

PowerEnJoy

Requirement Analysis and Specification Document

A.Y 2016/2017

Francesco Tinarelli (matr:806146)

Marco Wenzel (matr:878021)

Versione 1.1

General Description

We will project and implement PowerEnJoy, it’s a digital management system for a car-sharing service.

This system is based on mobile application and web application, there are two kind of different people that can use the system: Unregistered user or Registered user.

Unregistered user can only visit the web application and register himself on the system.

The system allows registered user to see the available cars in a specific zone or current position area using GPS and through this service is possible to make a reservation for up to 1 hour.

The user can unlock the reserved car logging on the web application and using “unlock car” button, the system provides this operation if the user is nearby to the reserved car.

The system registered some special safe area called: power grid area which is possible plug the car.

The system during the ride show the current charges money on the car screen.

More over the screen show power grid stations and permits to choose the path for the nearest grid station or ones specified by the user.

In addition to the functionality above, the system should incentivize the virtuous behaviours of the users, in fact, if the user provides to plug the car or in generally take care of the car, the system rewards him with discount.

There is no system before of this one.

Actors

* **Unregistered user:** a person no registered on system, he has to register on the system if he wants use system services.
* **Registered user:** a person registered on system, he can use all services and the system has personal data of him.

Goals

**Unregistered User:**

[G1]: Allow to register on the system

**Registered User:**

[G2]: Allow to log in on the system.

[G3]: Allow to find the locations of available cars within a certain distance from their current position or from a specified address.

[G4]: Allow to reserve a single car for up to 1 hour.

[G5]: Allow to pay a fee if he misses the reservation.

[G6]: Allow to tell the system she’s nearby.

[G7]: Allow to see the current charges.

[G8]: Allow to reach grid stations location.

[G9]: Allow to receive a discount if the user has at least two passengers.

[G10]: Allow to receive a discount if the user left the car with no more than 50% battery empty.

[G11]: Allow to receive a discount if the user left the car in a Power grid station and takes care of plugging the car.

[G12]: Allow to receive a fee if the user left the car more than 3km from the nearest power station or left the car with more than 80% of the battery empty.

[G13]: Allow to see an e-mail sent by the system with the bill ride and payment resume, at the end of this one.

[G14]: Allow to user to pay immediately after at the end of the ride.

Glossary

**Reservation:** it’s the ability to reserve an available car for up to 1 hour, provided only for registered user.

For reserve a car the user must choose the available car from the map and click on it.

For use this service, user must be logged on web/mobile app.

**Reservation Timer:** this timer count how much time remain for open the car.

**Unlock Car Button:** it’s a button on the mobile app that allow the system to check the user position and the reservation for unlock the car.

**Safe Area:** every zone where it’s possible park the car safely.

**Power Grid Area:** special safe area where it’s possible park the car safely and user takes care of plugging the car into power grid if he wants.

**Plugging Time:** the time that the user has to plug the car before the system create the bill.

**Bill:** Amount of cost that the user must pay for the ride, discount and/or fee included.

**System:** it’s the new system that we want implement. The system interacts with user for providing services, External Agency for payment, cars for receive ride data.

A piece of system software is installed into each car, so in this way the system and the cars can communicate and send data.

**External Agency:** It’s an agency that provides all type of payments service, when user must pay a fee or pays a ride. Interact with system for receive the amount to subtract on user bank account.

**User:** he is a client of the service. His personal data is already memorized into the database system (already registered on system), the personal data inserted:

* Name
* Surname
* E-mail
* Birth Date
* Driving License number
* Credit card number

**Discount**: it’s a reward for good behaviour.

**Ride**: service provided by the system that allow to engine ignites and start to charging the user for a given amount of money per minute.

**Fee**: amount of money that the user pays in addiction, fee can be assigned through missed reservation or bad behaviours.

**Car**: an automobile, in each car we have a screen and a GPS for provide services. Car send to the system the data of the ride.

