Politecnico di Milano

Software Engineering 2

Project Plan Document

Bressan G., de Santis S., Di Marco P.

POWER ENJOY

Professor Elisabetta Di Nitto

Academic Year 2016-2017

Contents

1 Introduction

1.1 Revision History

Version	Date	Authors	Summary
1.0	15/01/2017	Gabriele Bressan, Simone de Santis, Pietro Di Marco	Initial release

1.2 Purpose and Scope

Project Plan is the fourth part of the Power EnJoy project. During this part of the project we will introduce some definitions dedicated to costs and effort estimation. For calculating this economical parameters we need to use some concepts such Function Points and COCOMO definition. The scope of Project Plan can be divided in two principal parts. The first part we will discuss about two coefficients already introduced and they are Function Point and COCOMO, in particular thanks FP, once identify the functionality provided by Power EnJoy system, we can calculate the weight to implement them. Furthermore COCOMO give as an idea about the dimension of the project in terms of lines of code. During the second part of the Project Plan, using Gantt diagram will be show the schedule for the project and the specific time to execute all activities.

2 Definitions, Acronyms, Abbreviations

- RASD: Requirement Analysis and Specification Document.
- DD: Design Docuement.
- ITBP: Integration test Plan DocumentDBMS: Database Management System.
- DB: Database of Power EnJoy.
- Java EE: Java Enterprise Edition.
- Subsystem: The biggest (high level) part of Power EnJoy sytem that contains one or more component.
- Component: Low level part of Power EnJoy system conteined into Subsistem.
- GM: General Motors.
- Google Maps: Free service that allow to show the map information.

3 Project size, cost and effort estimation

In this section we provide some estimations of the costs, size and effort to develop the Power Enjoy System. The size of our project will be estimated using the function points, in particular we assign different FPs to the main functionalities of our System and finally we obtain the estimation of the correspondent amount of lines of code to be written in Java. At the end we use COCOMO for the estimation of cost and effort.

3.1 Size estimation: function points

The Function Points are associated to the program characteristics, in particular are analyzed: The Data structures; Inputs and outputs; The Inquiries; The External interfaces; Finally the total is computed by multiplying each "raw" count by the weight and summing all partial values. For Internal Logic Files and External Logic Files

	Data Elements		
Record Elements RET	1-19	20-50	Greater or equal than 51
Less than 2	Low	Low	Average
2-5	Low	Average	High
6 or more	Average	High	High

For External Output and External Inquiry

	Data Elements		
Files Type Referenced FTR	1-5	6-19	Greater than 20
Less than 2	Low	Low	Average
2-3	Low	Average	High
Greater or equal 4	Average	High	High

For External Input

	Data Elements		
Files Type Referenced FTR	1-3	5-15	Greater or equal than 16
Less than 2	Low	Low	Average
2-3	Low	Average	High
Greater or equal 4	Average	High	High

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

3.1.1 Internal Logic Files (ILFs)

Internal Logic Files (ILFs) are defined as user-identifiable group of logically related data or control information maintained within the boundary of the application. The primary intent of an ILF is to hold data maintained through one or more elementary processes of the application being counted. In our application we identified different Internal Logic Files related to different functionalities of the System. In particular the system has to store information about user, vehicle, history and reservation. First of all the user information that the system needs to store and that the user can change are: mail, password, licence, name, surname, fiscal code, bithday identification card, country, region, city address and sex. These data are very important to store, because in case of problem will be easy to identify the responsible. User data can be modify and update directly from the user using both mobile application and web application. History Data are updated every time the user perform a ride, they are maintained and handled by the system and the user can show it when he click on appropriate button into mobile application and web application. The user can also manager his history deleting a specified trip history. Also vehicle data are stored in a dedicated table of database in which contains all information about a identified car such as: licence plate, car status and vehicle position. Furthermore the vehicle data are took from General Motors database using dedicated APIs.

Reservation data are required to store all the request of reservation by the Mobile App or Web App, these data are permanently memorized in database. Just at the time of creation, User, Vehicle and Time are setted correctly, this let us keep track of the requesting information. In case of time for reservation is expired, time exceeded value will be setted true.

Position Data are relevant information that able the system to know every time all the position of the users, that are directly given by Mobile App. Ride Data are the information stored in order to take track of the all the important attributes of the Ride. Park Data are only the data used to describe the park around the city.

