

JetGo

Especificaciones de Requerimientos de Software

3.0

11/11/2025

Bautista Barbero
Lautaro Benegas
Lorenzo Chialvo
Juan de la Fuente
Franco Gregoris
Benicio Manzotti
Martino Nahuel
Franco Rosas
Valentin Zabala

Lead Software Developer

Prepared for: Benicio Manzotti / Valentin Zabala

Teacher: Juan Frattin / Pablo Canepa.

- **Revision History**

Date	Description	Author	Comments
5/20/2025	Version 1	Manzotti/Zabala	Version 1 del srs de JetGo
8/25/2025	Version 2	Manzotti/Zabala	Version 2 del srs
10/18/2025	Version 3	Martino/Gregoris	Version 3 del srs

- **Document Approval**

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date

Table of Contents

• Revision History.....	2
• Document Approval.....	2
1.1 Propósito.....	5
1.2 Alcance.....	6
1.3 Definiciones, Acrónimos y Abreviaturas.....	7
1.4 Referencias.....	7
1.5 Visión General.....	8
2.1 Perspectiva del Producto.....	8
2.2 Funcionalidad del Producto.....	8
2.3 Características de los Usuarios.....	9
2.4 Restricciones Generales.....	9
2.5 Suposiciones y Dependencias.....	10
3.1 Requisitos de Interfaz Externa.....	10
3.1.1 Interfaz de Usuario	
JetGo contará con una interfaz amigable, moderna y responsiva. Las principales pantallas serán:.....	10
3.1.2 Interfaz de Hardware.....	11
3.1.3 Interfaz de Software.....	11
3.1.4 Interfaz de Comunicación.....	11
3.2 Requisitos Funcionales (Ampliado).....	12
3.2.6 Editar o eliminar un viaje.....	12
3.2.7 Editar perfil.....	12
3.2.8 Chat entre usuarios de un viaje.....	12
3.2.9 Calificar usuarios después de un viaje.....	12
3.2.10 Notificaciones automáticas.....	12
3.2.11 Sugerencia de viajes compatibles.....	12
3.2.12 Marcar viaje como favorito.....	12
3.2.13 Verificación de identidad.....	13
3.2.14 Ver sección de ayuda / contacto.....	13
3.3 Casos de Uso (Ampliado).....	13
3.3.3 UC-03: Enviar mensaje en el chat del viaje.....	13
3.3.4 UC-04: Calificar a otro usuario.....	14
3.3.5 UC-05: Marcar viaje como favorito.....	14
3.3.6 UC-06: Ver notificaciones.....	14
3.4 Clases / Objetos (Ampliado).....	15
3.4.3 Calificación.....	15

3.4.4 Mensaje.....	15
3.4.5 Notificación.....	15
3.4.6 Solicitud.....	15
3.4.7 Preferencia.....	15
3.5 Requisitos No Funcionales (Ampliado).....	15
3.5.7 Usabilidad.....	15
3.5.8 Escalabilidad.....	15
3.5.9 Accesibilidad.....	16
3.6 Restricciones de Diseño (Ampliado).....	16
3.7 Requisitos Lógicos de Base de Datos (Ampliado).....	16
3.8 Otros Requisitos (Ampliado).....	16
4.1 Sequence Diagrams.....	17
4.1.1 Registro de Usuario.....	17
4.1.2 Inicio de Sesión.....	17
4.1.3 Solicitud de Reporte.....	18
A. Appendices.....	19
A.1 Appendix 1 – Acta de reunión con el cliente.....	19
A.2 Appendix 2 – Documento conceptual inicial.....	19

1. Introducción

Este documento constituye la Especificación de Requerimientos de Software (SRS) para el proyecto JetGo, una aplicación web que permite planificar y coordinar viajes compartidos entre usuarios, incluyendo aspectos como transporte, alojamiento y actividades.

El propósito de esta introducción es ofrecer una visión general del contenido y la organización del SRS, de manera que sirva como guía para todas las partes interesadas durante el ciclo de vida del desarrollo. Este documento proporciona toda la información necesaria para que un arquitecto de software pueda diseñar e implementar correctamente el sistema, en base a los requisitos establecidos.

Los requerimientos aquí descritos fueron elaborados siguiendo las recomendaciones del IEEE Std 830-1998, y están orientados a garantizar que el producto final cumpla con las expectativas del usuario, los objetivos del negocio y las restricciones técnicas impuestas por el entorno de desarrollo.

