida para determinar la disponibilidad. | Fecha de entrada, Fecha de salida | Disponibilidad de alojamiento |
| RF014 | Ingreso de cantidad de huéspedes | El sistema debe permitir al usuario ingresar la cantidad de adultos y niños para ajustar la disponibilidad. | Cantidad de adultos, Cantidad de niños | Opciones de alojamiento |
| RF015 | Especificación de número de habitaciones | El sistema debe permitir al usuario especificar el número de habitaciones necesarias. | Número de habitaciones | Disponibilidad de alojamiento |
| RF016 | Visualización de actividades turísticas | El sistema debe permitir al usuario visualizar un listado de actividades turísticas disponibles en el destino elegido. | Ninguna | Listado de actividades turísticas |
| RF017 | Realización de reservas | El sistema debe permitir al usuario realizar reservas directamente a través de la plataforma. | Información de la reserva | Confirmación de la reserva |
| RF018 | Visualización de reservas realizadas | El sistema debe permitir al usuario visualizar las reservas realizadas. | Ninguna | Lista de reservas realizadas |
| RF019 | Edición de reservas realizadas | El sistema debe permitir al usuario editar las reservas realizadas. | Información de la reserva | Confirmación de la edición de la reserva |
| RF020 | Eliminación de reservas realizadas | El sistema debe permitir al usuario eliminar las reservas realizadas. | Información de la reserva | Confirmación de la eliminación de la reserva |

REQUISITOS DE FAVORITOS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Entradas** | **Salidas** |
| RF021 | Sección de favoritos | El sistema debe permitir al usuario tener una sección de favoritos para guardar hoteles, destinos y actividades favoritas. | Selección de favoritos | Lista de favoritos |
| RF022 | Añadir a favoritos | El sistema debe permitir al usuario añadir hoteles, destinos y actividades a su lista de favoritos. | Información del elemento a añadir | Confirmación de añadido a favoritos |
| RF023 | Eliminación de favoritos | El sistema debe permitir al usuario eliminar elementos de su lista de favoritos. | Información del favorito a eliminar | Confirmación de eliminación de favorito |

REQUISITOS DE ACTIVIDADES TURISTICAS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Entradas** | **Salidas** |
| RF024 | Visualización de actividades turísticas | El sistema debe permitir al usuario la visualización de las actividades turísticas o los servicios adicionales. | Ninguna | Lista de actividades turísticas |
| RF025 | Creación de actividad turística | El sistema debe permitir la creación de un nuevo registro de actividad turística. | Información de la actividad | Confirmación de creación de actividad |
| RF026 | Actualización de actividad turística | El sistema debe permitir la actualización de la información de una actividad turística. | Información actualizada de la actividad | Confirmación de actualización de actividad |
| RF027 | Eliminación de actividad turística | El sistema debe permitir la eliminación de un registro de actividad turística. | Información de la actividad | Confirmación de eliminación de actividad |
| RF028 | Visualización de actividades turísticas | El sistema debe permitir la visualización de la información de actividades turísticas. | Ninguna | Información de la actividad |

