polimi-logo

PowerEnjoy

Requirements Analysis and Specifications Document

Version 1.0

Software Engineering 2 (A.A. 2016/2017)

- Simone Boglio (mat. 772263)
- Lorenzo Croce (mat. 807833)

Index

1. Introduction

1.1 Purpose

This document describes the RASD (Requirements Analysis and Specification Document) of the PoweEnjoy system. In the next pages we can find the description of the whole system; its components, the structure of the service, the interfaces with the old systems, requirement (functional and non) and the constraints. This document provides also some scenarios that represent some real situation and how these situations are managed. This document is useful for developers, managers, tester and you. It is also helpful to have a general view of the entire project.

1.2 Scope

PowerEnjoy is a digital management system for a car-sharing service thought to allow clients to search, reserve (and drive) a car using only a web service. In this way clients have user-friendly access to the car-sharing world in the city of Milan. Moreover PowerEnjoy provides only electric car and this is a good initiative to have an increasingly strong idea of a “green city”. As mentioned above, users can search and reserve a car using any devices with an internet access, like a pc or a smartphone. They can search a car using their current position (GPS localization) or submitting a specific address, the system will propose a list of the nearest cars and the clients will only have to choose which one reserve. Customer who reserve car have one hour to reach and start to drive it, otherwise the car will be set as available and a fee is accredited to the user. To access to service, clients must be registered into the system. When a client stops to use car, he will bring it in one of the specific areas and it will be set as available. The lock and unlock of the car is managed by the system, user haven’t any physical key. Others functionalities are explained in the rest of the document.

1.3 Identifying stakeholders

The main stakeholders for this project are the actual car-sharing company who want to increase number of clients and users of the service.

1.4 Identifying actors

The actors of the system are only the clients. After registrations, clients can search and reserve a car (and after drive it) using a mobile application.

1.5 Definitions, acronyms, abbreviations

1.5.1 Definitions

- Client: who want to access to service.
- User (or Customer): who use the service.
- Passengers: the user that reserved a car can bring with himself more travellers.
- Safe area: it's a specific area where cars are parked and can be parked by the user of the service.
- Special area: it's like a safe area but in this place there are power grid stations to recharge cars.
- Power grid station: it's look like a column where a car can be linked to it and in this way car's battery can be recharged.
- Ride: term used to indicate the utilization of a car by the user. The utilization starts when engine is power on and end when engine in power off end the car is located in a safe area.
- Current position: position identified by GPS coordinates (of the user or the car).
- Available: it's a status of the car. It means that the car can be chosen by someone for a ride (using the web app). If someone reserve an available car, this car's status is changed in "reserved".
- Reserved: it's a status of the car. It means that the car cannot be chosen by anyone because an user is still using it. When a reserved car is parked in a safe area and its engine is power off, the status change in "Available".

1.5.2 Acronyms

- RASD: Requirements Analysis and Specification Document
- API: Application Programming Interface.
- UI: User Interface.
- GPS: Global Position System.
- DBMS: Database Management System.
- RDBMS: Relational DBMS.
- DB: Database.
- HTTP: Hypertext Transfer Protocol.
- HTTPS: HTTP over SSL/HTTP Secure.
- OS: Operating System.
- JVM: Java Virtual Machine.

1.5.3 Abbreviations

- [Gn]: n(th) goal.
- [Rn]: n(th) functional requirement.
- [Dn]: n(th) domain assumption.

1.6 Goals

- [G1] The client shall be able to access the service through web service.
- [G2] The client shall be able to sign in and log in to the service.
- [G3] The user shall be able to manage his profile.
- [G4] The user shall be able to search cars in a specific zone.
- [G5] The user shall be able to reserve a car from a list up to one hour.
- [G6] The user shall be able to picks up and drives the reserved car.
- [G7] The user shall be able to know where are the safe area for parking the car.
- [G8] The user shall be able to know the current charges during the ride.
- [G9] The user shall be able to end the ride when he leaves the car.
- [G10] The system must incentivize the virtuous behaviours of the users.
- [G11] The system has to offer public APIs to enable the possibility to develop additional services on top of the basic ones.

1.7 Pre-existent company situation

Until now the electric-car company has a system where the client has to call a call centercommunicating his position, the operator search for the nearest available car and propose it to the client, if the client accepts the proposed car the operator reserve it for him using the internal information system. Cars are located in specific parking areas owned by the company. In each area there are some electric power stations for charge cars and a small office for an operator. The operator manages all the cars in his station. When client reach the indicated station, the operator checks the reservation and verify the client identity, after this give the key of the car to him. When client no need more the car he has to give it back in one of the specific parking areas and pay the ride to the operator. Also there are some operators entrusted to move the cars from one to another parking area when this are full or empty.