A lo largo del SRS se incluyen:

- Una descripción detallada del propósito y alcance del sistema.
- Definiciones clave, referencias relevantes y organización del documento.
- Requisitos funcionales y no funcionales del sistema.
- Interfaces externas e internas.
- Supuestos, limitaciones, dependencias y restricciones de diseño.

Este documento será utilizado como una fuente de verdad a lo largo del desarrollo del proyecto, sirviendo como base para las decisiones de arquitectura, diseño, programación, pruebas y evaluación del producto.

1.1 Propósito

Este documento especifica los requerimientos del sistema para JetGo, una aplicación web que permite a los usuarios planificar y coordinar viajes compartidos con otras personas, incluyendo transporte, alojamiento y actividades turísticas.

El propósito del presente SRS es comunicar de forma clara, precisa y completa las funcionalidades, restricciones, interfaces y requisitos del sistema a todos los actores involucrados: desarrolladores, testers, analistas, usuarios finales, docentes y gerentes de proyecto.

1.2 Alcance

JetGo es una plataforma de viajes colaborativos que conecta a personas interesadas en realizar trayectos compartidos y planificar viajes en grupo. A diferencia de otras aplicaciones de carpooling, JetGo permite coordinar viajes completos incluyendo transporte, estadías y actividades.

El sistema permitirá:

- Crear propuestas de viaje indicando destino, fechas, alojamiento, cupo y actividades.
- Buscar y unirse a viajes ya existentes.
- Comunicarse con otros viajeros mediante chat integrado.
- Gestionar solicitudes recibidas y enviadas.
- Calificar a los compañeros de viaje luego de finalizar la experiencia.
- Visualizar sugerencias de viajes compatibles en función de intereses.
- Guardar viajes como favoritos.
- Editar el perfil del usuario.
- Buscar y filtrar viajes por distintos criterios.
- Utilizar un chat en tiempo real dentro de cada viaje.
- Ver perfiles y calificaciones.
- Acceder a un historial de viajes realizados.

- Recibir notificaciones.

La aplicación está dirigida a personas mayores de 18 años interesadas en conocer nuevos destinos y compartir sus experiencias con otros viajeros. JetGo funcionará como una plataforma social de organización de viajes, similar en concepto a un "Tinder de viajes".

1.3 Definiciones, Acrónimos y Abreviaturas

- SRS: Software Requirements Specification
- UI: User Interface (Interfaz de Usuario)
- DB: Database (Base de datos)
- SPA: Single Page Application
- CRUD: Crear, Leer, Actualizar, Eliminar.
- WS: WebSocket.
- JWT: JSON Web Token.
- JetGo: Nombre del sistema a desarrollar
- Viaje: Propuesta publicada en la plataforma, que incluye destino, fechas, actividades, alojamiento y participantes
- Usuario: Persona registrada en el sistema, que puede crear o unirse a viajes
- Verificación: Proceso de validación de identidad mediante email, teléfono y DNI

1.4 Referencias

- IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications
- Plantilla oficial provista por el profesor
- Documentación del framework ReactJS y del backend Django
- Recomendaciones de buenas prácticas en diseño de sistemas web responsivos y seguros

1.5 Visión General

Este documento está organizado de la siguiente manera:

- Sección 1: Introducción al sistema JetGo, su propósito, alcance y contexto.
 - Sección 2: Descripción general del sistema, su funcionalidad principal, restricciones y suposiciones.
 - Sección 3: Detalle completo de los requisitos funcionales, no funcionales, de interfaz y de diseño.
 - Sección 4 (opcional): Modelos de análisis como diagramas de casos de uso, de clase o de base de datos.
 - Apéndices: Documentos de apoyo, ejemplos, prototipos o minutas relevantes.
2. Descripción General

2.1 Perspectiva del Producto

JetGo es una aplicación web independiente (no integrada con otro software externo) diseñada como una SPA utilizando ReactJS para el frontend y Django como backend. Se conectará a la base de datos de Supabase.

El sistema proporcionará una experiencia fluida y centralizada para organizar viajes sociales colaborativos.

2.2 Funcionalidad del Producto

Las funcionalidades clave que JetGo ofrecerá a sus usuarios incluyen:

- Registro de nuevos usuarios y autenticación segura.

