PP - Project Plan

Valeria Mazzola
Luca Marzi
Federico Nigro

January 22, 2017

Contents

1 Introduction

1.1 Hours of work

- Valeria Mazzola: 10h
- Luca Marzi: 10h
- Federico Nigro: 10h

1.2 Purpose and scope

This documents aims to provide a first estimation of the size of project and the effor required to implement all its functionalities (described in previous documents).

The purpose of this document is to serve as a guide in the analysis of physical resources and time that are necessary to obtain a deliverable product.

Neverthless, this documents aims to provide a possible schedule to follow for the completion of the different phases of the project, from the requirement analysis from the integration plan and testing.

1.3 Definitions & Acronyms

- RASD Document
- DD Document
- RASD: Requirements Analysis and Specification Document
- GPS: Global Positioning System
- SDK: Software Development Kit
- DBMS: Database Management System

2 Project size estimation

This section of the document provides an esitmation of the expected size of the project PowerEnJoy system.

The estimation relies on the attribution of the so called Functional Points to different characteristics of the project like:

- Data structures
- Input/Outputs

- Interfaces

For the cost and effort estimation we will instead rely on the COCOMO approach, using as initial guidance the amount of lines of code computed with the FP approach.

2.1 Function size estimation: function points

We will use the Function Point approach to estimate the size of the PowerEnJoy project.

We use the following table as a reference for the complexity weights used in evaluating the Function Points characterizing the project.

	Complexity		
Function Type	Low	Medium	High
Internal Logic Files	3	4	6
External Logic Files	4	5	7
External Inputs	3	4	6
External Outputs	7	10	15
External Inquiries	5	7	10

2.1.1 Internal Logic Files (ILFs)

PowerEnJoy relies on several types of ILFs to store information that is needed to support the main functionalities of the system.

In this section of the document we will analyze the main ILFs we have indentified.

The system as to store information about users and cars. All the information about users in condensed in a single table that holds all the information regarding the user's personal information like name, surname, driving license ect. The same table is used to handle all the information related to the authentication of the user and his/her association to a specific device.

A different table is used to record all the payment methods that are associated to a user (and that can be shared by different users); the system has been designed to accept only one type of payment method that is PayPal. Although we will consider all the interaction with PayPal services in terms of ELs, some information is condensed in a single table which store, for each payment method, the id of the user associated to the payment method, a token that has to be associated to each request towards the PayPal API.

A third table is used as support for the part of the business model that handles the cars; this table contains information such as: the name and the model of the car, its plate and a set of information that are related to the status of the car.

A fourth table is used to handle all the information concerning the part of the business model that handles the reservation and the association of active rides to reservations.

A fifth table is used to handle all safe zones and the power stations that are offered to the user as places where the car can be parked. This table contains some information about the status of each safe area (according to the business model) and its positioning inside a specific geographical context.

ILF	Complexity	FPs
Login data	Low	2
Payment data	Low	3
Car data	Medium	4
Reservation data	Medium	4
Safe Areas/PowerStations	Medium	4
	Total	25

2.1.2 External Logic Files (ELFs)

First of all, PowerEnJoy has to store a lot of information about the current state of each physical car of the system. This information includes telemetry information, status of the battery, status of other physical devices that are directly connected to the operation of the physical car and that are provided by the operating system on board.

This information is provided in real time by onboard system which as to communicate with different third-part applications that are installed onto the informatic system of the car.

ELF	Complexity	FPs
Telemetry data	Low	2
Status data	Medium	8
Other Car data	Low	2
	Total	12

2.1.3 External Inputs (EIs)

The PowerEnJoy systems receive inputs from several main actors such as the users and the cars.

In this paragraph we summarize the main inputs and the expected complexity for the handling of these inputs.

Users' inputs:

- Login/Logout/Register: simple operations that can be handled very easily. There are several available frameworks that offer a support for handling these kinds of inputs from the point of view of input sanitize ect. They contribute 3 FPs each.
- Change settings: it's a relatively simple operation. It contibutes 3 FPs.
- Change personal information: this operation requires some checks about the consistency of the new data but, overall, it's a simple operation. It contributes 3 FPs.

