Requirement Analysis and Specification Document for PowerEnJoy

Enrico Migliorini, Alessandro Paglialonga, Simone Perriello

November 10, 2016

Contents

1 Introduction

1.1 Purpose

This document presents the requirements of the *PowerEnJoy* system, aimed at powering a car-sharing service. Said requirements will be presented using both natural language and diagrams.

1.2 Intended Audience

This document is addressed to all the stakeholders in the *PowerEnJoy* project. This includes, but is not limited to, the CEO, the end users, the development committee, product designers and engineers, quality assurance and marketing.

1.3 Product Scope

The *PowerEnJoy* is a partially automated electric car-sharing service. The system keeps track of users and Cars, addresses users to available cars, locks cars when not in use, and charges the users for use and abuse of the Cars. The system also needs to keep track of the battery level of the Cars and dispatch personnel to connect low-on-battery Cars to the power grid.

It is important to notice that this document only describes the requirements for the software dealing with the cars, not the hardware on which the software will run or the management system.

1.4 Definitions, Acronyms and Abbreviations

1.4.1 Business terms glossary

[OBSOLETE: REFER TO GLOSSARY.TXT, THIS WILL BE UPDATED LATER]

1.4.1.1 Car-sharing A *Car-sharing* service allows *Users* to rent *Cars* for a limited amount of time, being charged a *Fee* according to time and possibly applying a *Discount* or an *Increase.*

1.4.1.2 Database A structure that holds informations linked logically according to relationships. For instance, a *Database* could hold records of every registered *User*, every available *Car* and every time a *User* rented a *Car*.

1.4.1.3 Discount A reduction in the *Fee* to be paid because of good behaviour on the part of the *User*, e.g. leaving the *Car* plugged or bringing it back with a mostly-full battery. The actions that constitute good behaviour are determined ad detailed further in the document. [ADD A REFERENCE WHEN YOU MAKE THE SECTION]

1.4.1.4 Fee The amount of money that the *User* will be charged for his usage of the *Car-sharing* service.

1.4.1.5 Increase An increase in the *Fee* to be paid because of improper behavious on the part of the *User*, e.g. bringing the *Car* back with a mostly-empty battery.

1.4.1.6 Management System An external system that allows administrative access to the internal *Database*.

1.4.1.7 System The automated software structure this document is about. It tracks *Users* and *Cars* and deals with all the details needed for *Car-sharing*, from GPS mapping to charging *Users* with *Fees*.

1.4.1.8 User A person registered on the *System*, who will use the *Cars* for a *Fee*.

1.4.1.9 Car An electric car owned by the *Car-sharing* service, rented to the *User* and tracked by the *System*.

1.4.2 Document specific terms

1.4.2.1 Alloy A descriptive language that allows to describe a set of structures through constraints.

1.4.2.2 DBMS

Data Base Management System. A software interface allowing to interact with the *Database*.

1.4.2.3 RASD

Requirements Analysis and Specification Document. This document, describing the *System* to be developed.

1.4.2.4 UC

Use Case. A description of interaction between *Users* and *System*.

1.4.2.5 UML

Unified Modeling Language. A language for modeling Object-Oriented software systems.

2 General Description

2.1 General Description

The PowerEnjoy system is created in order to provide a unified platform to be used for reserving and renting electrical Cars.

Access to the full System is reserved to the registered Users, although it is also possible for a Visitor to register themselves, becoming a User.

Once registered, a User will have easy access to the functionalities of PowerEnJoy, allowing them to see the locations of nearby Cars, reserve and rent them and enable money-saving options to become eligible for Discounts.

2.2 Product Perspective

The main System will be housed on a central elaborator, called server. This server will be responsible for all necessary operations, including communicating with the internal Database, keeping track of all the Cars and interacting with the Banking System in order to charge Fees to the Users. The user will access the System through a mobile app (client), connected to the main System via Internet, and interacting with the Smartphone's GPS System. Information exchange between the client and the server will be based on the HTTP protocol.

