Politecnico di Milano

PowerEnjoy Service - Project Plan Document

January 19, 2017

Version 1.1

Authors:

- Domenico FAVARO (Mat. 837995)
- Matheus FIM (Mat. 876069)
- Caio ZULIANI (Mat. 877266)

Prof. Elisabetta DI NITTO

Contents

1 Introduction

1.1 Revision History

This section records all revisions to the Document.

Version	Date	Authors	Summary
1.1	22/01/16	Domenico Favaro, Caio Zuliani, Matheus Fim	Initial Release

1.2 Purpose and Scope

The Project Plan Document (PPD) is fundamental to organize the production of the System. The main purpose of the Document is to study the complexity of the System, analize the difficulties that can be present in any stage of its development and deployment and give an overall estimation of the effort and cost that can be needed up to the final release of the product. The analysis will be done in terms of size, time and budget needed. This is fundamental to present our stakeholders a preview before the start of production in case they need to do any project changing decision before hand.
The second part of the Document helps the organization of our production by giving time estimations for each step and managing the workload of our resources accordingly. This serves our production team to make any project management decision but also our development team to know how they can organize themselves to follow the schedule and if by any reason the schedule can't be followed, they can notify the production. This is fundamental to keep track of the project developing at every stage of production.
Finally we present possible risks that can arise during the development and for each risk we have identified a contingency plan to avoid or minimize the possibility of harming the production.

1.3 Definitions and Abbreviations

- **RASD:** Reqirements And Specifications Document.
- **DD:** Design Document.
- **ITPD:** Integration Test Plan Document.
- **PPD:** Project Plan Document
- **App:** Application, refering to Web or Mobile App.
- **LOC:** Lines of Code, is a software metric used to measure the size of a computer program by counting the number of lines in the text of the program's source code.

- **FSM:** Functional Size Measurement, technique for measuring software in terms of the functionality it delivers.
- **FP:** Function Points. Unit of measurement used to express the amount of functionality for an information system. We'll be using them to compute a FSM of software.
- **VAF:** Value Adjustment Factor, used to adjust the Function Points estimation to the context of the project.
- **UFP:** Unadjusted Function Points, raw measurement of the total FP before the VAF is applied.
- **ILF:** Internal Logic Files, FP Function Type.
- **EIF:** External Interface Files, FP Function Type.
- **EI:** External Input, FP Function Type.
- **EO:** External Output, FP Function Type.
- **EQ:** External Inquiries, FP Function Type.
- **COCOMO:** Constructive Cost Model is a procedural software cost estimation model that we'll be using for this document.

For other concepts concerning the Service definition look in the **Glossary** section of the RASD, DD and ITPD.

1.4 Reference Documents

- Specification Document: Assignments AA 2016-2017.pdf
- PowerEnjoy Requirements And Specifications Document (RASD)
- PowerEnjoy Design Document (DD)
- PowerEnjoy Integration Test Plan Document (ITPD)
- Example Document - Project planning example document.pdf
- Project Planning Tools Documentation:
  - Function Points Table
    - `http://www.qsm.com/resources/function-point-languages-table`
  - COCOMO II - Manual
    - COCOMO Model Definition Manual
  - Project Management Software: Jira/Trello
    - `https://www.atlassian.com/software/jira`
    - `https://trello.com/`

2 Project Size, Cost and Effort Estimation

This section deals with the estimation of the expected size, cost and effort needed to develop the PowerEnjoy Service. For the size estimation we'll use the Function Points approach, once calculated we'll have an approximated size in LOC, and this measurement will be used later with COCOMO to predict the cost and amount of effort that will be required to develop the program.

2.1 Size Estimation: Function Points

The Function Mode approach is based in giving to each function needed in the code of the program a Category and a Complexity, once this is done, using Function Point Tables, a number of UFP is given to them. The VAF is then applied to the total of UFP. After the Total FP are calculated, it can then be translated into LOC. The Categories for FP are 5, and each has its own rating table for complexity according to the number of Data Elements present:

- Internal Logic Files (ILF):

  Translate to the sets of data used and managed by the application.