1.8 Reference Documents

This document refers to the project rules of the Software Engineering 2 project [***]and to the RASD assignment [***]. This document follows the IEEE Standard 830-1998 [*] for the format of Software Requirements specifications.

1.9 Overview

- 1: Introduction, it gives a brief description of the purpose, functionalities and goals of the application.
- 2: Overall Description, focuses more in-depth on features of the software, constraints and assumptions.
- 3: Specific Requirements, lists of requirements, typical scenarios and use cases, both with UML diagrams to provide an easy understanding at the several functionalities of the software.

2. Overall Description

2.1 Product perspective

2.1.1 User Interfaces

The clients can access to the service in two ways: web pages or mobile application. It is necessary to provide a common and uniform look and feel among the different hardware interfaces. All the interfaces shall be intuitive and user-friendly. They should not require the reading of long document or special skills or knowledge to be able to use the application.

2.1.2 Hardware Interfaces

The main hardware interface of the system consists in the access to the system of the car, in particular we need to check some transductors and interact with some actuators like:

- Transductors that checks status of the car engine.
- Transductors for every car seat that check if there is or not a seated person.
- Transductors that checks the status of the car doors (locked/unlocked).
- Actuator that enables the opening and closing of the car doors.
- GPS to know in every moment the position of the car. Also we need to access to the GPS position data in the user mobile application, device that can be able to respect these requirements it’s a smartphone.

2.1.3 Software interfaces

The mobile application must support Android and iOS (the most used mobile OS), for other devices (deprecated utilization) it is enough a web browser. The back-end store its data in a RDBMS and can run on every platform that support JVM. The back-end must offer programmatic interfaces (APIs) for user interfaces and external modules, like:

- Cars search
- Car reservation
- Online Payments
- Web interface

2.1.4 Communication interfaces

The Communication between users and system uses best known protocols. They are TCP, HTTP and

HTTPS. Those protocols must be support by the used devices.

2.2 Product function

The system allows user to search, reserve and use a car. Also system must be promoted the good behaviours of the user. This is a list of what the users of the service can do.
Client:

- Create an account

User:

- Login.
- Edit profile data.
- Delete account.
- Search cars in a determinate zone from his position or specific address.
- Reserve a car from a list.
- Ask to the system to unlock his reserved car when he is nearby.
- See where are safe areas where leave the car.
- Pay ride when he finishes to use the car.

2.3 User characteristics

We give for granted that users of the power-enjoy service have access to Internet. Passengers have to use the browser application or the mobile app. They move alone or with other people (passengers).

2.4 Actual information system

Actually already exist an information system where the company store all data, in particular we are interested to the database where are stored information about cars, payments and reservations. This database is used by:

- Administrator of the system that manage all the data about cars used by the company (like add, remove new car in the system).
- Call-center operators that search and reserve a car for client.
- Company that is responsible of car maintenance.
- Operators responsible of moving car in case of cars position imbalance in the city.
- Parking office operators that insert data about car reservation, client information and payments.

Also the cars used by the system are provided of GPS for check the current car position, it is used in some exceptional case (such someone try stole a car) by an external company for trace car on the order of system administrator.

2.5 Constraints

2.5.1 Regulatory policies & safety and security

All cars are provided of the whole necessary documents for traffic in the city: assurance, possession tax, revision and mechanic coupon. It’s user responsibility to ensure the traffic law and self-security (in addition to the safety of others). The system must guarantee user’s privacy both over profile data and rides. The system must guarantee to user the possibility to delete his profile. The mobile application requires only basic permission.

2.5.2 Hardware limitations

Every car must be provided of a GPS module to locate its position and an information system to communicate with the central. Every car must be provided of a control unit where are connected all the actuators and transductors. Every car must be provided of engine status sensor, doors state sensor, presence sensor for each seat. To access the service users must use device that can access to internet and with a GPS module. This device can be a smartphone.

2.5.3 Interfaces with other applications

PowerEnjoy requires to access the Internet and the Google Maps APIs to provide map visualizations and map-related services.

2.5.4 Interfaces with actual system

PowerEnjoy requires to interface with the database of the actual system where are registered all the data of the service and keep update all the relevant data since this system is used actually by different group of people.

2.5.5 Reliability of the services

The system must have a minimum availability of 99%.

2.5.6 Parallelism

The system will be able to support operations from different users at the same time without conflict.