ILF	Complexity	FPs
User Data	Low	7
Reservation Data	Average	10
Vehicle Data	Average	10
History Data	Low	7
Position Data	Average	10
Ride Data	Average	10
Park Data	Low	7
Total		61

3.1.2 External logic Files (ELFs)

In Power EnJoy system there are three different external logic file and they are:

- **General Motors Data:** This external logic file is stored inside General Motors system and consists in the core of vehicle management because thanks to APIs released by General Motors and executed by Power EnJoy system, in particular from vehicle manager, is possible to control each information about car thanks to OnStar hardware system installed into the car. In other word, inside the General Motors system is contained one external logic file, that contains all information about the interaction with Power EnJoy system such as all information about EnJoy vehicles.

- **Map Service Data:** Thanks to this external logic file is possible to store the information that came from a requests made by communication interface to Map service external component so de communication interface (user smartphone, user PC and iPad system) send their position to Map service and it respond with a JSON string (easy to pars) that contains the correct position of vehicle/s, parks and power stations on map. This is not the only functionality of Map service because it is used also to give information about navigation system from one start position and one end position.

- **External entity payment Data:** This is an external logic file contained inside an External entity payment and this external logic file contains specific information about the money transaction made by user to Power EnJoy system at the end of the ride. The payment is compute automatically by the system (the EnJoy system know credit card information) when the user use "park car" modality, so our system sent the payment information (credit card number, credit card expiration date, and credit card CCV code) to this external entity take the information, execute the transaction and save the information and the transaction status into external logic file.

ELF	Complexity	FPs
General Motors Data	Low	10
Map service Data	Low	10
External entity payment Data	Average	10
Total		30

3.1.3 External Inputs (EIs)

Power EnJoy system has got external inputs that came from the interaction with users and the system. The external inputs are:

Users

- **Login:** This operation involves all users that logs into the System. This is a simple operation and its contribution is 3FPs
- **Logut:** Also this is a simple operation and it complexity is 3FPs.
- **Modify personal information:** This operation require a set of checks on field during filling the form, so for this reason has an average complexity with 4FPs.
- **Create new account:** This operation require a set of checks on field during filling the form, so for this reason has an average complexity with 4FPs.
- **Delete account:** This is a not simple operation so for this reason has a high complexity with 6FPs.
- **Password recovery:** This operation requires the interaction between some components and send an email to user. Its has an average complexity with 4FPs.
- **Reserve a car:** The reservation car operation requires the interaction with different components. It has an average complexity with 6FPs.
- **Delete reservation of car:** the delete reservation is a simple operation and for this reason its complexity is low with 3FPs.
- **Insert, update and delete Park and Power Station:** These operation are made by system administrator when he want to insert, update and delete a park or power station. These operations are complex because works on database so their complexity is high with 6 FPS.
- **Insert, update and delete Vehicle:**These operations are performed by the Administrator when he need to insert update and delete a vehicle,These operations are complex because works on database so their complexity is high with 6FPS.

External Input	Complexity	FPs
Login	Low	3
Logout	Low	3
Password Recovery	Average	4
Create new account:	high	6
Insert park and power station	high	2x6
Update park and power station	high	2x6
Delete park and power station	high	2x6
Insert vehicle	high	6
Delete vehicle	high	6
Update vehicle	high	6
Delete account:	high	6
Total		85

3.1.4 External Inquiries

External Inquiries are elementary process with both input and output components that result in data retrieval from one or more internal logical files and external interface files. The input process does not update or maintain any FTRs (Internal Logical Files or External Interface Files) and the output side does not contain derived data. In our System we identified these EQ:

- **Show history trip:**The User performs this action to retrieve his History trip.
- **Request all Available cars in nearby position:** The User wants reserve a vehicle and he needs to get information about the nearby vehicles and their details.
- **Request information about battery level:**The User wants reserve a vehicle and he needs to get information about the nearby vehicles and their details.
- **Request information about distance:**The User wants reserve a vehicle and he needs to get information about the nearby vehicles and their details.
- **Request information about guests number:** The iPad App shows the detected guests number as a recap of the current ride.
- **Request information about currently charge:** The iPad system show the current charge.
- **Request information about vehicle status:** The Mobile App and Web App shows both the information about status of retrieved vehicle.

