POLITECNICO
MILANO 1863

PowerEnJoy
Integration Test Plan Document

Erba Alessandro
Leveni Filippo
Lodi Luca

A.A 2016/2017

Contents

1 Introduction **4**
1.1 Purpose and Scope . . . 4
1.2 List of Definitions and Abbreviations . . . 4
1.2.1 Rent . . . 4
1.2.2 Car Sharing . . . 4
1.2.3 PowerEnJoy . . . 4
1.2.4 Guest User . . . 4
1.2.5 Registered User . . . 4
1.2.6 PowerEnJoy User . . . 4
1.2.7 Staff . . . 5
1.2.8 Third Party Developer . . . 5
1.2.9 Service . . . 5
1.2.10 Safe Area . . . 5
1.2.11 Power Plug . . . 5
1.2.12 Power Plug Slot . . . 5
1.2.13 Parking solution . . . 6
1.2.14 Reservation . . . 6
1.2.15 Stopover . . . 6
1.2.16 Fare . . . 6
1.2.17 Personal Identification Number (PIN) . . . 6
1.3 List of Reference Documents . . . 6

2 Project size, cost and effort estimation **6**
2.1 Size estimation: function points . . . 6
2.1.1 Internal Logic Files (ILFs) . . . 7
2.1.2 External Interface Files (EIFs) . . . 9
2.1.3 External Inputs (EIs) . . . 10
2.1.4 External Inquiries (EQs) . . . 12
2.1.5 External Outputs (EOs) . . . 13
2.1.6 Overall estimation . . . 14
2.2 Cost and effort estimation: COCOMO II . . . 14
2.2.1 Scale Drivers . . . 14
2.2.2 Cost Drivers . . . 15
2.2.3 Effort equation . . . 22
2.2.4 Schedule estimation . . . 23

3 Schedule **24**
3.1 Global . . . 25
3.2 RASD . . . 26
3.3 DD . . . 27
3.4 IT . . . 28
3.5 PP . . . 29
3.6 First release . . . 30
3.7 Second release . . . 31

3.8 Third release 32

4 Resource allocation **33**

5 Risk management **33**
5.1 Business risks 33
5.2 Project risks 35
5.3 Technical risks 35

6 Effort Spent **36**

1 Introduction

1.1 Purpose and Scope

This document will provide the guideline to accomplish the project planning for the *PowerEnjoy* service, it will provide specifically:

- an estimation of the project size
- an estimation of the effort required to develop the project
- the schedule for the project
- the task assignment policy
- description of risk management approach

1.2 List of Definitions and Abbreviations

1.2.1 Rent

A *rent* is the activity beginning with the pick up and ending with the release of a car. It can include some *stopover* .

1.2.2 Car Sharing

car sharing is a model of car rental where people rent cars for short periods of time, often by the hour.

1.2.3 PowerEnJoy

PowerEnjoy is the *car sharing* brand object of this document.

1.2.4 Guest User

A *guest* is a person not logged in to the system.

1.2.5 Registered User

A *registered user* is a person registered to *PowerEnjoy* .

1.2.6 PowerEnJoy User

A *PowerEnjoy User* is a *registered user* can access to the *car sharing* through his smart device.

1.2.7 Staff

staff is the set of people that are *registered user* but can perform special operations. They are divided in three different categories.

- Field *staff*
  - They own a passepartout for cars reporting any issues

    (e.g: cars with low battery).
- Emergency *staff*
  - They manage users issues. They are able to directly contact users.
- Management *staff*
  - They manage the system, configuring its parameters. They can add new *safe areas* , modify *car sharing* fares and create new *staff* accounts.

1.2.8 Third Party Developer

third party developer is a person that can fulfill operations on *PowerEnjoy* system via API.

1.2.9 Service

The *service* is the group of actions that a *PowerEnjoy User* can fulfill trough his smart device.

1.2.10 Safe Area

A *safe area* is a geographical place[1] on the map in which parking is allowed. *safe area* are saved into the system. A *rent* can end parking the car only in safe areas.

1.2.11 Power Plug

A *power plug* is the station that charges the battery of an electric car. All *power plugs* are displayed in the map on-board *PowerEnjoy* cars. Each *power plug* is in a *safe area* . A *PowerEnjoy User* who plugs his car into a *power plug* receives a discount according to *PowerEnjoy* rules.

