POLITECNICO
MILANO 1863

Politecnico di Milano
A.A. 2016 – 2017
Software Engineering 2: “PowerEnJoy”
Requirements Analysis and Specification Document

Marco Festa

November 19, 2016

Contents

1 Introduction

1.1 Purpose

The goal of the Requirement Analysis and Specification Document (RASD) is to define and entirely describe all of the system's functionalities and requirements. Specifications are set according to the customer's needs in order to create an accurate model for the system and simulate the typical use cases and scenarios occurring after development.

1.2 Scope

The aim of the project is to develop a digital management system for a car-sharing service that exclusively employs electric cars. All registered users must be verified and provide valid driving license in order to get access to the system. Once they get approved they may use all of the most common features available for every existing car sharing service. GPS and other positioning systems are crucial to provide the customer with the best experience possible, ensuring accurate car localization and position related features. The eco-friendly attitude of the company focuses on fuel-efficiency and smart-transportation topics imposing the system to capture specific customer behaviors and eventually encourage them through bonuses or lowered rates. The iteration between the customer and the system is brought out through 3 different platforms, the web interface, the smart phone app, and the on-board display. These applications are in continuous communication with the centralized system which keeps track of all sensitive information.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

- *User (or Registered User):* a person registered to the system. The driving license has been verified and the customer info is correctly added into the database. Registered users are the only entities eligible for car renting.

- *Driver:* the user who physically unlocks an electric car using his own credentials and starts driving it becomes automatically the driver. Registered users are actually the only possible persons to potentially become drivers.
- *Guest:* a person that is not necessarily registered to the system.
- *Safe area:* geographical area where cars are authorized to be parked giving the user the chance to end the renting.
- *Power Grid Station:* inside the safe areas are located power station where electric cars can be plugged in and have the central battery recharged.
- *Free Car:* a car which is not being used by any registered user and is not under any pending reservation is considered available or free.
- *Reserved Car:* each registered user has the ability to choose a free car from the smartphone app or web interface and have it reserved for at most one hour. During this phase the car disappears from the list of free cars and can be only opened and unlocked by the user who made the reservation. The reservation state of a car ends either when the car is unlocked by the user who automatically becomes the driver, the one-hour limit is reached, or the state is ended by the user itself.
- *Opening procedure:* a user can open a free car, or one he had reserved, by using the dedicated feature on his smartphone app. An opened and locked car has to be considered a reserved car.
- *Unlocking procedure:* car unlocking is achieved via a PIN code entered through the touch screen display inside the vehicle.
- *PIN code (or PIN):* a 4-digit secret code chosen by the user during registration phase.
- *In-use Car:* a car that has been unlocked by a registered user and is now able to be turned on by the driver.
- *CAN bus:* the physical network connecting each present electronic device: sensors, micro-controllers, actuators and instruments.

1.3.2 Acronyms

- **RASD**: Requirement Analysis and Specification Document.
- **DB**: Database
- **PGS**: Power Grid Station
- **CAN bus**: Controller Area Network bus

1.3.3 Abbreviations

- **[Gx]**: Goal
- **[RE.x]**: Functional Requirement
- **[UC.x]**: Use Case

1.4 Reference Documents

- ISO/IEC/IEEE Std. 29148:2011, “Systems and software engineering - Life cycle processes – Requirements engineering”
- Specification document: “Assignments AA 2016-2017.pdf”

1.5 Document Overview

Document structure:

- **Section 1 – Introduction**: presentation of the document and product.
- **Section 2 – Overall Description**: specifies the background of assumptions and constraints necessary to describe the software product.
- **Section 3 – Specific Requirements**: this section gets deeper into product requirements. Lists a variety of possible scenarios and implemented features. Use cases, UML diagrams and eventually mockups are shown to give a clearer vision of the final product aspect and functionality.
- **Section 4 – Appendix**: in this section is shown an alloy model generated on the application domain, various tools used to write the document and the work hours of all the document relators.

2 Overall Perspective

2.1 Product Perspective

2.1.1 Integration with external systems

Being the system practically a standalone implementation, integrations with external resources are few. Basic API interfaces are used to handle driving license verification and payment processing. All of the necessary sensors and instruments are already configured and linked to the CAN bus of the electric cars ready to be handled and interpreted by our custom software installed on the onboard device.

