M.Sc. in Computer Science and Engineering

Software Engineering 2

Project Plan Document
(PP)

Version 1.0
Released on: 21st January 2017

Authors:
Marilena Coluccia
Burcu Cesur
Mustafa Çeçe

Reference Professor: Elisabetta Di Nitto

INDEX

Tables

Figures

1 INTRODUCTION

1.1 Revision History

Release No.	Date	Revision Description
Rev. 0	16/01/2017	Project Plan Template and Checklist
Rev. 0.1	18/01/2017	Schedule revision
Rev. 1.0	21/01/2017	Final revision and internal approval

Table 1.1 - Revision History

1.2 Purpose and Scope

The PowerEnJoy Project Plan includes information related to staffing, budgets, timelines, deadlines, goals and measurements. It also provides individual and group objectives and instruction on how each aspect of the plan is to be a carried out.

The plan also serves as a tool for keeping everyone associated with the project on track and focusing on the same details and information.

The intended audience of the PowerEnJoy Project Plan is all project stakeholders including the project sponsor, senior leadership and the project team.

1.3 List of Definitions and Abbreviations

Acknowledgment: signal passed between communicating processes or computers to signify acknowledgement, or receipt of response, as part of a communications protocols. It can contain a note.

API: Application Programming Interface is used for interacting and communicating with other existing systems.

Customer: A person that, after registering, has an account. Customer register to the system with name, surname, mobile number, credit card number and drive license then an access to the system is provided with a password that by sending an SMS to the mobile number used in registration.

DD: *Design Document*

Drivers module: considered as the form of dummy modules which are always distinguished as "calling programs", that is handled in bottom up integration testing, it is only used when main programs are under construction.

GPS: The Global Positioning System is a satellite-based navigation system that transmits radio signals. It is used to provide geolocalization and time information. The GPS has a specific role in the system, as it both receives and provides the coordinates of the cars and customers.

Guest: A possible customer who hasn't logged in or registered yet. Guests can have access to the system functionalities only after making a registration and logging in.

Function point: "unit of measurement" to express the amount of business functionality an information system (as a product) provides to a user. Function points are used to compute a functional size measurement of software.

JDBC: Java Database communication

RASD: *Requirements Analysis and Specifications Document*

Reservation: It can be made for a single car for at most one hour. Once the customer has reserved a car, the reservation is created immediately and expires after one hour.

SMS: Short Message Service is a form of text-messaging communication based on phones and mobile phones.

Record Element Type (RET): A RET is recognizable subgroup of data elements within an ILF or an ELF. It is best to look at logical groupings of data to help identify them. The concept of RET will be discussed in detail in the chapters that discuss internal logical file and external logical files.

File Type Referenced (FTR): A FTR is a file type referenced by a transaction. An FTR must also be an internal logical file or external logical file.

Data Element Type (DET): A DET is a unique user recognizable, non-recursive (non-repetitive) field. A DET is information that is dynamic and not static. A dynamic field is read from a file or created from DET's contained in a FTR. Additionally, a DET can invoke transactions or can be additional information regarding transactions. If a DET is recursive then only the first occurrence of the DET is considered not every occurrence.

EI: External Inputs.

EO: External Outputs.

EQ: External Inquiries.

ELF: External Logic Files.

ILF: Internal Logic Files.

SLOC: System Lines Of Code.

KSLOC: KiloSLOC or 1000 SLOC

1.4 List of Reference Documents

The following references were used to specify this document:

- Requirements Analysis and Specifications Document;
- Design Document;
- Integration testing plan document;
- Function point manual;
- COCOMO II model manual;
- COCOMO II;
- Gearing factors table

2 PROJECT SIZE, COST AND EFFORT ESTIMATION

The Function point estimation will be used to evaluate the project size. Regarding that, main functionality and code lines are fundamental to have a valid estimation.

For the cost and the effort estimation COCOMO II model will be used, using as initial guidance the amount of lines of code computed with the Function Points approach.

2.1 Size estimation: function points

The Function Point estimation approach, is based on the number of functionalities in a software and their complexity. Function Points estimators are useful since they are based on information that are available early in the project life cycle.

All the components are rated based upon DET's, and either RET's or FTR's.

Component	*RET's*	*FTR's*	*DET's*
External Inputs (EI)		✓	✓
External Outputs (EO)		✓	✓
External Inquiries (EQ)		✓	✓
External Logic Files (ELF)	✓		✓
Internal Logic Files (ILF)	✓		✓

Table 2.1 - Function Points Components

In the world of Function Point Analysis, systems are divided into five large classes and general system characteristics. The first three classes or components are External Inputs, External Outputs and External Inquires. Each of these components transact against files therefore they are called transactions. The next two Internal Logical Files and External Logical Files are where data is stored that is combined to form logical information. The general system characteristics assess the general functionality of the system.

The estimation is based on the statistical analysis of real projects, which have been properly normalized and condensed in the following tables:

	Data elements		
Record Elements	1-19	20-50	51+
1	Low	Low	Avg
2-5	Low	Avg	High
6+	Avg	High	High

Table 2.2 - FP Counting Weight for ILFs and ELFs

	Data elements		
File Types	1-5	6-19	20+
0-1	Low	Low	Avg
2-3	Low	Avg	High
4+	Avg	High	High

Table 2.3 - FP Counting Weight for EOs and EQs

	Data elements		
File Types	1-4	5-15	16+
0-1	Low	Low	Avg
2-3	Low	Avg	High
4+	Avg	High	High

Table 2.4 - FP Counting weight for EIs

Function type	**Complexity Weight**		
	Low	Average	High
External Inputs (EI)	3	4	6
External Outputs (EO)	4	5	7
External Inquiries (EQ)	3	4	6
External Logical Files (EIF)	5	7	10
Internal Logical Files (ILF)	7	10	15

Table 2.5 - UFP Complexity Weights

2.1.1 Internal Logical Files (ILFs)

An Internal Logical File is a group of logically related data that resides entirely within the application boundary and is maintained through External Inputs. An internal logical file has the inherent meaning it is internally maintained, it has some logical structure and it is stored in a file.