EQ	Complexity	FPs
Show history trip	Low	3
Request all Available cars in nearby position	Low	3
Request information about battery level	Low	3
Request information about distance	Low	3
Request information about guests number:	Low	3
Request information about currently charge:	Low	3
Request information about vehicle status:	Low	3
Total		21

3.1.5 External Outputs (EOs)

In the external output are described all notification that have to show to the user in Mobile Application, Web Application and iPad System.

- Notify **Payment result** and all summary information pertinent to the Reservation.
- Notify a **Reservation** that has been done correctly.
- Notify a **PinCode** that enable user to turn on the vehicle.
- Notify **Time Exceeded** for reservation.

External Output	Complexity	FPs
Payment result	Low	4
Reservation done	Low	4
PinCode	Low	4
Time Exceeded	Low	4
Total		16

3.1.6 Overall estimation

Down here is reported the table describing all the results of the estimation activity.

Function Type	Value
Internal Logic Files	61
External Logic Files	30
External Inputs	85
External Inquiries	21
External Outputs	16
Total	213

Lower Bound
SLOC:213*46=9798
Upper Bound
SLOC:213*67=14271

3.2 Cost and effort estimation: COCOMO II

3.2.1 Scale Drivers

Scale Factors	Very Low	Low	Nomianl	High	Very High	Extra High
Prec SFj	thoroughly unprecedented 6.20	largely unprecedented 4.96	somewhat unprecedented 3.72	generally familiar 2.48	largely familiar 1.24	thoroughly familiar 0.00
FLEX SFj	rigorous 5.07	occasional relaxation 4.05	some relaxation 3.04	general conformity 2.03	some conformity 1.01	general goals 0.00
RESL SFj	little (20 7.07	some (40 5.65	often (604.24	generally (752.83	mostly (901.41	full(1000.00
TEAM SFj	very difficult interactions 5.48	some difficult interactions 4.38	basically cooperative ineractions 3.29	largely cooperative 2.19	highly cooperative 1.10	seamless interactions 0.00
PMAT SFj	Level 1 Lower 7.80	Level 1 Upper 6.24	Level 2 4.68	Level 3 3.12	Level 4 1.56	Level 5 0.00

Description of scale factor

- Precedentedness: This parameter is used to indicate the past experiences to develop software in a large scale projects. In our case this indicator is **low** because we don't have much experience.
- Development flexibility: this is the level of flexibility during the development period beside external specification and requirements. This value will be **low** because is used a prescribed process.

- Risk resolution:Reflects the extent of risk analysis carried out. This value will be set to **high**.
- Team cohesion: This parameter is used to indicate how our team can collaborate and work together in a cooperative way. In our case this indicator assume **high** value.
- Process maturity: In the initial part of the project (RASD) we spend a lot of time to discuss about project settings and we didn't write any document. After a correctness evaluation of requirements we achieved the goals specified in the RASD document and all functionalities required work.

Scale Driver	Factor	Value
Precedentedness (PREC)	Low	4.96
Development flexibility (FLEX)	Low	4.05
Risk resolution (RESL)	High	2.83
Team cohesion (TEAM)	High	2.19
Process maturity (PMAT)	Level 3	3.12
Total		17.17

3.2.2 Cost Drivers

- Required Software Reliability:

  The RELY cost Driver is set to high because our business is completely dependent on our system and a fault in the system could cause important financial losses.

RELY Cost Drivers						
RELY Descriptors	slightly inconvenience	easily recoverable losses	moderate recoverable losses	high financial loss	risk to human life	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	0.82	0.92	1.00	1.10	1.26	

- Database size: This measure considers the effective size of our database. Our estimates are resulting in the DATA cost driver being high, this is due to the

ratio D/P we assume in fact that the size of the database would be of 5GB and the SLOC.

DATA Cost Drivers						
DATA Descriptors		D/P¡10	$10 \leq D/P \leq 100$	$100 \leq D/P \leq 1000$	D/P>1000	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers		0.90	1.00	1.14	1.28	

- Product complexity

  According to the value obtained from COCOMO II, complexity is setted to high.

CPLX Cost Driver						
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	0.73	0.87	1.00	1.17	1.34	1.74

- Required reusability:

  In our case the developing process is limited to the project so the value is nominal

RUSE Cost Driver						
RUSE Descriptors		None	Across project	Across program	Across product line	Across miltiple product lines
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers		0.95	1.00	1.07	1.15	1.24

- Documentation match to life-cycle needs:

  In this case, in the documentation, all life-cycle needs are covered in a right sized way, this means that parameter is setted to nominal value.