1.2.12 Power Plug Slot

A *power plug slot* is a parking slot reserved for the *power plug* charge of cars.

[1] defined by a set of geolocation positions

1.2.13 Parking solution

Either a *safe area* or a *power plug slot* .

1.2.14 Reservation

A *reservation* is the action of booking a car, within an hour from its pick up.

1.2.15 Stopover

A *stopover* is a temporary power-off of a rented car. During a *stopover* the *PowerEnjoy User* can leave the car, and re-unlock it later through his smart device. It is subject to a different fare than the rest of rent time.

1.2.16 Fare

A policy to compute price of a rent, complete of involved parameters.
E.g: proportional to time, 0.67€/minute during active rent time

1.2.17 Personal Identification Number (PIN)

personal pin is a number of 5 digits that the user inserts on the car screen to enable driving commands.

1.3 List of Reference Documents

References

[1] Erba A., Leveni F., Lodi L., *Requirement Analysis Specification Document*, 2016.

[2] Erba A., Leveni F., Lodi L., *Design Document*, 2016.

[3] Erba A., Leveni F., Lodi L., *Design Document*, 2017.

2 Project size, cost and effort estimation

2.1 Size estimation: function points

For Internal Logic Files and External Logic Files

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

For External Output and External Inquiry

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

For External Input

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

UFP Complexity Weights

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

2.1.1 Internal Logic Files (ILFs)

PowerEnjoy uses ILF in order to save informations required to offer service functionalities. We have identified at least 10 categories of ILF that will be necessary.

1. **Reservation:** the data about a *reservation* are composed by information about the *PowerEnjoy User* who have made the reservation request, the reservation's target *car info* and the *time* in which the reservation will expire.

2. **Rent:**the data about a *rent* are composed by information about the *PowerEnjoy User* who is doing the *rent* , the info about the *car* the that is the target of the rent, the *kilometers travelled*, the *cost* of the rent, the *discounts* applied for this rent, the

3. **Payment:***PowerEnjoy User* , time stamp, account, payed, description

4. **Car:**The data about the car are GPS position, battery autonomy, availability status, parked, charge level percentage.

5. **User:** All the user related information info, user name, password, *personal pin* , e-mail, Driving License, ID, credit card information, the user status (active or suspended)

6. **Staff:** All the staff related information info, privilege, name, surname, contacts, user name, password, task queue

7. **Map:** The data stored for the map have a Hierarchical structure, in Figure 1 how data are organized.

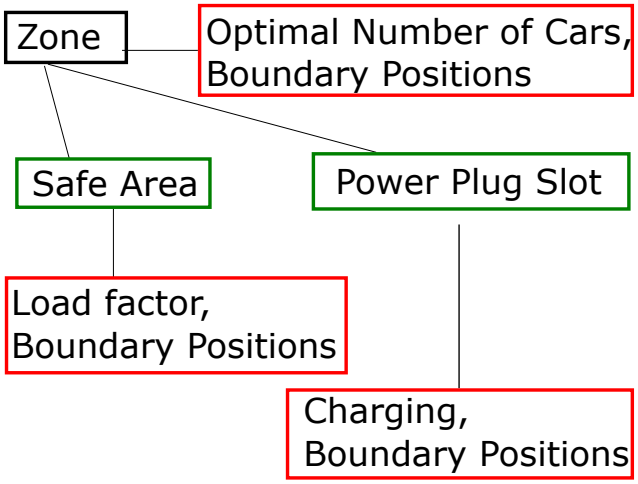


Figure 1: Map data Structure

8. **BPM & Relocations:** Computed Discounts, Pending Relocations, Fuel Consumption Weights, Safe areas info

9. **Fare and Discounts DSL:**Are all the files that contains the DSL scripts and compiled DSL scripts to compute fare and discounts.

10. **API auth Data:** All the informations about external developers are stored adn the data are strutured in a table containing these info App ID, developer info, Detailed list about Available API.

given all the data structures just explained above and using ILF related FP table we can come up with this function points analysis.

ILF	Complexity	FPs
Reservation data	Low	7
Rent data	High	15
Payment data	Low	7
Car data	Avg	10
User data	Low	7
Staff data	Avg	10
Map data	High	15
BPM & Relocations data	Avg	10
Fare and Discounts data	Low	7
API auth data	Low	7
Total		95

2.1.2 External Interface Files (EIFs)

PowerEnjoy uses EIF in order use external services. We have identified at least 5 categories of EIF that will be necessary.