2.3 Product Functions

Here are explained the main needs of the PowerEnJow system.

Need	**Priority**	**Function**
Registration	?	Allows a Visitor to register their credentials, becoming a User
Login	?	Lets a User access the system
Find Nearby Cars	?	Shows Available Cars in a location
Reservation	?	Allows a user to set a nearby Car as Reserved
Proximity Unlock	?	Unlocks the car when the User who Reserved it is close

Need	Priority	Function
Computation	?	Computes the Fee to be paid, including Discounts and Increases
Payment	?	Charges the user for the computed Fee
Detect Parking	?	Detects whether a Car has been left in a Parking Area
Detect Charging	?	Detects whether a Car is Plugged to a Charging Area
Administrative Access	?	Allows an administrator to modify the information in the Database or parametres of the System

2.4 User Classes and Characteristics

Name	Description	Responsibility
Visitor	A person who needs to undergo Registration to become a User	Performs a Registration
User	Someone who is registered in the system and can access its functionalities	Can find, reserve and rent Cars
Administrator	A specialized worker for the Company	Can modify the Database and the application's various parameters

2.5 Operating Environment

Users and Visitors will access the System through a smartphone application. The smartphones used to access the application will need to have a sufficiently advanced hardware in order to run it, and the application will require both an Internet connection and a GPS signal. The mobile application will, however, only be required in order to run the Find Nearby Cars, Reservation and Proximity Unlock functions.

The Cars will also need to be modified by adding a device capable of getting GPS signal and remotely communicate with the central System.

The central System will need a suitably powerful elaborator to run on.

2.6 Design and Implementation Constraints

The system will employ the HTTP protocol in order to have the client software, both the software application and the one installed on the Cars, communicate with the central System. As mentioned above, a network connection will be necessary. In order to maximize availability, it is advised to use the best available hardware.

User ID and password will be required in order to access the System. In order to register as a User, a Visitor needs to undergo Registration.

2.7 Assumptions and Dependencies

[INTEGRATE FROM THE LIST]

3 External Interface Requirements

The interfaces the system should interact with.

3.1 User Interface

The User Interface is provided by the client smartphone application, and allows the user to perform all of their actions.

3.2 Hardware Interface

The client application must be developed so as to have access to both the phone's network connection and its GPS locator.

The central server must be provided with one or more sufficiently advanced computers that may run the server-side application, and allow it access to a high-speed network connection.

The Cars will be fitted with the device mentioned in 2.5, allowing access to various informations, such as battery status, location, and eventual presence of Passengers.

3.3 Software Interface

The System does not need to interface with external software. Database management can be performed via the Administrative Access function.

3.4 Communication Interface

As mentioned above, the System heavily uses Internet communications protocols, mainly the HTTP protocol. HTTP requests to and from the server will be mostly carried by mobile network connections.

4 Functional Requirements

The functionality for the various users.

4.1 Use cases specification

[12pt]article [utf8]inputenc longtable caption

—p0.2 p0.8— labelformat=empty **Register Account**

ID UC1

Description Visitor wants to create an account for the Car-Sharing Service.

Actors Visitor.

Description Visitor wants to create an account for the Car-Sharing Service.

Pre-Conditions The Visitor connects to the Company's Car Sharing Web Site/ Application.

Flow of events

1. The Visitor selects the function *"Sign Up"*.
2. The System returns a form to enter all the required data: Name, Surname, Birth date, Social Security Number, ID Card Number, Driving License number and Credit Card number. It also asks for an Email address and a password which will be used for the future logins.
3. The Visitor fill the forms with all the required information.
4. The System stores the request together with all the data provided with it, generates a random activation URL and asks the Mail System to forward this/her URL to the email address of the Visitor.

Post Conditions The Mail System sends the activation URL to the Visitor's email provided in the registration form.

