Power EnJoy

Project **P**lan

Software Engineering 2

A.A. 2016/2017

Version 1.0

Emanuele Ghelfi (mat. 875550)
Emiliano Gagliardi (mat. 878539)

January 19, 2017

Contents

1 Introduction

1.1 Revision History

Version	Date	Author	Summary
1.0	22/01/17	Emanuele Ghelfi & Emiliano Gagliardi	Initial Release

1.2 Purpose and Scope

The Project Plan document aims at giving an estimation of the total effort requested by the project using Function Points and COCOMO II. In this way the project has also an estimation for the duration and for requested budget. The overall duration is used in giving a schedule to the project and in allocating resources.

In section 2 the Function Points and COCOMO approaches are used to estimate the size, the cost and effort of the project. The size in given in terms of SLOC (Source Lines Of Code) and the effort is given in terms of PM (Person Month).

In section 3 (//todo: add reference) a schedule for the project is proposed by taken into account all possible activities required by the project and the relationship precedence between them.

In section 4 (// todo: add reference) the resources (in terms of people) are assigned to each activity. The resource allocation is done by taken into account predispositions and preferences of the team members.

In section 5 risks related to the project are analyzed. For each risks is given a probability and a possible strategy in order to avoid it or to recover it. Notice that some risks are unpredictable or unrecoverable.

1.3 Definitions, Acronyms and Abbreviations

- FP: Function Point
- PM: Person Month
- SLOC: Source Lines Of Code
- KSLOC: Kilo Source Lines Of Code
- ILF: Internal logic file
- ELF: External logic file
- EI: External Input
- EO: External Output
- EQ: External Inquiries
- DBMS: Database Management System

- API: Application Programming Interface
- ETA: Estimated Time of Arrival
- UI: User Interface
- GUI: Graphical User Interface
- SE: Schedule Equation
- PP: Project Plan

1.4 Reference Documents

Notice that documents in RELEASE folder are related to the project and documents in Reference Documents are external documents used as references.

- Assignments AA 206-2017.pdf (In Reference Documents folder)
- PP.pdf : Project Plan from previous year (In Reference Documents folder)
- CII_modelman2000.pdf: Manual of COCOMO II (In Reference Documents folder)
- Power EnJoy RASD (in RELEASE folder)
- Power Enjoy DD (in RELEASE folder)
- Power EnJoy ITPD (in RELEASE folder)

2 Project size, cost and effort estimation

As the title says this section provide an estimation of the project size, cost and effort that have to be spent in the development of the PowerEnjoy project. The size estimation is made with the function points (FP) technique. The cost estimation is instead based on the COCOMO II statistical approach, and the effort and schedule estimation is based on the two previous.

2.1 Size estimation

The function point cost estimation approach is based on the amount of functionality in a software project and a set of individual project factors [Behrens 1983; Kunkler 1985; IFPUG 1994]. Function points are useful estimators since they are based on information that is available early in the project life-cycle. Here the function types are listed and analyzed for the PowerEnjoy project, assigning to each of them a the complexity level is determined. Then, following the COCOMO II definition, the FP are assigned to each function type.

2.1.1 Internal Logical Files (ILF)

The internal logical files represent each major logical group of user data or control information stored and used by the software system in order to carry out all the operations.
The ILF are the following:

- User information: login information, history of the system usage, the weight is HIGH.
- Operator information: login information, competence area, the weight is LOW.
- Admin information: login information, the weight is LOW.
- Car information: status, cached information (used by car proxy), the weight is HIGH.
- Safe area information: position and eventual plugs, the weight is LOW.

In the following table there are the assignment of the weight and the FPs:

ILF	WEIGHT	FPs
User information	AVERAGE	15
Operator information	LOW	7
Admin information	LOW	7
Car information	HIGH	15
Safe area information	LOW	7

The resulting FPs are: 51

2.1.2 External Interface Files (EIF)

The external interface files represent user data or control output type that leaves the external boundary of the software system.
The EIF are the following:

- Users payment information: AVERAGE
- Users drive license: AVERAGE
- Mobile map service call: LOW

And in the table the assigned weights and FPs:

EIF	WEIGHT	FPs
Users payment information	AVERAGE	7
Users drive license	AVERAGE	7
Mobile map service call	LOW	4

The resulting FPs are: 18

2.1.3 External Inputs (EI)

In this section are listed the functionality that the system has to provide to the different categories of users, and for each of them the FP assignment.
Functionality provided to each category of users:

- Login/logout: simple operations that involves only some check, so the weight is LOW

Functionality provided only to users:

- Reserve a car: operation that involve some check of information and some update, the weight is AVERAGE
- Lock/Unlock a car: this operations involve the check of positions and car communication, the weight is HIGH
- Registration: operations that consist in the check of some information, and in the use of different external API, the weight is AVERAGE

Functionality provided to operators:

- Manage car status: this operation consists in the update of some information, the weight is LOW

Functionality provided to administrators, they are all low because consist only in simple data manipulation operation, as update and drop in the database:

- add/delete administrators: LOW
- add/delete operators: LOW

- add/delete safe areas: LOW

In the following table are listed functionalities and related FPs:

EI	WEIGHT	FPs
Login/Logout	LOW	3x2
Registration	AVERAGE	4
Car search	HIGH	6
Car reservation	AVERAGE	4
Lock/unlock	HIGH	6x2
Manage car status	LOW	3
Add/delete administrators	LOW	3x2
Add/delete operators	LOW	3x2
Add/delete safe areas	LOW	3x2

The resulting FPs are: 53

2.1.4 External Outputs (EO)

The external outputs consist in the user data or control output type that leaves the external boundary of the software system.
The EO involved in the PowerEnjoy system are:

- Users and operators notification: simple information sent to the user, the weight is AVERAGE

- Ride cost: operation that involves the computation of the cost and the communication of it, the weight is AVERAGE

In the table are listed the EOs and the related FPs:

EO	WEIGHT	FPs
User and operator notification	AVERAGE	5
Ride cost	AVERAGE	5

The resulting FPs are: 10

2.1.5 External Inquiries (EI)

The EI are the input-output combinations, where input causes and generates an immediate output.
The EI in the PowerEnjoy system are:

- Consult profile: this functionality is provided only consulting the database, the weight is LOW

- Car research: this is one of the most difficult functionality to provide, that involves the interrogation to the cars, so the weight is HIGH

- Malfunction communication: the input is the description of a malfunction of a car, the output is the notification to a operator, the weight is AVERAGE
- Consult car status: this functionality is provided communicating with the cars, the weight is AVERAGE

The following table reports the EIs and the corresponding FPs:

EI	WEIGHT	FPs
Consult profile	LOW	3
Car research	HIGH	6
Malfunction communication	AVERAGE	4
Consult car status	AVERAGE	4

The resulting FPs are: 17

2.1.6 Overall estimation

FUNCTION TYPE	VALUE
Internal logic files	51
External interface files	18
External inputs	53
External outputs	10
External inquiries	17

The total amount of FPs is 149, that considering Java Enterprise Edition as a development platform (low SLOC/UFP is 15, high SLOC/UFP is 67) brings to the following estimation:

$$SLOC_{LOWER} = 15 \times 149 = 2235$$

$$SLOC_{UPPER} = 67 \times 132 = 9983$$

2.2 Cost and Effort Estimation: COCOMO II

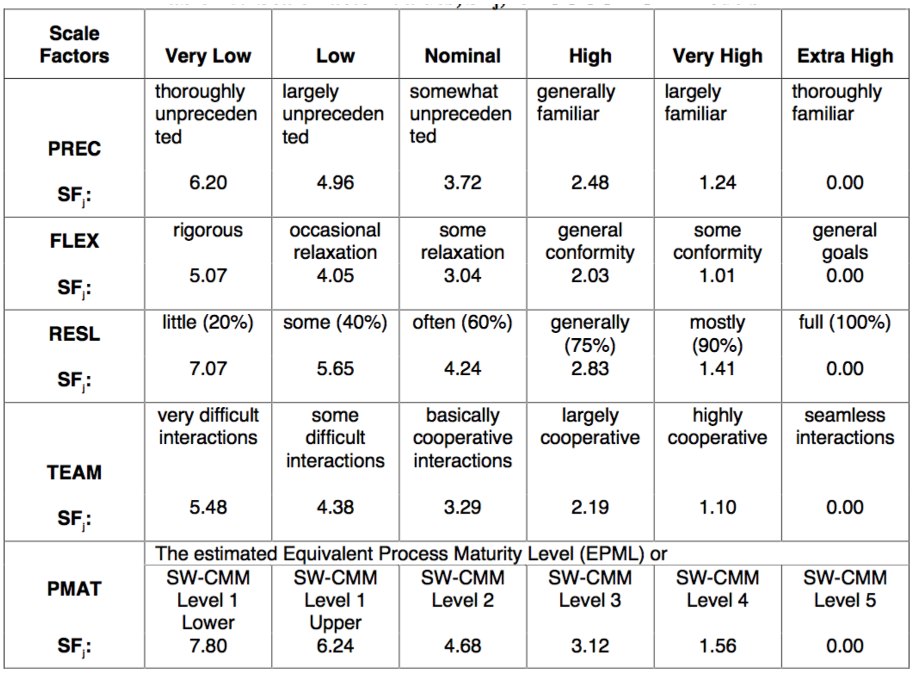
COCOMO stands for COnstructive COst MOdel. It provides an estimation of the cost and effort needed to develop the system to be.

The estimation is achieved through a complex, non linear model that takes into account the characteristics of product and process.

Tables in this sections are taken from [2].

2.2.1 Scale Drivers

This is the official table for the values of scale drivers from COCOMO II:



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

Figure 1: Scale Drivers

Description of scale drivers:

- Precedentedness: High if a product is similar to several previously developed projects. In this case powerEnjoy is a novel project for the organization and this value is LOW.
- Development Flexibility: High if there are no specific constraints to conform to pre-established requirements and external interface specs. In this project there are several constraints related on the functionalities so this value will be LOW.
- Architecture / Risk Resolution: High if we have a good risk management plan, clear definition of budget and schedule, focus on architectural definition. In this project there is a valuable focus on the architecture and there is also a focus on the risk management. Budget hasn't been specified at the starting of the project. Because of these motivation this value will be HIGH.
- Team Cohesion: High if all stakeholders are able to work in a team and share the same vision and commitment. Since the team members work well together this value VERY HIGH.
- Process Maturity: Refers to a well known method for assessing the maturity of a software organization: CMMI. This is one of the first project

for the organization, so the Level is set to 2: Managed. The organization has a reactive approach. Processes are planned, documented, performed, monitored, and controlled at the project level.

Results:

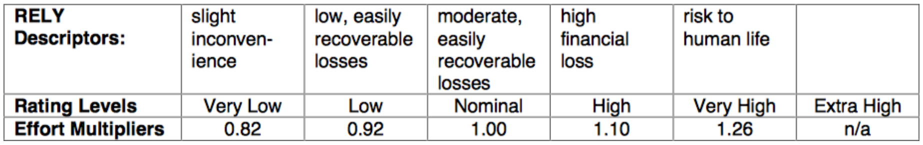
Scale Driver	Factor	Value
PREC	LOW	4.96
FLEX	LOW	4.05
RESL	HIGH	2.83
TEAM	VERY HIGH	1.10
PMAT	Level 2	4.68
Total		17.62

2.2.2 Cost Drivers

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. There are five product factors, and complexity has the strongest influence on estimated effort.

Required Software Reliability (RELY)

This is the measure of the extent to which the software must perform its intended function over a period of time. A malfunctioning in this software could lead to important financial losses because some of our client could select another company that offers the same product. This value is HIGH.

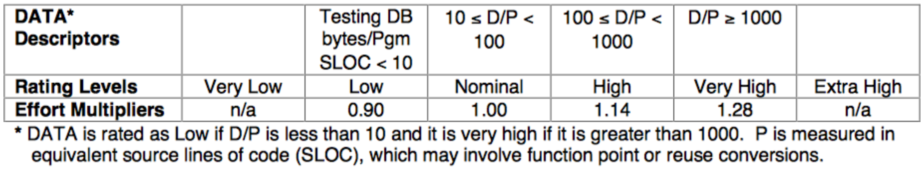


RELY Descriptors:	slight inconvenience	low, easily recoverable losses	moderate, easily recoverable losses	high financial loss	risk to human life	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	0.82	0.92	1.00	1.10	1.26	n/a

Figure 2: RELY

Data Base Size (DATA)

DATA is capturing the effort needed to assemble and maintain the data required to complete test of the program. At this stage is very difficult to estimate the size of the DB but an estimation should be around 4GB. The ratio D/P where D is the size in bytes of the database and P is the SLOC estimation is around 400, so this value is 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

* DATA is rated as Low if D/P is less than 10 and it is very high if it is greater than 1000. P is measured in equivalent source lines of code (SLOC), which may involve function point or reuse conversions.

Figure 3: DATA

Product Complexity (CPLX)

- Control Operation: HIGH because there could be highly nested structured programming operators with many compound predicates, queue and stack control, homogeneous, distributed processing. This is because there is the need to manage many request by the users about the status of the cars.

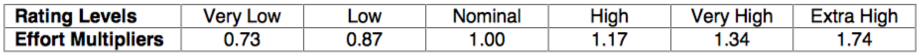
- Computational Operator: LOW because there could be an evaluation of moderate-level expressions: e.g., D=SQRT(B**2- 4.*A*C). This is due to the calculation of time and costs.

- Device-Dependent Operations: VERY LOW. There are simple read, write statements with simple formats.

- Data Management Operations: HIGH. There are simple triggers activated by data stream contents.

- UI Management Operations:LOW. Use of simple graphic user interface (GUI) builders.

At the end of the complexity analysis this could be set to NOMINAL as weighted average.



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

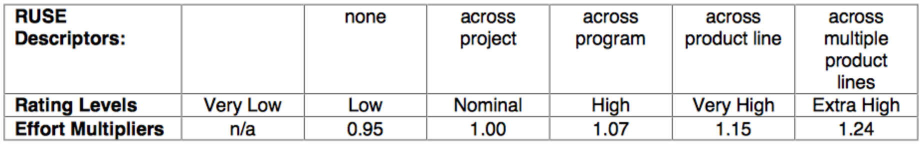
Figure 4: CPLX

Developed for Reusability (RUSE)

This cost driver accounts for the additional effort needed to construct components intended for reuse on current or future projects.

Development for reusability imposes constraints on the project's RELY and DOCU ratings. The RELY rating should be at most one level below the RUSE rating. The DOCU rating should be at least Nominal for Nominal and High RUSE ratings, and at least High for Very High and Extra High RUSE ratings.

In PowerEnJoy case Reusability is not very important since it isn't in the requirements of the project. This means that this value is set to NOMINAL.

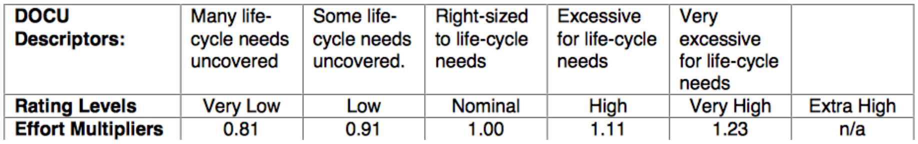


RUSE Descriptors:		none	across project	across program	across product line	across multiple product lines
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	0.95	1.00	1.07	1.15	1.24

Figure 5: RUSE

Documentation Match to Life-Cycle Needs (DOCU)

Attempting to save costs via Very Low or Low documentation levels will generally incur extra costs during the maintenance portion of the life-cycle. For this reason this value is set to NOMINAL.



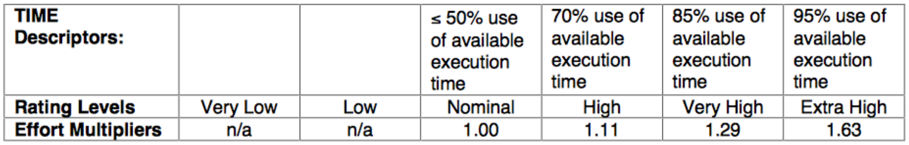
DOCU Descriptors:	Many life-cycle needs uncovered	Some life-cycle needs uncovered.	Right-sized to life-cycle needs	Excessive for life-cycle needs	Very excessive for life-cycle needs	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	0.81	0.91	1.00	1.11	1.23	n/a

Figure 6: DOCU

Execution Time Constraint (TIME)

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.

Since Power Enjoy is a quite big project this level is set to HIGH.

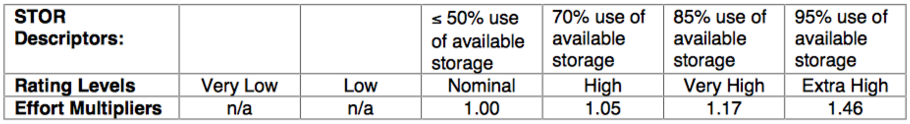


TIME Descriptors:			≤ 50% use of available execution time	70% use of available execution time	85% use of available execution time	95% use of available execution time
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	n/a	1.00	1.11	1.29	1.63

Figure 7: TIME

Main Storage Constraint (STOR)

This rating represents the degree of main storage constraint imposed on a software system or subsystem. This value is set to HIGH since the amount of storage required could grow up based on the users quantity.

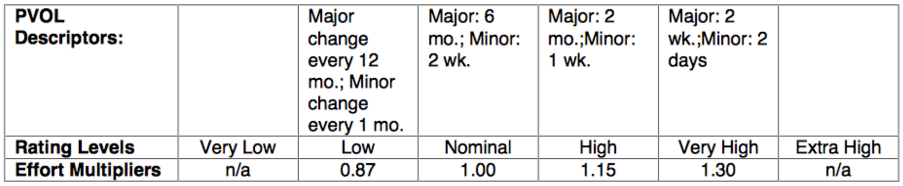


STOR Descriptors:			≤ 50% use of available storage	70% use of available storage	85% use of available storage	95% use of available storage
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	n/a	1.00	1.05	1.17	1.46

Figure 8: STOR

Platform Volatility (PVOL)

In this project there could be a Major Release every 6 month caused by changes in the platform (Mobile OS) and a Minor release every two weeks for bug fixing. So this value is set to NOMINAL.



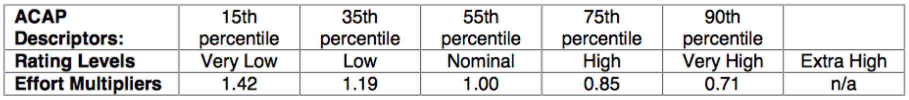
PVOL Descriptors:		Major change every 12 mo.; Minor change every 1 mo.	Major: 6 mo.; Minor: 2 wk.	Major: 2 mo.;Minor: 1 wk.	Major: 2 wk.;Minor: 2 days	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	n/a	0.87	1.00	1.15	1.30	n/a

Figure 9: PVOL

Analyst Capability (ACAP)

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.

Analysts of the organization are prepared and estimated as 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

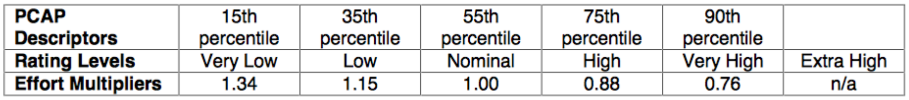
Figure 10: ACAP

Programmer Capability (PCAP)

Evaluation is based on the capability of the programmers as a team rather than as individuals.

Major factors considered in the rating are ability, efficiency and thoroughness, and the ability to communicate and cooperate.

The programmer capability of programmers of the organization are HIGH since they are used to work together.

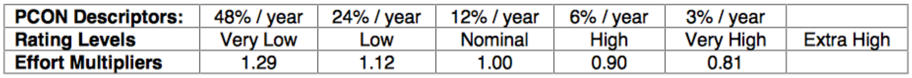


PCAP Descriptors	15th percentile	35th percentile	55th percentile	75th percentile	90th percentile	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.34	1.15	1.00	0.88	0.76	n/a

Figure 11: PCAP

Personnel Continuity (PCON)

The rating scale for PCON is in terms of the project's annual personnel turnover. Since this project is the unique project for our organization this value is set to HIGH.

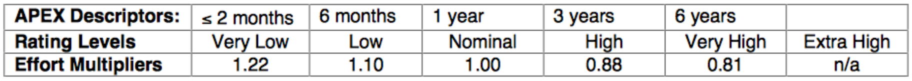


PCON Descriptors:	48% / year	24% / year	12% / year	6% / year	3% / year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.29	1.12	1.00	0.90	0.81	

Figure 12: PCON

Applications Experience (APEX)

JEE applications are a new kind of applications for programmers of our organization. For this reason this parameter is set to VERY LOW.

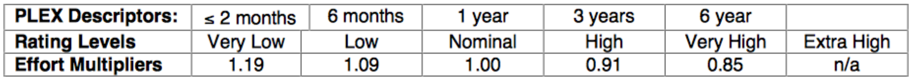


APEX Descriptors:	≤ 2 months	6 months	1 year	3 years	6 years	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.22	1.10	1.00	0.88	0.81	n/a

Figure 13: APEX

Platform Experience (PLEX)

JEE platform is a new platform for programmers of our organization but they are experienced in DB , networking and GUI. For this reason this value is set to NOMINAL.

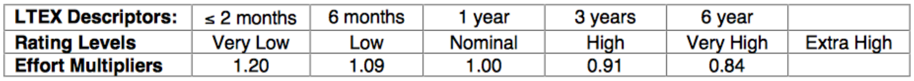


PLEX Descriptors:	≤ 2 months	6 months	1 year	3 years	6 year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.19	1.09	1.00	0.91	0.85	n/a

Figure 14: PLEX

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. This value is assumed to be NOMINAL for our project.

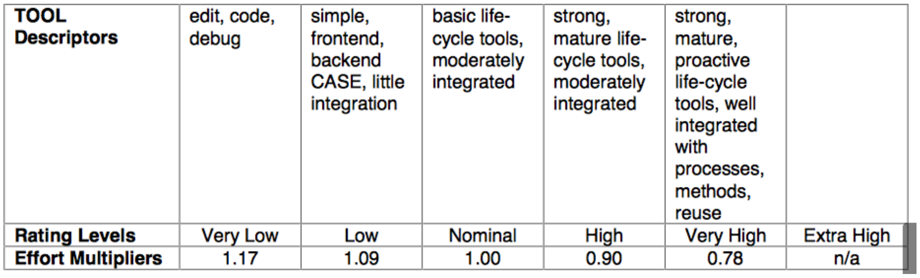


LTEX Descriptors:	≤ 2 months	6 months	1 year	3 years	6 year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.20	1.09	1.00	0.91	0.84	

Figure 15: LTEX

Use of Software Tools (TOOL)

The tool rating ranges from simple edit and code, very low, to integrated life-cycle management tools, very high. In this project there is the need of integrated life-cycle management tools so this value is set to HIGH.

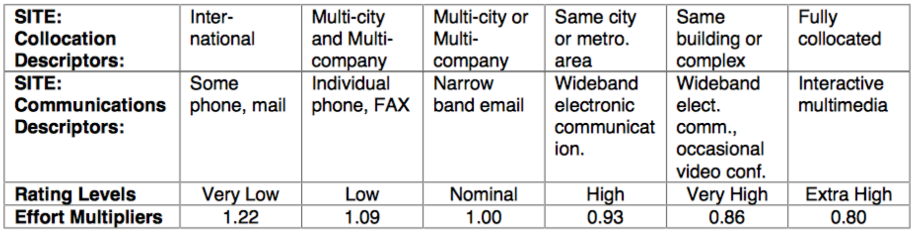


TOOL Descriptors	edit, code, debug	simple, frontend, backend CASE, little integration	basic life-cycle tools, moderately integrated	strong, mature life-cycle tools, moderately integrated	strong, mature, proactive life-cycle tools, well integrated with processes, methods, reuse	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.17	1.09	1.00	0.90	0.78	n/a

Figure 16: TOOL

Multisite Development (SITE)

The team is located in the same metro area so this value is set to HIGH.



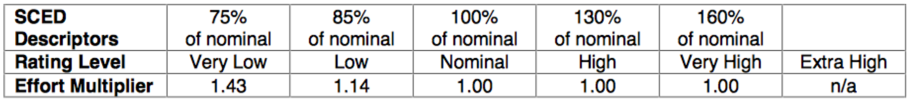
SITE: Collocation Descriptors:	Inter-national	Multi-city and Multi-company	Multi-city or Multi-company	Same city or metro. area	Same building or complex	Fully collocated
SITE: Communications Descriptors:	Some phone, mail	Individual phone, FAX	Narrow band email	Wideband electronic communication.	Wideband 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

Figure 17: SITE

Required Development Schedule (SCED)

This rating measures the schedule constraint imposed on the project team developing the software. Accelerated schedules tend to produce more effort in the earlier phases to eliminate risks and refine the architecture, more effort in the later phases to accomplish more testing and documentation in parallel. Stretch-outs do not add or decrease effort. Their savings because of smaller team size are generally balanced by the need to carry project administrative functions over a longer period of time.

SCED is set to NOMINAL.



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

Figure 18: SCED

Overall Result:

Overall results are summarized in the following table:

Cost Driver	Factor	Value
RELY	HIGH	1.10
DATA	HIGH	1.10
CPLX	NOMINAL	1.00
RUSE	NOMINAL	1.00
DOCU	NOMINAL	1.00
TIME	HIGH	1.11
STOR	HIGH	1.05
PVOL	NOMINAL	1.00
ACAP	HIGH	0.85
PCAP	HIGH	0.88
PCON	HIGH	0.90
APEX	VERY LOW	1.22
PLEX	NOMINAL	1.00
LTEX	NOMINAL	1.00
TOOL	HIGH	0.90
SITE	HIGH	0.93
SCED	NOMINAL	1.00
Total		0.9694

2.2.3 Effort Equation

The Effort is given by the following equation:

$$Effort = A \cdot EAF \cdot KSLOC^E \tag{1}$$

Where :

$A = 2.94$ for COCOMO II

$EAF = \prod EM_i$ (where EM is the effort multiplier for each Cost Driver)

$E = B + 0.01 \cdot \sum SF_i$ (where $B = 0.91$ for COCOMO II and SF are the Scale Factors)

After our analysis:

$EAF = 0.97$

$E = 0.91 + 0.01 \cdot 17.62 = 1.0862$

And now the Effort value can be computed, with a lower bound:

$$Effort = 2.94 \cdot 0.9694 \cdot 2.235^{1.0862} = 6,827 \approx 7PM$$

And the upper bound is:

$$Effort = 2.94 \cdot 0.9694 \cdot 9,983^{1.0862} = 34.69 \approx 35PM$$

2.2.4 Schedule Estimation

The final schedule estimation is given by the following formula:

$$Duration = 3.67 \cdot Effort^{SE} \tag{2}$$

Where SE is the schedule equation component derived from the scale factors:

$$SE = 0.28 + 0.2 \cdot (E - B) = 0.32524$$

The lower bound for the duration is:

$$Duration = 3.67 \cdot 6.827^{0.32524} = 6.8547 months$$

As an upper bound:

$$Duration = 3.67 \cdot 34.69^{0.32524} = 11 months$$

3 Schedule

In this section is provided a general schedule of the PowerEnjoy project, that has not to be considered .

4 Resource Allocation

5 Risk Management

This section address the problem of finding the main risks that can incur during the project development.

In order to characterize risks they are divided into categories.

5.1 Project Risks

Product Size Risks

Product Size risks are related to the size of the project.

The size of the project in terms on SLOC could be an underestimation. This could cause a reduced scheduling and to a reduced cost estimation. A possible strategy for tackle this risk could be planning for two releases, the first one with a limited set of functionalities and the second with the full set of functionalities.

The number of user is an important factor because an underestimation of the number of user. If the actual number of user is many times the number of expected user could be a problem. This problem must be taken into account in the definition of the software architecture. The architecture must be modular and must support extensions.

People Management Risks

These risks are related to the management of people involved in the project. Developers of the company need to be part of the whole, involved in important decision regarding technology or approach to problems. If developers become un-motivated, they become a risk because the whole project (and the future projects) could be compromised.

Key points are:

- Consistency
- Respect
- Inclusion
- Honesty

Unfeasible feature

Some feature requested by the customer could be unfeasible and this could be discovered only at the developing time. In order to avoid this risk all requirements must be analyzed in a proper way by technical experts. Unfeasible requirements must be discovered at an early stage of the project and there must be a discussion with the customer.

Acceptance of the System

Not acceptance of the system from users and company could be a risk. This product could be difficult to use by people not used to these services. In order to avoid this risk the market strategy is very important. The launch of the product might be correlated with discounts, tutorials or special offers.

5.2 Technical Risks

CarOS Risks

Technical risks can be caused by a lack of support by the carOS provider. The carOS provider could terminate the support for its technology. In this case the organization should search another provider of the same technology or thinking about developing its own technology. For developing a new technology could be reasonable to hire some experts in embedded technology since could be difficult and could require some knowledge.

Software dependency Risks

The reason made for the carOS problem should be extended also for the other software dependencies like driving license validation and payment. The contingency plan could be also the same: change dependency or developing a new technology that does the same things.

Technology to be built

The technology to use could be new for team members and the difficulty of development can be underestimate. Also with the selected technology some functionality of the project could be difficult to implement. There can be new technologies more effective and useful. For this reason all the team members need to follow technology trends. They need to follow also upgrading courses in order to improve competencies and achieve an high level of competencies in other technologies not used in the project.

5.3 Business Risks

The followings are business risks that for definition can't have a recovering strategy. They are related to market changes or environmental changes.

Strategic Risks

If the development of the product will be slow could happen some strategic risk. The market car sharing could become saturated and the product could not fit anymore the strategy of the company.

Market Risks

Another risk could be a change in the behavior of our stakeholders. Car sharing could be no more useful because of other services more efficient and less expensive.

Budget Risks

The budget could become reduced, this could cause some issues related to the quality of the software to be.

5.4 Risks Summary

In the following tables risks are summarized, a probability is given to each risks and the impact is estimated. Notice that the strategy is not reported because is already described in the related section.

Project Risks

Risk	Probability	Impact
Product Size	High	Average
People Management	Average	High
Unfeasible Feature	Low	High
Acceptance of the system	Average	High

Technical Risks

Risk	Probability	Impact
Car OS	Low	High
Software Dependencies	Low	High
Technology to be built	Low	High

Business Risks

Risk	Probability	Impact
Strategic	Low	High
Market	Average	High
Budget	Low	Average

6 Appendix

6.1 Hours of Work

Emanuele Ghelfi:

- 17/01/17: 4 h, Overview of the problem, COCOMO
- 18/01/17 2 h, Risks Managements
- 19/01/17: 2 h, Introduction

Total hours: 28 h

Emiliano Gagliardi:

- 17/01/17: 5 h, Size estimation

Total hours: 28 h

6.2 Used Tools

The tools used to create this RASD document are:

- Github: for version control.
- Lyx: to redact and organize this document.
- StarUML: to create UML diagrams (component diagram, data model diagram, deployment diagram, sequence diagram, state-chart diagram, use case diagrams).
- Excel: to compute formulas.

References

[1] COCOMO || Model Definition Manual

[2] http://www.ifpug.org