1. **Google Map:** Are the files needed to getthe Google Maps API's working, especially for Maps JavaScript API, Google Maps Geocoding API, Google Places API, GoogleMaps Geolocation API, Google Maps Directions API
2. **Car Navigation:** these are used to compute path, find safe areas, and all the operations needed for the service
3. **Car Controller:** All the info retrieved from the car Controller, these are Car status, sensor data, position, number of passengers. The stucture of this data is complex especially for the sensor data and are continuously send from the cars to the system.
4. **Payment:** Are the data of the credit card that the external pyment service needs in order to perform the payment.
5. **Document Validation:** Are the documents data that are passed to the external service in order to make the validation possible.

given all the data structures just explained and using EIF related FP table we can come up with this function points analysis.

EIF	Complexity	FPs
Google Map	Low	5x3
Car Navigation	Low	5
Car Controller	High	10
Payment	Low	5
Document Validation	Low	5
Total		40

2.1.3 External Inputs (EIs)

PowerEnjoy service supports many kind of interaction features with different categories of users. Especially we have found 6 categories of users.

1. All users:
   - **Login/Logout** simple operation for each type of user.
   - **password retrieval** average complexity operation performed by each type of user, as it involves a number of steps in order to be sure the user is really entitled to retrieve his password. For this reason, it contributes 4 FPs for each type of user.
2. *PowerEnjoy User* :
   - **Reserve** average complexity operation because it includes the elaboration of the route to the car
   - **Cancel** low complexity operation because it include a small operation.
   - **Rent** this operation is divided in sub-operations each one has its contribute to the FP's definition. Each one has a Low complexity apart of Charge operation. It involves the interaction with the power plug slot and it is classified Average.
     - Start
     - Unlock
     - Stopover
     - Park
     - Charge
   - **Register** Classified as high complexity operation because it includes a lot of data to be inserted and involves the call to external services for document verification.
   - **Update info** As the registration it can involve a lot of operations in the backend
   - **Recover Pin** An average complexity procedure because it involves the response by the system and the generation of the new code.
3. External Developers:
   - **New App** A low complexity operation.
4. Emergency Staff:
   - **Assign to Field Staff** classified as average because of the interaction with component that it implies.

- **Set Car Availability** classified as low because it only changes the car status

5. Field Staff:
   - **Unlock Car** Classified as low because they open the car using their passepartout.
   - **Start Engine** Classified as low because it implies only the pin insertion
6. Management Staff:
   - **Modify**
     - *Discount* defined High because it is a complex object to be configured
     - *Fare* defined High because it is a complex object to be configured
     - *City Zone* Defined low because it is not so difficult to be configured
     - *Safe Area* Defined as average because you have to insert set of coordinates
     - *power plug* Defined as average because you have to insert set of coordinates
     - *Car* Defined low because it is not so difficult to be configured
     - *term of conditions* Defined low because it is not so difficult to be configured
     - *staff account* Defined as average because you have to insert define a lot of informations
     - *grant/ remove privilege* defined High because it is a complex object to be configured

EI	Complexity	FPs
Login/Logout	Low	3x12
Password retrieval	Avg	4x6
Reserve	Avg	4
Cancel	Low	3
Money Saving Option	Low	3
Rent	-	-
-Start	Low	3
-Unlock	Low	3
-Stopover	Low	3
-Park	Low	3
-Charge	Avg	4
Register	High	6
Update info	High	6
Recover Pin	Avg	4
New App	Low	3
Assign to field staff	Avg	4
Set Car Availability	Low	3
Unlock car	Low	3
Sart engine	Low	3
Modify	-	-
-Discount	High	6
-Fare	High	6
-city zone	Low	3
-safe area	Avg	4
-power plug	Avg	4
-car	Low	3
-term of conditions	Low	3
-staff account	Avg	4
-grant/ remove privilege	High	6
Total		157

2.1.4 External Inquiries (EQs)

As speci

ed by the FP guidelines, an inquiry is essentially a data retrieval request performed by an user. *PowerEnjoy* supports interactions of this type:

- *PowerEnjoy User* must be able to Retrieve near cars
- *PowerEnjoy User* must be able to Retrieve Reserved Car position
- emergency *staff* must be able to Retrieve Task
- field *staff* must be able to Retrieve Manteinance/Relocation Requests