Exceptions

- The System recognizes invalid or missing data in the form compiled by the Visitor and informs him/her of the error. The flow of events restarts from point 1.
- The Visitor inserts in the form a Social Security Number, or ID Card Number, or Driving License number, or Email Address, which is already present in the System. The System shows an error message saying that some of those credentials were already been inserted into the System for another account. The flow of events restarts from point 1.

USE CASE : Activate Account.

Id: UC2 Description The Visitor wants to activate his/her/her account.

Actors Visitor.

Pre-Conditions The Visitor has received the activation URL on his/her mail box.

Flow of events

1. The Visitor clicks on the received activation URL. 2. The System acknowledges that the Visitor has arrived in his/her activation Web Page and activates his/her account.

Post-Conditions The Visitor is now become the User which can access the System using the credentials (Email, password) he provided during the registration phase. Exceptions : The Activation URL expires after 10 days it has been generated. The Visitor?s data are cancelled from the System and the Visitor will have to perform the Registration (UC1) again.

USE CASE : Log in

Id: UC3

Description The User wants to log in the System.

Actors User.

Pre-Conditions The User connects to the Company?s Car Sharing Web Site/Application .

Flow of events 1. The User selects the function ?Login?. 2. The System shows the User a login form, asking him/her to insert his/her account Email and password. 3. The User inserts the pair (Email, Password) used during the registration phase and selects the function ?Log me in?.

Post-Condition The System verifies the existence of an account associated with that pair (Email, Password) and logs the User in. The User has now access to the System User?s dedicated functionalities. Exceptions : 4 The System doesn?t find an account associated with that pair (Email, Password) and shows an error message, the flow of events starts from point 1.

USE CASE : Log out

Id: UC4 Description The User which is logged in the System wants to log out.

Actors User

Pre-Conditions The User is logged in the System.

Flow of events

1. The User selects the function ?log out?. 2. The System performs the User?s logout. Post-Condition The System shows the confirmation of the logout to the User. The User is now not able to use the System functionalities dedicated to Users anymore (until he logs in again).

USE CASE : Locate available cars. Id: UC5

Description The User wants to locate a Car.

Actors User.

Pre-Condition The User must be logged into the System.

Flow of events

1 The User selects the function ?Locate Cars?. 2 The System shows a text box asking the User to provide an address near which he would like to see the Cars whose state is Available. 3 The User inserts the desired address and selects the ?Locate? function.

Post-Condition The System shows the User a map containing all the Cars, whose state is Available, which are inside a 5KM distance range from the provided address or User?s GPS position. Alternative flow of events 3a The User selects the function ?Near me? instead of inserting a specific address and sends his/her GPS Coordinates to the System.

Exceptions :

4b The System does not find the inserted address and informs the User. The Flow of Events starts from point 1. 4c There are no available Cars in the specified address/User?s Position. The System informs the User. The Flow of Events start from point 1.

USE CASE : Reserve available car.

Id: UC6

Description The User wants to reserve a Car.

Actors User

Pre-Condition The User must be logged into the System and the System must have found cars when the User activated the ?Locate available cars? function.

Flow of events 1. The User chooses a specific Car between the showed ones in the map. 2. User selects the function ?Reserve this Car?.

Post-Condition The System stores the Reservation of the Car, changing the Car status in Reserved. The System activates a countdown of 1 hour during which the User will have the possibility to unlock the reserved Car.

USE CASE : Unlock Car. ID : UC9

DESCRIPTION : The User wants to ask the System to open the doors of the Car in order to enter it. PARTECIPATING ACTORS : User PRE-CONDITION : The User must be logged in the System and must have reserved a car. FLOW OF EVENTS : 1 The User activates the function ?Unlock Car?. 2 The User sends his/her GPS coordinates to the System; 3 The System checks that the GPS coordinates of the specific User ?s Reserved Car and the GPS coordinates of the User himself are in a 15 meters distance range. EXCEPTIONS : If one hour has passed since the reservation has been done and the User didn?t unlock the Car because he wasn?t in the 15 meters distance range or didn?t activate this function : 1. the reservation expires and the User cannot unlock the car anymore (unless with another reservation). 2. The System changes the Car status to Available. 3. The System communicates to the Banking System the amount of money (corresponding to the fee of 1 EUR) to charge to the User. 4. The System now allows the User to perform another reservation. POST-CONDITION : The System has verified that User is nearby the car (inside the specified distance range) and unlocks the Car doors. The System now changes the Car status to In Use and sets the Plugged Field False. The User enters the Car.