The PowerEnJoy application include several ILFs. First, the system has to store information about the customers: name, surname, mobile number, password, credit card and driver license. They can be stored in a unique table where the first column is represented by the CustomerID.

Another data subgroup is composed by car identifier, position, battery and availability. As written in the RASD assumptions, the rest of car information are already stored thanks to the old system, so they should only be joined.

After a reservation acceptance, the following data have to be stored in a reservations table: reservation identifier, customer identifier, car identifier, starting position, proposed destination, date and time of the reservation. If the trip is completed there are also data about the final destination, travel time and the cost to charge.

Finally, the system keeps a list of application and plugin identifiers that have access to the privileged API. Each identifier is associated with the contact information of the developer, a description of what the application or plugin does and the exhaustive list of methods that can be called.

Using the previously defined tables, this is the count obtained:

ILF	**Complexity**	**FP**
Customer	Low	7
Car	Low	7
Reservation	Average	10
API permissions	Low	7
Total		**31**

Table 2.6 - FP cost of ILFs

2.1.2 External Logical Files (ELFs)

An External Logical File is a group of logically related data that is used for reference purposes only. The data resides entirely outside the application boundary and is maintained by another application external inputs.

The system receives information from GPS components and request location information like the full address to an open-map service. Information about car battery and eventually maintenance unavailability is received too.

ELF	**Complexity**	**FP**
Location	Average	10
Car information	Low	7
Total		**17**

Table 2.7 - FP cost of ELFs

2.1.3 *External Inputs (EIs)*

An External Input is an elementary process in which data crosses the boundary from outside to inside. This data is coming external to the application. The data may come from a data input screen or another application. The data may be used to maintain one or more internal logical files. The data can be either control information or business information. If the data is control information it does not have to maintain an internal logical file.

About the PowerEnJoy system, the following EIs can be defined:

- Registration;
- Login / logout;
- Profile information changes;
- Car reservation;
- Delete reservation

EI	Complexity	FP
Registration	Low	3
Login	Low	3
Logout	Low	3
Profile information changes	Low	3
Car reservation	High	6
Delete reservation	Low	3
Total		**21**

Table 2.8 - FP cost for EIs

2.1.4 *External Inquiries (EQs)*

An External Inquiry is an elementary process with both input and output components that result in data retrieval from one or more internal logical files and external logical files. The input process does not update or maintain any FTR's and the output side does not contain derived data.

The PowerEnJoy system support some interactions of this type[1]:

- Map access page, information about near cars are showed on the map with the help of pins to localize them;
- Account management page, customer profile information in which there is the reservation history.

Using the previously defined tables, this is the count obtained:

[1] Those elements are well showed in the mockup document

EQ	Complexity	FP
Map access	High	6
Account management	Low	3
Total		**9**

Table 2.9 - FP cost of EQs

2.1.5 External Outputs (EOs)

An External Output is an elementary process in which derived data passes across the boundary from inside to outside. Additionally, an EO may update an ILF. The data creates reports or output files sent to other applications. These reports and files are created from information contained in one or more internal logical files and external logical files.

The password first receive and the password retrieval are located among this kind of processes.

EO	Complexity	FP
First password	Low	4
Password retrieval	Low	4
Total		**8**

Table 2.10 - FP cost of Eos

2.1.6 Overall estimation

In the *Table 2.11* - FP total costthe total function points cost is showed.

	FP
Internal Logical Files (ILF)	31
External Logical Files (EIF)	17
External Inputs (EI)	21
External Inquiries (EQ)	9
External Outputs (EO)	8
Total	**86**

Table 2.11 - FP total cost

FPs can be converted in line of code according to the following function:

$$LOC = FPs \cdot gearing\ factor$$

For PowerEnJoy system a gearing maximum factor equal to 67 and a gearing minimum factor equal to 46 will be considered.

$$LOC_{MAX} = 86 \cdot 67 = 5762 \quad LOC_{MIN} = 86 \cdot 46 = 3956$$

2.2 Cost and effort estimation: COCOMO II

In this section COCOMO II approach will be used to estimate the cost and effort needed to develop the PowerEnJoy system.

2.2.1 Scale Drivers

According the COCOMO II model manual, it's necessary to establish some Scale Factors:

- Precedentedness(PREC), value of the team experience;
- Development flexibility (FLEX), functional and technology flexibility;
- Risk resolution (RESL), level of awareness and reactiveness with respect to risks
- Team cohesion (TEAM);
- Process maturity (PMAT).

Scale Factors	***Very Low***	***Low***	***Nominal***	***High***	***Very High***	***Extra High***
PREC	thoroughly unprecedented	largely unprecedented	somewhat unprecedented	generally familiar	largely familiar	thoroughly familiar
SF_j	6.20	4.96	3.72	2.48	1.24	0.00
FLEX	rigorous	occasional relaxation	some relaxation	general conformity	some conformity	general goals
SF_j	5.07	4.05	3.04	2.03	1.01	0.00
RESL	little (20%)	some (40%)	often (60%)	generally (75%)	mostly (90%)	full (100%)
SF_j	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	very difficult interactions	some difficult interactions	basically cooperative interactions	largely coop.	highly coop.	seamless interactions
SF_j	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	The estimated Equivalent Process Maturity Level (EPML) or					
	SW-CMM Level 1 Lower	SW-CMM Level 1 Upper	SW-CMM Level 2	SW-CMM Level 3	SW-CMM Level 4	SW-CMM Level 5
SF_j	7.80	6.24	4.68	3.12	1.56	0.00

Table 2.12 - Scale Factor Values, SFj, for COCOMO II Models

According to Table 2.12 it is estimated a low SPEC and a nominal TEAM, due to small experience of all team members, but each of them have soft cooperative skills.