- field *staff* must be able to Retrieve assigned cars
- emergency*staff* and field *staff* must be able to Retrieve Car info
- *third party developer* developers must be able to Retrieve app privileges
- all users must be able to Retrieve map info

The resulting table is the following:

EQ	Complexity	FPs
Retrive near cars	High	6
Retrieve Reserved Car position	Low	3
Retrieve Task	Avg	4
Retrieve Manteinance/Relocation Requests	Avg	4x2
Retrieve assigned cars	Low	3
Retrieve Car info	Avg	4x2
Retrieve app privileges	High	6
Retrieve map info	Low	3x6
Total		56

2.1.5 External Outputs (EOs)

As part of its normal behavior, *PowerEnjoy* service occasionally needs to communicate with the user outside the context of an inquiry. These occasions are:

- Notify emergency *staff* of the raise of a New Emergency
- Notify Field *staff* of a New Manteinance request
- Notify field *staff* a New Relocation Request
- Notify *PowerEnjoy* users of Updated Fare/Terms Of Service/Discounts
- Display Rent Info in the car
- Notify the *PowerEnjoy User* sending a Reservation Reminder
- Notify Staff for not Payed Rents

EO	Complexity	FPs
New Emergency	Low	4
New Manteinance request	Low	4
New Relocation Request	Low	4
Update Fare/Terms Of Service/Discounts	Low	4x3
Rent Info in cars	Avg	5
Reservation Reminder	Low	4
Suspension for Rent not Payed	High	7
Total		40

2.1.6 Overall estimation

The following table summarizes the results of our estimation activity:

Function Type	Value
Internal Logic Files	95
External Logic Files	40
External Inputs	147
External Inquiries	56
External Outputs	40
Total	378

Given 378 as total FP estimation we calculate the Source Line Of Code using J2EE parameters as follows:

$$\text{SLOC} = 378 * 46 = 17388$$

and an upper bound of

$$\text{SLOC} = 378 * 67 = 25326$$

2.2 Cost and effort estimation: COCOMO II

2.2.1 Scale Drivers

In order to evaluate the values of the scale drivers, we refer to the following official COCOMO II table:

Scale Factor values, SF_j, for COCOMO II Models

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 cooperative	highly cooperative	seamless interactions
SF_j	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	Level 1 Lower	Level 1 Upper	Level 2	Level 3	Level 4	Level 5
SF_j	7.80	6.24	4.68	3.12	1.56	0.00

The results of our evaluation is the following:

Scale Driver	Factor	Value
Precedentedness (PREC)	Nominal	3.72
Development flexibility (FLEX)	High	2.03
Risk resolution (RESL)	Low	5,65
Team cohesion (TEAM)	High	2,19
Process maturity (PMAT)	Low	6,24
Total		19,83

2.2.2 Cost Drivers

- Required Software Reliability:

  Since the system represents the only way to reserve, unlock and use cars it must reliable 24/7 so we set this parameter as High.

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

- Database size:

  This measure attempts to capture the affect large data requirements have on product development. The reason the size of the database is important to consider it because of the effort required to generate the test data that will be used to exercise the program.

  The value will be set to *nominal*.

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

In accordance with the new COCOMO II CPLEX rating scale, this value will be high.

- Product complexity:

  Set to very high according to the COCOMO II rating scale.

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

- Required reusability:

  This cost driver accounts for the additional effort needed to construct components intended for reuse on the 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. We have chosen a *nominal* value because the architecture is quite dependent to the goals required.

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

- Documentation match to life-cycle needs:

  In accordance with the new COCOMO II DOCU rating scale, this value will be *high*.

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

- Execution time constraint:

  This parameter describes the expected amount of CPU usage with respect to the computational capabilities of the hardware. In accordance with the new COCOMO II TIME rating scale, this value will be *high*.

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

- Storage constraint:

  This parameter describes the expected amount of storage usage with respect to the availability of the hardware.In accordance with the new COCOMO II STOR rating scale, this value will be high. this value is set to nominal.

STOR Cost Driver						
STOR Descriptors			$\leq$ 50% use of available storage	70% use of available storage	85% use of available storage	95% use of available storage
Rating level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	n/a	n/a	1.00	1.05	1.17	1.46

- Platform Volatility:

  For what concerns the core system, we don't expect our fundamental platforms to change very often. However, the client applications may require at least a major release once every six months to be aligned with the development cycle of the main mobile operating systems. For this reason, this parameter is set to nominal.

