COMPONENT INTERFACE VIEW

- **GPS Interface**

To convert addresses to GPS coordinates we use the google Maps Geocoding API. As well described on the website, Geocoding is the process of converting addresses into geographic coordinates which you can use to place markers on a map, or position the map Also Data is available in JSON and XML format. It will possible to use Google Maps Roads API.
The Google Maps Roads API allows you to map GPS coordinates to the geometry of the road, and to determine the speed limit along those road segments. The API is available via a simple HTTPS interface, and exposes two services:

- **DBMS Interface**

The DBMS Interface is thinking as a generic one because it can be of many type(MySQL,MsSQL etc).The DBMS Controller is not related to the specific DBMS, because it's only an abstraction of this interface and it should work with everyone. This interface is fundamental to do operation on the real database.

- **REST API Interface** (API Controller to front-ends)

The front-ends of the system (the web application and the mobile app) shall communicate with the application server using the back-end programmatic interface implemented as a RESTful interface over the HTTPS protocol. The RESTful interface is implemented in the application server using JAX-RS and uses XML as the data representation language.

- **Algorithm Interface**

This Interface allows the call to the method of some algorithm for example the save money algorithm to manage the discount or the addition charging to the user.

- **APP Interface**

This interface links the component Application Controller and the rest of application: It allows the communication between API controller and the Mobile Application from a side, and the logic of the application on the other side. It Also permit the communication between the Algorithm and the DBMS with the logic part of the Application.

- **UI Interface**

This interface Is the set of commands or menus through which the user communicates with the program. So it is all the thinks displayed on the app to make the user and the program interacting.

- **USER interface**

Offers the methods useful to log in a User. It interacts with PersonalDataManager component to complete the login and start the user session. Provides the methods for the insertion of a new customer in the database, validating his data. Provides the methods that check the personal credentials in the database. Provides all the methods to validate personal data, for instance the correctness of the name (it cannot contain numbers) or of a birthdate (it shall not be in the future).

- **UTILITY Interface**

Offers the method to detract money for the payment and sent the email notification of the bill or a notification of some event. Provides the methods for the notification of client systems.

- **RESERVATION Interface**

Provides the methods to make a reservation. It interacts with other components for data validation and storing. provides methods to see car state information and to unlock the car.

- **SENSOR Interface**

Sensor interface ICs connect directly to a sensor element and provide the needed signal conditioning to extract an accurate signal for monitoring and control systems. Fully integrated sensor signal conditioners connect directly to the sensor, perform conditioning, and optionally fulfill data communication functions.

- **ADMIN Interface**

This interface permit to: Manage basic software configuration, Create and configure groups, Create and manage databases, Back up and restore content, manage all the maintenance of the system of carSharing, organize the work for the employee, control the bureaucracy part of the system as fine or insurance expiry , control the equal distribution of cars in the city.

- **EMPLOYEE Interface**

This interface offers the methods to organize the work of the employee, like the relocation of cars, the emergency problem etc.