- Position providing: thanks to the API provided the Android and iOS operating systems, this operation is relatively simple. It contributes 3 FPs.
- Reservation insertion: it's a very complex operation because it requires the exclusive acquisition of some resources and the cooperation of several architectural modules; this operation must be handled properly because it represents the core of the business model. This operation contributes 8 FPs.
- Reservation deletion: it's a relatively simple operation because it associated to the releasing of some resources and not thery acquisition. It contributes 3 FPs
- Start active ride: it's a reather complex operation that requires the cooperation of different architecturale modules and different actors (the user and the car). This operation requires many conditions to be satisfied and involves a clear effect in the physical world. It contributes 10 FPs.

Cars' inputs:

- Status update: it's a relatively a simple operation. It contributes 3 FPs.
- Command result: it's a simple operation. It contributes 3 FPs.

EI	Complexity	FPs
Login/Logout/Register	Low	3*3
Change settings	Low	3
Change personal information	Low	3
Position providing	Low	3
Reservation insertion	High	8
Reservation deletion	Low	3
Start active ride	Low	3
Car's status update	Low	3
Command result	Low	3
	Total	38

2.1.4 External Inquiries (EIs)

In this paragraph we consider the main data retrieval requests that can be performed by the users and their relative and their contribution in terms of FPs.

- Retreive list of cars nearby: overall it's a simple operation that requires some coordination between different architectural modules. Under the hypothesis that the business model is consistent, this data retrieval can be performed by means of a single query that operates with trigonometric data. It contributes 3 FPs.

- Retreive reservation information: simple operation, it contributes 3 FPs.
- Retreive list of safe areas: simple operation, it contributes 3FPs.
- Retreive retreive payments: simple operation, it contributes 3FPs

EQ	Complexity	FPs
Retreive list of cars cars nearby	Low	3
Retreive reservation information	Low	3
Retreive list of safe areas	Low	3
Retreive list payments	Low	3
	Total	12

2.1.5 External Outputs (EO)

In this paragraph we discuss the main outputs that the system has to produce with respect to the environment. Most of the environment is represented by the users. Notifications consist both of email notifications and push-notifications.

Here are the fundamental forms of communications that the user receives from the system:

- Notification of the expiration of a reservation.
- Notification of the beginning of an active ride.
- Notification of the end of an active ride.
- Notification of the end of a reservation.
- Notification receipts and penalities.

EQ	Complexity	FPs
Notification of the expiration of a reservation	Low	3
Notification of the beginning of an active ride	Medium	4
Notification of the end of an active ride.	Low	3
Notification of the end of a reservation.	Medium	4
Notification receipts and penalities	Low	3
	Total	17

2.1.6 Overall estimation

The following table summarizes the results of our estimation activity:

Function Type	Total FPs
Internal Logic Files	25
External Logic Files	12
External Inputs	38
External Inquiries	12
External Outputs	17
	104

We consider Java 2E as main platform for the development of the core of the business logic; we have decided to consider also the development of the mobile applications as part of the development of the business logic for the estimation of the expected lines of code.

Lower bound:

$SLOC$= 104* 50 = 5200

Upper bound:

$SLOC$= 104* 70 = 7280

2.2 Cost estimation with COCOMO II

2.2.1 Introduction to the COCOMO II method

As mentioned, we are using the COCOMO II method to estimate the effort the PowerEnJoy project will require. Such effort is calculated with the following formula:

$$Effort = A * KSLOC^{E} * \prod_{i=1}^{n} EM_i \tag{1}$$

where:

- $A = 2,94$ is the productivity constant for COCOMO II
- $KSLOC$ is the size of the project computed with the Function Points method in section 2.1
- EM_i terms are the effort multipliers, to be computed from cost drivers in section 2.2.2
- E is the scale factors parameter, to be computed in section 2.2.3

2.2.2 Cost drivers

Here is our estimation of the post-architecture cost drivers:

Cost driver	Description	Level	Multiplier
RELY	Required Software Reliability	High	1.10
DATA	Database Size	Nominal	1.00
CPLX	Product Complexity	High	1.17
RUSE	Developed for Reusability	Nominal	1.00
DOCU	Documentation Match to Life-Cycle Needs	Low	0.91
TIME	Execution Time Constraint	High	1.11
STOR	Main Storage Constraint	Nominal	1.00
PVOL	Platform Volatility	Nominal	1.00
ACAP	Analyst Capability	Nominal	1.00
PCAP	Programmer Capability	High	0.88
PCON	Personnel Continuity	Low	1.12
APEX	Applications Experience	High	0.88
PLEX	Platform Experience	Nominal	1.00
LTEX	Language and Tool Experience	Nominal	1.00
TOOL	Use of Software Tools	High	0.90
SITE	Multisite Development	Nominal	1.00
SCED	Required Development Schedule	Nominal	1.00
	Overall product of all cost drivers		1.01477