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

- Analyst Capability:

  In accordance with the new COCOMO II ACAP rating scale, this value will be *low*.

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

- Programmer Capability:

  In accordance with the new COCOMO II PCAP rating scale, this value will be *low*.

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

- Application Experience:

  In accordance with the new COCOMO II APEX rating scale, we rate this parameter as *low*.

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

- Platform Experience:

  We don't have any experience with the Java EE platform. For this reason, we're going to set this parameter to *low*.

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

- Language and Tool Experience:

  In accordance with the new COCOMO II LTEX rating scale, this value will be *high*.

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

- Personnel continuity:

  In accordance with the new COCOMO II PCON rating scale, this value will be *high*.

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

- Usage of Software Tools:

  Our application environment is complete and well integrated, so we'll set this parameter as *high*.

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

- Multisite development:

  In accordance with the new COCOMO II SITE rating scale, this value will be *extra high*.

SITE Cost Driver						
SITE Collocation Descriptors	International	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 level	Very low	Low	Nominal	High	Very High	Extra High
Effort multipliers	1.22	1.09	1.00	0.93	0.86	0.80

- Required development schedule:

  In accordance with the new COCOMO II SCED rating scale, this value will be *nominal*.

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

Overall, our results are expressed by the following table:

Cost Driver	Factor	Value
Required Software Reliability (RELY)	High	1.10
Database size (DATA)	Nominal	1
Product complexity (CPLX)	High	1,17
Required Reusability (RUSE)	Nominal	1.00
Documentation match to life-cycle needs (DOCU)	High	1,11
Execution Time Constraint (TIME)	High	1,11
Main storage constraint (STOR)	Nominal	1.00
Platform volatility (PVOL)	High	1.15
Analyst capability (ACAP)	Low	1,19
Programmer capability (PCAP)	Low	1,15
Application Experience (APEX)	Low	1.10
Platform Experience (PLEX)	Low	1.09
Language and Tool Experience (LTEX)	Highl	0.91
Personnel continuity (PCON)	Very high	0.81
Usage of Software Tools (TOOL)	High	0.90
Multisite development (SITE)	Extr high	0.80
Required development schedule (SCED)	Nominal	1.00
Total		1.58798

2.2.3 Effort equation

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

$$\text{Effort} = A * EAF * \text{KSLOC}^{E}$$

where:

A = 2.94 (for COCOMO II)
EAF = product of all cost drivers (1.58798)
E = exponent derived from the scale drivers. It is computed as:
$B + 0.01 * \sum_i SF[i] = B + 0.01 * 19,83 = 0.91 + 0.1983 = 1.1083$
in which B is equal to: 0.91 for COCOMO II.

With this parameters we can compute the effort value, which has a lower bound of:

$$\text{Effort} = A * EAF * \text{KSLOC}^{E} = 2.94 * 1.58798 * 17.338^{1.1083} = 110,249 \text{ PM} \approx 110 \text{ PM}$$

and an upper bound of:

$$\text{Effort} = \text{A} * \text{EAF} * \text{KSLOC}^{E} = 2.94 * 1.58798 * 25.326^{1.1083} = 167{,}79 \text{ PM} \approx 168 \text{ PM}$$

2.2.4 Schedule estimation

In order to estimate the real project timespan, we should divide the effort by the members' number (divide by 3 in our case).

$$\text{Effort} = 110{,}249 \text{ PM}$$
$$\text{Duration} = 110{,}249 \text{ PM} / 3 = 36{,}74 \text{ months}$$

$$\text{Effort} = 167{,}79 \text{ PM}$$
$$\text{Duration} = 167{,}79 \text{ PM} / 3 = 55{,}93 \text{ months}$$

We think this excessively wide timespan (4 years and 6 months) is incompatible with the business viability of the project, in fact all the assumptions made in the early analysis could became invalid because of new competitors or changes in laws and regulations. Thus we need to hire qualified staff to reduce project's timespan to a reasonable amount (time to market).

Cocomo II provides us with the right set of formulas to estimate a feasible and reasonable time to market for our project, non-regarding on the number of people working on the project. We can divide the effort by this computed time to market to decide how many people to hire (approximately) to finish in that time.