It is predicted a Level 3 PMAT and a high RESL. About FLEX, it should be low due to the RASD document that somehow fixes functionalities.

	SF_j
Precedentedness(PREC)	4,96
Development flexibility (FLEX)	4,05
Risk resolution (RESL)	2,83
Team cohesion (TEAM)	3,29
Process maturity (PMAT)	3,12
Total	**18,25**

Table 2.13 - Scale drivers SF_j count

2.2.2 Product Factors

Post-Architecture effort multipliers (EM) are used in the COCOMO II model to adjust the nominal effort, Person-Months, to reflect the software product under development. Product factors account for variation in the effort required to develop software caused by characteristics of the product under development. A product that is complex, has high reliability requirements, or works with a large testing database will require more effort to complete.

2.2.2.1 Required Software Reliability (RELY)

This is the measure of the extent to which the software must perform its intended function over a period. A system malfunctioning does not involve high risks because in the area considered there are other competitors in the service. Surely there would be a loss about the city pollution being competitors using pollutants for their service. For this it is considered to adopt a high rating level.

RELY Descriptors:	slight inconvenience	low, easily recoverable losses	moderate, easily recoverable losses	high financial loss	risk to human life	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	0.82	0.92	1.00	1.10	1.26	n/a

Table 2.14 - RELY multipliers

2.2.2.2 Data Base Size (DATA)

This cost driver attempts to capture the effect large test data requirements have on product development. The rating is determined by calculating D/P, the ratio of bytes in the testing database to SLOC in the program. As assumed in the RASD, the system will store maximum 10GB a day. As average record, 7 GB will be considered and the DATA rating level will be very high.

DATA* Descriptors		Testing DB bytes/Pgm SLOC < 10	10≤D/P<100	100≤D/P<1000	D/P≥1000	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	0.90	1.00	1.14	1.28	n/a

Table 2.15 - DATA multipliers

2.2.2.3 Product Complexity (CPLX)

Complexity is divided into five areas: control operations, computational operations, device-dependent operations, data management operations, and user interface management operations. The combination of areas characterizes the product or the component of the product to rate. In the PowerEnJoy system that combination sets the CPLX to a nominal rating level.

Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	0.73	0.87	1.00	1.17	1.34	1.74

Table 2.16 - CPLX multipliers

2.2.2.4 Developed for Reusability (RUSE)

This cost driver accounts for the additional effort needed to construct components intended for reuse on current or future projects. This effort is consumed with creating more generic design of software, more elaborate documentation, and more extensive testing to ensure components are ready for use in other applications. Being the PowerEnJoy system a across project, the RUSE rating level is set to Nominal

RUSE Descriptors:		none	across project	across program	across product line	across multiple product lines
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	0.95	1.00	1.07	1.15	1.24

Table 2.17 - RUSE multipliers

2.2.2.5 Documentation Match to Life-Cycle Needs (DOCU)

In COCOMO II, the rating scale for the DOCU cost driver is evaluated in terms of the suitability of the project's documentation to its life-cycle needs. DOCU rating level is calibrated to nominal.

DOCU Descriptors:	Many lifecycle needs uncovered	Some lifecycle needs uncovered.	Right-sized to life-cycle needs	Excessive for life-cycle needs	Very excessive for life-cycle needs	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	0.81	0.91	1.00	1.11	1.23	n/a

Table 2.18 - DOCU multipliers

2.2.3 Platform Factors

The platform refers to the target-machine complex of hardware and infrastructure software.

2.2.3.1 Execution Time Constraint (TIME)

This is a measure of the execution time constraint imposed upon a software system. The rating is expressed in terms of the percentage of available execution time expected to be used by the system or subsystem consuming the execution time resource. TIME is thought having a high rating level.

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 Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	n/a	1.00	1.11	1.29	1.63

Table 2.19 - TIME multipliers

2.2.3.2 Main Storage Constraint (STOR)

This rating represents the degree of main storage constraint imposed on a software system or subsystem. As mentioned in paragraph 2.2.2.2 , an average record will be considered to predict this factor, leading to consider a high rating level for STOR.

STOR Descriptors:			≤ 50% use of available storage	70% use of available storage	85% use of available storage	95% use of available storage
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	n/a	1.00	1.05	1.17	1.46

Table 2.20 - STOR multipliers

2.2.3.3 Platform Volatility (PVOL)

"Platform" is used here to mean the complex of hardware and software (OS, DBMS, etc.) the software product calls on to perform its tasks, including any compilers or assemblers supporting the development of the software system. PVOL is set to Nominal according to the following table.

PVOL Descriptors:		Major change every 12 mo.; Minor change every 1 mo.	Major: 6 mo.; Minor: 2 wk.	Major: 2 mo.; Minor: 1 wk.	Major: 2 wk.; Minor: 2 days	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	0.87	1.00	1.15	1.30	n/a

Table 2.21 - PVOL multipliers

2.2.4 *Personnel Factors*

The Personnel Factors are for rating the development team's capability and experience – not the individual. These ratings can change during the project develop, reflecting the gaining of experience or the rotation of people onto and off the project.

2.2.4.1 *Analyst Capability (ACAP)*

Analysts are personnel who work on requirements, high-level design and detailed design. The major attributes that should be considered in this rating are analysis and design ability, efficiency and thoroughness, and the ability to communicate and cooperate. The PowerEnJoy system analysis was done considering even real world aspects, but admitting the human mistake, ACAP is set to high.

ACAP Descriptors:	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.42	1.19	1.00	0.85	0.71	n/a

Table 2.22 - ACAP multipliers

2.2.4.2 *Programmer Capability (PCAP)*

The Programmer Capability should consider ability, efficiency and thoroughness, and the ability to communicate and cooperate. The experience of the programmer should not be considered here. PCAP rating level is set to high.