**Reference Documents:**

For creating this RASD document, we used:

* Specification Document: Requirement Engineering III.pdf
* IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.
* Example document: MyTaxiService.pdf

**Overview**

The document is composed by 3 part:

1. **Introduction:** brief introduction, specific Goals and the glossary, in this way we have an overview of main services provided and some nomenclature.
2. **Overall Description:** Constraints, Domain and text assumption for specify the world that we want study.
3. **Specific Requirements:** analysis of Consistency about Domain assumption and Goals.

**Overall Description**

**Domain assumptions**

* All the GPS always give the right position.
* The GPS of the cars cannot be switched off.
* A user doesn’t make a reservation until he pays last ride of fee.
* A user drive only if has got driving license. ok
* There is a software that allow to recognize the right number of passengers.
* System battery car always show the right battery.
* Each car is registered on system.
* Every car is the same type.
* If a user parks in a not-safe area zone and take a fee, he will pay this fee.
* The payment is provided by an external agency.
* When the user finishes the ride and the plugging time is over, the external agency provides an immediate payment if the user has enough money for pay.
* If the user doesn’t have enough money on the bank count, the external agency notifies the system and this one notifies user with an e-mail.
* Power grid area location is registered on the system.
* Car state could be only available or not-available
* There aren’t pause during the ride.
* If a car is left with more 80% of battery empty, the car state ’ll change into not-available state and a maintenance guy will go to the place for plugging the car.
* If a user takes more than 1 discount he takes only the biggest one.
* If a car is recharging, its state is not-available until the battery left is almost 70%.
* There is a car crash insurance for every car.
* The potential discounts will be counted only after plugging time.
* The plugging time is 3 minutes.
* A car is nearby if the distance between this one and the user is less than 3 meters.
* The amount of the payments is always right.
* If a user takes a discount and a fee, at the end of the ride system counted only the fee.

Product Perspective

User External Interfaces:

The product consists in a web and mobile application both based on web.

Web application should be providing register system; a user can register on the system only via web application.

Moreover, it’s possible for registered user to see the payment history.

The mobile application should be use to check the user position and unlock the car using specific button.

In addition, the mobile application permits to reserve a car in the same way than web application.

So, the use of mobile application is mandatory if they want ride a car.

Moreover, in each car there is a screen that permits to find the grid station locations and reach them through specified address or the nearest one.

The screen permits also to see the current ride charge.

System External Interfaces:

The system interacts with an external agency for providing the payment service.

At the end of the ride system receives the amount of money that user has to pay, it sends this amount to external agency, external agency tries to do the transaction and notifies the system if the transaction is successes or is failed (user doesn’t have enough money in bank account).

Constraints

**Regulatory policies**

The system must require to user the permission to get his position and he has to manage sensible data (position, private data, credit card number) respecting the privacy law. Furthermore, the system must not use this data to send SPAM e-mail respecting the privacy law.

**Hardware Constraints**

Each car must have GPS and a screen to show charging ride.

Mobile application must be installed on a smartphone that has enough space for this app and almost 3G connection.

**Parallel Operation**

The server supports parallel operations from different users who use different cars and different power grid stations.