2.2.3 Scale factors

Here is our estimation of the five scale factors:

Precedentedness

Scale factor	Precedentedness
Description	Reflects the previous experience of the organization with this type of project. Varies from the throughly unprecedented case to the very familiar.
Factor weight	Low: we don't have previous experience with a project this big nor with the areas of experties required for it, such as the Java2E framework. We do however have experience with Java and with medium-sized projects.
Value assigned	4.96

Development flexibility

Scale factor	Development flexibility
Description	Reflects the flexibility of the constraints on the project requirements.
Factor weight	Nominal: the specification given to us has left us with a great deal of choices in the interpretation of the specification itself.
Value assigned	3.04

Risk resolution

Scale factor	Risk resolution
Description	Reflects the level of accuracy and depth of the risk analysis performed, and the level of reactiveness to risks that can be achieved during the project.
Factor weight	Nominal: we have performed some risks analysis in 4.1 and 4.2. However, our experience in the field is not solid enough to be sure that the analysis performed will cover the actual risks we may face during the future phases of the project.
Value assigned	4.24

Team cohesion

Scale factor	Team cohesion
Description	Reflects the level of cohesion of the stakeholders, their ability to share a vision and to commit to the choices taken so far in the project.
Factor weight	High: we believe the requirements and design choices made so far have been successful in putting together the needs and interests of all stakeholders.
Value assigned	2.19

Process maturity

Scale factor	Process maturity
Description	Reflects the process maturity of the organization and its ability to abide to CMMI guidelines.
Factor weight	Nominal: we believe our team is able to structure its work according to a level 2 Capability Model.
Value assigned	4.68

Computation of the E parameter

The E parameter, as described by the COCOMO II standard, is computed as such:

$$E = B + 0.01 * \sum_{i=1}^{5} SF_i \tag{2}$$

where:

- $B = 0.91$ is an empirical constant valid for COCOMO II
- SF_i is the i-th scale factor

When taking into account the previously identified values of the scale factors, we get:

$$E = 0.91 + 0.01 * 19.11 = 1.1011 \tag{3}$$

thus obtaining a value of 1.1011.

2.2.4 Computation of the estimated project effort

We now apply the equation shown in section 2.2.1 twice to get both a lower bound and an upper bound estimate of the project effort.

$$Effort_{lower\,bound} = 2.94 * 5.2^{1.1011} * 1.01477 = 18.3276 \tag{4}$$

$$Effort_{upper\,bound} = 2.94 * 7.28^{1.1011} * 1.01477 = 26.5465 \tag{5}$$

We get a lower bound of 18.32 person-months and an upper bound of 26.55 person-months.

2.2.5 Computation of the estimated project duration

We estimate the duration of the project with the following equation:

$$Duration = 3.67 * Effort^{0.28+0.2*(E-0.91)} \tag{6}$$

where:

- $E = 1.1011$ is the scale factors aggregation identified earlier in section 2.2.3
- $Effort$ is the effort in person-months, computed in section 2.2.4

Once again, we compute the estimate twice for a lower and upper bound.

$$Duration_{lower\,bound} = 3.67 * 18.32^{0.28+0.2*(1.1011-0.91)} = 9.3 \tag{7}$$

$$Duration_{upper\,bound} = 3.67 * 26.55^{0.28+0.2*(1.1011-0.91)} = 10.4 \tag{8}$$

We therefore get an estimate of between 9.3 and 10.4 months.