PCAP Descriptors	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.34	1.15	1.00	0.88	0.76	n/a

Table 2.23 - PCAP multipliers

2.2.4.3 *Personnel Continuity (PCON)*

The rating scale for PCON is in terms of the project's annual personnel turnover: from 3%, very high continuity, to 48%, very low continuity. The PCON level is nominal.

PCON Descriptors:	48% / year	24% / year	12% / year	6% / year	3% / year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.29	1.12	1.00	0.90	0.81	

Table 2.24 - PCON multipliers

2.2.4.4 Applications Experience (APEX)

The rating for this cost driver is dependent on the level of applications experience of the project team developing the software system or subsystem. The ratings are defined in terms of the project team's equivalent level of experience with this type of application. Regarding the PowerEnJoy team the APEX is considered Nominal.

APEX Descriptors:	≤ 2 months	6 months	1 year	3 years	6 years	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.22	1.10	1.00	0.88	0.81	n/a

Table 2.25 - APEX multipliers

2.2.4.5 Platform Experience (PLEX)

The Post-Architecture model broadens the productivity influence of platform experience, by recognizing the importance of understanding the use of more powerful platforms, including more graphic user interface, database, networking, and distributed middleware capabilities. PLEX is set to a high rating level.

PLEX Descriptors:	≤ 2 months	6 months	1 year	3 years	6 year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.19	1.09	1.00	0.91	0.85	n/a

Table 2.26 - PLEX multipliers

2.2.4.6 Language and Tool Experience (LTEX)

This is a measure of the level of programming language and software tool experience of the project team developing the software system or subsystem. LTEX is considered nominal.

LTEX Descriptors:	≤ 2 months	6 months	1 year	3 years	6 year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.20	1.09	1.00	0.91	0.84	

Table 2.27 - LTEX multipliers

2.2.5 *Project Factors*

Project factors account for influences on the estimated effort such as use of modern software tools, location of the development team, and compression of the project schedule.

2.2.5.1 *Use of Software Tools (TOOL)*

The tool rating ranges from simple edit and code, very low, to integrated life-cycle management tools, very high. An emerging extension of COCOMO II is in the process of elaborating the TOOL rating scale and breaking out the effects of TOOL capability, maturity, and integration. TOOL is considered high.

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 life-cycle tools, well integrated with processes, methods, reuse	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.17	1.09	1.00	0.90	0.78	n/a

Table 2.28 - TOOL multipliers

2.2.5.2 *Multisite Development (SITE)*

Given the increasing frequency of multisite developments, and indications that multisite development effects are significant, the SITE cost driver has been added in COCOMO II.

Determining its cost driver rating involves the assessment and judgement-based averaging of two factors: site collocation (from fully collocated to international distribution) and communication support (from surface mail and some phone access to full interactive multimedia). The SITE has a high rating level.

SITE: Collocation Descriptors:	International	Multi-city and Multicompany	Multi-city or Multi-company	Same city or metro. area	Same building or complex	Fully collocated
SITE: Communications Descriptors:	Some phone, mail	Individual phone, FAX	Narrow band email	Wideband electronic communication.	Wideband elect. comm., occasional video conf.	Interactive multimedia
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.22	1.09	1.00	0.93	0.86	0.80

Table 2.29 - SITE multipliers

2.2.5.3 Required Development Schedule (SCED)

This rating measures the schedule constraint imposed on the project team developing the software. The ratings are defined in terms of the percentage of schedule stretch-out or acceleration with respect to a nominal schedule for a project requiring a given amount of effort. It is considered high.

SCED is the only cost driver that is used to describe the effect of schedule compression / expansion *for the whole project.*

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 Multiplier	1.43	1.14	1.00	1.00	1.00	n/a

Table 2.30 - SCED multipliers

2.2.6 *Cost drivers' overall*

According to the previous paragraph, this table can be redacted.

Cost Driver	**Factor**	**Value**
Required Software Reliability (RELY)	High	1,10
Database size (DATA)	Very high	1,28
Product complexity (CPLX)	Nominal	1,00
Developed for Reusability (RUSE)	Nominal	1,00
Documentation Match to Life-Cycle Needs (DOCU)	Nominal	1,00
Execution Time Constraint (TIME)	High	1,11
Main Storage Constraint (STOR)	High	1,05
Platform Volatility (PVOL)	Nominal	1,00
Analyst Capability (ACAP)	High	0,85
Programmer Capability (PCAP)	High	0,88
Personnel Continuity (PCON)	Nominal	1,00
Applications Experience (APEX)	Nominal	1,00
Platform Experience (PLEX)	High	0,91
Language and Tool Experience (LTEX)	High	1,00
Use of Software Tools (TOOL)	High	0,90
Multisite Development (SITE)	High	0,93
Required Development Schedule (SCED)	High	1,00
Total		**0,934939**

Table 2.31 - Cost driver's overall

2.2.7 *Effort equation*

This final equation gives us the effort estimation measured in Person-Months (PM):

$$Effort = A \cdot EAF \cdot KSLOC^{\alpha}$$

Where, according to COCOMO II manual, A, EAF *and* α can be defined as:

$$A = 2.94$$

$$EAF = product\ of\ all\ cost\ drivers\ (0{,}8508\ )$$

$$\alpha = \beta + 0{,}01 \cdot \sum_j SF(j) = 1{,}0925 \qquad with \qquad \beta = 0{,}91$$

The PowerEnJoy effort range will be:

$$Effort_{Max} = A \cdot EAF \cdot KSLOC^{\alpha} = 2{,}94 \cdot 0{,}934939 \cdot 5{,}762^{1{,}0925} = 18{,}6233 = 19\ PM$$

$$Effort_{Min} = A \cdot EAF \cdot KSLOC^{\alpha} = 2{,}94 \cdot 0{,}934939 \cdot 3{,}956^{1{,}0925} = 12{,}3490 = 13\ PM$$

2.2.8 *Schedule estimation*

The schedule can be estimated fitting the following formula:

$$Duration = \gamma \cdot Effort^{\varepsilon}$$

Where $\gamma\ and\ \varepsilon$ can be defined as:

$$\gamma = 3{,}67 \qquad \varepsilon = 0{,}30928$$

The schedule rage is:

$$Duration_{Max} = 3{,}67 \cdot 18{,}6233^{0{,}30928} = 9{,}0671 \Rightarrow 10\ months$$

$$Duration_{Min} = 3{,}67 \cdot 12{,}3490^{0{,}30928} = 7{,}9852 \Rightarrow 8\ months$$

3 SCHEDULE

In this chapter a high-level project schedule will be provided. During the project, due to internal organization of development phases, advanced schedules will be defined.