- External Interface Files (EIF):

  Similar to ILF set of data used by the application but generated and maintained by other applications.

  Their complexity table is the same as ILF.

Record Types	**1-5**	**6-19**	**> 19**
0-1	Low	Low	Avg
2-3	Low	Avg	High
> 3	Avg	High	High

- External Inputs (EI):

  Defined as operations to receive and manage data coming from the external environment.

File Types	**1-4**	**5-15**	**>15**
0-1	Low	Low	Avg
2	Low	Avg	High
> 2	Avg	High	High

- External Outputs (EO):

  They are elementary operations that generates data for the external environment. It usually includes managing data from logic files

- External Inquiries (EQ):

  Operation that involves both input and output, managing the data themselves without significant intervation from logic files.

  Their complexity table is the same as EO.

File Types	**1-5**	**6-19**	**>19**
0-1	Low	Low	Avg
2-3	Low	Avg	High
> 3	Avg	High	High

To measure the complexity in FP, weights are given to each Function Type using the following table:

Function Type	**Low**	**Avg**	**High**
ILF	7	10	15
EIF	5	7	10
EI	3	4	6
EO	4	5	7
EQ	3	4	6

2.1.1 Internal Logic Files (ILF)

The PowerEnjoy System is organized to store a number of information to provide its functionalities. This information corresponds to the Entities that compose the System and will be manage by the Entity Beans in the Controllers *(see DD)*. The ILF will then correspond to managing the information of these entities, which are:

- Cars: Cars will be stored in the system DB and replicated by persistent entities in the JEE2 environment for managing their status. Their information include CarID, Location, Status, BatteryStatus, if it's PluggedIn and Number of Passengers.

  Car element count is 6 Fields.

- Users: Users tuples will be added by the System in the DB as well for the Registration feature, their attributes include UserID, if it's LoggedIn, UserName, Password, Email, DriverLicense, PaymentInfo and Location.

  User count is 8 Fields.

- CRM: Although not directly added by the System, the loggedIn status of the CRMs is important for the Chat feature. It stores their CRMID, Password, Name, Email and if it's LoggedIn.

  CRM count is 5 Fields.

- Reservations: Will link Users with Cars and will have a status to know wether the reservation is active or not. Their attributes are ReservationID, UserID, CarID, Status, ReservationTime, PickUp-Time.

  Reservation elements count is 6 and references 2 Record Types.

- Rides: Once the Reservation has been confirmed a Ride is created, it refers to the Reservation and it tracks the use of the Car. It has RideID, ReservationID, CarID, Status, StartTime, EndTime, LocationPath.

  Ride count is 7 and references 2 Record Types.

- Payments: When the Ride is finished it generates the Payment for the User to pay, it tracks the input made by the Car, Rides and Users and include also eventual Discounts and Extra Fees. It has PaymentID, RideID, TotalFee, PassengerDiscount, BatteryDiscount, PluggedInDiscount, LowBatteryFee, NoPlugNearFee.

  Payment has 8 fields, references 1 Record Type.

- User Reports: Tracks any fault in the Service by Users or Cars and are handled by CRMs. They have the following fields: UserReportID, UserID, CRMID, CarID, Description.

  User Report has 5 fields, references 3 Record Types.

- Locations: Although not managed by the System, the record of safe parking and plug in areas has to be kept and accessed by the LocationHelper. They have basic Info like LocationID, Latitude, Longitude, Type.

  Locations have 4 fields.

Using the Complexity Table for the ILF we obtain the following.

ILF	**Complexity**	**FP**
Car Data	Low	7
User Data	Low	7
CRM Data	Low	7
Reservations	Avg	10
Rides	Avg	10
Payments	Low	7
User Reports	Low	7
Locations	Low	7

The Total UFP for ILF is **62**.

2.1.2 External Interface Files (EIF)

Our only external data source comes from Google Maps API, that offers the map service. It's used by the Location Helper and by the User and CRM Interface Components. It'll be accessed to retrieve the map interface find directions and show the location of the User and the Cars, as well as to retrieve walking and driving distances. The response is given in JSON and it provides numerous fields, however it doesn't reference any record types so we count only the number of parameters the response gives us.