3 Project plan

3.1 Identifying milestones and tasks

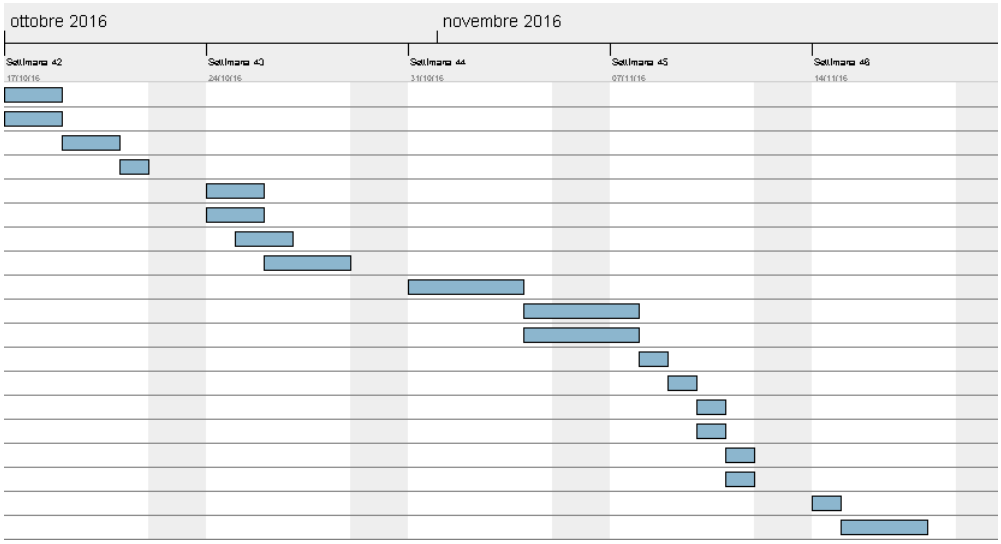
This section aims to point out some of the main concepts and guide lines from which tasks and activities that will be showed afterwards were born. References to the DD and RASD documents are made respecting the dead lines of each document, in the various Gantt there are several tasks to complete which directly or indirectly are associated with the creation of the two documents and the deals made with the set of Stakeholders. Furthermore, Stakeholders comprehend at least one person to represent all the actors exposed in the RASD and, in particular, a possible User, some Finantial Partners and Sponsors, a possible Government mediator, a possible member of the company of the Transactional System, a possible Car society's member, a possible ISP's member. Accordingly with all these people, will be possible to design "legally" and with a full integration with the external environment. Another topic to discuss about is surely the importance of costs: they are highlighted in the middle of some phases of the Gantt, but a cost monitoring is always necessary and other strategies have to be formulated in particular cases (also in relation of the budget coming from the Partners and the Sponsors). At the end of the Development phase (in which there is the "concrete" implementation and testing of the three parts of the Software of our System), there is the Deployment phase: it is the range of time necessary for prepearing the entire System to be launched on the market. Finally the term Maintenance hints the necessity to continue to work on the implementation in periodic steps after the Start up phase too. Precise steps of revisioning of the Project and of the code behind it will be fixed following a precise scheme, as well as, the first feedbacks released by the Users. Maybe there will be some bugs to fix and that were not identified in the testing range of time, or maybe it will be indispensable an updated version of the Software with some new interactive functionalities for the Users. In any case the approach to develop the System will be also considering the future and the new tendences.

3.2 Schedule

In this section the Gantt files related to the principal phases of the Project will be emphasised. An assumption on these schemes are that they are designed in an abstract way, as long as their purpose is purely to give a first sketch of what will be the project itself to the reader. Internal blocks, associated to the specific activities, will be analized afterwards and eventually modified during the particular task of reference.