Cocomo II's results are:

$$\text{Duration} = 3.67 * \text{Effort}^{F}$$

As a lower bound, we consider

$$\text{F} = 0.28 + 0.2 * (\text{E} - \text{B}) = 0{,}31966$$
$$\text{Effort} = 110{,}249 \text{ PM}$$
$$\text{Duration} = 3.67 * (110{,}249)^{0{,}31966} = 16{,}5 \text{ months}$$

while as an upper bound, we consider

$$\text{F} = 0.28 + 0.2 * (\text{E} - \text{B}) = 0{,}31966$$
$$\text{Effort} = 167{,}789 \text{ PM}$$
$$\text{Duration} = 3.67 * (167{,}789)^{0{,}31966} = 18{,}9 \text{ months}$$

Results

Software Development (Elaboration and Construction)

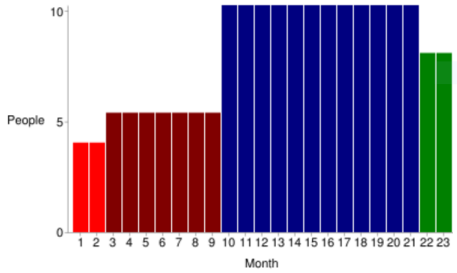
Effort = 167.8 Person-months
Schedule = 18.9 Months
Cost = $0

Total Equivalent Size = 25326 SLOC

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	10.1	2.5	4.0	$0
Elaboration	40.3	7.5	5.4	$0
Construction	127.5	12.4	10.3	$0
Transition	20.1	2.5	8.1	$0

Staffing Profile



Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	1.4	4.8	12.8	2.8
Environment/CM	1.0	3.2	6.4	1.0
Requirements	3.8	7.2	10.2	0.8
Design	1.9	14.5	20.4	0.8
Implementation	0.8	5.2	43.4	3.8
Assessment	0.8	4.0	30.6	4.8
Deployment	0.3	1.2	3.8	6.0

From this result we can estimate in normal estimation and upper bound cases:

$$\begin{aligned} \text{People} &= \text{Effort} \ / \ \text{Duration} \\ &= 110{,}249 \text{ PM} \ / \ 16.5 \text{ months} \\ &= 6.68 \text{ (approx 7 people)} \end{aligned}$$

$$\begin{aligned} \text{People} &= \text{Effort} \ / \ \text{Duration} \\ &= 167{,}789 \text{ PM} \ / \ 18{,}9 \text{ months} \\ &= 8.87 \text{ (approx 9 people)} \end{aligned}$$

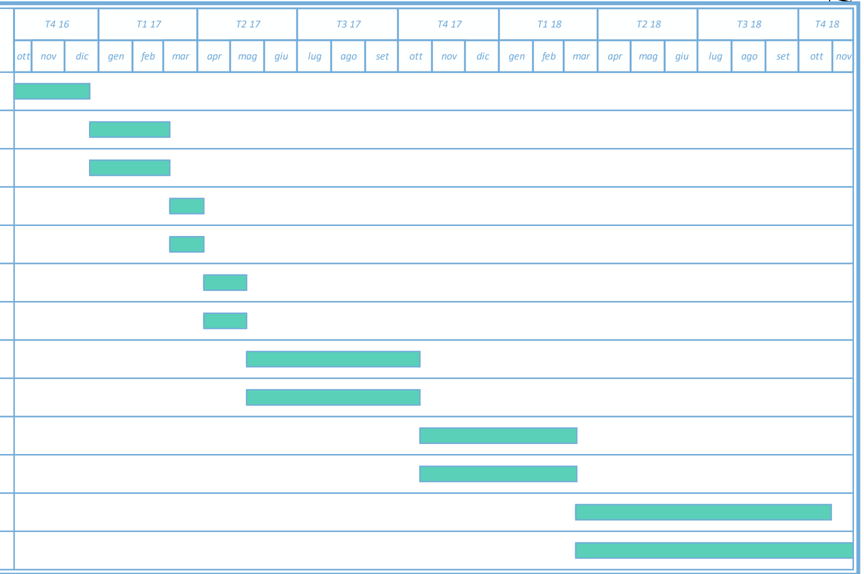
Therefore we could hire additional 5 people to be able to go to market in approximately 16.5 months (with an uncertainty of 2 months). In case of delay, we could mitigate by balancing the amount of features in each one of the planned 3 releases.

