Especificación de requisitos de software

Proyecto: Hoteles Sena
Revisión: 1

10/20024

Contenido

Introducción

El documento de Especificación de Requisitos de Software (SRS) para el proyecto Hoteles SENA describe las funciones, características y limitaciones del sistema que será desarrollado en Python. Este documento busca proporcionar una visión clara del software para facilitar su diseño, desarrollo y posterior implementación.

1.1 Propósito

El propósito de este documento es definir de manera clara y comprensible los requisitos del sistema Hoteles SENA, destinado a ofrecer servicios de hospedaje a los estudiantes del SENA. Este documento servirá como guía para los desarrolladores, instructores y usuarios interesados en el proyecto

1.2 Alcance

El sistema Hoteles SENA permitirá registrar usuarios, gestionar habitaciones, realizar reservas, procesar pagos, generar reportes y administrar una lista de usuarios. Está diseñado para mejorar la experiencia de hospedaje de los estudiantes y facilitar la gestión administrativa del alojamiento. El software será desarrollado en Python, con interfaz amigable y orientado al entorno académico del SENA.

1.3 Personal involucrado

Nombre	Andrés Serna
Rol	Programador
Categoría profesional	Analista y Desarrollador de software
Responsabilidades	Programar y Desarrollar parte del código, revisar su correcto funcionamiento
Información de contacto	Cmiloserna@gmail.com
Aprobación	Correspondiente

Nombre	Diego Silva
Rol	Programador
Categoría profesional	Analista y Desarrollador de software
Responsabilidades	Programar y Desarrollar parte del código, revisar su correcto funcionamiento
Información de contacto	
Aprobación	Correspondiente

Nombre	Maicol Steven
Rol	Analista de datos
Categoría profesional	Analista y Desarrollador de software
Responsabilidades	Encargado de analizar y manejar de manera eficiente los daros requeridos para el proyecto.
Información de contacto	
Aprobación	Correspondiente

Nombre	Frank
Rol	Programador
Categoría profesional	Analista y Desarrollador de software
Responsabilidades	Programar y Desarrollar parte del código, revisar su correcto funcionamiento
Información de contacto	
Aprobación	Correspondiente

1.4 Definiciones, acrónimos y abreviaturas

SENA: Servicio Nacional de Aprendizaje.
SRS: Software Requirements Specification (Especificación de Requisitos de Software).
Usuario: Persona que utiliza el sistema Hoteles SENA, ya sea estudiante o administrador.

1.5 Resumen

El resto del documento describe de forma estructurada la descripción general del sistema, los requisitos específicos, y los requisitos no funcionales del proyecto Hoteles SENA.

2 Descripción general

2.1 Perspectiva del producto

El sistema Hoteles SENA es una aplicación independiente desarrollada en Python. Está orientada a gestionar el alojamiento de estudiantes mediante módulos de registro, reservas, pagos y reportes. Puede integrarse en un futuro con otros sistemas académicos del SENA.

2.2 Funcionalidad del producto

El software permitirá: registrar usuarios, crear y editar habitaciones, realizar reservas, registrar pagos, consultar reportes y generar listados de usuarios. También validará datos y mostrará información de forma clara.

2.3 Características de los usuarios

Tipo de usuario	Común
Formación	Bachiller
Habilidades	Manejo solicitado del sistema
Actividades	Uso cotidiano del sistema y sus diferentes funcionalidades.

Tipo de usuario	Avanzado
Formación	Tecnólogo
Habilidades	Manejo solicitado del sistema
Actividades	Uso experto del sistema y sus diferentes funcionalidades.

Tipo de usuario	Administrador
Formación	Técnico
Habilidades	Control del sistema
Actividades	Administrador del sistema y sus diferentes funcionalidades.

2.4 Restricciones

- **Lenguaje y paradigma obligatorio:**

- El sistema **debe implementarse en Python**.
- Es **obligatorio aplicar principios de Programación Orientada a Objetos (POO)**: encapsulamiento, herencia, polimorfismo y composición.

Interfaz del sistema:

- No se permite el uso de interfaces gráficas complejas (GUI).
- La **interacción será únicamente por consola** (entrada y salida de texto).

Tiempo de desarrollo:

- El proyecto **debe completarse en un plazo máximo de 15 días**.

Tamaño y estructura del equipo:

- Los equipos estarán conformados por **tres o cuatro integrantes**.
- Cada miembro debe cumplir un rol definido dentro de Scrum (Product Owner, Scrum Master, Development Team).