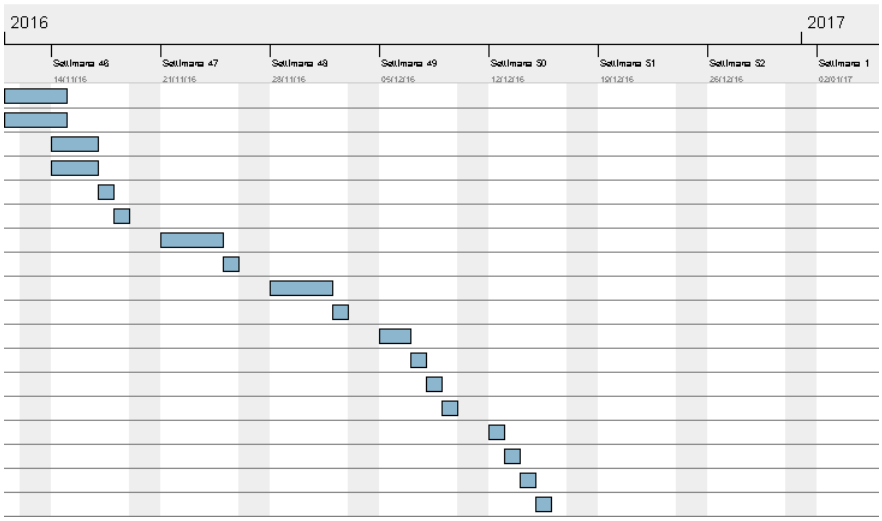
- **RASD document Gantt Diagram**

Nome	Data d'inizio	Data di fine
Identification of goals and requirements and detailed scheme	17/10/16	18/10/16
Identification of properties and domain assumptions	17/10/16	18/10/16
Identification of the constraints	19/10/16	20/10/16
Identification of the main cases of analysis	21/10/16	21/10/16
Sequence Diagrams: first analysis	24/10/16	25/10/16
Use Cases: first analysis	24/10/16	25/10/16
Mockups for UI	25/10/16	26/10/16
Researching and selecting stakeholders	26/10/16	28/10/16
Meeting with each of the stakeholder	31/10/16	03/11/16
Refining goals and requirements	04/11/16	07/11/16
Redefining the constraints	04/11/16	07/11/16
Final meeting with the stakeholders (final presentation of the RASD)	08/11/16	08/11/16
Clearifing convergences among the various stakeholders	09/11/16	09/11/16
Identifying responsabilities, tasks and roles in a first abstract way	10/11/16	10/11/16
Integrating and scheduling of the tasks	10/11/16	10/11/16
First Cost estimation	11/11/16	11/11/16
Further refinements	11/11/16	11/11/16
Final meeting with all the stakeholders	14/11/16	14/11/16
Final Refinements	15/11/16	17/11/16



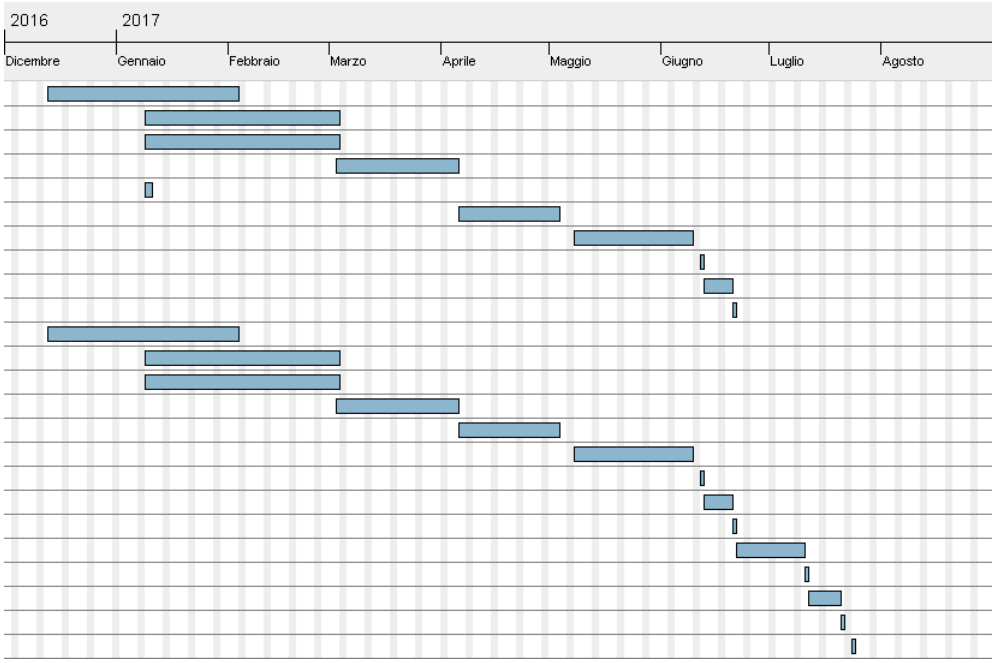
- **DD document Gantt Diagram**

Nome	Data d'inizio	Data di fine
First view of the architecture w.r....	11/11/16	14/11/16
Identifying the components of t...	11/11/16	14/11/16
Identifying the nature of the link...	14/11/16	16/11/16
Integration of the various comp...	14/11/16	16/11/16
First versions of the UI Mockups ...	17/11/16	17/11/16
First meeting with all the stakeh...	18/11/16	18/11/16
Technical meeting with each of t...	21/11/16	24/11/16
Revision of components w.r.t. th...	25/11/16	25/11/16
Subcomponents analysis and de...	28/11/16	01/12/16
Revision of the links and the arc...	02/12/16	02/12/16
Final version of the entire archite...	05/12/16	06/12/16
Further, more detailed runtime e...	07/12/16	07/12/16
Component Interfaces and UI M...	08/12/16	08/12/16
Second meeting with all the stak...	09/12/16	09/12/16
Crearifying dubts/Final guide lin...	12/12/16	12/12/16
Allocation of Costs	13/12/16	13/12/16
Final refinements	14/12/16	14/12/16
Final version of the Architecture/...	15/12/16	15/12/16