- Creación de propuestas de viaje, con detalles completos.
- Búsqueda avanzada de viajes por distintos criterios.
- Solicitud de participación en viajes.
- Gestión de solicitudes (aceptar/rechazar).
- Sistema de calificaciones posterior al viaje.
- Edición de perfil de usuario.
- Sugerencias de viajes personalizadas.
- Guardado de viajes como favoritos.
- Recepción de notificaciones importantes.
- Acceso a la sección de ayuda/contacto.
- Pré Visualizar clima del lugar deseado.
- Chat en tiempo real.
- Calificación entre usuarios.
- Notificaciones.

2.3 Características de los Usuarios

Los usuarios del sistema serán personas mayores de edad, con conocimientos básicos en el uso de navegadores web. Se espera que puedan:

- Registrarse y completar un perfil.
- Gestion de amigos y solicitudes.
- Navegar por interfaces claras e intuitivas.
- Interactuar con formularios, botones y filtros.
- Leer información sobre viajes y enviar solicitudes.

- Crear, Unirse, modificar, filtrar viajes.
- Sistema de reviews y calificaciones.
- Gestionar miembros del viaje.
- Publicar, seguir, dar me gusta, guardar, posts y historias.
- Crear, unirse a chats.
- Enviar mensajes.
- Mandar ubicación en el mapa.
- Leer notificaciones.
- Visualizar el clima en la zona.

No se requieren conocimientos técnicos para utilizar la plataforma.

2.4 Restricciones Generales

- La interfaz se desarrollará en ReactJS.
- El backend se implementará en Django.
- Se utiliza supabase como Cliente de base de datos
- La base de datos será relacional (PostgreSQL).
- El sistema debe ser accesible desde navegadores actuales (Chrome, Firefox, Edge).
- El diseño debe ser responsive, adaptado a escritorio, inicialmente no para dispositivos móviles.
- Solo se permitirá el uso de la plataforma a usuarios verificados con email, teléfono y DNI.

2.5 Suposiciones y Dependencias

- Todos los usuarios deberán contar con conexión a Internet estable.
- El acceso a funciones sociales (chat, calificaciones, favoritos, anotarse a viajes, social) requerirá una cuenta verificada.
- Algunas funcionalidades dependen del uso de APIs externas (por ejemplo, para validación de identidad, mapas, etc.).
- No se aplicarán restricciones geográficas: los viajes pueden ser locales o internacionales.
- El sistema estará inicialmente disponible sólo en idioma español.

3. Requisitos Específicos

3.1.1 Autenticación y gestión de usuarios

- **RF01:** El sistema debe permitir registrar nuevos usuarios mediante email, nombre, apellido y contraseña o cuentas de google.
- **RF02:** El sistema debe validar que el mail no esté previamente registrado.
- **RF03:** El sistema debe enviar un correo de verificación.
- **RF04:** El usuario debe poder iniciar sesión mediante email y contraseña
- **RF05:** El sistema debe permitir recuperar contraseña vía email.
- **RF06:** El usuario debe poder editar su perfil (bio, foto, número de teléfono, redes sociales).
- **RF07:** El usuario debe poder ver perfiles de otros usuarios.
- **RF08:** El sistema debe mostrar la reputación/calificaciones del usuario

3.1.2 Interfaz de Hardware

JetGo está pensado como una aplicación web. No requiere hardware específico más allá de:

- *Un dispositivo con acceso a Internet.*
- *Navegador moderno compatible con HTML5, CSS3 y JavaScript.*

3.1.3 Interfaz de Software

JetGo interactuará con los siguientes módulos/software:

- Sistema de autenticación JWT para sesiones seguras.
- APIs RESTful para comunicación frontend-backend.
- Base de datos PostgreSQL.
- API de validación de identidad(APOSE).
- API de mapas (No determinada por el momento).

3.1.4 Interfaz de Comunicación

JetGo se comunicará usando protocolo HTTPS.

Las solicitudes entre el cliente y el servidor serán realizadas vía API REST.

Para el chat, se utilizará WebSocket para comunicación en tiempo real.

3.2 Requisitos Funcionales (Ampliado)

3.2.6 Editar o eliminar un viaje

Un usuario que ha creado un viaje podrá editar los detalles del mismo o eliminarlo antes de su inicio. Esto actualizará automáticamente la información visible a otros usuarios.