DOCU Cost Driver						
DOCU Descriptors	Many life-cycle needs uncovered	Some life-cycle needs uncovered	Right-sized to life-cycle needs.	Excessive for life-cycle needs	very excessive for life-cycle needs	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	0.81	0.91	1.00	1.11	1.23	n/a

- Execution time constraint This parameter describe the CPU usage that our system requires, in this case because the Power Enjoy is quite complex the value is set to very-high

TIME Cost Driver						
TIME Descriptors			¡= 50% use of available execution time	70% use of available execution time	85% use of available execution time	95% use of available execution time
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	n/a	n/a	1.00	1.11	1.29	1.63

- Storage constraint

  This rating describes the level of storage constraint enforced on software system. This value is setted to nominal, data will use less than 50% of available storage.

STOR Cost Driver						
STOR Descriptors			¡= 50% use of available execution time	70% use of available execution time	85% use of available execution time	95% use of available execution time
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	n/a	n/a	1.00	1.05	1.17	1.46

- Platform Volatility

  This parameter measures the volatility of the application, in our case we do not expect that the program will change very often anyway the App and Web App should be surely updated periodically, that's why the value is set to nominal

PVOL Cost Driver						
PVOL Descriptors		Major change every 12 mo., minor change every 1 mo.	Major: 6mo; minor: 2wk.	Major: 2mo, minor:1wk	Major: 2wk; minor 2days	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	n/a	0.87	1.00	1.15	1.30	n/a

- Analist Capability:

  The analysis work has been done thinking of a real application domain, system is intended to work well in a real world. According to this, value is setted to high.

ACAP Cost Driver						
ACAP Descriptors	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.42	1.19	1.00	0.85	0.71	1

- Programmer Capability:

  We have not implemented the project yet, moreover considering that we do not have much experience as programmers we'll set this parameter to nominal

PCAP Cost Driver						
ACAP Descriptors	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.34	1.15	1.00	0.88	0.76	n/a

- Application Experience We have some experience in the development of Java applications, but we never tackled a Java EE system of this kind. For this reason we're going to set this parameter to low.

APEX Cost Driver						
APEX Descriptors	¡=2months	6 months	1year	3years	6years	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.22	1.10	1.00	0.88	0.81	n/a

- Platform Experience

  We don't have any experience with the Java EE platform,so we'll set this parameter to nominal.

APEX Cost Driver						
APEX Descriptors	$\leq$ 2*months*	6 months	1year	3years	6years	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.19	1.09	1.00	0.91	0.85	n/a

- Language and Tool Experience:

  Our developer group has no past experience with Java Enterprise Edition platform, but there are many skills in Java programming language, relational databases so the value is nominal.

LTEX Cost Driver						
LTEX Descriptors	$\leq$ 2*months*	6 months	1year	3years	6years	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.20	1.09	1.00	0.91	0.84	n/a

- Personnel continuity This parameter is critical for us because we have to deliver this project soon, so the parameter will be set to low.

PCON Cost Driver						
PCON Descriptors	48year	24year	12year	6year	3year	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.29	1.12	1.00	0.90	0.81	n/a

- Usage of Software Tools:

  We don't have any experience with the Java EE platform, so we'll set this parameter to low

TOOL Cost Driver						
TOOL Descriptors	edit, code, debug	simple, frontend, backend CASE, little integration	basic lifecycle tools, moderately integrated	strong, mature lifecycle tools, moderately integrated	strong, mature, proactive lifecycle tools, well integrated with processes, methods, reuse	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.17	1.09	1.00	0.90	0.78	n/a

- Multisite development

  Multisite development is a parameter to be taken carefully. In our case, we have done many different remote collaboration, and there is a efficient communication support, also site collocation are close, so this let us estimate and set parameter to very high.

SITE Cost Driver						
SITE Collocation Descriptors	International Some pgone, mail	Multicity and multicompany Individual phone, fax	Multicity or multicompany Narrow band email	Same city or metro area Wideband electronic communication	Same building or complex Wideband elect. comm., occasional video conf.	Fully collocated Interactive multimedia
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.22	1.09	1.00	0.93	0.86	0.80

- Required development schedule: the definition of all the required documentation took a consistent amount of time, especially for the requirement analysis and the design phases. For this reason, this parameter is set to high.