Specific Requirements

External interface requirements

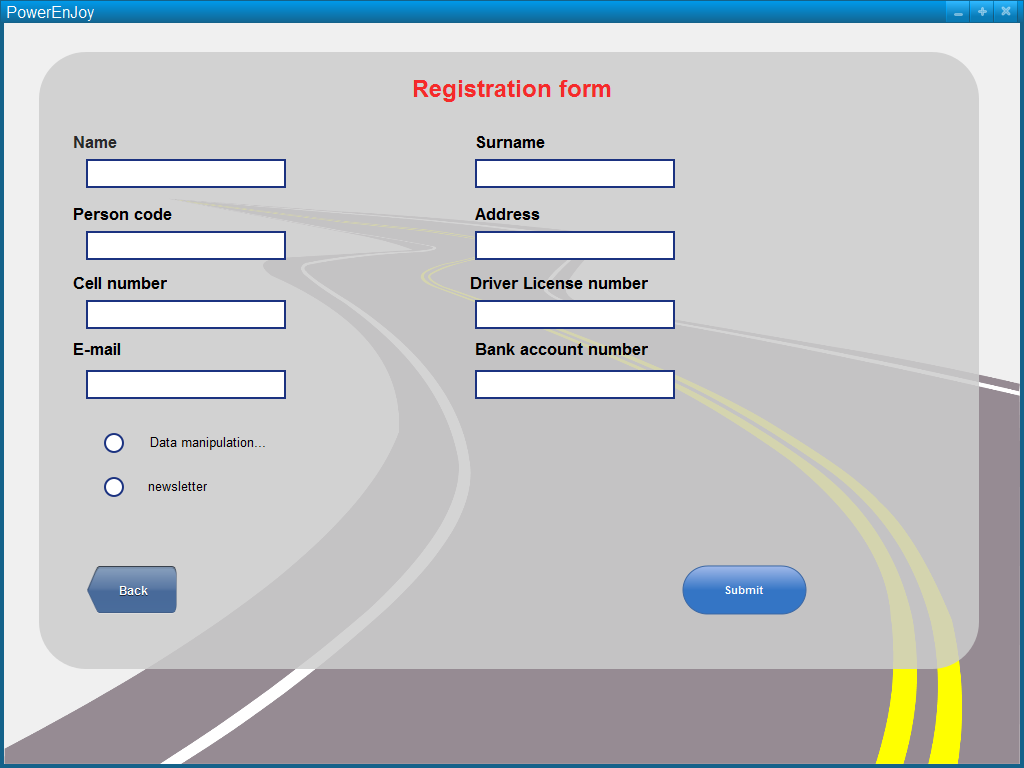
**User Web Application:**

* **Log in**

****

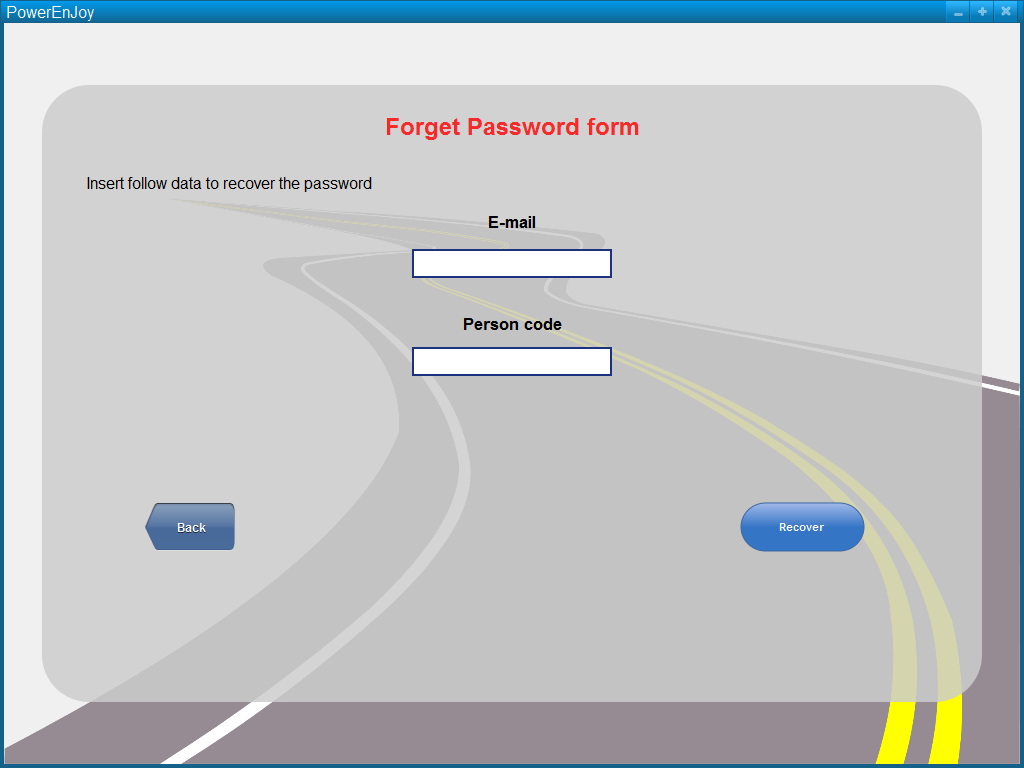
As we said Log in required only e-mail and the password generated by the system when user was registered.