3.2.7 Editar perfil

El usuario podrá modificar su nombre, foto, descripción, intereses, email y contraseña desde su panel de configuración.

3.2.8 Chat entre usuarios de un viaje

Los miembros aceptados en un viaje podrán enviar mensajes a través de un chat en tiempo real. Solo estarán disponibles dentro de ese viaje específico.

3.2.9 Calificar usuarios después de un viaje

Finalizado un viaje, los usuarios podrán evaluar la experiencia y a los demás participantes, utilizando un sistema de estrellas y comentarios.

3.2.10 Notificaciones automáticas

El sistema notificará automáticamente a los usuarios sobre nuevas solicitudes, aceptación/rechazo, mensajes, viajes recomendados o calificaciones pendientes.

3.2.11 Sugerencia de viajes compatibles

El sistema recomendará viajes compatibles con los intereses, historial o ubicación del usuario mediante un algoritmo de filtrado colaborativo.

3.2.12 Marcar viaje como favorito

Los usuarios podrán guardar viajes de interés haciendo clic en un ícono de favorito. Estos se almacenarán en su perfil para futura referencia.

3.2.13 Verificación de identidad

JetGo requerirá que los usuarios verifiquen su identidad subiendo una foto del DNI y correo electrónico. La cuenta no estará habilitada hasta la verificación.

3.2.14 Ver sección de ayuda / contacto

Se proporcionará una sección con preguntas frecuentes, guía de uso y formulario para contacto directo con los administradores del sistema.

3.2.15 Ver sección social

Se proporcionará una sección dedicada al apartado social, donde el usuario podrá compartir sus experiencias mediante historias, posts, además de seguir las experiencias de sus amigos

3.3 Casos de Uso (Ampliado)

3.3.3 UC-03: Enviar mensaje en el chat del viaje

Actor: Usuario autenticado

Precondición: El usuario debe ser parte del viaje.

Flujo principal:

1. El usuario accede al chat del viaje.
2. Escribe un mensaje en el campo correspondiente.
3. Hace clic en "Enviar".
4. El sistema transmite y guarda el mensaje.

3.3.4 UC-04: Calificar a otro usuario

Actor: Usuario autenticado

Precondición: El viaje debe haber finalizado y ambos usuarios deben haber participado.

Flujo principal:

1. El usuario accede a la sección de calificaciones.
2. Selecciona al usuario a calificar.
3. Elige una puntuación y deja un comentario.
4. El sistema guarda y muestra la calificación.
5. El sistema guarda la información
Todos los datos del viaje, chat, perfiles, calificaciones y favoritos se almacenarán en una base de datos relacional segura y encriptada.

3.3.5 UC-05: Marcar viaje como favorito

Actor: Usuario autenticado

Flujo principal:

1. El usuario visualiza un viaje.
2. Hace clic en el ícono de favorito.
3. El sistema guarda la referencia en su perfil.

3.3.6 UC-06: Ver notificaciones

Actor: Usuario autenticado

Flujo principal:

1. El usuario accede al centro de notificaciones.
2. El sistema muestra un listado cronológico de alertas.

3.4 Clases / Objetos (Ampliado)

3.4.3 Calificación

Atributos: usuario_origen, usuario_destino, estrellas, comentario, fecha

3.4.4 Mensaje

Atributos: viaje, emisor, contenido, timestamp

3.4.5 Notificación

Atributos: usuario, tipo, mensaje, leída, timestamp

3.4.6 Solicitud

Atributos: viaje, usuario, estado (pendiente, aceptado, rechazado), fecha

3.4.7 Preferencia

Atributos: usuario, categoría, valor (por ejemplo: naturaleza, cultura, aventura)

3.5 Requisitos No Funcionales (Ampliado)

3.5.7 Usabilidad

La interfaz será clara, simple y accesible. Se priorizará una buena experiencia de usuario y navegación intuitiva.

3.5.8 Escalabilidad

El sistema debe poder adaptarse al crecimiento en la cantidad de usuarios y viajes sin pérdida de rendimiento.

3.6 Restricciones de Diseño (Ampliado)

- Uso exclusivo de tecnologías web.
- Debe soportar múltiples sesiones simultáneas.
- No se permitirá el registro sin verificación de identidad.
- El sistema deberá ser modular y mantenible.

3.7 Requisitos Lógicos de Base de Datos (Ampliado)

- Modelo relacional con claves foráneas y restricciones de integridad.
- Índices en campos de búsqueda frecuente.
- Encriptación de contraseñas y datos sensibles.