**REQUISITOS NO FUNCIONALES**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Tipo** |
| RN001 | Recuperación automática de fallos no críticos | El sistema debe ser capaz de recuperarse automáticamente de fallos no críticos sin afectar las operaciones normales. | Confiabilidad |
| RN002 | Máximo de fallas mensuales | El usuario no puede experimentar más de 3 fallas mensuales en la aplicación. | Confiabilidad |
| RN003 | Duración promedio de fallas no críticas | El promedio de duración de fallas no críticas no podrá ser mayor a 15 minutos. | Confiabilidad |
| RN004 | Disponibilidad continua | El sistema debe estar disponible continuamente, es decir con un nivel de disponibilidad para los usuarios 7 días por 24 horas. | Disponibilidad |
| RN005 | Tiempo de actividad del sistema | El sistema debe apuntar a un tiempo de actividad del 99.9999% durante las horas de operación normales. | Disponibilidad |
| RN006 | Manejo del aumento de carga | El sistema estará en la capacidad de manejar un aumento del 20% en la carga de trabajo sin degradación significativa del rendimiento. | Disponibilidad |
| RN007 | Ventanas de mantenimiento planificado | Se programarán ventanas bimestrales específicas de 2 horas, para llevar a cabo actividades de mantenimiento planificado. | Disponibilidad |
| RN008 | Longitud y complejidad de claves | La longitud de las claves de la aplicación debe ser de mínimo 8 caracteres e incluir símbolos, al menos una mayúscula y al menos un número. | Seguridad |
| RN009 | Cifrado de comunicación | Toda la comunicación entre el cliente y el servidor estará cifrada utilizando protocolos seguros como TLS/SSL. | Seguridad |
| RN010 | Rendimiento de servidores | Los servidores tendrán un rendimiento que garantice tiempos de respuesta rápidos para las solicitudes del usuario. | Infraestructura |
| RN011 | Ancho de banda adecuado | Se garantizará un ancho de banda adecuado para satisfacer las necesidades de tráfico actual y futuro. | Infraestructura |
| RN012 | Pruebas de rendimiento | Se realizarán pruebas de rendimiento para evaluar la capacidad de la red y se realizarán mejoras según sea necesario. | Infraestructura |
| RN013 | Minimización de latencia | Se implementarán medidas para minimizar la latencia y mantenerla dentro de los límites aceptables. | Infraestructura |
| RN014 | Políticas de acceso en cortafuegos | Se establecerán políticas de acceso en los cortafuegos para permitir el tráfico necesario y bloquear cualquier actividad no autorizada. | Infraestructura |
| RN015 | Tiempo promedio para reparar errores | El tiempo promedio para reparar un error no debe ser mayor a 8 horas. | Mantenibilidad |
| RN016 | Estándares de codificación y documentación | El código fuente seguirá estándares de codificación claros y estará bien documentado para facilitar su mantenimiento. | Mantenibilidad |
| RN017 | Documentación del sistema | El sistema contará con documentación que permitirá al personal encargado del mantenimiento informarse sobre cómo está constituido el sistema. | Mantenibilidad |
| RN018 | Metodologías ágiles para mantenimiento | Se usarán metodologías ágiles que proporcionarán prácticas y principios para la evolución continua del sistema. | Mantenibilidad |
| RN019 | Independencia de navegadores | El sistema será independiente de navegadores, permitiendo su ejecución en diferentes exploradores web como Chrome, Firefox o Edge. | Portabilidad |
| RN020 | Adaptabilidad a resoluciones de pantalla | La interfaz de usuario será adaptable a diversas resoluciones de pantalla, incluyendo dispositivos móviles y tabletas. | Portabilidad |
| RN021 | Pruebas de portabilidad | Se realizarán pruebas de portabilidad regulares para garantizar que el sistema funcione de manera correcta en diferentes entornos. | Portabilidad |

**RESTRICCIONES**

|  |  |  |
| --- | --- | --- |
| **ID** | **Nombre** | **Descripción** |
| RT001 | Precisión en la disponibilidad de habitaciones | El sistema deberá actualizar la disponibilidad de habitaciones con una precisión del 99.5% después de cada reserva o cancelación. |
| RT002 | Exactitud en los detalles de reservas | El sistema deberá registrar y mostrar los detalles de las reservas con una exactitud del 99% en la información relacionada con fechas, tipo de habitación, número de huéspedes y monto pagado. |
| RT003 | Precisión en las tarifas | Las tarifas mostradas en el sistema deben ser precisas en un 99.5%, incluyendo detalles como impuestos aplicados y costos adicionales. |
| RT004 | Diseño responsivo | El sistema será compatible con dispositivos de diferentes tamaños y resoluciones, utilizando CSS flexible y técnicas de diseño responsivo. |
| RT005 | Nomenclatura y estructura del código | Se utilizarán nombres descriptivos para variables, funciones y archivos, y se organizará el código en estructuras lógicas para mejorar su legibilidad. |
| RT006 | Control de versiones | Se mantendrá un control de versiones para el desarrollo colaborativo, brindando un seguimiento efectivo de cambios en el código. |
| RT007 | Visualización de datos con Chart.js | El sistema utilizará la biblioteca Chart.js para visualizar datos de inventario mediante gráficos interactivos en tiempo real. |
| RT008 | Navegación clara y lógica | Los elementos de navegación estarán organizados de manera lógica y clara, permitiendo a los usuarios moverse de manera fácil y eficiente. |
| RT009 | Iconos y etiquetas comprensibles | Se utilizarán iconos y etiquetas fácilmente comprensibles para representar acciones y funciones. |
| RT010 | Diseño intuitivo en pantallas pequeñas | Se garantizará que la interfaz sea intuitiva y fácil de usar en pantallas más pequeñas, utilizando principios de diseño responsivo. |
| RT011 | Minimización de instrucciones externas | Se minimizará la necesidad de instrucciones externas o manuales detallados, permitiendo que los usuarios comprendan fácilmente cómo utilizar la aplicación. |
| RT012 | Página de error personalizada | Se proporcionará al usuario una página de error personalizada que sea clara y amigable, evitando mensajes técnicos confusos. |
| RT013 | Mensajes de error descriptivos | Se proporcionarán mensajes de error descriptivos y comprensibles para el usuario. |
| RT014 | Copias de seguridad y recuperación de datos | Se implementarán estrategias de copias de seguridad y recuperación de datos para garantizar que los datos críticos estén protegidos y puedan ser restaurados en caso de pérdida. |

