# Software Requirements Specification

##### Project Phase 2

##### TURBO Project

Document Version 1 Printing Date: 8/8/2023

Team

Rakan maher Joudeh Rand Dabbour

### Submitted to:

DR. Mohammed Nassar

### Software Engineering Amman Arab University

Revision History

|  |  |  |
| --- | --- | --- |
| Version | Date | Comments |
| 1.0 | 8/8/2023 | Initial version of  Software Requirements Specification |

# Table Of Contents

1-INTRODUCTION

* 1. Purpose ……………………………………………………………………1
  2. Project Scope……………………………………………………………1

2- Overall Description

2.1 Product Perspective …………………………………………………………3

2.2 Product Functions ……………………………………………………………..3

3- Specific Requirements

3.1.User Interfaces …………………………………………………………….6

3.2 Hardware Interfaces ……………………………………………………6

3.3 Software Interfaces ……………………………………………………6

3.4 Communications Interfaces ………………………………………….7

4. Software System Attributes

INTRODUCTION

The appeal of driving exotic supercars is a fantasy for many vehicle lovers in today's dynamic world of luxury and performance. The concept of a "Rent Supercars" platform arises as an interesting enterprise with the rapid growth of the sharing economy and the increasing demand for unique experiences. The goal of this project is to establish a comprehensive online platform that allows individuals to rent a varied choice of high-performance supercars, giving an exciting and unforgettable driving experience.

* 1. Purpose

The goal of the "Rent Supercars" project is to connect automobile enthusiasts with their aspirations of driving top-tier supercars. This platform will cater to individuals who want to experience the thrill of driving some of the most powerful and visually attractive vehicles ever made without committing to purchase. The idea promises to provide an unmatched experience that embodies elegance, speed, and refinement by allowing users to rent these vehicles for short periods of time.

The project also intends to raise awareness of the concept of sustainable luxury. Rather than emphasizing ownership, "Rent Supercars" promotes responsible sharing of high-performance vehicles, lowering the total environmental effect associated with mass manufacture and individual ownership.

* 1. Project Scope

The "Rent Supercars" project's scope includes the creation of a user-friendly and visually appealing web platform that facilitates the renting of supercars. The platform will have a wide range of features and functionalities to provide customers and administrators with a seamless and immersive experience.

Overall Description

2.1 Product Perspective

The "Rent Supercars" platform is intended to function within the framework of today's sharing economy, employing advanced technology to link supercar aficionados with their ideal driving experiences. It acts as a go-between for automobile owners and rentals, ensuring smooth transactions and outstanding experiences. The platform functions as a web-based application that is available from desktop and mobile devices, ensuring user convenience and accessibility.

The platform integrates with external systems for secure payment processing, email notifications, and user authentication. A backend database is also integrated to manage user profiles, car inventories, reservations, and administrative operations. The platform's user interface is visually appealing and easy, allowing customers to easily browse available supercars, choose rental dates, and finalize bookings.

2.2 Product Functions

1.User Registration and Authentication:

-Users can create accounts with unique credentials.

-Authentication ensures secure access to personalized features.

2.Car Catalog and Search:

-A comprehensive catalog of available supercars.

-Users can search, filter, and sort cars based on preferences.

3.Car Details and Imagery:

-Detailed information about each supercar, including specifications.

-High-quality images showcase the car's exterior and interior.

4.Reservation and Booking:

-Users can select desired rental dates from an availability calendar.

-Booking process includes payment, confirmation, and reservation details.

5.Pricing and Payment:

Transparent pricing for different supercar models and rental durations.

Secure payment processing for seamless transactions.

6.User Profiles and Preferences:

-Users can create profiles with personal and driving information.

-Preferences allow users to save favorite cars and view rental history.

7.Customer Support and Communication:

-Contact channels for customer inquiries and support.

-Email notifications for booking confirmation, reminders, and updates.

8.Admin Dashboard and Management:

-Administrative access to manage car inventory and user profiles.

-Overview of reservations, earnings, and platform performance.

9.Safety and Compliance:

-Ensures compliance with safety regulations and standards.

-Liability waivers and rental agreements are provided during booking.

Specific Requirements

3.1 User Interfaces

3.2Hardware Requirements:

Server Infrastructure:

High-performance servers to host the application and manage user requests.

Sufficient storage for user profiles, car inventory, and transaction data.

Database Server:

Powerful database server to handle data storage and retrieval efficiently.

Scalable architecture to accommodate growing user and car data.

Network Security Infrastructure:

Firewalls, intrusion detection systems, and encryption protocols to ensure data security.

Secure sockets layer (SSL) certificates for secure data transmission.

Web Hosting:

Reliable web hosting services with high uptime and low latency.

Fast content delivery networks (CDNs) to optimize user experience.

Mobile Devices and Computers:

Users require smartphones, tablets, or computers with modern web browsers.

Devices must support responsive web design for optimal user interface.

3.3 Software Requirements:

Web Application Framework:

Utilize a robust web application framework (e.g., Django, Ruby on Rails) for backend development.

Ensure compatibility with the chosen programming language and database system.

Database Management System:

Select a relational database management system (e.g., MySQL, PostgreSQL).

Ensure data integrity, scalability, and efficient data retrieval.