SCED Cost Driver						
SCED Descriptors	75% of nominal	85% of nominal	100% of nominal	130% of nominal	160% of nominal	
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.43	1.14	1.00	1.00	1.00	n/a

Cost Driver	Factor	Value
Required Software Reliability (RELY)	High	1.10
Database size (DATA)	High	1.14
Product complexity (CPLX)	High	1.17
Required Reusability (RUSE)	Nominal	1.00
Documentation match to life-cycle needs (DOCU)	Nominal	1.00
Execution Time Constraint (TIME)	very high	1.29
Main storage constraint (STOR)	Nominal	1.00
Platform volatility (PVOL)	Nominal	1.00
Analyst capability (ACAP)	high	0.85
Programmer capability (PCAP)	Low	1.00
Application Experience (APEX)	Low	1.10
Platform Experience (PLEX)	Nominal	1.00
Language and Tool Experience (LTEX)	Nominal	1.00
Personnel continuity (PCON)	Low	1.12
Usage of Software Tools (TOOL)	Low	1.09
Multisite development (SITE)	Very high	0.86
Required development schedule (SCED)	high	1.00
SCED Cost Driver		SCED Cost Driver

3.3 Effort Equation

The following formula estimate the Effort Person-Months (PM):

A=2,94 (COCOMO II)
EAF=1,8579229153 effort adjustment factor
B=0,91
E=0,91+0,01*17,17(sommSF)=1,0817
EFFORT Lower Bound=A*EAF*$KSLOC^{E}$=64 PM
EFFORT Upper Bound=A*EAF*$KSLOC^{E}E$=96 PM
Schedule Estimation
This is the duration equation based on Effort, this is in staff months.
Duration(months)=C x $Effort^{F}$
C=3,67
F=D+0,2*(E-B)
F=0,28+0,2*(1,0817-0,91)=0,31434
D=0,28
Duration(lower)=$3.67*64^{(0,31434)}$=13,56 months
Duration(High)=$3.67*96^{(0,31434)}$=15,4 months

3.4 Tasks and schedule

During this part of the project we will talk about the organization of our project, so this phase have a very big importance respect the cost and times already planned. For doing this part we will provide an high level project schedule using a Gantt diagram that contains all activities of the project documentation at a precise period of time. During tasks and schedule design phase we also considered that the requirement already described in the RASD document, they may change or be changed, or there might be other type problems.

The main task of this project are:

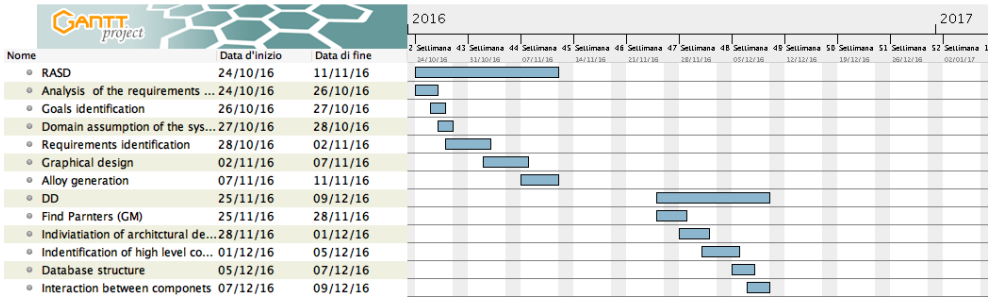
1. **Requirement Analysis and Specification Document:** is an high level part of the project that contains all the goals of the system, domain assumption function requirements and nonfunctional requirements.

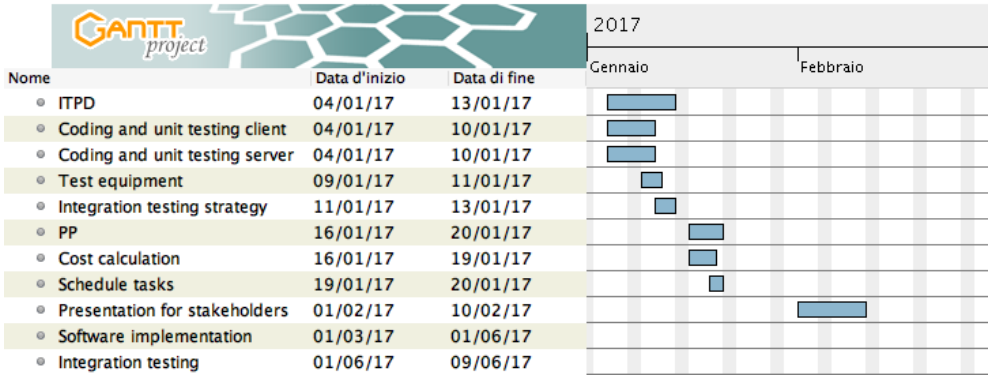
2. **Design Document:** this part of the project that contains information about how Power EnJoy system will be distributed on the hardware parts and explain what are the main software parts and what are their interactions.