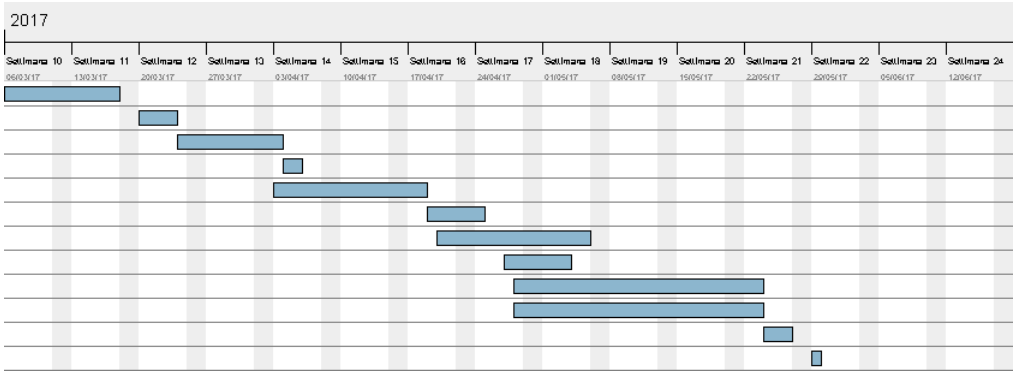
- **Development Gantt Diagram**

Nome	Data d'inizio	Data di fine
Subcomponent implementation (server)	13/12/16	05/02/17
White box tests for the subcomponents	09/01/17	05/03/17
Eventual refinements of code and structures	09/01/17	05/03/17
Black box tests for the subcomponents	03/03/17	05/04/17
Definition of the modalities of integration with the Bottom up analysis	09/01/17	10/01/17
Bottom up integration (step from 1 to 12 as ITPD indications)	06/04/17	03/05/17
System testing	08/05/17	11/06/17
First Internal Simulation (Server side analysis)	12/06/17	12/06/17
Refinements	13/06/17	20/06/17
Second Internal Simulation (Server side analysis)	21/06/17	21/06/17
Subcomponent implementation (Mobile App)	13/12/16	05/02/17
White box tests for the subcomponents	09/01/17	05/03/17
Eventual refinements of code and structures	09/01/17	05/03/17
Black box tests for the subcomponents	03/03/17	05/04/17
Bottom up integration (step from 1 to 12)	06/04/17	03/05/17
System testing	08/05/17	11/06/17
First Internal Simulation (Mobile App analysis)	12/06/17	12/06/17
Refinements	13/06/17	20/06/17
Second Internal Simulation (Mobile App analysis)	21/06/17	21/06/17
Concrete components integration	22/06/17	10/07/17
First Internal Demostration	11/07/17	11/07/17
Refinements/Code Inspections/Correction of Bugs	12/07/17	20/07/17
Second Internal Demostration	21/07/17	21/07/17
Meeting with the Stakeholders: Presentation of the App and Demostrations	24/07/17	24/07/17



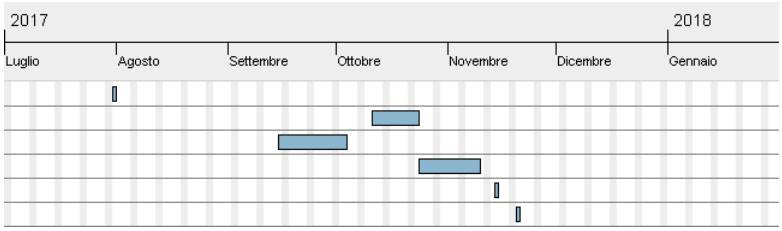
- **Deployment Gantt Diagram**

Nome	Data d'inizio	Data di fine
Projecting the Advertisment Campaign	06/03/17	17/03/17
Selecting the channels and media for the advertisment	20/03/17	23/03/17
Contacting/Dealing with the selected channels and media	24/03/17	03/04/17
Launching the Advertiment Campagn	04/04/17	05/04/17
Infrastructures configurations and normatives	03/04/17	18/04/17
Disposition of the Cars on territory	19/04/17	24/04/17
Configuration of the Power grid stations	20/04/17	05/05/17
Preparing courses	27/04/17	03/05/17
Formative Courses for Operators	28/04/17	23/05/17
Formative Courses for Administrators	28/04/17	23/05/17
Final dispositions and awarenesses	24/05/17	26/05/17
Final meeting with Staff and Stakeholders: decision of the day of start up	29/05/17	29/05/17