EIF	**Complexity**	**FP**
API Map Retrieval	Avg	10
Direction Retrieval	Low	7

This gives us that the UFP for EIF are**17**.

2.1.3 External Inputs (EI)

The Sytem relies on many interactions provided for the User to allow them the functionalities of the service, others involve or are shared by the CRM, we'll present all of them and give a complexity estimation for each one. They consist on a function activated by the User Input.

Shared:

- Login: Controls that the parameters are correct and logs in the User. Low Complexity.
- Logout: Signs out the connected User. Low Complexity.
- Reserve Car: Creates a new Reservation. It involves User, Car and Reservation. Complexity Average.

- Cancel Reservation: Cancels already created Reservation, interacts just with User and Reservation. Complexity Low.

Users:

- Register: Creates a new User and inserts his/her data in the system, has many fields but concers only the User. Complexity is Low.
- Confirm Reservation: Unlocks the Car and starts a new Ride. Average Complexity.
- End Ride: Finishes current Ride and creates a new Payment. Complexity is Average.

CRM:

- Create User Report: Links User, CRM and Car in case of any issue. Average Complexity.
- Lock Car: Signals the Car to Lock. Low Complexity.
- Unlock Car: Low Complexity.

EI	**Complexity**	**FP**
Login	Low	3 x2
Logout	Low	3 x2
Register	Low	3
Reserve Car	Avg	4 x2
Cancel Reservation	Low	3 x2
Confirm Reservation	Avg	4
End Ride	Avg	4
Create User Report	Avg	4
Lock Car	Low	3
Unlock Car	Low	3

This gives us a Total UFP for the EI of **47**.

2.1.4 External Outputs (EO)

PowerEnjoy External Outputs are limited to sending Emails in case of User Registration to send the password, End of Ride to send the Payment and Time Limit over the Reservation to notify the User.

- Email Password: Involves just the User, Low Complexity.
- Email Payment: Sends Payment to User, Low Complexity.
- Send Late Reservation Notification: Notifies status of Reservation to User, Low Complexity.

EO	**Complexity**	**FP**
Notify Password	Low	4
Notify Payment	Low	4
Notify Reservation	Low	4

The Total UFP for the EO is **12**.

2.1.5 External Inquiries (EQ)

Similar to EI but sends a data response instead for the User. Basically a query of our system. PowerEnjoy EQ are:

- Search Nearby Cars: Gives a List of close by avaliable Cars from the User Location. Has Low Complexity.
- Money Saving Option: Calculates the best Re-Charging station according to a standard Deviation Algorithm (*see DD*) and returns it to the User so a discount for the ride can be applied. Is of Average Complexity.
- Find Location: Gives a Path to arrive to the given Location.
- Find Cars: Returns all desired Cars for the CRM.
- Find Users: Returns all Users for the CRM.
- Find Reservations: Returns desired Reservations for the CRM.
- Find Rides: Returns all Rides for the CRM.
- Find Payments: Returns given Payments for the CRM.
- Find User Reports: Returns the User Reports for the CRM.

All the Finding Functions are considered to have Low Complexity.

EQ	**Complexity**	**FP**
Search Nearby Cars	Low	3
Money Saving Option	Avg	4
Find Location	Low	3
Find Cars	Low	3
Find Users	Low	3
Find Reservations	Low	3
Find Rides	Low	3
Find Payments	Low	3
Find User Reports	Low	3

This gives us a Total UFP for the EQ of **28**.

2.1.6 Overall Estimation

The following table shows the estimation results for each Function Type:

Function Type	**Value**
ILF	62
EIF	17
EI	47
EO	12
EQ	28

The added Total of UFP is **166**.