Figure 3.1 - RASD and DD schedule

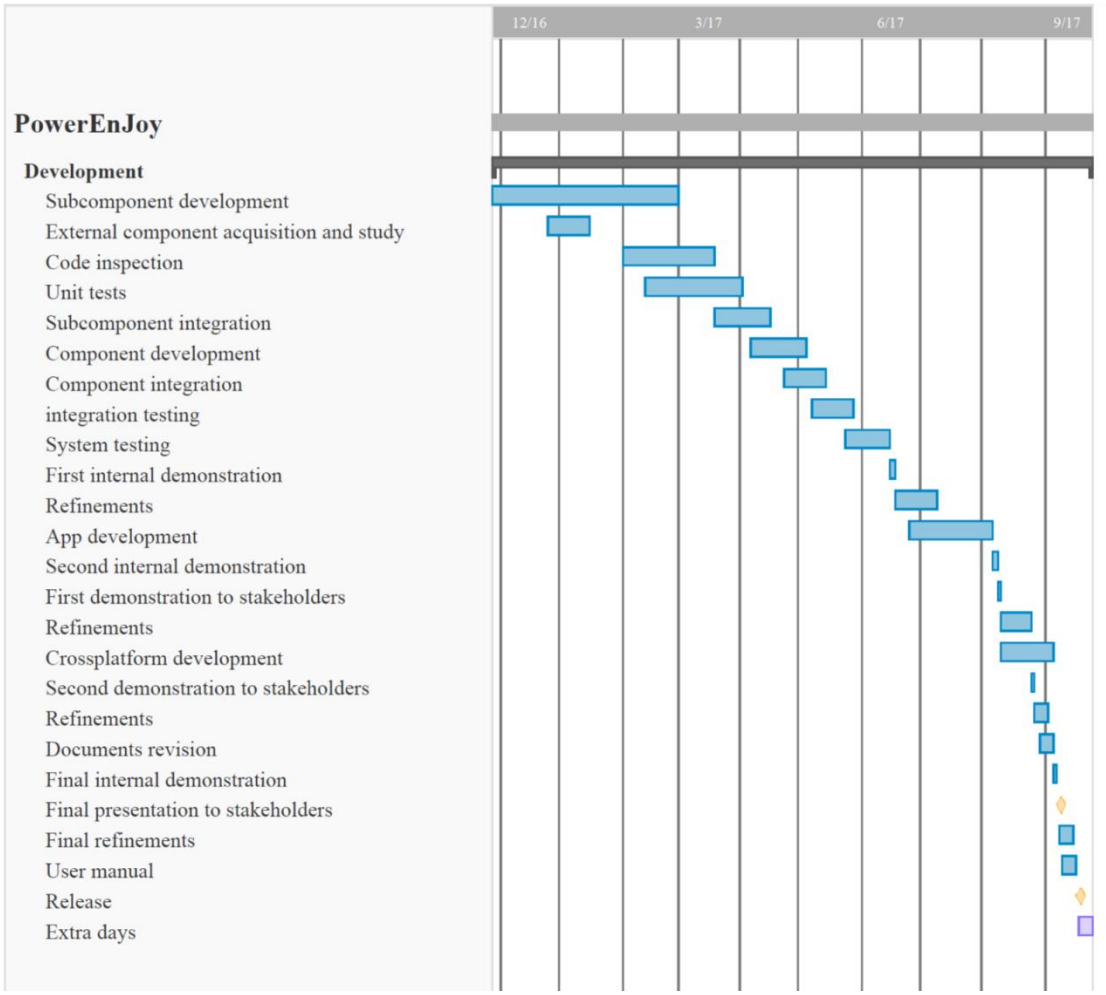

Figure 3.2 - Development schedule

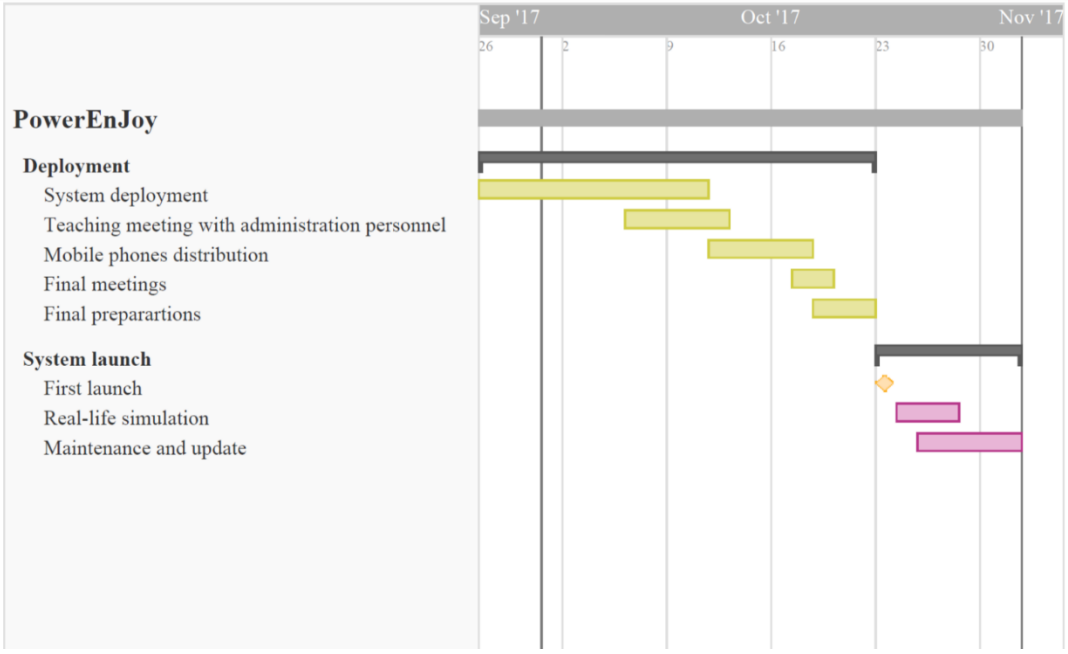

Figure 3.3 - Deployment and System launch schedule

4 RESOURCE ALLOCATION

In this chapter, it will be planned a general overview of how the team will split and/or cooperate on the tasks defined by the schedule in the previous section. During the project, due to internal organization of development phases, advanced schedules will be defined.

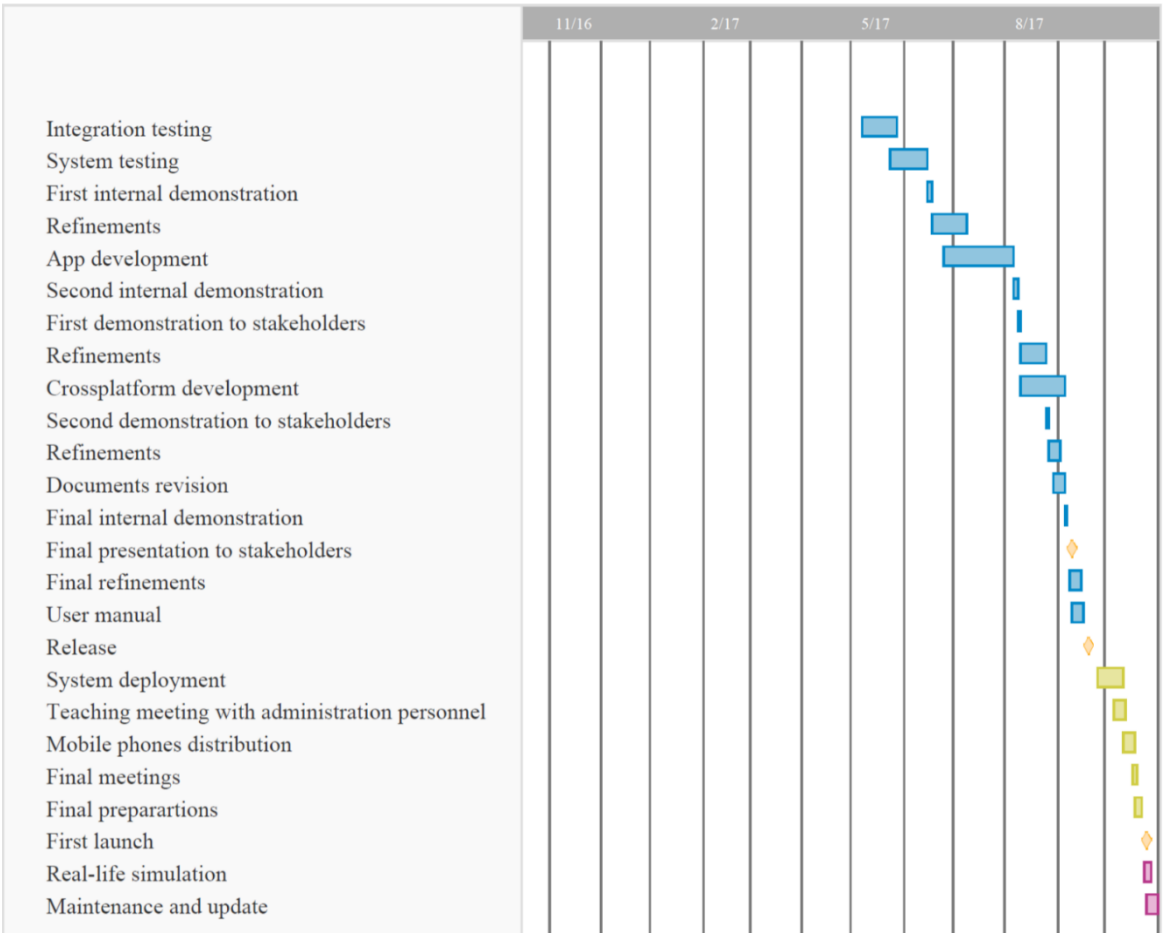

Figure 4.1 - Marilena Coluccia's schedule


11/16
2/17
5/17
8/17
PowerEnJoy
Burcu Cesur
Meeting with stakeholders
Definition of main functionality
Initial requirement draft
Initial assumptions
Initial draft of activity diagrams
Identification of use cases
Initial mockups of GUI
Meeting with stakeholders
Refinement of assumptions
Refinement of requirements
Refinement of activity diagrams
Refinement of GUI mockups
Meeting with stakeholders
Final requirements
Final assumptions
Final activity diagrams
Final GUI mockups
Initial cost, schedule and effort estimation
Meeting with stakeholders
Final refinemets
Revision of estimations
High level draft of architecture
Identification of main component
Meeting with stakeholders
Revision of architecture
Refinements of componets
Identification of subcomponents
First draft of component interfaces
Refinement of GUI mockups
Initial draft of algorithm design
Meeting with stakeholders
Final architecture design
Refinement of component interfaces
Refinement of algorithm design
Meeting with stakeholders
Final UI design
Final component interfaces
Final algorithm design
Meeting with stakeholders
Final refinements
Subcomponent development
External component acquisition and study
Code inspection
Unit tests
Subcomponent integration
Component development
Component integration