2.1.2 Domain model

State chart and class diagram are shown below. The car state chart explains in detail the various possible car states and the transitions between them

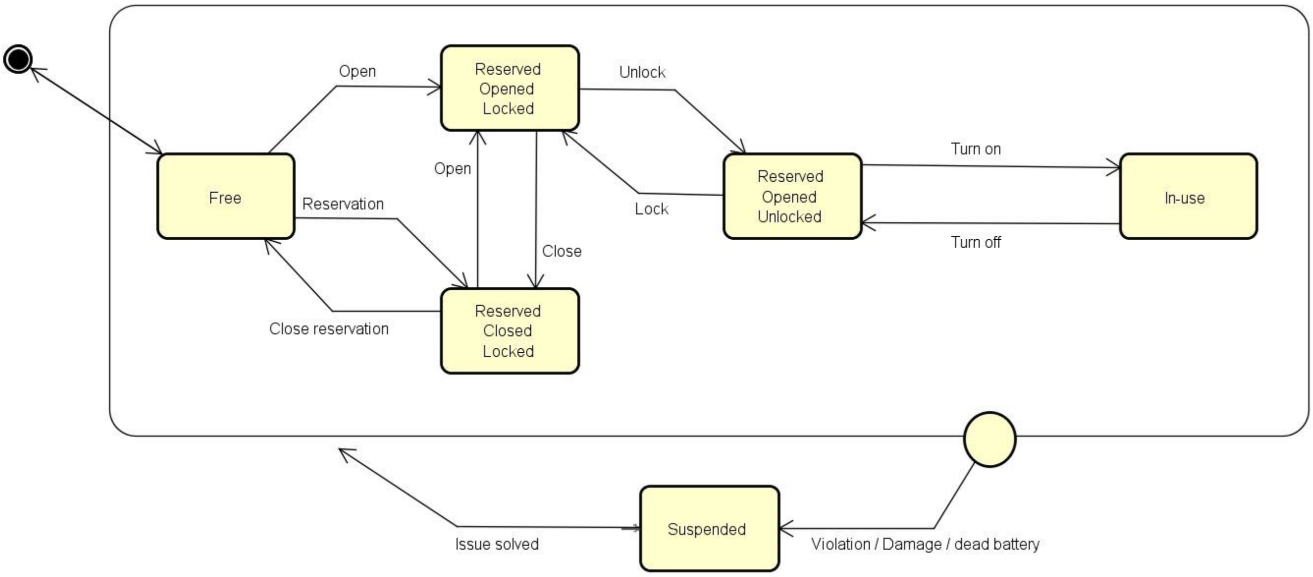

Figure 1: Car State Chart

The class diagram shown below displays the class structure chosen to manage all sensitive information for the system in order to make it reliable and efficient.