To have an idea of the size of our project in the code we translate the total of UFP to LOC using the Function Points Languages Table. Since we'll develop in the J2EE platform the measures are $Median = 49, High = 67$.

$$LowBoundLOC = 166 * 49 = \mathbf{8134}$$

$$HighBoundLOC = 166 * 67 = \mathbf{11122}$$

We can expect a software ranging from the size of 8200 to 11150 LOC.

2.2 Cost and Effort Estimation: COCOMO II

In the estimation of the cost and effort needed in the development of the PoweEnjoy Service it will be employed the COCOMO II.

2.2.1 Scale Drivers

In the evaluation of the Scale Drivers it was used the following table provided by the COCOMO II approach.

Scale Factors	**Very Low**	**Low**	**Nominal**	**High**	**Ver**
PREC SFj	thoroughly unprecedented 6.20	largely unprecedented 4.96	somewhat unprecedented 3.72	generally familiar 2.48	larg 1.42
FLEX SFj	rigorous 5.07	occasional relaxation 4.05	some relaxation 3.04	general conformity 2.03	som 1.01
RESL SFj	little (20%) 7.07	some (40%) 7.07	often (60%) 4.24	generally (75%) 2.83	mos 1.41
TEAM SFj	very difficult interactions 5.48	some difficult interactions 4.38	basically cooperative interactions 3.29	largely cooperative 2.19	high coop
PMAT SFj	SW-CMM Level 1 Lower 7.80	SW-CMM Level 1 Upper 6.24	SW-CMM Level 2 4.86	SW-CMM Level 3 3.12	SW- 3 1.

Analysing each point that compose the Scale Drivers:

- Precedentedness: It's about if the team/organization has a previous experience with this type of project. As the team don't have a large professional experience the item will be evaluate as low.
- Development flexibility: It's about the degree of flexibility in the development process. In this project there were several specifications for the system when becomes of functionalities being so the item will be evaluate as low.
- Risk resolution: It's about the risk analysis made. It was carried out several risk evaluations during the differents parts of the project so the item will be evaluated as high.
- Team cohesion: It's the evaluation of how well the development team knows each other and work together. As the team has achieved a excellent level of work since the division of the workload until the respect to the deadlines the item will be evaluated as very high.
- Process maturity: It's about the process maturity of the organization/team that is know as CMM. Considering that it's the team first project of this kind especially involving Java EE the item will be evaluated as Level 2. Moreover it's important to emphasize the proactive of the group even if it's not characteristic of the evaluation of Level 2 of CMM.

Being so the evaluation is:

Scale Driver	Factor	Value
Precedentedness (PREC)	Low	4.96
Risk resolution (RESL)	High	2.83
Team cohesion (TEAM)	Very high	1.10
Process maturity (PMAT)	Level 2	4.68
Total		13.57

