
Especificación de Requisitos de Software (SRS) Plan de Gestión de Proyecto (PGP)

Proyecto: Alquilando Web

Revisión: 1.0



Especificación de Requisitos de Software (SRS)

1) Introducción

a. Propósito y alcance

El presente documento tiene como propósito presentar la especificación de requerimientos con el objetivo de desarrollar una plataforma web para la empresa Alquilando. Igualmente se detallan las funciones determinadas con sus dueños y el equipo de desarrollo de software 4Byte.

Este documento está dirigido tanto al equipo de desarrollo como a los responsables de la toma de decisiones en la empresa Alquilando.

b. Definiciones, acrónimos y abreviaturas a considerar

SRS: Software Requirements Specification (Especificación de Requisitos de Software).

Check-out: Proceso mediante el cual un huésped finaliza su estadía en un alojamiento.

4Byte: Nombre del equipo de desarrollo responsable del proyecto.

Usuario registrado: Persona que ha creado una cuenta en la plataforma.

Administrador: Usuario con privilegios para gestionar usuarios y alojamientos.

Encargado: Usuario con permisos específicos para gestionar las salidas de los huéspedes.

Responsive: Capacidad del sistema de adaptarse a diferentes tamaños de pantalla y dispositivos.

Épicas: Historias de usuario de alto nivel que agrupan múltiples funcionalidades o requerimientos.

c. Referencias

Título	Fecha	Autor
Entrevista 1	12/03/25	4Byte
Entrevista 2	19/03/25	4Byte
Cuestionario	26/03/25	4Byte
Épicas	26/03/25	4Byte

2) Descripción general

a. Resumen de la idea del producto

El producto consiste en una plataforma digital para la gestión de alquileres temporales que permitirá a los usuarios realizar reservas de manera ágil y simple.

Además la plataforma incluirá funciones para la gestión de las reservas facilitando tareas como actualizar la disponibilidad de los alojamientos, registrar los check-out, entre otros.

b. Perspectiva del producto

El sistema a desarrollar será un producto independiente, puesto que puede operar de manera autónoma y no depende directamente de otros sistemas para su funcionamiento central. Sin

embargo, la plataforma requerirá integraciones con servicios externos, para llevar a cabo la gestión de transacciones económicas (por ejemplo, pagos de reservas y devoluciones).

c. Características de los usuarios

Rol de usuario	Administrador
Resumen de actividades	<ul style="list-style-type: none"> ● Subir departamento ● Bajar departamento ● Editar departamentos ● Listar usuarios ● Deshabilitar usuario ● Aplicar descuento ● Enviar Mensaje como Administrador

Rol de usuario	Encargado
Resumen de actividades	<ul style="list-style-type: none"> ● Realizar check-out ● Enviar Mensaje como Encargado

Rol de usuario	Usuario Registrado
Resumen de actividades	<ul style="list-style-type: none"> ● Iniciar Sesión ● Buscar alojamientos ● Filtrar alojamientos ● Ver información de alojamientos ● Hacer Reserva ● Cancelar Reserva ● Calificar alojamiento ● Enviar Mensaje como Usuario ● Cerrar Sesión ● Eliminar cuenta

Rol de usuario	Usuario no registrado
Resumen de actividades	<ul style="list-style-type: none"> ● Registrar usuario ● Buscar alojamientos ● Ver información de alojamientos

d. Evolución previsible del sistema

En el futuro el cliente desea que el sistema se integre con AirBnB o Booking para poder sincronizar las reservas o publicaciones.

3) Requisitos del Software

Requisitos de Interfaz

a. Interfaz de Usuario

El cliente nos indicó que el estilo sería libre, respetando los colores que se presentan en el logo de la empresa. La interfaz requerida será limpia y moderna, con un diseño minimalista que prioriza imágenes grandes y tipografía clara; con elementos visuales intuitivos para facilitar la navegación y la búsqueda de alojamientos.

b. Interfaces de Software

El producto se integrará con un servidor externo de un banco en el momento de efectuar el pago, permitiendo así procesar las transacciones de manera segura y en tiempo real.