3 Schedule

In this section we are going to present a probable project schedule. Obviously this is a prototype and will be refined during the development.
In order to mantain readability and to allow a better understanding, the project schedule will be first presented in a global way and then each part will be expanded.

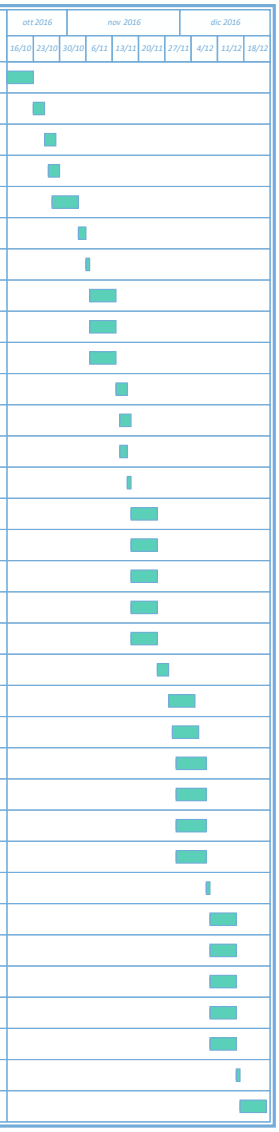
3.1 Global

ID	Nome attività	Inizio	Fine	Durata
1	Requirements Analysis and Specifications Document (RASD)	16/10/2016	23/12/2016	69g
2	Design Document (DD)	24/12/2016	06/03/2017	73g
3	RASD adjustments	24/12/2016	06/03/2017	73g
4	Integration Test Plan Document (ITPD)	07/03/2017	06/04/2017	31g
5	RASD and DD adjustments	07/03/2017	06/04/2017	31g
6	Project Plan Document (PPD)	07/04/2017	15/05/2017	39g
7	RASD, DD and ITPD adjustments	07/04/2017	15/05/2017	39g
8	First release	16/05/2017	20/10/2017	158g
9	Documents adjustments and PPD refinements	16/05/2017	20/10/2017	158g
10	Second release	21/10/2017	12/03/2018	143g
11	Documents adjustments and PPD refinements	21/10/2017	12/03/2018	143g
12	Third release	12/03/2018	30/10/2018	233g
13	Finalizing all documents	12/03/2018	08/12/2018	272g