- **Start up and Maintenance Gantt Diagram**

Nome	Data d'inizio	Data di fine
First start up	31/07/17	31/07/17
First check from the environment (Periodical plan)	11/10/17	23/10/17
Users experience, real-life simulations	15/09/17	03/10/17
Eventual fix of the first bugs (From the periodical check)	24/10/17	09/11/17
First System updates release	14/11/17	14/11/17
First of the periodical meetings for improving the app	20/11/17	20/11/17



3.3 Resource allocation

In this section the various activities explained in the previous two sub paragraphs will be splitted among the members of the group. This first sketch is an ideal start point from which a division of competences and responsabilities among people is figured out. In the section there are the activities listed, it is

important to notice that some of these activities don't regard only the implementation phase, but also everything that will allow the System to be integrated correctly within the environment as well. Finally, the criteria of allocations of the resources attempts to equally distributes the tasks and with the respect of the topic associated to them. It is clear that every member of the group can have its own team. It is totally up to the division manager the choice of the internal members and the further division of matters and topics contained in the bigger one.

3.3.1 Administrative roles

Making and revisioning the documents	Valeria Mazzola
Organizing the Meetings	Valeria Mazzola
Researching and selecting Stakeholders	Valeria Mazzola
Meeting with each of the Stakeholders	Valeria Mazzola
Presenting the RASD to the Stakeholders	Valeria Mazzola
Clearifing the convergences on the RASD among Stakeholders	Valeria Mazzola
Integrating and scheduling the tasks	Valeria Mazzola
UI Mockups	Valeria Mazzola
Clarifying dubts on the Project	Valeria Mazzola
Allocating Costs	Valeria Mazzola
Bottom up integration	Valeria Mazzola
Internal Simulations	Valeria Mazzola
Internal Demostrations	Valeria Mazzola
Demostrations to Stakeholders	Valeria Mazzola
Revisioning the Software documentation	Valeria Mazzola
Projecting the Advertisment Campaign	Valeria Mazzola
Selecting channels and media	Valeria Mazzola
Contacting/Dealing with the different channels and media	Valeria Mazzola
Launching the Adverstiment Campaign	Valeria Mazzola
Preparing courses	Valeria Mazzola
Final dispositions and awarenesses	Valeria Mazzola
User experience/real-life simulations	Valeria Mazzola
Periodical Meeting for improving the System functionalities	Valeria Mazzola
Infrastructure normatives and configurations	Valeria Mazzola
Technical meetings with each of the interested Stakeholder for the specific analysis of components and subcomponents	Luca Marzi
Scheduling of future activities	Valeria Mazzola

3.3.2 Projectual roles (documentation)

Sequence Diagrams analysis	Federico Nigro
Use Cases analysis	Luca Marzi
Component Interfaces	Federico Nigro
Further, more detailed examples to be integrated with Sequence Diagrams and Use Cases	Federico Nigro
Final Version of the components of the interface	Federico Nigro
Software Documentations	Federico Nigro

3.3.3 Projectual roles (relation with stakeholders and consequences)

Technical meetings with each of the interested Stakeholder for the specific analysis of components and subcomponents	Luca Marzi
Revision of components wrt the technical meetings	Luca Marzi

3.3.4 Projectual roles (common activities of design and modeling)

Indentifying goals and requirements and making the detailed scheme	Luca Marzi, Valeria Mazzola, Federico Nigro
Identifying the properties and the domain assumptions	Luca Marzi, Valeria Mazzola, Federico Nigro
Identifying the constraints	Luca Marzi, Valeria Mazzola, Federico Nigro
Indentifying the main cases of analysis	Luca Marzi, Valeria Mazzola, Federico Nigro
Eventual refinement of goals and requirements	Luca Marzi, Valeria Mazzola, Federico Nigro
Eventual re-definition of constraints	Luca Marzi, Valeria Mazzola, Federico Nigro
Identifying responsabilities, tasks and roles in a first abstract way	Luca Marzi, Valeria Mazzola, Federico Nigro
Further refinements of the Project	Luca Marzi, Valeria Mazzola, Federico Nigro