USE CASE : Drive Car. ID : UC10 DESCRIPTION : The User starts to drive the reserved Car. PARTECIPATING ACTORS : User PRE-CONDITION : The User has unlocked the doors of the car and entered the Car. FLOW OF EVENTS : 1 The User starts the engine of the Car. 2 The System starts the Ride Timer which indicates the time usage of the Car. 3 [Extension Point UC11]. 4 [Extension Point UC14]. 5 The System calculates the current fee charged to the User (calculated as a given amount of money per minute multiplied by the minutes of the Ride Timer) while showing it on the on-board screen. POST-CONDITION : The User drives the Car.

USE CASE : Drive with Passengers. ¡¡extends UC10¿¿ ID : UC 11 DESCRIPTION : The User picks up passengers to share the ride with. PARTECIPATING ACTORS : User PRE-CONDITION : The User is driving his/her reserved Car.

FLOW OF EVENTS : 1. The User picks up the desired passengers. 2. The Car detects the number of passengers.

POST-CONDITION : The User is sharing the ride with his/her passengers. The System stores the number of passengers in the ride and if they stayed in the Car for at least 3 minutes.

USE CASE : End ride. Id : UC12 DESCRIPTION : The User ends the ride and the System processes the fee. PARTECIPATING ACTORS : User PRE-CONDITION : The User parks the Car in one of the Parking Areas. FLOW OF EVENTS : 1. The User exits the Car. 2. The System verifies that no one is in the car. 3. The System checks the battery status. 4. The System checks, using the GPS coordinates, if the User has left the Car within a 3KM distance range from the nearest Charging Area. 5. The System checks if the User drove with passengers (UC11). 6. [Extension Point UC13]. POST-CONDITION : The System locks the doors of the Car and sets it as Available. The System communicates to the Banking System the final fee to charge to the User. ALTERNATIVE FLOW OF EVENTS: 7a The battery status is higher than 507b The User did plug the Car (UC13), the battery status is higher than or equal to 207c The User did plug the Car (UC13), the battery status is lower than 207d The User didn?t plug the Car (UC13), the battery status is higher than 507e The battery status is between 207f The battery status is lower than 207g The battery status is between 207h The battery status is lower than 207j The battery status is higher than 507k The battery status is between 20

EXCEPTIONS : The Ride ends and the Car stops moving when the battery status reaches 0

USE CASE : Plug the Car. (¡¡extends UC12¿¿) ID : UC13 PARTECIPATING ACTORS : User PRE-CONDITION : The User has parked the Car in one of the Charging Areas designated by the System. FLOW OF EVENTS : 1. The User plugs the Car into the socket of the Charging Area. 2. The System detects that the Car is plugged within 2 minutes since the User got off the Car. POST-CONDITION : The battery of the Car is charging and the System stores the User?s action for possible discounts. The System sets the Car Plugged Field True. USE CASE : Enable Money Saving Option. (¡¡Extends UC10¿¿) ID : UC14 PARTECIPATING ACTORS : User PRE-CONDITION : The User enables the Money Saving Option. FLOW OF EVENTS : 1. The System asks the User the address of his final destination showing a text box where to insert the address. 2. The User provides the address to the System.

POST-CONDITION : The System computes an algorithm which takes in consideration the distribution of the cars in the city, the final destination of the User and the availability of power plugs in the Charging Area. The result of this algorithm will be sent to the User providing him the address of the Charging Area where to leave the Car. (The User will still have to plug the Car in order to have a discount!)

EXCEPTIONS : If the Socket of the Charging Area has no more available plugs , the System informs the User and the Flow of Events starts from point 1.