2.6 Assumption and dependencies

- The user has a valid driving licence.
- The user have enough money at the end of the ride for pay the bill.
- The user can pay the fee in case of the cars isn’t picked up within one hour from the reservation.
- If a user switch off the engine and left car in a safe area, if he wants to catch again the car, he must start a new operation.
- If a user switch off the engine and left car in a no safe area the system continues to charge the user.
- The user drive car only in the city.
- The user will drive back the car into a safe area (doesn’t stole it).
- GPS position is always correct.
- If a car is set as “reserved” the system doesn’t propose it when user is searching for a car.
- The GPS installed on the car is always on.
- Each user can reserve only one car by one.
- Set of safe areas is pre-defined by the management system.
- Safe areas are equivalently distributed in the city.
- Only the user who reserved the car can drive it.
- Every available car has enough charge. *
- Special areas are at the same time safe areas.
- Available cars are always in a safe area.
- All cars are of the same model and have the same number of seats (1 driver, 4 passengers).
- Car maintenance is assigned to an external company.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Signin: This page represent the signin form. The client must provide his personal information.

Login: This page represent the login form. The user must provide his credentials to access to the service.

Research: This page represent the research form. The user can choose if whether to search the car by his current position or submitting a valid address.

Reservation: This page represent the reservation form. The user can choose one car from the list and reserve it.

Car screen: In each car there is a screen where some information are indicated. These information are referred to the battery level, the current charges of the ride and the positions of safe areas.

3.1.2 Hardware Interfaces

Every car need a screen where the user can see the current charge of the ride. No others is required, the client to access the service doesn’t need special hardware components it is enough a compatible smartphones (IOS and Android) or any supported browser on a computer.

3.1.3 Communication Interfaces

PowerEnjoy uses the TCP transport protocol and HTTP/HTTP over SSL application layer protocol to guarantee the top of the line security on the matter of transmission of private data. Also PowerEnjoy needs to able to access the control car unit through appropriate protocols for check state value and control the lock system of the car.

3.2 Functional Requirements

3.2.1 [G1] The client shall be able to access the service through web service.

1. [R1] The system must provide two different way to access to the service with same functionalities: web page and mobile app.

3.2.2 [G2] The client shall be able to sign in and log in to the service.

- [R1] The client must be able to submitted a valid email address, phone number and choose an available username, this fields are mandatory for successfully complete the registration.
- [R2] The system must be able to send back to the new user his password.
- [R3] The user can be able to log in into the service submitting his username and his password.
- [R4] The user must be able to submitted his credit card number in according with the privacy and security policies.
- [R5] The unregistered user can be see only sign in and log in pages.
- [R6] The system must implement a password retrieval mechanism.
- [D1] Phone number, email address and credit card data used for the registration must be valid.

3.2.3 [G3] The user shall be able to manage his profile.

- [R1] The logged user must be able to modify and delete his profile.
- [R2] The logged user must be able to edit his credit card number.

3.2.4 [G4] The user shall be able to search cars in a specific zone.

- [R1] The user must be able to search cars near his current position (using GPS coordinate).
- [R2] The user must be able to search cars near a specific address.
- [R3] The system must be able to verify that the submitted position is in the city.
- [R4] The system must be able to refuse request of searching car in position out of the city.

- [R5] The system must be able to provide a list of available cars in according with the position indicated by the user.
- [D1] There is always at least one available car thanks to the operators that manage the distribution of the cars in the city.
- [D2] The GPS coordinates of the car are always available and correct.
- [D3] All the cars have the same number of seats

3.2.5 [G5] The user shall be able to reserve a car from a list up to one hour.

- [R1] The user must be able to see the list of available car received as response to his request.
- [R2] The user must be able to select and reserve a car from the list for up to one hour.
- [R3] The user can reserve only one car by one.
- [R4] If after one hour of car reservation the user don’t pick up the car, reservation expired and the user pays a fee of 1 EUR.
- [D1] The user knows that the capacity of the car is 4 passengers.
- [D2] The user can’t delete a reservation.

3.2.6 [G6] The user shall be able to picks up and drives the reserved car.

- [R1] The user who reserved a car must be able to tell the system he is near the car.
- [R2] The system must be able to calculate the distance from the user and the car.
- [R3] If the distance between user and car is short, the system must be able to unlock the car.
- [R4] The system must be able to know if car’s engine is power on.
- [R5] The system must be able to notifies user of the current charges.
- [D1] The user doesn’t drive out of Milan more than 10km.
- [D2] The car doesn’t run out of battery during the ride.

3.2.7 [G7] The user shall be able to know where are the safe area for parking the car.

- [R1] The system allows to user to see the position of safe areas.
- [R2] The system allows to user to see the position of special areas.

3.2.8 [G8] The user shall be able to know the current charges during the ride.