Backups automáticos semanales.

3.8 Otros Requisitos (Ampliado)

- El sistema deberá estar alojado en un VPS (Hostinger, etc).
- Tendrá soporte técnico básico vía correo electrónico.
- Se implementó una política de privacidad.
- Gestión de sesiones:
 - Tokens JWT con expiración corta (15-30 minutos)

- Refresh tokens seguros
- Invalidación automática de sesión tras inactividad
- Protección contra ataques comunes:
 - SQL Injection prevention
 - XSS (Cross-Site Scripting) protection
 - CSRF (Cross-Site Request Forgery) tokens
 - Rate limiting en APIs para prevenir DDoS
- Auditoría y logs: Sistema de logging para acciones críticas (cambios de perfil, transacciones, reportes)

3.9 Privacidad y Cumplimiento Legal

- GDPR compliance (si planean expandirse internacionalmente)
- Ley de Protección de Datos Personales Argentina (Ley 25.326)
- Política clara de retención de datos
- Derecho al olvido (eliminación de cuenta y datos)
- Consentimiento explícito para uso de datos sensibles

Política de privacidad visible y comprensible

4. Analysis Models

Los modelos de análisis permiten representar visualmente cómo los actores interactúan con el sistema para cumplir ciertos requerimientos funcionales. En este apartado se incluyen diagramas de secuencia, los cuales son útiles para entender el comportamiento del sistema en distintos escenarios. Cada modelo tiene trazabilidad con uno o más requerimientos definidos previamente.

4.1 Sequence Diagrams

4.1.1 Registro de Usuario

Introducción

Este diagrama representa el flujo de interacción entre un usuario y el sistema al momento de registrarse.

Descripción narrativa

1. El usuario accede al formulario de registro.
2. Ingresa sus datos (nombre, correo, contraseña, etc.).
3. El sistema valida los datos (por ejemplo, que el correo no esté en uso).
4. Si la validación es exitosa, guarda al nuevo usuario en la base de datos.
5. Devuelve una confirmación de registro exitoso.

Trazabilidad

Requisito funcional RF-001: “El sistema debe permitir el registro de nuevos usuarios”.

4.1.2 Inicio de Sesión

Introducción

Este diagrama muestra cómo un usuario puede ingresar al sistema utilizando sus credenciales.

Descripción narrativa

1. El usuario accede al formulario de inicio de sesión.
2. Introduce su correo y contraseña.
3. El sistema verifica las credenciales.
4. Si son correctas, se genera una sesión activa.
5. El usuario es redirigido a su panel principal.

Trazabilidad

Requisito funcional RF-002: “El sistema debe permitir a los usuarios autenticarse mediante correo electrónico y contraseña”.

4.1.3 Solicitud de Reporte

Introducción

Este diagrama describe el proceso mediante el cual un usuario solicita un reporte del sistema.

Descripción narrativa

1. El usuario selecciona el tipo de reporte en el panel.

2. El sistema consulta los datos necesarios en la base de datos.
3. Procesa y genera el archivo del reporte.
4. Devuelve el archivo o lo muestra en pantalla para el usuario.

Trazabilidad

Requisito funcional RF-010: “El sistema debe permitir generar reportes detallados de uso”.

A. Appendices

A continuación se presenta información complementaria útil para comprender mejor el desarrollo del sistema. Salvo que se indique lo contrario, los apéndices no forman parte del conjunto obligatorio de requisitos del sistema.

A.1 Appendix 1 – Acta de reunión con el cliente

Resumen

En la reunión inicial con el cliente, se definieron los siguientes aspectos clave:

- El objetivo principal del sistema es permitir a los usuarios registrarse, iniciar sesión y generar reportes personalizados.
- Se acordó que la plataforma será accesible desde dispositivos móviles y de escritorio.
- El cliente priorizó la seguridad en el manejo de datos personales y la facilidad de uso.

A.2 Appendix 2 – Documento conceptual inicial

Resumen

El cliente entregó un documento en el que se describía la visión general del proyecto, incluyendo:

- Público objetivo: usuarios administrativos de pequeñas y medianas empresas.
- Funcionalidades principales: registro, login, panel de control, generación de reportes.

- Restricciones iniciales: sistema multilenguaje y con interfaz intuitiva.