* **Register**

****

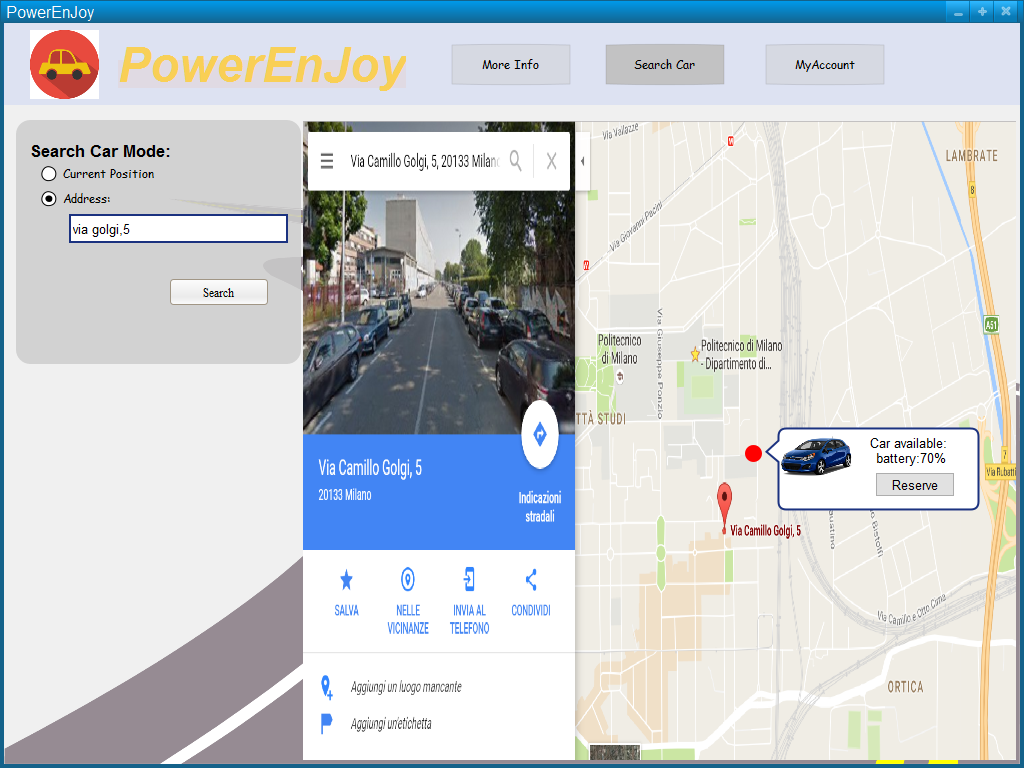
This functionality is provided only via web app.

* **Forget Password**



This functionality is provided only via web app.

* **Search available car and reserve**

****

This mock up represents the service: Search available cars near user position or near specified address, through this functionality is possible reserve a car click on the button like in the mock up.

* **See all Bill and Payments**

****

This functionality is provided only via web app.

This mock up shows the system service that permits to see the bills of the latest rides and through this user see how discount is reached.

This particular mock up shows the case which the user doesn’t have enough money on bank account, the external agency that provides payments service notifies the system and this one notifies the user.

