

1 Requirements

1.1 Functional Requirements

1.1.1 User Requirements:

- G.1)** First-Time Users must be able to register to the System creating an Account.
- ~~The System must allow an user that has not created an Account to register by providing valid credentials, an e-mail and payment information.~~ **registerPage.html, registrationServlet.java, EmailUtility.java**
 - The System must sent a password to the Registered User Email that can be used to Log into the Account.
- G.2)** Registered Users must be able to login to their Account at any time they want.
- The System must allow Registered Users to log into their Account only if they provide their correct e-mail and password. **index.html, loginServlet.java**
- G.3)** Users can save his/hers credentials in the System.
- The System must be able save Users credentials upon their request.
 - When the user choose to save his/her credential they don't need to inform it in every time they open the System.
- G.4)** A Registered User will be able to make a Reservation of any available Car near his/her current location or from an address that she/he can specify.
- The System must be able to locate the User via GPS.
 - The System must allow the Users to enter a determined address, in case they don't want to use their current location to locate the cars in the area.
 - The System must be able to locate all available Cars via GPS.
 - The System must show the User all the available Cars near the user given location.
 - The System must allow the Users to select one of the Cars showed to them to create a Reservation, the selected Car will be marked as reserved.

G.5) Users that have made a Reservation must be able to notify the System when they are nearby the Reserved Car so the System can unlock it.

- While a Reservation is active, the System must allow the User to notify that they are nearby their Reserved Car.
- The System must confirm the notification by checking that the User current location and the Car location coincide.
- The System must unlock the Reserved Car and set it to Ready to Use.

G.6) A User that has made a Reservation must be able to cancel it before 1 hour starting from the time when the Reservation was made.

- The System must check how long has passed from the reservation time, if it less then one hour the user will be able to open the reservation and cancel it without being charged for it.
- The System must confirm the user request and set the car as available again.

G.7) In case an User hasn't started using the Reserved Car at 45 minutes after the Reservation was made, he/she will be notified that either if the Reservation is not canceled or the Car is not used in 15 minutes, the Reservation will be automatically canceled and a 1 Euro fee will be charged to her/his Account.

- After 45 minutes from the reservation time has passed, the System must send a notification to the user's registered smartphone warning that he/she has 15 minutes left to pick up the car or to cancel the reservation.
- After the full hour has passed and the reservation is still active, the System must cancel the reservation, block this user account for the next hour and charge the fee of 1 Euro for not picking up the car.

G.8) When Users start using their Reserved Car, they must be able to see their current expenses on the service through a System screen inside the Car.

- The car System must keep track of the gross fee to be paid by the user.
- As soon as the car ignites, the car System must start showing the fee on the computer monitor and keep it this way until the user is finished using the car.

G.9) The User must be able to know where the safe parking areas are nearby his/her current location or any address that she/he can specify.

- The System must be able to show, when requested, a map with the safe parking zones and points with electricity plugs .

G.10) Users must be able to finish their use of the Car when leaving it in a Safe Parking Area and exiting the car. The User will then be charged for the use of the service. The used Car will be locked and freed for Users to be reserved.

- The System must verify if the car is parked in a Safe Parking Area. If the car isn't located in one, the System must notify the user on the screen of the car that it's not possible to finish the ride.
- The System must show a message on the car's screen asking if the user want to finish the ride or if want to continuing the use of the car.
- The user must answer to the request. If the user want to keep using it, the System will set the car as reserved for one hour. Nevertheless in terms of payment the System will charge the same amount of money in the final Transaction as if the car has been in a ride. If the user does not want to use the car the System will finish ride.
- Upon the user leaving of the car the System must lock the car and consider the user's answer to whether set the car as available or reserved.

G.11) The User will always be notified when any Transaction is made on his Account.

- The System must notify the user of every Transaction made on his Account by email. Further information related to the ride as the total time, the driven route and the charged Fees must compose the email.

G.12) Notify the Users that are currently using a Car of any available discounts on their ride if they abide by the 'Virtuous Behaviour Rules' and of the extra fee in case of not respecting the facilitation of the re-charging of the Car on site. These extra discounts/charges will be applied on the Total fee at the end of the ride.

- The System must notify the user on the Car's screen of the any information related with a discount or extra fee.

- The System must calculate at the end of the ride if any discount or extra fee is applicable. These fees are:
- G.12.1) Apply a 10% discount if the User takes at least 2 other passenger into the car.
 - The System must read from the status of the Car the number of passengers and if it's 2 or more, apply the 10% discount to the ride fee.
- G.12.2) Apply a 20% discount if the User leaves the Car with the battery at least half-full.
- G.12.5) Apply a 30% extra fee if the User leaves the Car with the battery less than 20% full.
 - The System must read from the status of the Car the amount of battery left and if it's 50% or more calculate the 20% discount else if it's less than 20% calculate the 30% extra fee.
- G.12.3) Apply a 30% discount if the User leaves and plugs the Car in a Re-Charging Station.
 - The System must detect from the status of the Car if it's plugged in to a Re-Charging Station and if it is, calculate the 30% discount.
- G.12.4) Apply a 30% extra fee if the User leaves the Car at more than 3KM from the nearest Re-Charging Station.
 - The System must read from the status of the Car its current location and the distance to the nearest Re-Charging Station, if it is more than 3KM then calculate the 30% extra fee.
- The System must apply these discounts or extra fees at the end of the ride over the total fee.
- G.13)** Users can activate the 'Money Saving' option on their Account to be notified of any nearby Re-Charging station on their arrival destination. Leaving the car at the end of the ride at this station and plugging it will register as a 'Virtuous Behavior' and will apply and extra discount when charging the User.
 - Users must be allowed to enable the 'Money Saving' option on their accounts at any time they choose.
 - The System must allow the user to input his/her destination at any time of the ride whether on the app or on the car's screen.
 - Upon the final destination of the user the System must direct the end of the ride considering a uniform distribution of parked

cars of the System, the nearest charging station in relation with the address informed and the availability of these charging stations.