c. Interfaces de Hardware

N/A

Requisitos funcionales

- Solicitar reserva -
- Aceptar reserva -
- Cancelar reserva -
- Registrar cuenta -
- Iniciar sesión -
- Cerrar sesión -
- Eliminar cuenta -
- Puntuar alojamiento -
- Añadir comentario -
- Responder comentario -
- Modificar reseña -
- Cambiar alojamiento
- Aplicar descuento -
- Listar usuarios -
- Recuperar contraseña -
- Deshabilitar usuario
- Habilitar usuario
- Ver historial de mensajes -
- Enviar Mensaje como Administrador
- Enviar Mensaje como Usuario
- Enviar Mensaje como Encargado
- Modificar datos de la tarjeta -
- Buscar alojamiento -
- Publicar alojamiento -
- Ver información de alojamientos -
- Editar publicación -
- Archivar publicación -
- Desarchivar publicación -
- Habilitar fechas -

- Deshabilitar fechas -
- Realizar check-out

Reglas de negocio más importantes:

- El usuario debe ser mayor de 18 años.
- Los pagos sólo se realizan con tarjetas de crédito.
- Los administradores pueden enviar mensajes desde la reserva hasta el inicio de la estadía.
- Solo se puede puntuar hasta 2 semanas después del check-out.
- Los usuarios pueden enviar mensajes desde el momento de la reserva hasta una semana después de terminada la estadía.
- Según el estado de la reserva, el mensaje se envía al administrador o al encargado.
- Los encargados pueden enviar mensajes desde el inicio de la estadía hasta una semana después de terminada la misma.
- Los usuarios pueden editar sus reseñas hasta 2 semanas después del check-out.
- Los usuarios no pueden comentar sin puntuar.
- No se puede eliminar una cuenta si el usuario tiene reservas en curso o futuras
- Se puede deshabilitar un usuario que no tiene una reserva en curso, si tiene reservas futuras éstas se cancelan.
- El usuario puede realizar reservas que se solapan con otras, pero deberá aclarar a los inquilinos.

Requisitos no funcionales

- Las contraseñas no serán visibles en ningún momento desde el sistema, ni para usuarios ni para administradores.
- El sistema implementará validaciones de acceso y roles de usuario para restringir el acceso a información según permisos.
- El código será estructurado y documentado siguiendo buenas prácticas de desarrollo, lo que facilitará su mantenimiento y evolución.
- La interfaz será intuitiva y amigable, buscando ofrecer una buena experiencia al usuario.
- El sistema será responsive, de modo que se pueda visualizar correctamente desde distintos dispositivos electrónicos (computadora, celular, etc).

Plan de Gestión de Proyecto (PGP)

1) Introducción

a. Propósito y alcance

El propósito del proyecto Alquilando es crear una página web en la que se puede realizar reservas de alquileres temporarios. Permitiendo a los usuarios buscar, reservar o cancelar alquileres de alojamientos. Y permitiendo que los administradores se encarguen fácilmente de la gestión de publicaciones, de reservas y de alojamientos.

Este documento va dirigido a los desarrolladores y a los dueños de la empresa Alquilando.

b. Definiciones, acrónimos y abreviaturas a considerar

PGP (*Plan de Gestión del Proyecto*): Documento que define cómo se organizará, ejecutará y controlará el proyecto. Incluye planificación de tareas, recursos, presupuesto, riesgos y entregables.

Product Owner: Representante del cliente. Es responsable de definir y priorizar las funcionalidades del sistema y validar los entregables durante todo el proyecto.

Scrum: Marco de trabajo ágil para el desarrollo de software. Organiza el trabajo en ciclos iterativos llamados sprints.

Sprint: Iteración de tiempo fijo (generalmente entre 2 y 4 semanas) donde se desarrolla y entrega una parte funcional del sistema.

Demo: Demostración del sistema funcional al finalizar un sprint o una etapa clave del desarrollo, con el objetivo de mostrar avances al cliente.

Frontend: Parte del sistema visible para el usuario. Incluye la interfaz gráfica y la interacción del usuario con la aplicación.

Backend: Parte del sistema que maneja la lógica de negocio, base de datos, servicios, y procesos internos que no son visibles para el usuario.

Hosting: Servicio que permite publicar un sitio web o sistema en Internet para que esté disponible públicamente.

Dominio: Dirección web única que identifica a la aplicación.