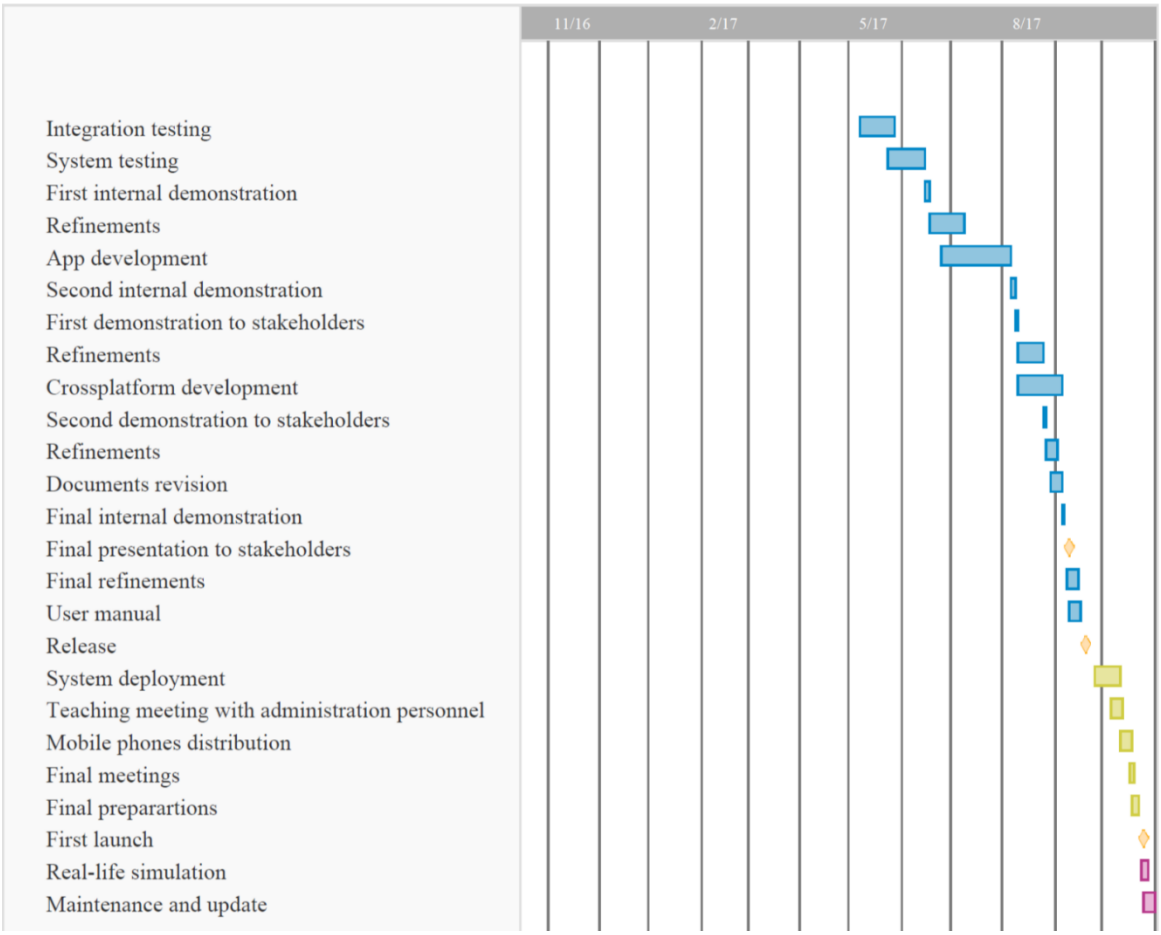

Figure 4.2 - Burcu Cesur's schedule


11/16
2/17
5/17
8/17
PowerEnJoy
Mustafa Çeçe
Meeting with stakeholders
Definition of main functionality
Initial requirement draft
Initial assumptions
Initial draft of activity diagrams
Identification of use cases
Initial mockups of GUI
Meeting with stakeholders
Refinement of assumptions
Refinement of requirements
Refinement of activity diagrams
Refinement of GUI mockups
Meeting with stakeholders
Final requirements
Final assumptions
Final activity diagrams
Final GUI mockups
Initial cost, schedule and effort estimation
Meeting with stakeholders
Final refinemets
Revision of estimations
High level draft of architecture
Identification of main component
Meeting with stakeholders
Revision of architecture
Refinements of componets
Identification of subcomponents
First draft of component interfaces
Refinement of GUI mockups
Initial draft of algorithm design
Meeting with stakeholders
Final architecture design
Refinement of component interfaces
Refinement of algorithm design
Meeting with stakeholders
Final UI design
Final component interfaces
Final algorithm design
Meeting with stakeholders
Final refinements
Subcomponent development
External component acquisition and study
Code inspection
Unit tests
Subcomponent integration
Component development
Component integration