2.2.2 Scale Drivers

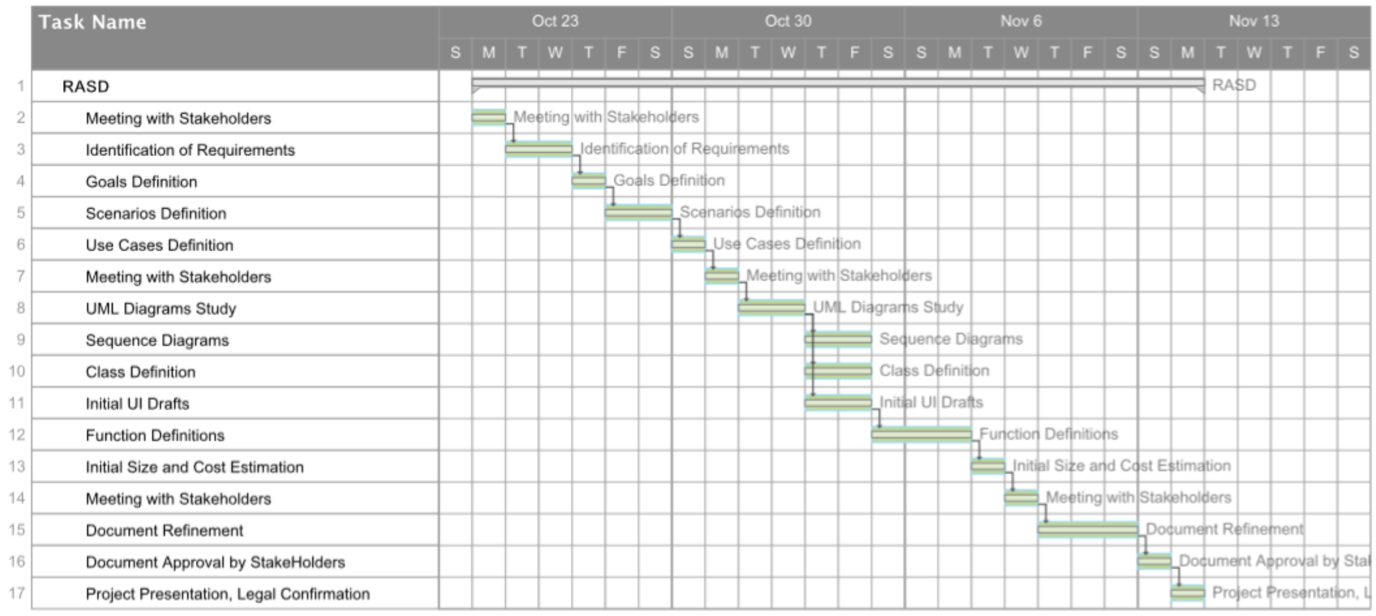
2.2.3 Cost Drivers

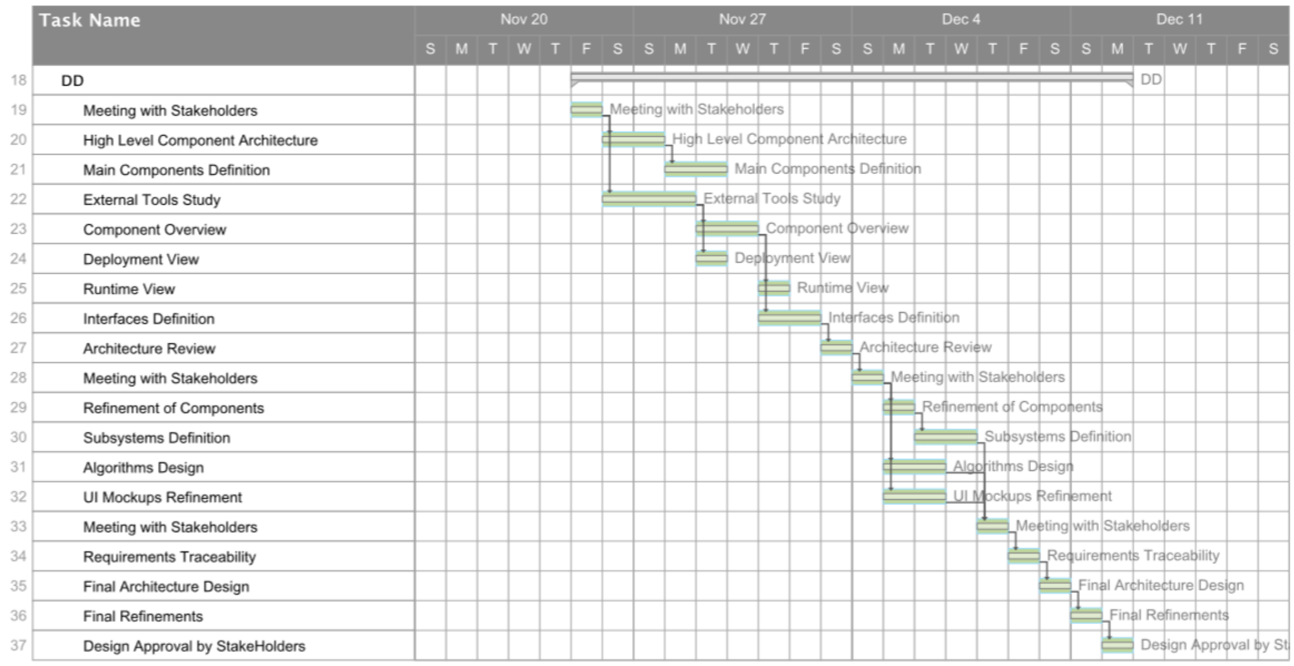
2.2.4 Effor Equation

2.2.5 Schedule Estimation

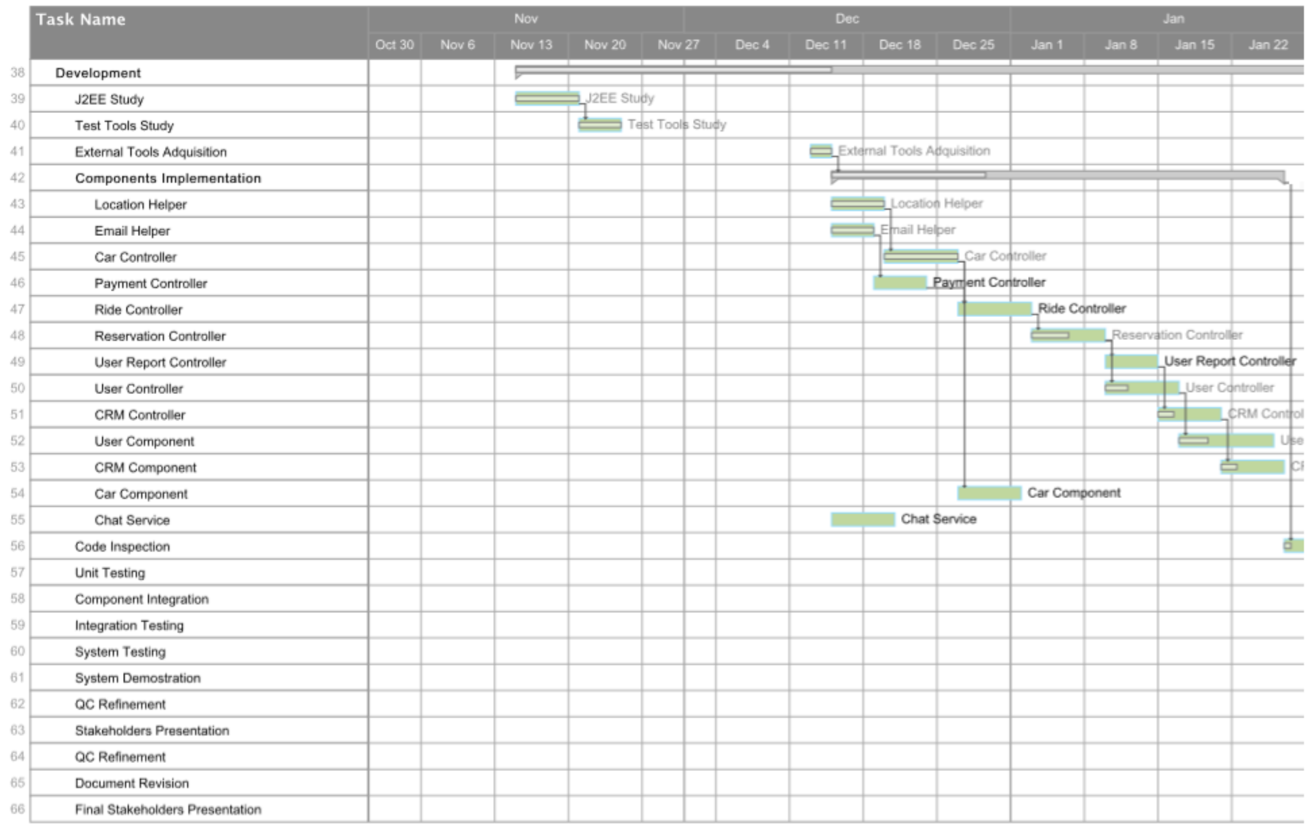
3 Project Plan Schedule

This section illustrate a high level view of the organization of the project schedule, showing the main phases of the production process. Should it be implemented, more detailed schedules for each of the phases and each of the sprints organized inside them will be defined during the project for better internal organization.
Due to the size of the Schedule Graph we separated in 4 different sections, RASD, DD, Development and Deployment.