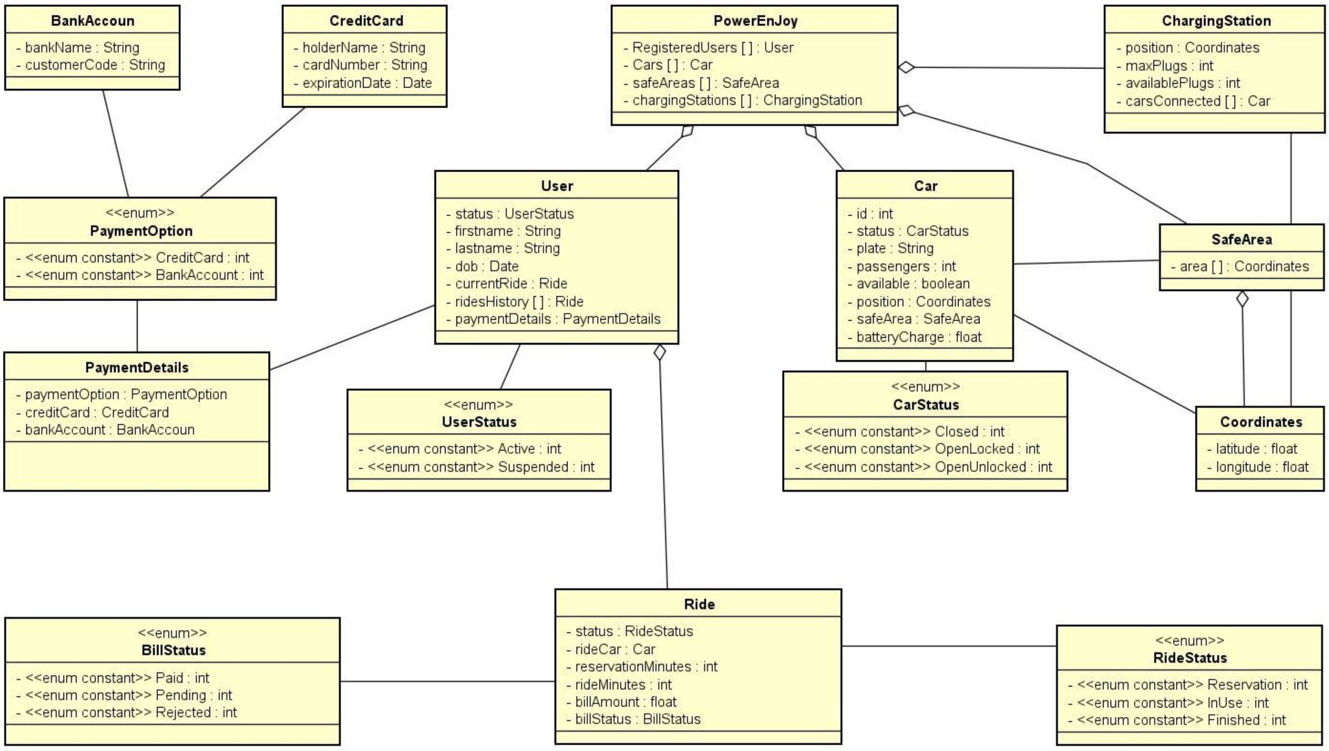
BankAccoun
- bankName : String
- customerCode : String
CreditCard
- holderName : String
- cardNumber : String
- expirationDate : Date
PowerEnJoy
- RegisteredUsers [ ] : User
- Cars [ ] : Car
- safeAreas [ ] : SafeArea
- chargingStations [ ] : ChargingStation
ChargingStation
- position : Coordinates
- maxPlugs : int
- availablePlugs : int
- carsConnected [ ] : Car
<<enum>>
PaymentOption
- <<enum constant>> CreditCard : int
- <<enum constant>> BankAccount : int
User
- status : UserStatus
- firstname : String
- lastname : String
- dob : Date
- currentRide : Ride
- ridesHistory [ ] : Ride
- paymentDetails : PaymentDetails
Car
- id : int
- status : CarStatus
- plate : String
- passengers : int
- available : boolean
- position : Coordinates
- safeArea : SafeArea
- batteryCharge : float
SafeArea
- area [ ] : Coordinates
PaymentDetails
- paymentOption : PaymentOption
- creditCard : CreditCard
- bankAccount : BankAccoun
<<enum>>
UserStatus
- <<enum constant>> Active : int
- <<enum constant>> Suspended : int
<<enum>>
CarStatus
- <<enum constant>> Closed : int
- <<enum constant>> OpenLocked : int
- <<enum constant>> OpenUnlocked : int
Coordinates
- latitude : float
- longitude : float
<<enum>>
BillStatus
- <<enum constant>> Paid : int
- <<enum constant>> Pending : int
- <<enum constant>> Rejected : int
Ride
- status : RideStatus
- rideCar : Car
- reservationMinutes : int
- rideMinutes : int
- billAmount : float
- billStatus : BillStatus
<<enum>>
RideStatus
- <<enum constant>> Reservation : int
- <<enum constant>> InUse : int
- <<enum constant>> Finished : int

Figure 2: Class Diagram

2.2 Product Functions

In this section are listed all the goals we aim to accomplish trough the different functionalities we plan to implement.

[G1] Users must be able to register to the system by providing all necessary credentials and payment information. They will receive back via e-mail a password to access the system.