Metodología de trabajo:

- Se debe **utilizar la metodología ágil Scrum**, con tres sprints de cinco días cada uno.
- Es obligatorio realizar los eventos: **Sprint Planning, Daily Scrum, Sprint Review y Sprint Retrospective**.

Módulos funcionales obligatorios:

- El sistema debe incluir y hacer funcionar correctamente los módulos de:
  - **Reservas**
  - **Clientes**
  - **Habitaciones**
  - **Pagos**

2.5 Suposiciones y dependencias

Se asume que los usuarios tendrán acceso a una computadora con conexión a internet y permisos para ejecutar el sistema. El software dependerá de bases de datos compatibles (por ejemplo, SQLite o MySQL).

2.6 Evolución previsible del sistema

En futuras versiones, el sistema podría incluir módulos de calificación de servicio, integración con correos electrónicos para confirmar reservas, y acceso móvil mediante aplicación Android o web.

3 Requisitos específicos

Número de requisito	RF 1
Nombre de requisito	Gestión de Usuarios

Tipo	[X] Requisito ☐ Restricción
Fuente del requisito	⍰ Registrar nuevos usuarios (clientes, empleados, administradores). ⍰ Iniciar y cerrar sesión con credenciales seguras. ⍰ Asignar roles y permisos según el tipo de usuario.
Prioridad del requisito	[X] Alta/Esencial ☐ Media/Deseado ☐ Baja/ Opcional

Número de requisito	RF 2
Nombre de requisito	Gestión de habitaciones
Tipo	[X] Requisito ☐ Restricción
Fuente del requisito	⍰ Registrar, modificar y eliminar habitaciones. ⍰ Definir tipo de habitación (sencilla, doble, suite, etc.). ⍰ Indicar el estado de la habitación (disponible, ocupada, mantenimiento).
Prioridad del requisito	[X] Alta/Esencial ☐ Media/Deseado ☐ Baja/ Opcional

Número de requisito	RF 3
Nombre de requisito	Reservas
Tipo	[X] Requisito ☐ Restricción
Fuente del requisito	⍰ Crear nuevas reservas (fecha de entrada, salida, habitación, cliente). ⍰ Consultar disponibilidad en un rango de fechas. ⍰ Modificar o cancelar reservas existentes.
Prioridad del requisito	[X] Alta/Esencial ☐ Media/Deseado ☐ Baja/ Opcional

Número de requisito	RF 4
Nombre de requisito	Pagos
Tipo	[X] Requisito ☐ Restricción
Fuente del requisito	⍰ Registrar pagos de clientes (efectivo, tarjeta, transferencia). ⍰ Generar facturas o comprobantes. ⍰ Calcular totales con impuestos o descuentos.
Prioridad del requisito	[X] Alta/Esencial ☐ Media/Deseado ☐ Baja/ Opcional

Número de requisito	RF 5
Nombre de requisito	Reportes
Tipo	[X] Requisito ☐ Restricción
Fuente del requisito	⍰ Generar reportes de ocupación, ingresos y clientes. ⍰ Exportar reportes a PDF o Excel.
Prioridad del requisito	[ ] Alta/Esencial [X] Media/Deseado ☐ Baja/ Opcional

Número de requisito	RF 6
Nombre de requisito	Lista de usuarios y clientes
Tipo	[X] Requisito ☐ Restricción
Fuente del requisito	⍰ Visualizar y buscar usuarios registrados. ⍰ Filtrar por tipo de usuario o estado de la cuenta.
Prioridad del requisito	[ ] Alta/Esencial [X] Media/Deseado ☐ Baja/ Opcional

3.1 Requisitos comunes de los interfaces

3.1.1 Interfaces de usuario

El requisito de la **interfaz de usuario** establece que el sistema debe contar con una presentación visual clara, intuitiva y fácil de usar, permitiendo que cualquier usuario —ya sea cliente, recepcionista o administrador— pueda navegar y realizar sus tareas sin necesidad de conocimientos técnicos avanzados. La interfaz debe mostrar de forma ordenada las opciones principales, como gestión de reservas, habitaciones, usuarios y reportes, garantizando una experiencia fluida y agradable. Además, debe mantener coherencia en colores, tipografía y botones, facilitando la comprensión y reduciendo errores de uso.

3.1.2 Interfaces de hardware

El requisito de las **interfaces de hardware** establece que el sistema debe ser compatible con los dispositivos físicos necesarios para su funcionamiento y operación en un entorno hotelero. Esto incluye equipos como computadores de escritorio o portátiles con teclado, ratón y monitor.