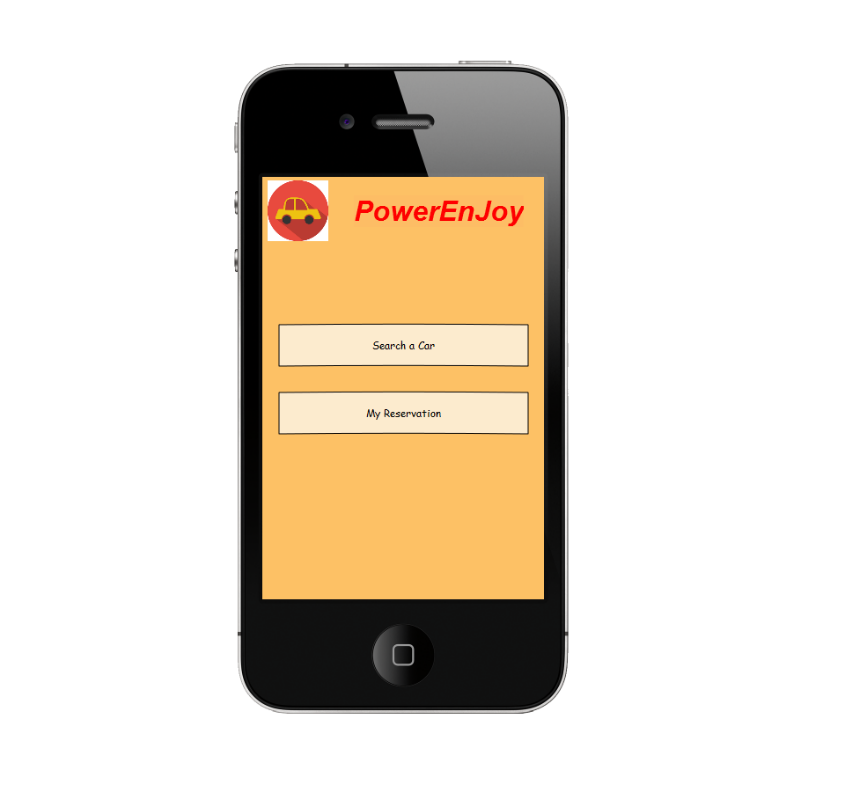
User Mobile Application

* Log in



This is the log in service via mobile application, as we already said mobile application don’t provide register service and forget password service.

* **Menu**

****

This screen don’t provide any service but we think it’s important to demonstrate that the mobile application is thin and provide only necessary service to control or make reservation.

* Search a car



The same service provides via Web Application.

* My reservation



This functionality is provided only via mobile app.

This screen permits to show if this particular user has a reservation, address and battery of reserved car and the time left until the reservation expired.

Unlock car button permits to check the closeness (near 5m) and the reservation data.

Car screen

* Reach grid station and see current charging money



These functionalities are provided only via car screen, these permits to choose grid station path and see the current charging money.

All of these mock up images represents only a few possible screen of web, mobile application, in this document miss the confirm screen for reservation and unlock car.

We have decided to insert the most representative screen.

Functional Requirements

**[G1]: Allow to register on the system**

[R1]: The system check if the input datas are correct and of those one are not already exist.

[Domain Assumptions]: An user drive only if has got driving license.

**[G2]: Allow to log in on the system.**

[R2]: The system must be able to check if the password and username are correct.

**[G3]: Allow to find the locations of available cars within a certain distance from their current position or from a specified address.**

[R3]: The system checks the syntactical correctness of the input address

[R4]: System must show only the available car into 2km.

[R5]: System must capture the current position user if he chooses this option.

[R6]: For each available car, the system must show how much battery has the car.

[Domain Assumptions]:

All the GPS always give the right position.

The GPS of the cars cannot be switched off.

System battery car always show the right battery.

Each car is registered on system.

Car state could be only available or not-available.

**[G4]: Allow to reserve a single car for up to 1 hour.**

[R7]: The system must show the reservation timer.

[R8]: The system must change the car state whenever a car is reserved.

[R9]: The system must forbids more reservation at the same time from the same user.

**[G5]: Allow to pay a fee if he misses the reservation.**

[R10]: The system must add a fee on the user bank account.

[R11]: The system must change the car state from not-available to available.

[R12]: The system must allow user makes a new reservation.