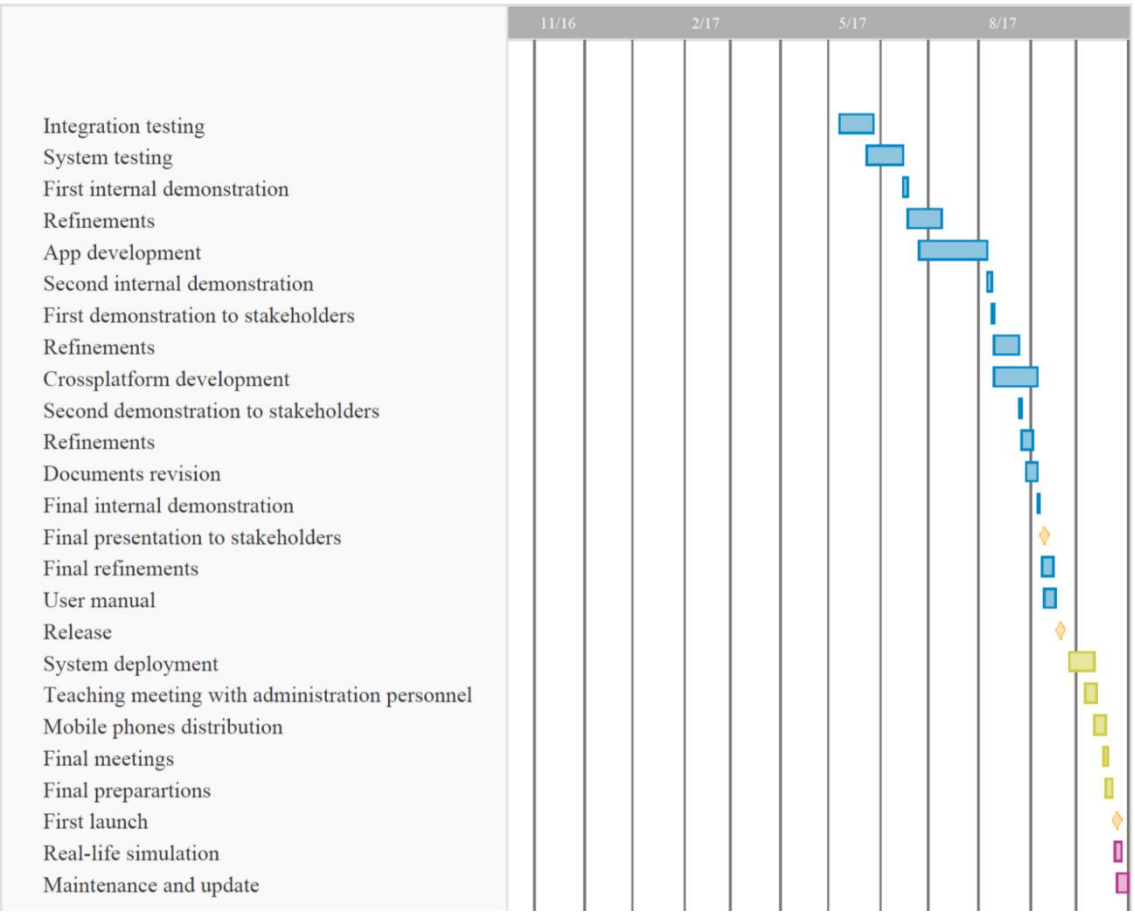

Figure 4.3 - Mustafa Çeçe's schedule

5 RISK MANAGEMENT

Risk management is a set of activities, methods and coordinates resources to direct and control an organization with regard to the risks. Every kind of risks have to be already detailed and well analyzed, in order to implement actions to deal with the possible ones when they arise. All identified risks are: risk assessment, risk monitoring, and risk mitigation.

The focus of this paragraph is to identify activities to reduce the probability and/or impact of the risks. Risks are categorized in tree main topics, which are *technical*, *political* and *operational*.

Several preventions should be considered regarding technical risks, which can lead to failure of functionality and performance. A technical analysis, followed by a deeply designed architecture, is crucial in this point. Prototyping and testing are supporters of the developing process in meaning of minimizing/eliminating technical risks. Technical risks may be faced during ongoing PoweEnJoy process so that design recovery has to be planned. Source code loss is calamity, therefore security boundaries have to be defined and data backups have to be made regularly. Other technical risks are lack stability of the technology components, which causes components to crash, and unextendible technology components in the meaning of capabilities. Regarding to those, alternative backup components will be replaced and/or fixes will be provided from backup teams to mitigate the risk. Changes, on terms and conditions of the inherited Mapping Service and/or API modifications, can also rise technical problems but adapting alternative resolutions are common and easily to handle.

Political risks are identified in law concerns for possible legislation modifications. Since this project offers electrical cars by taking the customers' credit card numbers, changes in the legislations have to be followed regularly to not encounter issues such that topic consideration changes as against law. Setting boundaries between obtaining cost from credit card and the data storage flow can be followed if such risk arise. Also, defined special parking areas may face issues if a relevant government rules are changed, so that these modifications will handled by changes on these offered parking areas.

Concerns on operational risks are about schedule, finance and structure. Estimated schedule is a fundamental of bone structure of the project process. Even though project schedule is provided in this document, it may show alterations while the ongoing process. For this reason, this risk is prevented by business case analysis and incremental development, allocating extra time for major activities. On the other hand, schedule risk is connected to the financial risk, so that schedule issues may also bring along budget problems like cost overruns. Schedule risks mainly have effect on project and finally on company economy and may lead to project failure. Regarding to that, changes on the schedule may ruin the expected/designed financial flow. Change in the pricing plans of the cloud infrastructure can also lead to specific issues on the financial and business side, but they can be easily handled with preventive known modifications. Also, teams are connected to each other's timetables so that when the schedule encounters a breakdown it may also corrupt the planned deadlines. Considerations are also made on company reorganization and changing focuses of the workers. For this reason, an alternative team or flow has to be provided. The solution is to design duty flexibility across workers so that the responsibilities can be shared or switched easily without corrupting the operation when a structural threat appear.

Other threats are structural risks, that have to be prevented by employing the most sufficient and compatible persons, rewarding usually, leading team formation, offering trainings and peer reviews.

6 EFFORT

Name & Surname	*16.01.17*	*18.01.17*	*21.01.17*	*Outside of Group*	*Total*
Marilena COLUCCIA	2h	2h	2h	21 h	27 hour
Burcu CESUR	2h	2h	2h	16 h	22 hour
Mustafa ÇEÇE	2h	2h	2h	6 h	12 hour