3.1.3 Interfaces de software

Producto de software: Sistema de Base de Datos

Descripción:
Motor de base de datos relacional encargado de almacenar toda la información del sistema, incluyendo usuarios, habitaciones, reservas..

3.2 Requisitos funcionales

1. Registrar nuevos usuarios (clientes, empleados, administradores)
2. Registrar, modificar y eliminar habitaciones

3. Definir tipo de habitación (sencilla, doble, suite, etc.)
4. Indicar el estado de la habitación (disponible, ocupada, mantenimiento)
5. Crear nuevas reservas (fecha de entrada, salida, habitación, cliente)
6. Consultar disponibilidad en un rango de fechas
7. Modificar o cancelar reservas existentes

3.2.1 Requisito funcional 1

1. Registrar nuevos usuarios (clientes, empleados, administradores)

- Permite agregar nuevos clientes, empleados y administradores al sistema, guardando su información básica y credenciales.

3.2.2 Requisito funcional 2

2. Registrar, modificar y eliminar habitaciones

- Permite registrar, modificar o eliminar habitaciones, indicando su tipo, precio y estado (disponible, ocupada o en mantenimiento).

3.2.3 Requisito funcional 3

3. Definir tipo de habitación (sencilla, doble, suite, etc.)

- Permite clasificar las habitaciones según sus características, como sencilla, doble o suite, definiendo precios y servicios asociados a cada tipo para facilitar la gestión y reservas.

3.2.4 Requisito funcional 4

4. Indicar el estado de la habitación (disponible, ocupada, mantenimiento)

- Muestra las habitaciones disponibles según fechas seleccionadas, facilitando la planificación de reservas.

3.2.5 Requisito funcional 5

5. **Crear nuevas reservas (fecha de entrada, salida, habitación, cliente)**

- Permite crear, actualizar o cancelar reservas, registrando fechas, cliente, habitación y estado de la reserva.

3.2.6 Requisito funcional 6

6. Consultar disponibilidad en un rango de fechas

- Permite al usuario seleccionar un rango de fechas específico (entrada y salida) para realizar búsquedas, reservas o reportes, asegurando que no se generen conflictos entre las reservas existentes.

3.2.7 Requisito funcional 7

7. Modificar o cancelar reservas existentes

- Permite crear, actualizar o cancelar reservas, registrando fechas, cliente, habitación y estado de la reserva.

3.3 REQUISITOS NO FUNCIONALES

3.3.1. Usabilidad

- El sistema debe ofrecer una interfaz gráfica sencilla, clara e intuitiva, de modo que cualquier usuario —cliente, recepcionista o administrador— pueda operar las funciones básicas sin capacitación técnica avanzada.

3.3.2. Rendimiento

- El sistema debe responder de forma rápida ante las acciones del usuario, con tiempos de carga y consulta inferiores a 3 segundos en operaciones estándar como registro, búsqueda o generación de reportes.

3.3.3. Seguridad

- Debe garantizar la protección de los datos de los usuarios y del hotel mediante el uso de contraseñas cifradas (por ejemplo, con `bcrypt`) y control de acceso basado en roles. Además, las operaciones sensibles deben requerir autenticación.

3.3.4. Confiabilidad

- El sistema debe funcionar de forma estable, sin errores críticos durante su ejecución, y conservar la integridad de los datos ante fallos o cierres inesperados.

3.3.5. Escalabilidad

- El diseño debe permitir agregar nuevas funcionalidades en el futuro (por ejemplo, inventario, mantenimiento o reportes avanzados) sin afectar la estructura del sistema.

3.3.6. Mantenibilidad

- El código debe estar organizado, documentado y estructurado bajo el modelo **MVC (Modelo - Vista - Controlador)**, para facilitar su comprensión, modificación y depuración.

3.3.7. Compatibilidad

- El sistema debe poder ejecutarse en diferentes sistemas operativos (Windows, Linux o macOS) y con versiones de Python 3.8 o superiores.

3.3.8. Portabilidad

- Debe ser posible trasladar el sistema de un equipo a otro sin pérdida de datos, siempre que se mantenga la misma configuración de entorno y base de datos.

3.3.1 9. Disponibilidad

- El sistema debe estar disponible para su uso en cualquier momento dentro del entorno local del hotel, y garantizar un acceso continuo siempre que el equipo esté encendido.

3.3.2 10. Interoperabilidad

- El sistema debe ser capaz de comunicarse con otros programas o servicios compatibles, como motores de base de datos (SQLite, MySQL) y herramientas externas de generación de reportes o envío de correos electrónicos (ReportLab, SMTP).