Base de Datos: Sistema que almacena la información del sistema, como usuarios, alojamientos, reservas, etc.

c. Referencias

Título	Fecha	Autor
Entrevista 1	12/03/25	4Byte
Entrevista 2	19/03/25	4Byte
Cuestionario	26/03/25	4Byte
Épicas	26/03/25	4Byte

2) Planes generales

a. Entregables del proyecto

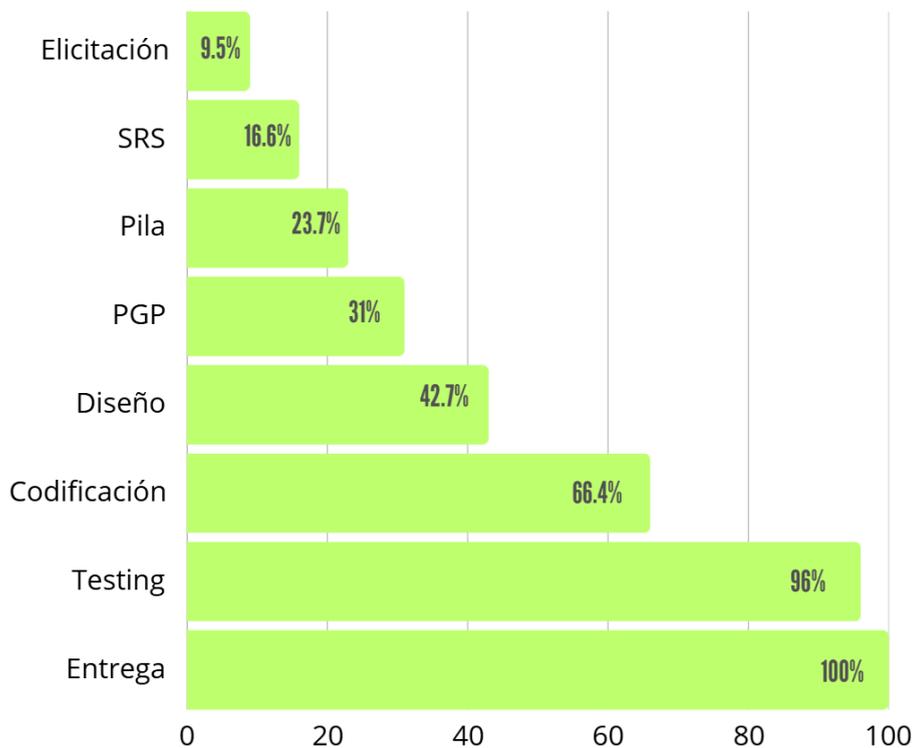
Se realizará 2 entregas de documentación del producto y 2 entregas del sistema (a las que llamaremos demos) Las entregas quedaron pautadas para las siguientes fechas:

- Entrega 1: 28/03/25
- Entrega 2: 25/04/25
- Demo 1: 06/06/25
- Demo 2: 02/07/25

b. Calendario y resumen del presupuesto

El proyecto de desarrollo será de aproximadamente 5 meses. El desarrollo del sistema inicia desde el día 05/03/25 hasta el 02/07/25. El presupuesto final destinado a este proyecto será de 16857 dólares.

Porcentaje del presupuesto para cada etapa:



c. Plan del personal

Analistas Funcionales

- Cantidad: 3
- Duración estimada: Primer mes del proyecto (Fases de relevamiento y análisis)
- Responsabilidades: Entrevistas con el cliente, definición de requerimientos, elaboración del SRS y PGP.

Desarrolladores Full-Stack

- Cantidad: 3
- Duración estimada: 5 meses (todo el proyecto)
- Responsabilidades: Implementación de la lógica de negocio, desarrollo de la interfaz de usuario, integración con servicios externos y base de datos.

Scrum Master

- Cantidad: 1
- Duración estimada: 5 meses
- Responsabilidades: Coordinación general del equipo, seguimiento de avances, gestión de tareas, reuniones con el cliente y resolución de impedimentos.

Tester

- Cantidad: 2
- Duración estimada: Desde el tercer mes hasta el final del proyecto (3 meses)
- Responsabilidades: Diseño y ejecución de casos de prueba, testing funcional y de regresión, automatización de pruebas, control de calidad del producto final.

Participación del cliente

- Rol: Dueño del producto (Product Owner)
- Duración estimada: Todo el proyecto
- Responsabilidades: Validación de funcionalidades, revisiones en cada sprint, toma de decisiones funcionales.

