

Software Requirements Specification (SRS)

1. Introduction

1.1 Document Purpose

The "Locate a Socket" web application's functional and non-functional requirements receive detailed definition within this Software Requirements Specification (SRS) document. The document functions as a shared basis for all stages of software development starting from design through deployment to stakeholders and programmers and project administrators.

1.2 Product Scope

"Locate a Socket" serves as a location-based service for EV drivers to quickly discover accessible charging stations during their journeys. The application provides three main features consisting of station locator and both access information and secure payment functionality. Easy access to charging facilities serves as the purpose to improve the overall EV driving experience.

1.3 Document Overview

The document presents an overview of the system design while specifying functional needs and non-functional needs alongside system limitations and necessary prerequisites. This document consists of three sections which include:

- Section 2: Overall description
- Section 3: Specific requirements
- Section 4: Supporting information

1.4 Definitions, Acronyms, and Abbreviations

- **EV:** Electric Vehicle
- **API:** Application Programming Interface
- **GPS:** Global Positioning System
- **UI:** User Interface

2. Overall Description

2.1 Product Perspective

The application "Locate a Socket" connects to existing GPS mapping features and delivers current information about EV charging infrastructure. The system functions independently yet it supports connection to payment gateways together with mapping services.

2.2 Product Function

The functionalities are:

- Real-time location tracking of charging stations
- Detailed information about charging station features (e.g., types of chargers, availability, pricing)
- Secure payment processing
- User account management
- Route customisation based on charging station locations

2.3 User Characteristics

End User: Electric vehicle owners serve as end users of “Locate a Socket” because they need to find charging stations easily when they travel.

Administrative Users: Administrative Users maintain the charging stations database by ensuring proper accuracy and timeliness of the data.

Financial Transaction Handlers: They execute secure payment handling throughout the app to meet requirements from financial regulations and data protection standards.

2.4 Constraints

- The system needs to operate on iOS along with Android platforms.
- The solution needs to integrate with securely connected payment processing systems.
- Dependent on the availability and accuracy of GPS data.

2.5 Assumptions and Dependencies

- Assumes users have access to mobile devices with internet connectivity.
- Dependent on third-party services for mapping and payment processing.

3 Specific Requirements

3.1 External Interfaces

- **User Interface:** Mobile and web applications
- **Hardware Interfaces:** GPS-enabled devices
- **Software Interfaces:** Integration with Google Maps and payment gateways like PayPal and Stripe
- **Communication Interfaces:** RESTful APIs for third-party integrations

3.2 Functional Requirements

- Users should be able to establish secure accounts through the system and then access the platform through security protocols.

- Users will find on the system a map which shows live charging station positions.
- Users will have the ability to select charging stations through filtration methods using charger type along with cost and availability options.
- The system needs to generate step-by-step directions for the chosen charging station.
- Users can safely make payments via the system which will generate transaction receipts.

3.3 Non-Functional Requirements

- Performance degradation should not occur when the system grows to handle rising numbers of users and increasing data quantities. The system needs capabilities that enable both automatic growth and contraction of resources in response to the current demand.
- The structure of the system codebase needs proper documentation to enable efficient maintenance tasks and simple updates. The system enables modifications to charging station data along with payment systems to deploy quickly while preserving continuous service availability.
- Real-time monitoring with secure coding practices and regular security audits among multiple security measures need to be implemented by the application to respond to security threats.
- The application needs to initialize its load within 2 seconds while supporting simultaneous user interactions. Users must experience immediate responses when they interact with search and navigation features of the system.

4. Supporting Information

4.1 References

Pareek, Surbhi, et al. "Electric vehicle charging station challenges and opportunities: A future perspective." 2020 International Conference on Emerging Trends in Communication, Control and Computing (ICONC3). IEEE, 2020.

<https://ieeexplore.ieee.org/abstract/document/9117473>

Shah, Anushi, et al. "Maximizing service uptime of smartphone-based distributed real-time and embedded systems." 2011 14th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing. IEEE, 2011.

<https://ieeexplore.ieee.org/abstract/document/5753585>

Saifan, Ahmad A., and Areej Al-Rabadi. "Evaluating maintainability of android applications." 2017 8th International Conference on Information Technology (ICIT). IEEE, 2017.

<https://ieeexplore.ieee.org/abstract/document/8080052>

Sharma, Akash, et al. "Security of android banking mobile apps: Challenges and opportunities." International conference on cyber security, privacy and networking. Cham: Springer International Publishing, 2021.

https://link.springer.com/chapter/10.1007/978-3-031-22018-0_39