- [R1] The system allows to user to see the current charges by a screen installed on the car.
- [R2] The system must be able to send to the car the current charges.

3.2.9 [G9] The user shall be able to end the ride and pay it when he leaves the car in safe area.

- [R1] The system must be able to know if the car’s engine is power of and the car is parked in a safe area.
- [R2] The system must be able to stop charging user if the car is parked in a safe area.
- [R3] The system must be able to notify the user the account of the ride.
- [R4] The user must be able to see the amount of the ride.

- [R5] The system must charge the total amount of the ride on the credit card user.
- [D1] The set of safe areas is pre-defined by the system.
- [D2] The user has enough money on his credit card.

3.2.10 [G10] The system must incentivize the virtuous behaviours of the users.

- [R1] The system must be able to know the number of passenger in the car.
- [R2] The system must be able to know the battery level of the car.
- [R3] The system must be able to know if the car is in a special area.
- [R4] The system must be able to detect the car's position.
- [R5] The system must be able to apply a discount of 10% on the last ride if it detects that there are at least two other passengers onto the car.
- [R6] The system must be able to apply a discount of 20% on the last ride if it detects that the car is left with no more than 50% of the battery empty.
- [R7] The system must be able to apply a discount of 30% on the last ride if it detects that the car is parked in a special area and it is plugged into a grid power.
- [R8] The system must be able to apply a charges of 30% more on the last ride if it detects that the car is left at more than 3Km from the nearest power grid station or with no more than 80% of the battery empty.
- [R9] Bonus for one ride are cumulative.
- [R10] The discount percentage can't exceed 100% of the total amount.

3.2.11 [G11] The system has to offer public APIs to enable the possibility to develop additional services on top of the basic ones.

- [R1] The system offers APIs to third party applications using web APIs as technology.
- [R2] The system replies using current industry standards.
- [D1] Access to APIs functionalities is provided only using the HTTPS protocol.

3.4 Scenarios

3.4.1 Scenario 1

Marco lives in Milan and he want to go to shopping but he doesn't want to take public transports. Marco is already registered to PowerEnjoy service and he has the mobile app installed on his Android smartphone. He decided to use it. He starts the application, log in to the system and start searching a car nearest him using his GPS position. In few seconds a map with the list of available car is showed on his smartphone and he must just select and reserve one of them. When he reaches his reserved car he takes his phone and using the PowerEnjoy app asks to unlock the car. The car is immediately unlocked and he can drive it. When the engine is on the system starts charging and Marco can see current charges on the screen installed in the auto. When Marco arrived to the centre, he searches a safe-area, parks the car and the system accredit the amount of the ride on the Marco's credit card.

3.4.2 Scenario 2

Davide lives in Milan and today he has to go to the post office for send a letter, after he wants to go to his friend Luca. He wants to use public transports to go to the post office and uses a car for reach Luca's house since it isn't reached by anyone public transports. So he decided to use PowerEnjoy service from the PC of his house to search cars by address in the nearest of post office. Immediately a list of cars in the nearby submitted address appears on the pc screen and Davide can choose which car reserve. In this way he has up to one hour to go to the post office and pick up the reserved car for reach his friend Luca.

3.4.3 Scenario 3

Andrea and his friends Lucia and Federico want join a party tonight but they don't have a car and it's too late for take the metro, also they are young and don't want spend a lot of money for take a taxi, so they decide to use PowerEnjoy. After registration and reservation of the nearest car Federico and his friends pick up the car and since they want save some money they decide to find the nearest special area to the party location, through the map on the screen of car, for parking and plug the car into the power grid station. In this way Federico get two bonus: one for bring at least others 2 passengers and one for let the car in the special area plugged on the power station. Federico and his friends save some money and now they can happily join the party in few minutes.

3.4.6 Scenario 4

Luigi arrives this morning in Milan by a train for join a conference of his company, he doesn't know which public transport take since the location it is far more than one hour by foot, so he decides sadly to walking to the destination, after few metres he see one of the car of PowerEnjoy service and reads on one of its side “PowerEnjoy, green car-sharing for You, download our app and start to drive”, intrigued by the message he searches and downloads the app on his Android smartphone, since he isn’t already registered, he insert his data (name, email, number of phone, number of drive license and credit card) and after few second he get back the password for join the system. Through his GPS position the system proposes him quickly the car in front of him, in one-click he reserve the car and the system confirm the prenotation, at this point he click the button for signal he is nearest the car and the system unlock the doors of the car and he finally can drive to the conference destination. Luigi is positively surprised that it took less than 5 minutes for do all the things. After 15 minutes he arrives to the destination and thanks to the map on the screen he finds the nearest safe area where leaves the car, he turns off engine and exit the car and receive the bill automatically on his banking account. Luigi easily arrived 40 minute before and in this time he can relax and have a coffee, he will use again PowerEnjoy service for sure.