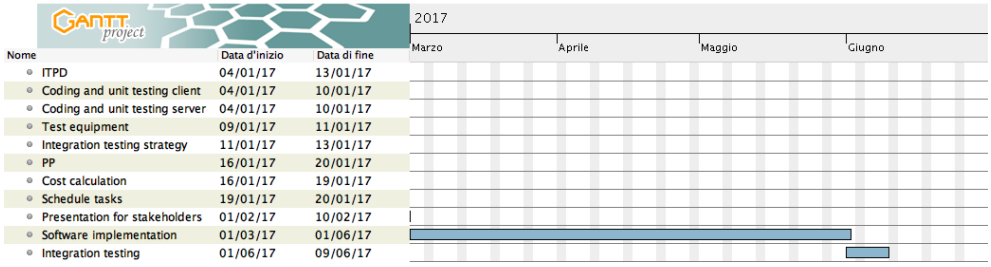
3. **Integration Testing Plan Document:** this part of the project contains the information about testing part, than what software parts have been tested and what are the tools used.

4. **Project Plan:** this part contains the information about the calculation of the Power EnJoy system costs.

5. **Presentation for stakeholders:** part of the project dedicated to prepare the slide for explain to our investors the idea and the necessary founding.
6. **Software implementation**
7. **Integration testing of the system**







4 Resource allocation to task

This chapter is entirely dedicated to describe how ha s been distributed the all project tasks between each member of the group. The information reported in the above tables are an high level view, because often we worked in the studio classroom or through Skype and we focalized together at the same part or chapter to find a better idea from those proposals. Work together to the same part is more expensive in terms of time but this strategy permit us to have a clear vision of all project and not that one of us is "specialized" only in one part of the system, so each decision of the project was made all together to avoid inconsistencies between the various parts of the project. The tables above show the division of the part for each document and for each member of the group.

from 22/10/2016 to 13/11/2016					
Resource	1st Step	2nd Step	3rd Step	4th Step	5th Step
Gabriele	Goals identification	Functional requirements	Non functional requirements (iPad System)	Use case and Sequence diagrams	Alloy
Simone	Goals identification	System vehicle interaction (OnStar and GM)	Non functional requirements (Web App)	Sequence Diagram and Scenario description	Alloy and Car state diagram
Pietro	Detailed description of the problem	Functional requirements	Non functional requirements (Mobile App)Sequence Diagram and Scenario Description	Sequence Diagram and Scenario Description	Alloy

RASD Resource Allocation

from 25/11/2016 to 11/12/2016					
Resource	1st Step	2nd Step	3rd Step	4th Step	5th Step
Gabriele	Introduction and Document Structure	High Level components and their interactions	Entity relationship model representation of database and Presentation Layer	Component Interfaces	User Interface Design (iPad System) and Requirements Traceability
Simone	Architectural Design decision	High Level components and their interactions	Java Beans component of Business Layer	Overall Architecture	User Interface Design (Web App) and Requirements Traceability
Pietro	Architectural Design decision	High Level components and their interactions	Deployment view diagram	Algorithm Design	User Interface Design (Mobile App) and Used Tools

DD Resource Allocation

from 2/1/2017 to 15/1/2017				
Resource	1st Step	2nd Step	3rd Step	4th Step
Gabriele	Introduction and Integration Strategy (Entry Criteria Elements to be Integrated)	Sequence of Component (Power EnJoy manager and Park manager)	Individual Steps and test Description (User Manager, History Manager, Power Enjoy Manager)	Tools used and App Performance Analysis
Simone	Integration Strategy (Integration Testing Strategy)	Sequence of Component (Payment Handler Manager)	Individual Steps and test Description (Park Manager, Payment Handle Manager) and Integration Between Subsystem	Test Equipment
Pietro	Sequence of Component (Data Access Utilities and Hitstory manager)	Sequence of Component (Subsystem integration Sequence)	Integration between subsystem	required Program Stubs and Test Data