Final refinements	Luca Marzi, Valeria Mazzola, Federico Nigro
First view of architecture wrt the final offers from/for each Stakehoder	Luca Marzi, Valeria Mazzola, Federico Nigro
Identifying the components of the architecture	Luca Marzi, Valeria Mazzola, Federico Nigro
Identifying the nature of links	Luca Marzi, Valeria Mazzola
Integration of the various components wrt the nature of links	Luca Marzi, Valeria Mazzola
Eventual revisions of the links	Luca Marzi, Valeria Mazzola
Final version of the entire architecture	Luca Marzi, Valeria Mazzola, Federico Nigro
First check from the environment	Luca Marzi, Federico Nigro
Subcomponents analysis and detailed architecture	Luca Marzi, Valeria Mazzola

3.3.5 Implementative roles (related to the implementation of the Server)

Subcomponents implementation	Luca Marzi
White box testing for each subcomponent	Luca Marzi
Eventual refinement of code and internal structures	Luca Marzi
Eventual fixing of bugs	Luca Marzi
Next updates releases	Luca Marzi

3.3.6 Implementative roles (related to the implementation of the Mobile App and the Car's System)

Subcomponents implementation	Federico Nigro
White box testing for each subcomponent	Federico Nigro
Eventual refinement of code and internal structures	Federico Nigro
Eventual fixing of bugs	Federico Nigro
Next updates releases	Federico Nigro

3.3.7 Implementative roles (integrations)

Defining the concrete modalities of integration	Luca Marzi, Valeria Mazzola
Concrete components integration	Luca Marzi, Valeria Mazzola

3.3.8 Implementative roles (common aspects)

Refinements of code after the integration	Luca Marzi, Valeria Mazzola, Federico Nigro
Refinements of code following the Stakeholders further specifications	Luca Marzi, Valeria Mazzola, Federico Nigro
Refinements of code after the various testing phases	Luca Marzi, Valeria Mazzola, Federico Nigro

3.3.9 Implementative roles (black box testing)

Black box testing of Server subcomponents	Luca Marzi
Black box testing of the Mobile App and Car's System subcomponents	Federico Nigro
System testing (simulated approach for each component)	Luca Marzi, Valeria Mazzola, Federico Nigro
System testing (real approach for each component)	Luca Marzi, Valeria Mazzola, Federico Nigro
System testing (testing the entire System)	Luca Marzi, Valeria Mazzola, Federico Nigro

3.3.10 Additional Aspects (single and common aspects)

Releases	Luca Marzi, Valeria Mazzola, Federico Nigro
Formative courses for the Operators	Valeria Mazzola
Formative courses for the Administrators	Luca Marzi
Disposition of the Cars on the territory	Luca Marzi
Configuration of the power grid stations	Federico Nigro

4 Risk management

4.1 Identifying risks

4.1.1 Business risks

One market-related risk that may arise after the project implementation is that the PowerEnJoy system may not be competitive enough, expecially considering the ever increasing number of similar car sharing services now present on the market, particularly in big cities. Although this scenario can potentially lead to a great loss of investment, we believe the probability of it actually becoming reality are not so high, especially considering this is, up until now, the only service offering electric cars.

4.1.2 Implementation phase behind schedule

One possible risk of technical nature is the potential overestimation of our own programming skills, as well as the quality of our design. This would result in a significant delay on the schedule we have defined, which would translate to a great loss of stakeholders' investment.

This risk cannot be understated, and could possibly have a great impact on the project schedule. We should therefore create a contingency plan in case our skills prove to not be enough for the challenge the project offers.

4.1.3 Loss of application data

A risk that must not be underestimated is the possibility of losing core application data due to hardware failures or human errors. This risk can be avoided by setting up a proper backup and recovery system, possibly with the help of external professionals, and it will be a priority after deployment.

4.1.4 User data leaks

In case a security exploitation happens in the system's servers after deployment, there is a risk that user data may be leaked as the result. We believe the chance of this happening are not to be underestimated, however we have designed the system in such a way that the data leaked will not be disastrous for the users.

First, no passwords will be leaked, as they will be stored in the database using proper hashing and salting techiques. Second, no credit card information can be leaked, as the system is designed to only use PayPal as payment method. Therefore, the most private information that can leak from our databases will be the relation between user's names, email addresses and driving license ID, which is not in itself secret information.

4.2 Contingency plan development

4.2.1 Contingency plan for scenario described in section 4.1.2

If our programming skills turn out to have been overestimated, we should fall back to hiring professional programmers with knowledge and experience of the J2E framework.