3) Presupuesto

a. Principales actividades del proyecto

El proyecto tendrá las siguientes tareas:

1. Elicitación de requerimientos: se realizarán 2 entrevistas para obtener los requerimientos del cliente.
2. Elaboración del SRS: se realizará la especificación de los requisitos de software.
3. Pila: realización de las historias de usuario.
4. Planificación: se llevará a cabo el documento PGP, el cual permite estimaciones razonables de recursos, costo y planificación temporal.
5. Diseño
6. Codificación (backend, frontend)
7. Testing
8. Entrega del producto

b. Asignación de esfuerzo

Actividad	Personas involucradas	Esfuerzo unitario	Esfuerzo total
Elicitación de requerimientos (2 entrevistas)	4	8 horas (2 entrevistas de 40 minutos + tiempo de análisis de las entrevistas)	$4 \times 8 = 32$ horas.
Elaboración del SRS	4	6 horas	$4 \times 6 = 24$ horas
Pila	4	6 horas	$4 \times 6 = 24$ horas
Planificación: documento PGP	4	6 horas	$4 \times 6 = 24$ horas
Diseño	4	10 horas	$4 \times 10 = 40$ horas
Codificación (backend, frontend)	4	20 horas	$4 \times 20 = 80$ horas
Testing	4	25 horas	$4 \times 25 = 100$ horas
Entrega del Producto	4	2 horas	$2 \times 4 = 8$ horas
Cantidad de horas destinadas al proyecto:			332 horas

c. Presupuesto final

- Cantidad de horas del proyecto: 332 horas.
- Precio por hora: 50 dólares.
- Recursos adicionales : 257 dólares.
 - Dominio web (.com): 12 dólares/año.
 - Hosting básico: 120 dólares/año.
 - Base de datos: 25 dólares/mes x 5 meses.

El valor del presupuesto total será: 16857 dólares.

4) Riesgos

1. Cambios de requerimientos

→ Responsable: Líder del proyecto

→ Probabilidad: 80%

→ Impacto: 2

→ Mitigación:

- Definir y documentar los requerimientos de forma clara y completa desde el inicio.
- Firmar un documento de aprobación de requerimientos previo al desarrollo.

→ Plan de contingencia:

- Priorizar los cambios más críticos.
- Tener seguimiento de las consecuencias de aplicar los cambios de requerimientos.

2. Enfermedad de los desarrolladores

→ Responsable: Líder del proyecto

→ Probabilidad: 70%

→ Impacto: 2

→ Mitigación:

- Mantener el área de trabajo higienizado.
- Tener las vacunaciones al día.

→ Plan de contingencia:

- Tener una distribución de trabajo diferente en caso de ausencias.
- Tener un personal de la salud de cabecera.

3. Fallo en la estimación del cronograma

→ Responsable: Líder del proyecto

→ Probabilidad: 50%

→ Impacto: 2

→ Mitigación:

- Agregar un margen de seguridad de tareas críticas o inciertas
- Actualizar estimaciones periódicamente
- Realizar estimaciones colaborativas con todo el equipo

→ Plan de contingencia:

- Repriorizar tareas
- Postergar funcionalidades no críticas
- Renegociar fechas de entrega

4. Problemas de integración

→ Responsable: Equipo de desarrollo

→ Probabilidad: 40%

→ Impacto: 3

→ Mitigación:

- Mantener una comunicación constante con propietarios de sistemas externos.
- Realizar pruebas de integración continuamente.

→ Plan de contingencia:

- Tener otro sistema de respaldo.

5. Fallos de seguridad

→ Responsable: Equipo de desarrollo

→ Probabilidad: 10%

→ Impacto: 1

→ Mitigación:

- Encriptar contraseñas
- Usar una comunicación segura (como HTTPS)

→ Plan de contingencia:

- Cambiar las credenciales afectadas.
- Revisar registros de actividad para rastrear accesos sospechosos.

6. Fallos de la base de datos

→ Responsable: Equipo de desarrollo

→ Probabilidad: 5%

→ Impacto: 1

→ Mitigación:

- Realizar copias de seguridad periódicamente.
- Optimizar las consultas más críticas para evitar deadlocks.
- Monitoreo constante.

→ Plan de contingencia:

- Restaurar la última copia de seguridad válida.
- Investigar el causante del problema y resolverlo.