3.5

The client shall be able to sign in to the service Actor Client Goal [G2] Entry conditions There isn’t any entry condition Flow of events - The client goes to the home of the service (using the app or the website).

- The client chooses “Sign in” option.
- The client compile requested fields.
- The client clicks on the button “Signin”.
- The system saves information on a DB and send back to the client his password. Exit condition The client became a PowerEnjoy user and he can start to use the service. Exceptions - The submitted e-mail address is already registered on the service.- The submitted username is already used by someone.- Some fields are not compiled

The client shall be able to log in to the service Actor Client Goal [G2] Entry conditions The client has already a valid account Flow of events - The client goes to the home of the service (using the app or the website).

- The client chooses “Log in” option.
- The client inserts username and password.
- The client clicks on the button “Login”.
- The system check if the credentials are correct. Exit condition The user is redirected to the search page. Exceptions - The credentials are incorrect.

The user shall be able to manage his profile Actor: user Goal: [G3] Entry condition: the user is already logged into the system. Flow of events:

- The user selects “Profile” option in the homepage.
- The user modifies information (like credit card number or telephone number).
- The user clicks “Save” button.
- The system updates new data. Exit condition: the system show the update message and redirect

user to the home. Exception:

- The user submits a blank field.

The user shall be able to search cars in a specific zone Actor: user Goal: [G4] Entry condition: the user is already logged into the system. Flow of events:

- The user selects “search” option on the homepage.
- The user chooses kind of research: by current position or by address.
- If user chose to search by address he submits the address information.
- The user clicks “Search” button. Exit condition: the system shows on the user’s smartphone a list of available cars. Exception:
- If user chose to search by address and he doesn’t submit any address, system return an error.

The user shall be able to reserve a car from a list up to one hour Actor: user Goal: [G5] Entry condition: the user has already sent a research request. Flow of events:

- The user selects what car he prefers to reserve.
- The user confirms his choice clicking “reserve” button. Exit condition: the system show the reservation message and redirect user to the home. Exception:
- If another user reserves the same car first that the user completes the operation, system shows a message and asks for a new selection.

The user shall be able to picks up and drives the reserved car Actor: user Goal: [G6] Entry condition: the user is logged to the system and has already sent a reservation request. Flow of events:

- When the user is in the car’s neighbourhood, it appears on his smartphone a notification and the he clicks on “Unlock” button.
- The system unlocks the user’s reserved car.
- The user opens the car and drive it. Exit condition: the user can drive his reserved car. Exception: there isn’t any exception.

The user shall be able to know where are the safe area for parking the car Actor: user Goal: [G7] Entry condition: the user is already driving the car. Flow of events:

- The user can see where the safe (and special) areas are clicking on the “Safe area” option on the screen’s car. Exit condition: a map of the city appear on the screen and in it are represented the safe and special areas. Exception: there isn’t any exception.

The user shall be able to know the current charges during the ride Actor: user Goal: [G8] Entry condition: the user is already driving the car. Flow of events:

- The user can see the current charges clicking on “Ride information” option on the screen’s car. Exit condition: on the car’s screen appears information about charges and battery status. Exception: there isn’t any exception.

The user shall be able to end the ride and pay it when he leaves the car in safe area Actor: user Goal: [G9] Entry condition: the user is already driving the car. Flow of events:

- The user drive the car until a safe or special area.
- The user, if he is in a special area can plug the car into the power grid for take a discount.
- The user power off the engine.
- The user left the car.
- The system checks that the car is parked in a safe area, engine is power off and there isn’t anyone on the car and then locks it.
- The system accredits the ride (applying discount if it is required) and notifies the user. Exit condition: the car is locked and its status is changed to available. Exception: there isn’t exception.

Performance Requirements PowerEnjoy is a system that want to help the largest number of people to move into Milan. For this scope the system must be able to support a large number of requests (many of these simultaneous) and it must be able to responds to any request in few seconds. So the server must be correctly chosen to guarantee the most possibly efficiency. It must be scalable.

Software System Attributes Reliability The system must guarantee that if an user make a reservation, his reservation is available within the deadline. Availability The system must guarantee that the service is available 24/7 Usability The user interface must be user-friendly to guarantee a very quick access to the service in both cases; access by app and access by web page. Security Since the user must insert his credit card number at the registration moment, the system must guarantee that this information is hidden from anyone and the only permit use for it is to pay the ride. The system must guarantee too the privacy of all registered users.