[G2] User are able to login using their personal login credentials (e-mail/password).

[G3] User must be able to search for available cars nearby or close to every chosen location. For each available car battery charge and specific safe area must be visible.

[G4] User must be able to reserve a single car for up to one hour before they pick it up. If the reserved car is not picked-up within the one-hour limit reservation in cancelled and the user pays a 1 EUR fee.

[G5] A user has the chance to cancel his reservation at any given moment.

[G6] A user near to a free or reserved (by him) car must be able to communicate his position in order to let the car open if he's detected close to it.

[G7] Once the car has been opened the user must be able to unlock it through using his private 4-digits PIN code.

[G8] The first time a registered user gets into a car he has to be prompt with a message allowing him to choose his personal PIN code.

[G9] As soon as the car is unlocked the "reservation" state ends, the engine can now be turned on and the system starts charging the user for a given amount of money per minute.

[G10] At each moment the user has to be notified with the current amount he's being charged.

[G11] Within the car safe area, the user must be able to stop the vehicle, lock it and close it to finish his ride.

[G12] The system must be able to close the car automatically once the user is detected outside the vehicle.

[G13] The system stops charging the client as soon as the car is empty and closed.

[G14] The system must lock the car automatically when the user exits the vehicle.

[G15] The system must apply a discount of 10% on the last ride if the user took at least two other passengers onto the car.

[G16] The system must apply a discount of 20% on the last ride if the car is left with more than 50% of remaining battery charge.

[G17] The system has to suspend a user if the ride payment fails and restrict access to cars until the issue is solved.

2.3 User Characteristics

All of the registered users are in possession of a valid driving license and have accepted all of the terms and conditions imposed by the company during registration phase. Are in possession of valid credentials to login the system and start searching for available cars. Since the only way to open PowerEnJoy electric cars is via the smartphone app each user is required to have an Android or IOS device.

2.4 Constrains

2.4.1 Regulatory policies

All sensitive data and user information are acquired by the company under the accepted terms and conditions. This data is stored in the company DB; it's use and transmission to third parts society is regulated in accordance to the law.

2.4.2 Hardware limitations

Except for an internet connection and a compatible smartphone, the system does not require any specific hardware limitation.

2.4.3 Interfaces to other applications

Specific API integrations is used to perform credit card payments and check the user's driving license validity in real time.

2.4.4 Parallel operations

PowerEnJoy system must support parallel operations from different users when working with the DB in order to avoid collision or any other integrity issue.

2.5 Assumptions and Dependencies

2.5.1 Dependencies

Car hardware: in order to provide all the services and functionalities specified the all of the electric cars must have:

- GPS tracker;
- Mobile internet connection;
- A powerful embedded system to provide all the required functionalities and communicate to the central system;
- A touch screen monitor to handle user input to the embedded device;
- Accurate sensors to collect data from engine, battery and instruments, a weight sensor positioned under each seat to calculate the number of present passengers.
- A hardware connection between the embedded system and the CAN bus of the car to control door locking/unlocking and receive data from the sensor.

Mobile application: access to the positioning system has to be granted by the user in order to open the car. Internet connection is also mandatory to communicate the position to the central system to manage the request.

Web application: there are no specific dependencies other than an internet connection to use the web application. No specific plugin or browser extension are required. Browser location detection is optional to find “around me” cars.

Power grid station: each power grid station is connected to the central system through and internet connection to send information about free recharging spots or possible damaged plugs.

2.5.2 Assumptions

Users: the user who has reserved or directly opened the car is the only user entitled to drive it. All users are considered honest and not interested in cheating the weight sensors to achieve the tariff discount. Users always reports damages or bad interior conditions for each car they use.

Cars: cars are always connected to the internet and the GPS signal is to consider stable. An external company is in charge to provide the said services to PowerEnJoy: cleaning the car if necessary, recharging the batteries if cars are left outside a PGS and run out of charge, reintegrate abandoned cars inside the specific safe area. A car has the ability to maintain enough battery charge to perform crucial operations: closing itself if abandoned, send to the central system the current position, send as much data as possible to the central system if any misuse is detected. For safety reasons a car cannot close itself if any passenger is detected inside the vehicle.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces