Version 1.4

Software Engineering Project

Project Management Document

POLITECNICO
MILANO 1863

Vianney Payelle - Rémi Rigal - Noëlie Ramuzat

Revision Notice

Versions	Date	Description	Modifications
V1.1	16/01/2017	Creation of the document	Adding of the Function Points part
V1.2	18/01/2017	Creation of : • The COCOMO part • The Introduction part	Update of the Function points part
V1.3	21/01/2017	Creation of : • The Task Schedule part • The Resources Allocation part	Update of the Function points part
V1.4	22/01/2017	• Creation of the Risks Management part • Validation of the document	Update of the COCOMO part

Table 1 : Testing Document versions

Table of contents

List of tables

List of figures

1. Introduction

1.1. Purpose

The Project Management Document provides a planning estimation and schedule of the PowerEnjoy project development. It also presents a proposal of task assignment and minimal needed budget. The aim of this document is to be a complete support for the project manager to ensure that the software will be delivered in time.

For that purpose, it requires to take into account the context, requirements and the interaction of economic, social, technical and organizational aspects.

1.2. Scope

The scope of the project PowerEnjoy, which is a service based on mobile application, is to manage, design, build, and implement a service aimed at facilitating public transportation. The application provides to its target, the client, a way to research an electric car near a position, reserve it and pick it for a ride. At the end the application sends the ride's bill to the client. The PowerEnjoy application needs the client to be registered in its database before he can reserve a car, for security and payment reasons (credential information, driver license, and identity card). When the client logs in, the mobile application allows him to reserve a car around an address or his GPS position. Then it provides him details about his reservation on the main page. The application also allows the client to cancel his reservation, unlock the reserved car when he is near it, and access his account details to modify it.

After the ride the application locks the car and sends an email to the client with the bill of the ride. The mobile application can moreover give discounts and charges in function of the client's ride such as sharing the car or plug it in power grid station. The PowerEnjoy application is built in order to ensure an easy and clear reservation service as well as an effective data collecting and saving. This refers the PowerEnjoy application simplifies the customer's uses, optimises the time to reserve a car and adjusts the price of the ride according to the driver.

1.3. Definitions, Acronyms, Abbreviations

Name	Definition
RASD	Requirements Analysis and Specifications Document
DD	Design Document
BCE	Business Controller Entity
ITD/ ITPD	Integration Testing / Test Plan Document
DBSM	Database Management System
API	Application Programming Interface.
SMS	Short message service: notification sent to a mobile phone SMS gateway is needed to use it
OS	Operating Systems

Table 2 : Glossary of the Testing Document

1.4. Reference documents

Name	Publication Date	Authors	Contents
Assignment AA 2016-2017 Software Engineering 2	14/10/2016	Elisabetta di Nitto	Project goal, schedule and rules
RASD	16/11/2016	Rémi Rigal Vianney Payelle Noëlie Ramuzat	Requirements Analysis and Specifications Document of the project
DD	15/12/2016	Rémi Rigal Vianney Payelle Noëlie Ramuzat	Design Document of the project
ITPD	15/01/2017	Rémi Rigal Vianney Payelle Noëlie Ramuzat	Integration Test Plan Document

Table 3 : Reference documents used in the Testing Document

1.5. Used Tools

Name	Use
Github & SourceTree	Control the document versions
Edraw Max 8.4	Create the Architecture models
Adobe Acrobat Reader DC	Create the Integration document PDF

Table 4 : Description of the tools used to create the Testing document

1.6. Document structure

The document is scheduled in the following way:

Project size, cost and effort estimation

In this section are describing the Function Points of the PowerEnjoy project, to estimate its size, and the estimations of the effort and cost needed to achieve the project, with COCOMO II.

Tasks Schedule

This part presents the different tasks of the project and their schedule according to the results found in the previous part, so that each part of the project is covered.

Resource Allocation

The third part allocates each member of our group to the various tasks, according to our availability and skills.

Risk Management

In this section are given the risks that the PowerEnjoy project may encounter and their relevance. Then some general recoveries will be given.

Finally the hours of work repartition follows this section.

2. Project size, cost and effort estimation

In this section are described the estimations made to calculate the cost, the size and the effort needed to produce the PowerEnjoy software system. To achieve the size estimation, the Function Points approach is chosen. It permits to characterize the dimension of the PowerEnjoy software by using its major functionalities and the average code line produced, here, in Java.

To perform this point, only the business logic part was used because the user interface one is less representative of the project. Indeed, from a general point of view, it is only composed of the connections to the API and the creation of the interface.

Secondly, concerning the cost and effort estimation, the COCOMO II model is applied. By taking into account the characteristics of product and process of the system it gives back a result based on statistical variables.

2.1. Size estimation: function points

Function type	Weight		
	Low	Average	High
Inputs	3	4	6
Outputs	4	5	7
Inquiry	3	4	6
ILF	7	10	15
ELF	5	7	10

2.1.1. Internal Logic Files (ILFs)

Drivers	1 element to record (email) and less than 20 data element	**low complexity**
Cars	1 element to record (email) and less than 20 data element	**low complexity**
Invoices	1 element to record (email) and less than 20 data element	**low complexity**
Ride Information	1 element to record (email) and less than 20 data element	**low complexity**
Research Information	1 element to record (email) and less than 20 data element	**low complexity**
Reservation Information	1 element to record (email) and less than 20 data element	**low complexity**

2.1.2. External Logic Files (ELFs)

The application has to communicate with the embedded system of the car to retrieve information and send orders. The communications with the embedded car system can be considered as an **average complexity**.

2.1.3. External Inputs (EIs)

The application may allow the customer to login, logout, create and modify an account. And also search for a car and make a reservation. Those actions: login/logout/create-modify account/search a car/make a reservation have respectively complexity of **low complexity** x2, **average complexity** x2 and **high complexity** x2.

The company have to manage the cars and the drivers' information. And also modify the rate of the rides if needed. The car and driver management have an **average complexity** and change the ride's rate have a **low complexity.**

2.1.4. External Inquiries (EQs)

The customer can request to check the profile of their account and the history of his rides. Checking the profile or the ride history are actions of **low complexity**. The company can request to check the cars and customers data. The car & customer totalize actions of **average complexity**.

2.1.5. External Outputs (Eos)

The system has to communicate with the sms/push gateways and the bank services. Using the gateways are **low complexity** actions. The creation of invoices and communication with bank services are **high complexity** actions.

2.1.6. Overall estimation

Software analysis				Total
Inputs	3	4	2	37
Outputs	3	0	1	19
Inquiry	2	2	0	14
ILF	6	0	0	42
ELF	0	1	0	7
			Total	119

Considering that Java will be used to code the system, we can estimate the total amount of line of codes by multiplying the total function points by the conversion rate factors; therefore we obtain those high and low estimations:

High estimation	Low estimation
4998	7973

2.2. Cost and effort estimation: COCOMO II

2.2.1. Scale Factors

	Very Low	Low	Nominal	High	Very High	Extra High	Average
Precedentness	6,2	4,96	3,72	2,48	1,24	0	
#Elements	1	0	1	0	0	2	2,48
Flexibility	5,07	4,05	3,04	2,03	1,01	0	
#Elements	0	0	2	0	0	1	2,02666667
Resolution	7,07	5,65	4,24	2,83	1,41	0	
#Elements	0	3	4	0	0	0	4,84428571
Team cohesion	5,48	4,38	3,29	2,19	1,1	0	
#Elements	0	0	0	1	1	0	1,645
Process Maturity	7,8	6,24	4,68	3,12	1,56	0	
#Elements	0	0	0	1	0	0	3,12
						SF Total	**14,1159524**

Precedentness

Table 11. Precedentedness Rating Levels

Feature	Very Low	Nominal / High	Extra High
Organizational understanding of product objectives	General	Considerable	Thorough
Experience in working with related software systems	Moderate	Considerable	Extensive
Concurrent development of associated new hardware and operational procedures	Extensive	Moderate	Some

For our project we have a considerable understanding (Nominal), moderate experience (very low) and some concurrent procedures (Extra high). We can also assume than the need for innovative procedures is minimal (Extra high).

Flexibility

Table 12. Development Flexibility Rating Levels

Feature	Very Low	Nominal / High	Extra High
Need for software conformance with pre-established requirements	Full	Considerable	Basic
Need for software conformance with external interface specifications	Full	Considerable	Basic
Combination of inflexibilities above with premium on early completion	High	Medium	Low

For the flexibility, we have to comply with usual norms, car's components and mobile applications norms. Thus the needs for software conformance are considerable (Nominal x2) and premium need on early completion are low (Extra high).

Architectural/risk resolution

Table 13. RESL Rating Levels

Characteristic	Very Low	Low	Nominal	High	Very High	Extra High
Risk Management Plan identifies all critical risk items, establishes milestones for resolving them by PDR or LCA.	None	Little	Some	Generally	Mostly	Fully
Schedule, budget, and internal milestones through PDR or LCA compatible with Risk Management Plan.	None	Little	Some	Generally	Mostly	Fully
Percent of development schedule devoted to establishing architecture, given general product objectives.	5	10	17	25	33	40

Table 13. RESL Rating Levels

Characteristic	Very Low	Low	Nominal	High	Very High	Extra High
Percent of required top software architects available to project.	20	40	60	80	100	120
Tool support available for resolving risk items, developing and verifying architectural specs.	None	Little	Some	Good	Strong	Full
Level of uncertainty in key architecture drivers: mission, user interface, COTS, hardware, technology, performance.	Extreme	Significant	Consider-able	Some	Little	Very Little
Number and criticality of risk items.	> 10 Critical	5-10 Critical	2-4 Critical	1 Critical	> 5Non-Critical	< 5 Non-Critical

Our Risk Management Plan includes some risk items and solution, a schedule with milestone has been defined and budget is partially calculated using COCOMO, so it is partially covered (Nominal x2). We can assume that 10% of time (low) has been devoted to establish the architecture (RASD + DD). There is some tool support within the framework (Nominal). Given the experience of the authors of design documents, we can assume that the level of uncertainty is considerable (nominal). There are between 5 and 10 critical item (low).

Team cohesion

Table 14. TEAM Rating Components

Characteristic	Very Low	Low	Nominal	High	Very High	Extra High
Consistency of stakeholder objectives and cultures	Little	Some	Basic	Consider-able	Strong	Full
Ability, willingness of stakeholders to accommodate other stakeholders' objectives	Little	Some	Basic	Consider-able	Strong	Full
Experience of stakeholders in operating as a team	None	Little	Little	Basic	Consider-able	Extensive
Stakeholder teambuilding to achieve shared vision and commitments	None	Little	Little	Basic	Consider-able	Extensive

We can assume given the few numbers of stakeholders and their common origins, there is a strong consistency in objectives and cultures (very high). They know each other quite well and they all want to achieve the project so they can considerably accomodate to other stakeholders' objectives (high). This is not their first project with other stakeholders so they can basically act as a team (high). And so the teambuilding is considerable (very high).

Process maturity

Table 15. PMAT Ratings for Estimated Process Maturity Level (EPML)

PMAT Rating	Maturity Level	EPML
Very Low	CMM Level 1 (lower half)	0
Low	CMM Level 1 (upper half)	1
Nominal	CMM Level 2	2
High	CMM Level 3	3
Very High	CMM Level 4	4
Extra High	CMM Level 5	5

According to a SW-CMM level of the company of 3 the rating is high.

2.2.2. Cost Drivers

	Very Low	Low	Nominal	High	Very High	Extra High	Average
RELY	0,82	0,92	1	1,1	1,26		
#Elements	0	0	0	1	0	0	1,1
DATA		0,9	1	1,14	1,28		
#Elements	0	0	0	1	0	0	1,14
RUSE		0,95	1	1,07	1,15	1,24	
#Elements	0	0	1	0	0	0	1
DOCU	0,81	0,91	1	1,11	1,23		
#Elements	0	0	0	1	0	0	1,11
TIME			1	1,11	1,29	1,63	
#Elements	0	0	0	0	1	0	1,29
STOR			1	1,05	1,17	1,46	
#Elements	0	0	1	0	0	0	1
PVOL		0,87	1	1,15	1,3		
#Elements	0	1	0	0	0	0	0,87
ACAP	1,42	1,19	1	0,85	0,71		
#Elements	0	0	1	0	0	0	1
PCAP	1,34	1,15	1	0,88	0,76		
#Elements	0	0	1	0	0	0	1
PCON	1,29	1,12	1	0,9	0,81		
#Elements	0	0	0	0	1	0	0,81
APEX	1,22	1,1	1	0,88	0,81		
#Elements	0	0	0	1	0	0	0,88
PLEX	1,19	1,09	1	0,91	0,85		
#Elements	0	0	0	0	1	0	0,85
LTEX	1,2	1,09	1	0,91	0,84		
#Elements	0	0	0	1	0	0	0,91
TOOL	1,17	1,09	1	0,9	0,78		
#Elements	0	0	0	1	0	0	0,9
SITE	1,22	1,09	1	0,93	0,86	0,8	
#Elements	0	0	0	0	0	1	0,8
SCED	1,43	1,14	1	1	1		
#Elements	0	0	1	0	0	0	1
CPLX	0,73	0,87	1	1,17	1,34	1,74	
#Elements	0	0	2	1	2	0	1,17
						EM product	**0,7255642**

- **Product factors**

Reliability (RELY)

The reliability of the software can ensure high financial loss like losing a car so the rating level is high.

Data base size (DATA)

There is no defined testing database as we may use a simulator for testing, but we can assume that the cost to develop a simulator is quite similar to a large database so rating level is high.

Reusability (RUSE)

The reusability of components will be fixed to the project itself for some piece of program that will be close enough for cars and users. So we can use a nominal rating level.

Documentation (DOCU)

As the project could be extended to some other cities, an excessive for life-cycle needs documentation is asked in order to improve it easily so this is a high rating level.

- ## Platform factors

Execution time constraints (TIME)

As we may not make our customer waiting too much, at least 85% use of available execution time should be reached so we have a very high rating level.

Main storage constraints (STOR)

There is no remarkable constraint about the use of storage so a nominal rating value seems fine.

Platform volatility (PVOL)

As we are using JEE we may not have major releases more than every year so a low rating level.

- ## Personal factors

Analyst capability (ACAP)

We will consider that we have average analyst, so from the 55th percentile and a nominal rating level.

Programmer capability (PCAP)

We will consider that we have average programmer, so from the 55th percentile and a nominal rating level.

Personnel continuity (PCON)

The personnel have nice environment to work and the management is flexible so there is a low rate of turnover: less than 3%, a very high rating level.

Application experience (APEX)

We have many experienced developers and few new so the average APEX descriptors is about 3 years and a high rating level.

Platform experience (PLEX)

They are all using this platform since many years so the average PLEX descriptors are 6 years and a very high rating level.

Language and tool experience (LTEX)

We have many experienced developers and few new so the average LPEX descriptors is about 3 years and a high rating level.

- ## Project factors

Use of software tools (TOOL)

The tools are well know from the teams and well integrated so we have a high rating level.

Multisite development (SITE)

Our team are fully collocated with a lot of communication so this is an extra high rating level.

Required development schedule (SCED)

The scheduled have been well planned so we have nominals SCED descriptors.

Complexity (CPLX)

The average complexity of control operations imply some very high and extra high rating level (especially for car control) but this is not the case for most of the parts of the software. Many nested structure programming will be involved, so the average rating level can be set to very high.

Our system may not involve strong analysis of elements so the average rating level of computational operations can be set to nominal. Our software will manage many device dependent operations (cars, smartphones) but they do not require high complexity so a high rating level is enough. The data management operation is very important in our system, especially because it involve search and repartition optimisation. Thus a very high rating level can be applied. The user interface management operation will be quite simple as smartphone interface are well known so it will be a nominal rating level.

2.2.3. Effort equation

The effort equation in Person-Months is:

Effort = A * EM * KSLOC^{E}

Where:

A= 2.94 for COCOMO II

EM = 0.73

E = B + 0.01 * SUM(SF) = 1.051 exponent derived from the scale factor with B = 0.91 for COCOMO II

With the Lower Bound:

KSLOC = 4.998

Effort = 11.58 Person-Month

With Upper Bound:

KSLOC = 7.973

Effort = 18.91 Person-Month

2.2.4. Schedule estimation

Considering the general schedule, the duration will be:

Duration = 3.67 * Effort^{F}

Where:

F = 0.28 + 0.2 * (E-B) = 0.28 + 0.2 * (0.141) = 0.3082

With the Lower Bound:

Duration = 7.81 Months

With Upper Bound:

Duration = 9.08 Months

3. Tasks Schedule

In this part is given the general PowerEnjoy project schedule. This one presents the different tasks of the project with the estimate time and resources required to complete them. This plan will be regularly revise to take into account the changes (such as requirements, or business goals ones) that may occurs during the evolution of the project and affect the entire project. The schedule has been made to avoid delays and organize concurrency between tasks, however it contains an estimation of the Development, Deployment and Start-up parts conduct because they can't actually be done by us (like the meeting with the stakeholders).

ID	Task Name	Start	Finish	Duration
1	RASD	16/10/16	10/11/16	20.0 d.
2	Goals + Constraints + Assumption	16/10/16	26/10/16	9.0 d.
3	Requirements + Scenarios + Use Cases	27/10/16	2/11/16	5.0 d.
4	Alloy Modelling	27/10/16	9/11/16	10.0 d.
5	Meeting with stakeholders	11/11/16	11/11/16	0.0 d.
6	DD	14/11/16	8/12/16	19.0 d.
7	Architectural Design	14/11/16	22/11/16	7.0 d.
8	Algorithm Design + Requirements tracability	22/11/16	5/12/16	10.0 d.
9	User Interface + BCE	14/11/16	6/12/16	17.0 d.
10	Meeting with stakeholders	9/12/16	9/12/16	0.0 d.
11	ITD	12/12/16	16/1/17	26.0 d.
12	Integration Strategy + Tools & Equipements required	12/12/16	1/1/17	15.0 d.
13	Tests Description	20/12/16	16/1/17	20.0 d.
14	Development	12/12/16	13/7/17	154.0 d.
15	External Components aquisition & study	12/12/16	2/1/17	16.0 d.
16	Unit tests + Code Inspection	11/1/17	2/7/17	123.0 d.
17	Components development	12/12/16	15/3/17	68.0 d.
18	Components integration	16/3/17	18/4/17	24.0 d.
19	Integration + System testings + Corrections + Internal demonstrations	18/4/17	31/5/17	32.0 d.
20	Stakeholders demonstrations + Corrections + User manual	2/6/17	11/7/17	28.0 d.
21	Release	15/7/17	15/7/17	0.0 d.
22	Deployement	16/7/17	15/8/17	23.0 d.
23	System deployement + Meeting with administrators	16/7/17	10/8/17	20.0 d.
24	Mobile phone deployement + Final meetings	16/7/17	15/8/17	23.0 d.
25	Start-up	16/8/17	31/8/17	12.0 d.
26	System first start	16/8/17	16/8/17	0.0 d.
27	Updates	17/8/17	31/8/17	11.0 d.

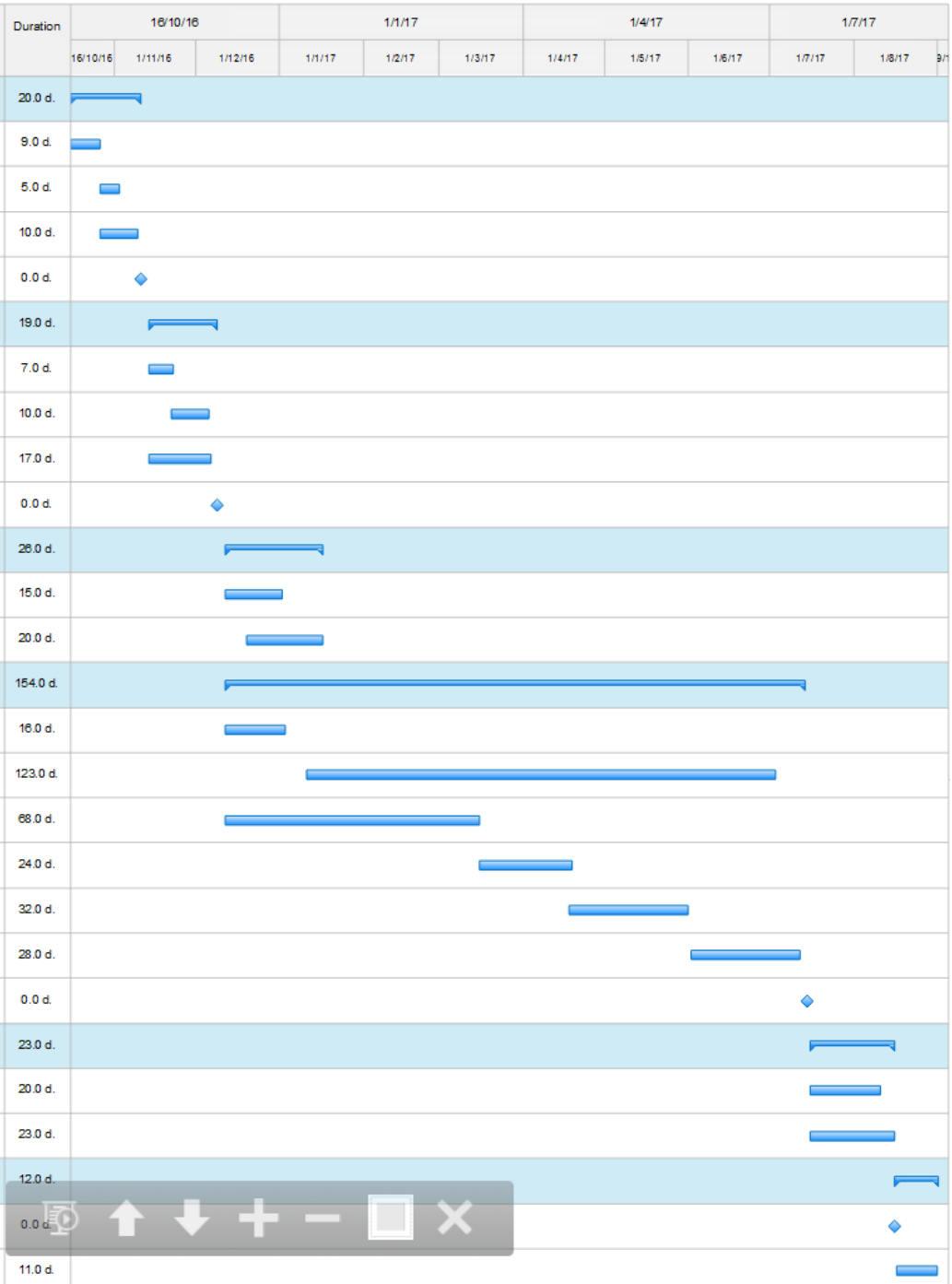

Figure 1 - Task Schedule

4. Resource Allocation

This section presents how the tasks, defined in the previous part, will be assigned to the different members of the PowerEnjoy project team. This allocation includes the Development, Deployment and Start-up parts as the Tasks Schedule to assure a realistic presentation of the project development. Moreover, for the same reason, the ITD tasks have been divided in a different way than what occurs in reality. Finally, for sake of clarity, the RASD and DD tasks have been grouped in two more general parts.

In the Resource Allocation table below, the colours represent the group of tasks defined in the tasks schedule: RASD, DD, ITD, Development, Deployment and Start-up.

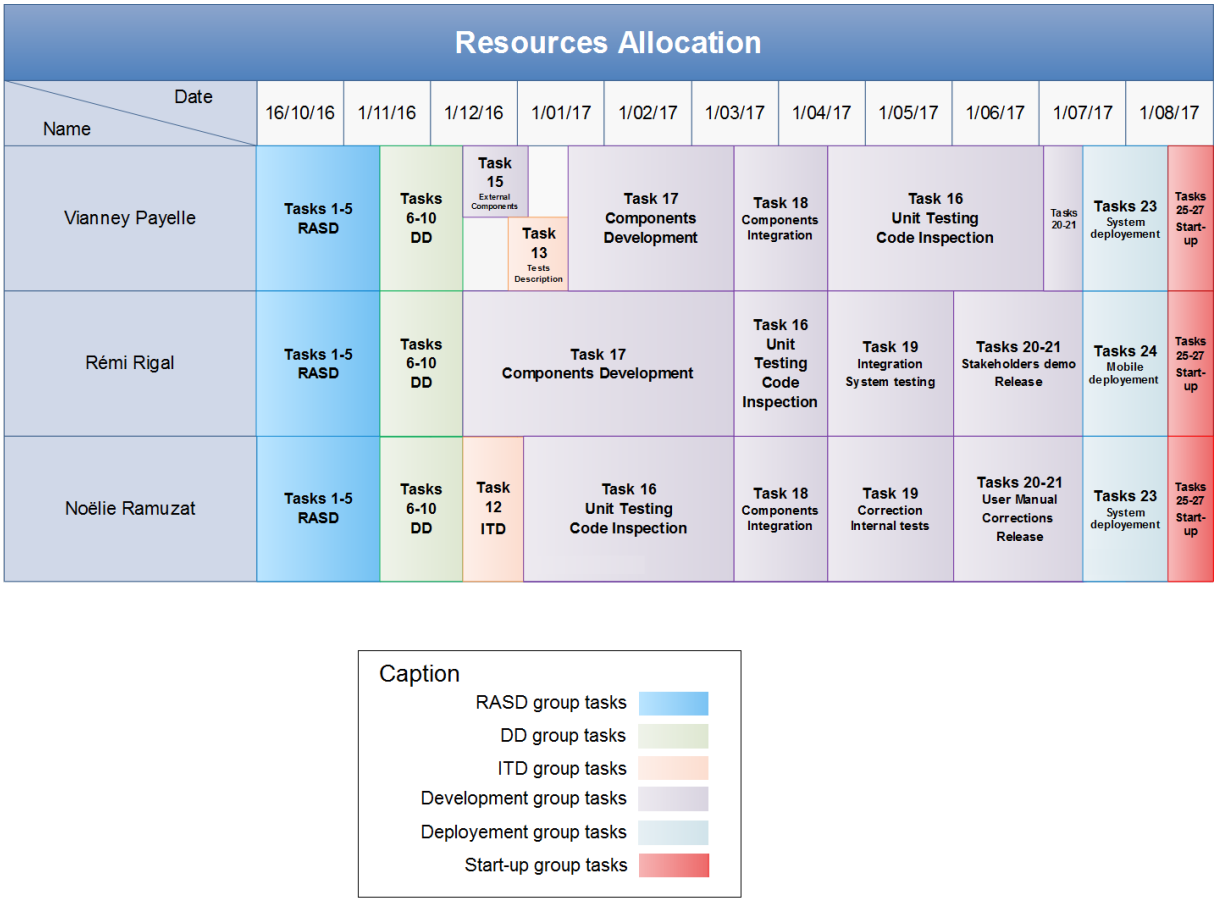

Figure 2 - Resource Allocation

5. Risk Management

All along the development process and after the deployment the project may face several risks; this section will make an exhaustive list of those risks.

5.1. Risks during the development

Our development team is composed of only three members, therefore a big risk can be someone being ill or absent for a long period of time. This type of issue is likely to cause delays, especially if the concerned employee is managing a very specific part of the project. Another related contingency can be someone quitting the company; in this case the resulting delays can be significant if no applicant with enough skills is found to replace the vacant position. The best solution to this is to make sure that every employee has enough responsibility within the project and that everyone is capable of doing all the parts of the project in order to have a flexible team.

Similarly, the versioning systems that will be used must be well organized in order to optimize commits and so gain a significant amount of time for the development process.

Our system also depends on a big panel of external components such as the bank transaction service or the SMS gateway, we then need to be sure that those components will work and be reliable for a reasonable amount of time. An active support and regular updates are also preferable.

If we take into account all these possible sources of delay, it's reasonable to envisage that certain tasks will take longer than expected, especially during some period of the year where there are a lot of holidays. This is why the time allocated to all the different tasks of the project is slightly longer than the one predicted by the COCOMO method.

5.2. Risks after the deployment

Once the PowerEnjoy service is deployed, it can still face many risks such as not being accepted by the taxi driver lobby, in fact, the PowerEnjoy service is providing cars accessible to any individual capable of driving; therefore it causes harm to the taxi drivers by taking off a big part of their customer base. Being a powerful lobby, the best arrangement is to include the union of the taxi drivers in the meeting with the stakeholders in order to find compromise with them.

Another important aspect that we need to take into account for the PowerEnjoy service is the legislative one. Indeed, the laws regarding driving license and cars may be

subject to changes, therefore it is important to stay aware of the new laws that may be problematic for our system. For example the usage of navigation systems with screen may become forbidden, we would then adapt the service that informs the drivers about the power grid positions.

Then, since the PowerEnjoy service is only accessible through a smartphone our customer base is limited. According to the Pew Research Center, in 2015 in Italy, 60% of adults own a smartphone and 88% of the Italians aged between 18 and 34 own one. The risk is that many possible customers will not be able to access the system only because of not owning a smartphone, but the figures are increasing exponentially every year and should not be a concern anymore in several years.

6. Hours of work

6.1. Vianney Payelle

16/01/2017: 2h
18/01/2017: 2h
22/01/2017: 4h

6.2. Rémi Rigal

21/01/2017: 2h
22/01/2017: 3h

6.3. Noëlie Ramuzat

16/01/2017: 30min
18/01/2017: 1h
21/01/2017: 6h
22/01/2017: 2h