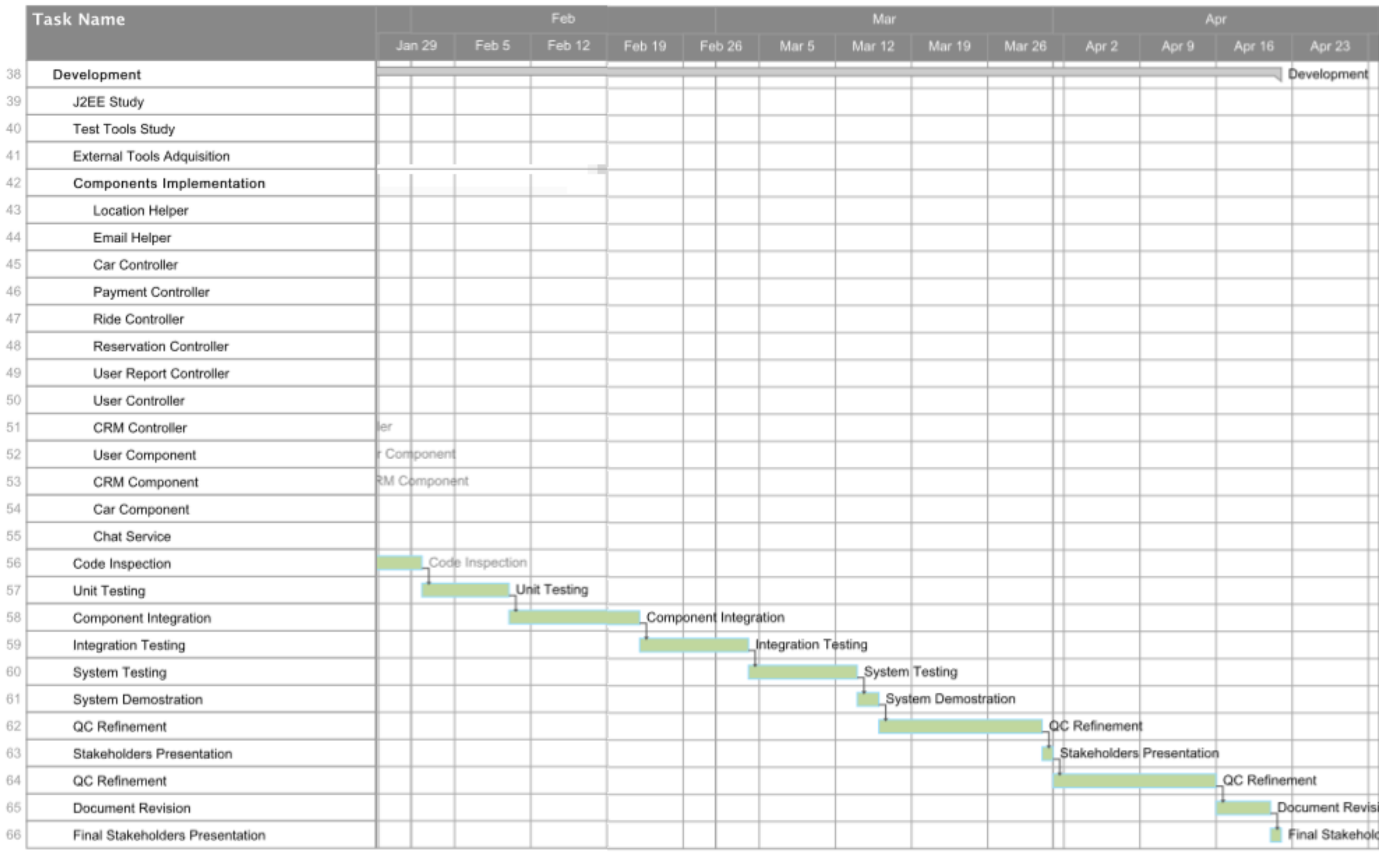




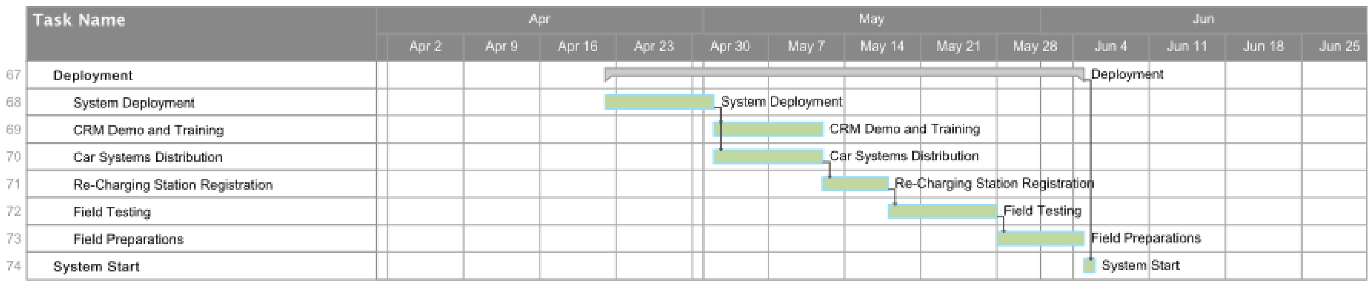
	Task Name
18	DD
19	Meeting with Stakeholders
20	High Level Component Architecture
21	Main Components Definition
22	External Tools Study
23	Component Overview
24	Deployment View
25	Runtime View
26	Interfaces Definition
27	Architecture Review
28	Meeting with Stakeholders
29	Refinement of Components
30	Subsystems Definition
31	Algorithms Design
32	UI Mockups Refinement
33	Meeting with Stakeholders
34	Requirements Traceability
35	Final Architecture Design
36	Final Refinements
37	Design Approval by StakeHolders



	Task Name
38	Development
39	J2EE Study
40	Test Tools Study
41	External Tools Adquisition
42	Components Implementation
43	Location Helper
44	Email Helper
45	Car Controller
46	Payment Controller
47	Ride Controller
48	Reservation Controller
49	User Report Controller
50	User Controller
51	CRM Controller
52	User Component
53	CRM Component
54	Car Component
55	Chat Service
56	Code Inspection
57	Unit Testing
58	Component Integration
59	Integration Testing
60	System Testing
61	System Demostration
62	QC Refinement
63	Stakeholders Presentation
64	QC Refinement
65	Document Revision
66	Final Stakeholders Presentation