Frontend Technologies:

Use HTML, CSS, and JavaScript for creating responsive user interfaces.

Incorporate frontend libraries and frameworks (e.g., React, Angular) for enhanced interactivity.

Payment Processing Gateway:

Integrate a secure payment gateway (e.g., PayPal, Stripe) for processing transactions.

Ensure compliance with industry-standard security measures for payment data.

Authentication and Security:

Implement user authentication using industry-standard protocols (e.g., OAuth, JWT).

Apply encryption and hashing techniques to safeguard sensitive user data.

Email Notification Services:

Integrate email service providers (e.g., SendGrid, Amazon SES) for sending automated notifications.

Send booking confirmations, reminders, and updates to users' email addresses.

Content Management System (CMS):

Use a CMS to manage and update website content (e.g., WordPress).

Allow administrators to easily modify text, images, and other content.

3.4 Communication Interface Requirements:

User-Friendly Interface:

Provide an intuitive user interface for browsing cars, making reservations, and managing accounts.

Ensure consistency in design elements and navigation across all pages.

Responsive Design:

Design the platform to be responsive, adapting to various screen sizes and devices.

Provide optimal user experience on desktops, tablets, and smartphones.

Booking and Reservation System:

Allow users to select rental dates, view availability, and book supercars seamlessly.

Provide real-time feedback on car availability during the booking process.

Secure Payment Interface:

Integrate a secure payment gateway to handle transactions.

Ensure user data encryption during payment processing.

Contact and Support Channels:

Provide visible contact information for customer support inquiries.

Include a contact form or live chat for direct communication.

4. Software System Attributes

4.1 Reliability:

The "Rent Supercars" platform strives to give its users with a highly dependable service. Reliability is critical to ensuring that people can book and rent supercars with confidence without experiencing unexpected disruptions. To accomplish this, follow these steps:

-Implement extensive platform testing to discover and eradicate probable problems and errors.

-Use a reliable booking and reservation system that accurately represents vehicle availability and eliminates double bookings.

-Monitor the system's functioning on a regular basis to detect and address any issues as soon as possible.

-Provide users with explicit cancellation, refund, and booking change terms and conditions.

4.2 Availability:

High availability ensures that consumers may access the "Rent Supercars" platform whenever they need it. To increase availability:

To reduce downtime, host the platform on dependable and redundant servers.

Implement a load balancing method to disperse user requests and keep the server from becoming overloaded.

To optimize content delivery and reduce latency, use a content delivery network (CDN).

Regular maintenance should be performed during off-peak hours to minimize user impact.

4.3 Security:

Security is critical for protecting user data and ensuring a secure online environment. The "Rent Supercars" platform will have strict security procedures in place:

Encrypt data communication between users and the platform using HTTPS.

To prevent unwanted access, use strong authentication and authorisation procedures.

Update and patch software components on a regular basis to address known vulnerabilities.

To protect against cyber dangers, install a strong firewall and intrusion detection system.

Comply with data protection standards and obtain user permission before collecting and processing data.

4.4 Maintainability:

The platform's maintainability assures that it can be easily managed, updated, and extended over time. To improve maintainability:

Make use of clean and modular code standards to make future enhancements and bug fixes easier.

To track and manage code changes, use version control.

Provide thorough developer documentation, including API documentation for potential integrations.

Conduct regular code reviews to detect and address issues with code quality.

4.5 Portability:

The platform's portability allows it to be deployed and accessed across multiple contexts and devices. To increase portability:

Make the user interface responsive and adaptable to different screen widths.

Compatibility with major web browsers and operating systems should be ensured.

Containerization technologies such as Docker can be used to bundle the program and its dependencies.

4.6 Performance:

Performance is essential for providing a consistent user experience and quick reaction times. To improve performance:

To reduce data retrieval time, optimize database queries and apply caching methods.

Reduce website load times by using content compression and minification.

Monitor server and database performance and scale resources to manage traffic spikes as needed.

To distribute content and reduce latency, use content delivery networks (CDNs).

5.Database Requirements:

Choose a dependable and scalable database management system (DBMS) that satisfies the requirements of the project. Consider adopting relational databases such as MySQL, PostgreSQL, or other systems that are well-known for their dependability and performance.

5.1 Database Management System:

Choose a dependable and scalable database management system (DBMS) that satisfies the requirements of the project. Consider adopting relational databases such as MySQL, PostgreSQL, or other systems that are well-known for their dependability and performance.

5.2 Data Schema:

Create a well-structured database schema that reflects the system's numerous entities and relationships. This includes the following:

Cars: Save details on each available vehicle, such as its make, model, year, features, rental price, availability status, and photographs.

Users: Keep track of user information such as name, contact information, and login credentials.

Bookings: Keep track of user bookings, including the car booked, the booking dates, the user ID, and the status.

Transactions: Save information about financial transactions such as bookings and payments.

Create tables to hold user evaluations and ratings for cars and the overall service, if necessary.

5.5 Data Security and Privacy:

Implement stringent data security safeguards to safeguard sensitive user and transaction data. This includes the following:

Encryption: To prevent unauthorized access, encrypt important data such as user passwords and payment information.

Access Control: Create user roles and permissions to limit access to specific database tables and operations.

Anonymization of Data: If necessary, anonymize or pseudonymize user data to safeguard user privacy.