ITPD Resource Allocation

from 15/1/2017 to 22/1/2017			
Resource	1st Step	2nd Step	3rd Step
Gabriele	Introduction and Project Size (Function Points)	COCOMO II	Task and Schedule and Resource Allocation to Tasks
Simone	Project Size (Internal Logic File, External Logic Files, External Inputs)	COCOMO II	Resource Allocation to Tasks
Pietro	Project Size (External Inquiries, External Outputs and Overall estimation)	COCOMO II	Risk management

PP Resource Allocation

from 1/2/2017 to 10/2/2017	
Resource	1st Step
Gabriele	Slide preparation
Simone	Slide preparation
Pietro	Slide preparation

Presentation for stakeholders Resource Allocation

from 1/3/2017 to 1/6/2017				
Resource	1st Step	2nd Step	3rd Step	4th Step
Gabriele	Database	Business tier	iPad System	Test Business tier
Simone	Business tier	Business tier	Web App	Test Web App and iPad System
Pietro	Business tier	Mobile App	Test Database	Test Mobile App

Implementation Resource Allocation

from 1/6/2017 to 9/6/2017			
Resource	1st Step	2nd Step	3rd Step
Gabriele	Business-Data	iPad-Business	Business-Map service
Simone	Business-Data	Web-Business	Business-ExternalPayment
Pietro	Business-Data	Mobile-Business	Business-GM

Integration Testing Resource Allocation

5 Risk associated with the project

In this section we are going to describe the risks that the project development may face

5.1 Project risks

Decision Quality: the decision about quality of pattern used and code write could be a long term problem, because when the system becomes of large dimension with complex functionality and the initial pattern of the system is not strong or not properly used ,could be the risk to lose time to redo work already done.

Requirements change: During the development of the project the requirements could be changed, so this implies to lose time to adapt the existing system with the new functionalities.

Organization: the organization of the team is one of important factor of the project risks. In each moment of the project each member of the group have to know what is his work but also have to know what is the part that other members are doing

Communication between members: the tamos often works remotely and this can lead to misunderstandings in fundamental decision. Those difficulties can be overcome by defining the responsibilities of each team member.

Experience in programming: Our group actually know Java but is not very confident with the framework of Java EE, so we are slow in development of the software.

5.2 Technical risks

App/Web App usability: Another important risk is that people could find our service difficult to use, in particular we should pay attention about the App/Web App usability. A countermeasures for this problem is surely solved letting the customers have an active role in the development phase of the project, this could be done proposing surveys and creating a Facebook page that could have a role of "customer care" to assist our clients. Moreover the website and the App should include a section where is explained and illustrated step by step the service.

Quality of the Vehicles: Another important risk is surely due to the quality degradation of our vehicles, in fact because this service is proposed to many customers this aspect become important. We should take care about the condition of every vehicle, this problem can be mitigated planning an internal organization whose work is to control every day their conditions and in case to solve related problems.

Downtime: it happens for different reasons due to one or more bug into the system, excessive load of the server or hardware failure. The downtime could bring in two different ways. The first is that the customers stop to use the service and we lose users. The second way is that the downtime can bring to lose money.

Data loss: it is an important problem to avoid for two important reasons. The first reason is that if someone from outside can take the data, our users privacy is violated, and the Powe EnJoy company can destroy its image. The second reason is that without data is impossible to use the service. For these reasons the backups are made frequently.

Scalability: our system have to resist to the high number of user that want to use our service, so for this reason, after some analysis, can be used by a large number of customers.

5.3 Economical Risks

Bankruptcy: it happens when stakeholders made a bad decisions, so there is no money to maintain Power EnJoy server. This problem can be avoid with a good feasibility study helps.

Good competitors: competitors can often "stole" customers, so to avoid this situation there is a solution that consists in study the market where it will be applied the system and check if already exists competitors. In affirmative case is important to study their functionality and make our functionalities better than their.

5.4 Political Risks

Change Legislation: This problem concerns the possible changes of the national legislation, in fact in the future could be emanated laws related to the minimum driving age and this could affect our business. Unluckily this problem is also unpredictable and difficult to solve, anyway the best way to protect our business by these issues is to be flexible and always prepared to change and modify the structure and the service of the system.