G.14) Users must have the option to contact a CRM at any moment, the System must provide a Customer Service area that offers a Chat feature within the App or a phone number.

- The System must show a Customer Service number in its Customer Service Area.
- The System must have a Chat feature that will allow the User contact directly a CRM.

1.1.2 CRM Requirements:

G.15) CRM must be able to log in into the System and see a list of all the Cars available, reserved, in use or unavailable.

- The System must allow log in to CRM if they have provided valid username and password.
- The System must show to CRM a list of all the Cars that can be sorted by Status.

G.16) CRM has to be able to receive User Reports via chat or outside the app via phone call and be able to register it to the System.

- When a User uses the Chat feature to contact a CRM the System will redirect the chat to the first available CRM and notify him/her that a User is contacting her/him.
- The System must allow the CRM to create a User Report that will link a User with a Car and if necessary allow the option to add a Transaction.

G.17) CRM upon request from an User or any major cause can Reserve or cancel any active Reservation.

- The System must allow from the list of all Cars showed to the CRM, the option to Reserve any available Car for a User or cancel a Reservation of a Reserved Car. This action does not include a User Report.

G.18) CRM must be able to change the Status of a Car given a User Report and tag it, in case it's unavailable, if it's due to Re-Charge, Fix, or Removal.

- The System must allow from the list of all Cars showed to the CRM, the option to Reserve any available Car for a User or cancel a Reservation of a Reserved Car. This action does not include a User Report.

G.19) Upon the completion of an User Report CRM must be able to apply extra fees or refund the payment to any User in case the situation demands it.

- The System at the end of an User Report must ask for confirmation of the CRM in case the Report includes a Transaction. This can be an extra fee that the User must pay or a refund the service owns the User.

1.2 Non Functional Requirements

1.2.1 User Interface

For our User Interface (UI) we'll offer a mobile App for Users and a desktop web App for Users and CRM. They will offer:

- **LogIn Page:** ~~First screen of the app with the LogIn and SignIn options.~~
- **Register Page:** ~~First screen of the app with the LogIn and SignIn options.~~
- **Main Page:** Main page of the app where the user sees the map his location and selects the available cars. In Case of CRM, it can see all Cars.
- **Car Details Page:** Once a Car is selected this page allows to reserve a car and contains informations for the decision.
- **Reservation Page:** Once the Reservation is made, the User can see the details of her/his Reservation and have the option to cancel it.
- **Ride Page:** Once a Ride started this page shows the details of the current ride, it's present also in the Car screen App.
- **Email Page:** While not part of the App UI, the System should respond with an email to the User at the end of the Ride showing the final details for it.
- **Contact Page:** Page where the User can contact the CRM operator.

1.2.2 Documentation

To organize this project workflow and development the following Documents will be drafted:

- Requirements And Specifications Document (RASD): Describes the Goals of the project and how to achieve them by specifying the Requirements.
- Design Document (DD): Defines the real structure of the System. Architecture, Components and design decisions will be explained here.
- Project Testing Document: Will detail the Test process for the system before deployment. Test case specifications, procedures, incidents, reports will be considered Documentation Testing.
- Project Management Documentation: Specific assignment of tasks, estimations and goals for each of the development phases.

1.2.3 System Architecture

Project will be developed using a JEE Platform based in a 3-Tier Client/Server Architecture. First idea consists in a Distributed Logic structured layers where the Client Apps will interface with the Server that will store the logic and will connect with the Data Layer of the System. The architecture as explained briefly in the prop system section will consist of:

- Client Tier:
 - Web Client on a browser with abilitated JavaScript for the Web App use.
 - Application Client for the Mobile App (Android and iOS), with GPS functionalities. We will consider the screen present in the Car as a Client App of the System that'll implement some logic to interface to the Car Software to Lock, Unlock, read the Battery level and other functions.
- Server Tier:
 - Web Server that'll answer connections for Web Clients. It will interface with the Application Server.
 - JavaEE Application Server, that'll implement the logic of the System. Application Clients (Users and Cars) will connect directly to it. Will implement JavaBeans to handle request from multiple Clients at a time.
- Enterprise Information System (EIS) Tier:
 - DataBase Server, accessed by the Application Server to store all the data refering to the System.

1.2.4 Hardware Consideration

As stated in the Architecture Considerations we'll need a server able to store our Application and DataBase and be accessible by multiple Clients at a time. We'll consider the Server to be always accesible through an stable connection to Internet. As an extension of the System we have the Clients running in the Cars, they'll need to be accessible via 3G and have GPS feature for localization.

1.2.5 Incident Handling

We'll consider our data to be persistent. If by any case any data is lost or corrupted we expect not to create big losses to the System, all transactions of data will be considered atomic we expect no loss during any kind of transaction.

1.2.6 Security

Data Backups will be considered daily for data security. Accounts will be protected by hashed passwords. As one of the requirements of the project asked to send Users a password via mail we assume that it'll be handled by secure servers (IMAPS) so we're assured just the User can see it. Firewalls will be considered for our Servers connections. In case of breach, User Accounts can be blocked by CRMs. CRM Accounts are introduced directly into the DataBase by a separate system that register the Company employees, we assume it safely as well.