**DISEÑO Y ARQUITECTURA  
HERRAMIENTAS O LENGUAJES**

Para la realización del sistema se usarán las siguientes herramientas:

**Node.js**

Entorno de ejecución para JavaScript del lado del servidor, basado en el motor V8 de Google Chrome. Permite la creación de aplicaciones web y servidores de manera eficiente y escalable.

Este será usado debido a características como:

**Asincronía y No Bloqueo:** Node.js utiliza un modelo de entrada/salida no bloqueante, lo que permite manejar múltiples conexiones simultáneamente sin esperar a que se complete cada operación, resultando en un rendimiento más rápido y eficiente.

**Facilidad de Integración con JavaScript:** Al usar JavaScript tanto en el cliente como en el servidor, Node.js simplifica el desarrollo y mantenimiento del código, facilitando la integración entre el front-end y el back-end.

**Soporte para NPM (Node Package Manager):** Node.js cuenta con un vasto ecosistema de módulos y bibliotecas a través de NPM, lo que acelera el desarrollo al permitir la reutilización de paquetes y herramientas existentes.

**Escalabilidad y Rendimiento:** Node.js es ideal para aplicaciones que requieren alta escalabilidad y rendimiento en tiempo real, gracias a su arquitectura basada en eventos y su capacidad para manejar una gran cantidad de conexiones simultáneas.

**Amplia Comunidad y Documentación:** Node.js tiene una activa comunidad de desarrolladores y una extensa documentación, lo que facilita el aprendizaje y la resolución de problemas.

**Versatilidad:** Puede utilizarse para el desarrollo de aplicaciones web, APIs RESTful, y otros servicios del lado del servidor, ofreciendo flexibilidad en la creación de diferentes tipos de aplicaciones.

**Express**

Framework minimalista y flexible para Node.js que proporciona un conjunto robusto de características para el desarrollo web y la creación de APIs. Se utilizará para el diseño y arquitectura del sistema debido a sus ventajas como:

**Simplicidad y Eficiencia:** Express simplifica la configuración del servidor y el manejo de rutas, permitiendo construir aplicaciones web de manera rápida y eficiente.

**Middleware:** Permite el uso de middleware para manejar solicitudes, procesar datos y gestionar sesiones, proporcionando una estructura modular y extensible.

**Enrutamiento:** Ofrece un sistema de enrutamiento robusto y fácil de usar, lo que facilita la creación y gestión de rutas dentro de la aplicación.

**Integración con Módulos y Librerías:** Se integra fácilmente con otros módulos y bibliotecas de NPM, ampliando las capacidades del sistema sin complicaciones adicionales.

**Documentación y Comunidad:** Posee una buena documentación y una comunidad activa que contribuye al desarrollo de soluciones y mejores prácticas para la implementación.

**MYSQL Licencia pública general:**

Sistema de gestión de bases de datos relacional (RDBMS) de código abierto. Es ampliamente utilizado para almacenar y gestionar datos en aplicaciones web. MySQL utiliza el lenguaje de consulta estructurado (SQL) para realizar operaciones en la base de datos, como la inserción, actualización, recuperación y eliminación de datos.