Risk	Probability	Effects
Decision Quality	Moderate	Severe
Requirements change	Low	Moderate
Organization	Low	Moderate
Communication between members	Low	Moderate
Experience in programming	Moderate	Moderate

Risk	Probability	Effects
App/Web App usability	High	Severe
Quality of the Vehicles	High	Severe
Downtime	Moderate	Catastrophic
Scalability	Moderate	Catastrophic

Risk	Probability	Effects
Bankruptcy	Low	Catastrophic
Good competitors	Moderate	Severe

Risk	Probability	Effects
Change Legislation:	Low	Catastrophic

6 Risk associated with the project

In this section we are going to describe the risks that the project development may face

6.1 Project risks

Decision Quality: the decision about quality of pattern used and code write could be a long term problem, because when the system becomes of large dimension with complex functionality and the initial pattern of the system is not strong or not properly used ,could be the risk to lose time to redo work already done.

Requirements change: During the development of the project the requirements could be changed, so this implies to lose time to adapt the existing system with the new functionalities.

Organization: the organization of the team is one of important factor of the project risks. In each moment of the project each member of the group have to know what is his work but also have to know what is the part that other members are doing

Communication between members: the tamos often works remotely and this can lead to misunderstandings in fundamental decision. Those difficulties can be overcome by defining the responsibilities of each team member.

Experience in programming: Our group actually know Java but is not very confident with the framework of Java EE, so we are slow in development of the software.

6.2 Technical risks

App/Web App usability: Another important risk is that people could find our service difficult to use, in particular we should pay attention about the App/Web App usability. A countermeasures for this problem is surely solved letting the customers have an active role in the development phase of the project, this could be done proposing surveys and creating a Facebook page that could have a role of "customer care" to assist our clients. Moreover the website and the App should include a section where is explained and illustrated step by step the service.

Quality of the Vehicles: Another important risk is surely due to the quality degradation of our vehicles, in fact because this service is proposed to many customers this aspect become important. We should take care about the condition of every vehicle, this problem can be mitigated planning an internal organization whose work is to control every day their conditions and in case to solve related problems.

Downtime: it happens for different reasons due to one or more bug into the system, excessive load of the server or hardware failure. The downtime could bring in two different ways. The first is that the customers stop to use the service and we lose users. The second way is that the downtime can bring to lose money.

Data loss: it is an important problem to avoid for two important reasons. The first reason is that if someone from outside can take the data, our users privacy is violated, and the Powe EnJoy company can destroy its image. The second reason is that without data is impossible to use the service. For these reasons the backups are made frequently.

Scalability: our system have to resist to the high number of user that want to use our service, so for this reason, after some analysis, can be used by a large number of customers.

6.3 Economical Risks

Bankruptcy: it happens when stakeholders made a bad decisions, so there is no money to maintain Power EnJoy server. This problem can be avoid with a good feasibility study helps.

Good competitors: competitors can often "stole" customers, so to avoid this situation there is a solution that consists in study the market where it will be applied the system and check if already exists competitors. In affirmative case is important to study their functionality and make our functionalities better than their.

6.4 Political Risks

Change Legislation: This problem concerns the possible changes of the national legislation, in fact in the future could be emanated laws related to the minimum driving age and this could affect our business. Unluckily this problem is also unpredictable and difficult to solve, anyway the best way to protect our business by these issues is to be flexible and always prepared to change and modify the structure and the service of the system.

Risk	Probability	Effects
Decision Quality	Moderate	Severe
Requirements change	Low	Moderate
Organization	Low	Moderate
Communication between members	Low	Moderate
Experience in programming	Moderate	Moderate

Risk	Probability	Effects
App/Web App usability	High	Severe
Quality of the Vehicles	High	Severe
Downtime	Moderate	Catastrophic
Scalability	Moderate	Catastrophic

Risk	Probability	Effects
Bankruptcy	Low	Catastrophic
Good competitors	Moderate	Severe

Risk	Probability	Effects
Change Legislation:	Low	Catastrophic

7 Bibliography

References of the documents used to redact this document:

- RASD
- DD
- ITPD
- MyTaxiService Project Plan Document of last Year
- Project Management Basics slides

8 Hours of work

To redact this document, we spent 25 hours per person.