[Domain Assumptions]:

A user doesn’t make a reservation until he pays last ride of fee.

Car state could be only available or not-available.

//e.a. assumptions?

**[G6]: Allow to tell the system she’s nearby.**

[R13]: The system must capture the user position when he wants unlock the car.

[R14]: The system must forbid the car unlocking if the distance between user position and the car is not near or he doesn’t have a reservation for that car.

[R15]: The system must stop the reservation timer

[Domain Assumptions]:

All the GPS always give the right position.

The GPS of the cars cannot be switched off.

A car is nearby if the distance between this one and the user is less than 3 meters.

**[G7]: Allow to see the current charges**

[R16]: The system must charge money when the motor ignites.

[R17]: The system must notify in real time how much the client is paying for the ride.

[R18]: The system stops charging when the engine is shut down.

[Domain Assumptions]:

There are no pause during the ride.

The potential discounts will be counted only after plugging time.

**[G8]: Allow to reach grid stations location.**

[R19]: The system during the ride visualize the position of grid station to the map (through the car screen).

[R20]: The system must calculate the path to the nearest grid station or grid station chosen by the user.

[Domain Assumptions]:

Power grid area location is registered on the system.

**[G9]: Allow to recive a discount if the user has at least two passengers.**

[R21]: The system must apply a discount of 10% for the last ride if there aren’t any other discount.

[R22]: The system must be able to recognize if there are passengers.

[Domain Assumptions]:

There is a software that allow to recognize the right number of passengers.

If a user takes more than 1 discount he takes only the biggest one.

The potential discounts will be counted only after plugging time.

If a user takes a discount and a fee, at the end of the ride system counted only the fee.

**[G10]: Allow to receive a discount if the user left the car with no more than 50% battery empty.**

[R23]: The system must read the state of the battery.

[R24]: The system apply a discount of 20% if battery left is more than 50%.

[Domain Assumptions]:

If a user takes more than 1 discount he takes only the biggest one.

The potential discounts will be counted only after plugging time.

If a user takes a discount and a fee, at the end of the ride system counted only the fee.

System battery car always show the right battery.

**[G11]: Allow to receive a discount if the user left the car in a Power grid station and takes care of plugging the car.**

[R25]: The system must recognize if a car is parked in a power grid station

[R26]: The system applies a discount of 30% if a car is left in a power grid station and user takes care of plugging the car.

[Domain Assumptions]:

If a user takes more than 1 discount he takes only the biggest one.

The potential discounts will be counted only after plugging time.

If a user takes a discount and a fee, at the end of the ride system counted only the fee.

Power grid area location is registered on the system.

If a car is recharging, its state is not-available until the battery left is almost 70%.

The plugging time is 3 minutes.

**[G12]: Allow to receive a fee if the user left the car more than 3km from the nearest power station or left the car with more than 80% of the battery empty.**

[R27]: The system must calculate the distance from the nearest power grid station.

[R28]: The system must read the state of the battery.

[R29]: The system must apply a fee of 30% on last ride if user lefts the car more than 3km from the nearest power station or left the car with more than 80% of the battery empty.

[Domain Assumptions]:

If a car is left with more 80% of battery empty, the car state ’ll change into not-available state and a maintenance guy will go to the place for plugging the car.

If a user takes a discount and a fee, at the end of the ride system counted only the fee.

All the GPS always give the right position.

The GPS of the cars cannot be switched off.

**[G13]: Allow to see an e-mail sent by the system with the bill ride and payment resume, at the end of this one.**

[R30]: The system must send an e-mail containing the bill at the end of the plugging time and the resume of the bank transaction.

**[G14]: Allow to user to pay immediately after at the end of the ride.**

[R31]: The system must send the amount of money that user pays to external agency.

[Domain Assumptions]:

The payment is provided by an external agency.

When the user finishes the ride and the plugging time is over, the external agency provides an immediate payment if the user has enough money for pay.

If the user doesn’t have enough money on the bank count, the external agency notifies the system and this one notifies user with an e-mail.