	Task Name
38	Development
39	J2EE Study
40	Test Tools Study
41	External Tools Adquisition
42	Components Implementation
43	Location Helper
44	Email Helper
45	Car Controller
46	Payment Controller
47	Ride Controller
48	Reservation Controller
49	User Report Controller
50	User Controller
51	CRM Controller
52	User Component
53	CRM Component
54	Car Component
55	Chat Service
56	Code Inspection
57	Unit Testing
58	Component Integration
59	Integration Testing
60	System Testing
61	System Demostration
62	QC Refinement
63	Stakeholders Presentation
64	QC Refinement
65	Document Revision
66	Final Stakeholders Presentation



	Task Name
67	Deployment
68	System Deployment
69	CRM Demo and Training
70	Car Systems Distribution
71	Re-Charging Station Registration
72	Field Testing
73	Field Preparations
74	System Start

4 Resource Allocation

5 Risk Management

6 Effort Spent

Date	Domenico	Caio	Matheus
15/01/17	2h	2h	2h
16/01/17	-	-	-
17/01/17	4h	-	-
18/01/17	4h	-	-
19/01/17	-	6h	-
20/01/17	-	-	-
21/01/17	-	-	-
22/01/17	-	-	-

7 Changelog

As the project and design decisions may change during the development this document is also prone to change. We'll document every version in this part.

- **Version 1.1:** 22/01/2017