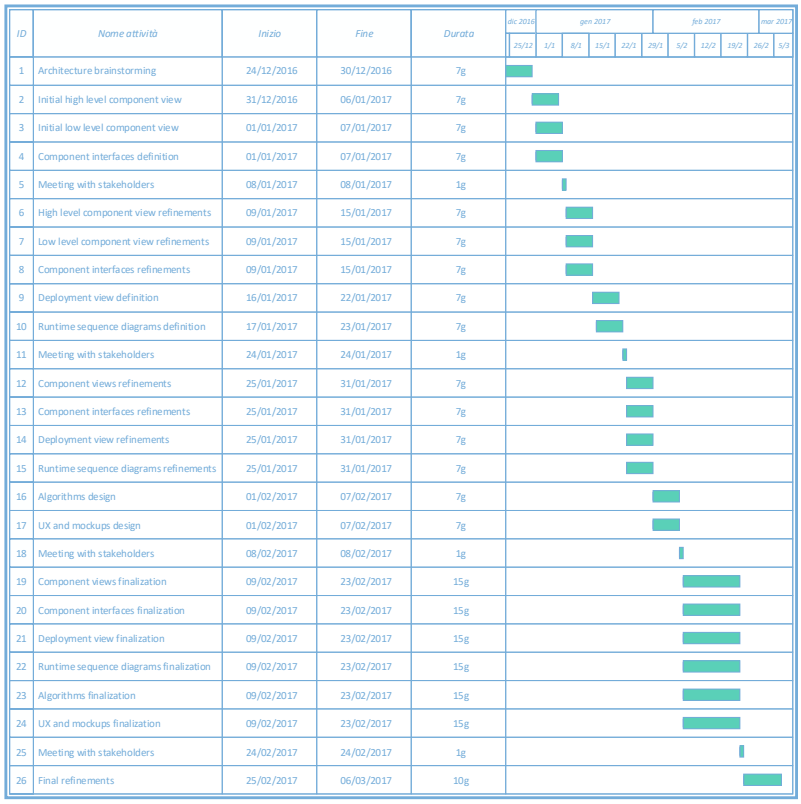
3.2 RASD

ID	Nome attività	Inizio	Fine	Durata
1	Meetings with local governement and stakeholders	16/10/2016	22/10/2016	7g
2	Brainstorming and solution overview	23/10/2016	25/10/2016	3g
3	Goals definition	26/10/2016	28/10/2016	3g
4	Domain assumptions definition	27/10/2016	29/10/2016	3g
5	Requirements definition	28/10/2016	03/11/2016	7g
6	Scenarios description	04/11/2016	05/11/2016	2g
7	Meeting with stakeholders	06/11/2016	06/11/2016	1g
8	Goals refinement	07/11/2016	13/11/2016	7g
9	Domain assumptions refinement	07/11/2016	13/11/2016	7g
10	Requirements refinement	07/11/2016	13/11/2016	7g
11	Identification of use cases	14/11/2016	16/11/2016	3g
12	Sequence diagrams definition	15/11/2016	17/11/2016	3g
13	Initial mockups draft	15/11/2016	16/11/2016	2g
14	Meeting with stakeholders	17/11/2016	17/11/2016	1g
15	Goals refinement	18/11/2016	24/11/2016	7g
16	Domain assumptions refinement	18/11/2016	24/11/2016	7g
17	Requirements refinement	18/11/2016	24/11/2016	7g
18	Use cases refinement	18/11/2016	24/11/2016	7g
19	Mockups refinement	18/11/2016	24/11/2016	7g
20	External interfaces description	25/11/2016	27/11/2016	3g
21	Early class diagram definition	28/11/2016	04/12/2016	7g
22	Consistency verification through Alloy	29/11/2016	05/12/2016	7g
23	Goals revision	30/11/2016	07/12/2016	8g
24	Domain assumptions revision	30/11/2016	07/12/2016	8g
25	Requirements revision	30/11/2016	07/12/2016	8g
26	UML diagrams revision	30/11/2016	07/12/2016	8g
27	Meeting with stakeholders	08/12/2016	08/12/2016	1g
28	Goals finalization	09/12/2016	15/12/2016	7g
29	Domain assumptions finalization	09/12/2016	15/12/2016	7g
30	Requirements finalization	09/12/2016	15/12/2016	7g
31	UML diagrams finalization	09/12/2016	15/12/2016	7g
32	External interfaces finalization	09/12/2016	15/12/2016	7g
33	Meeting with stakeholders	16/12/2016	16/12/2016	1g
34	Final refinements	17/12/2016	23/12/2016	7g



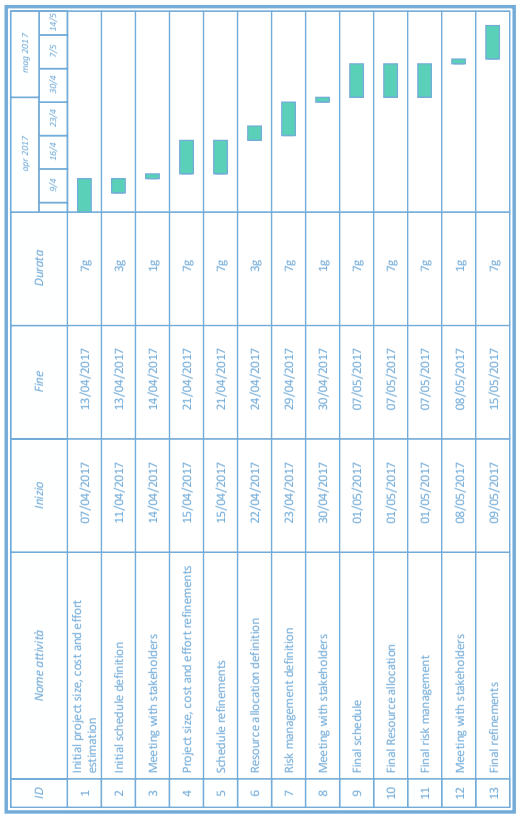
3.3 DD

ID	Nome attività	Inizio	Fine	Durata
1	Architecture brainstorming	24/12/2016	30/12/2016	7g
2	Initial high level component view	31/12/2016	06/01/2017	7g
3	Initial low level component view	01/01/2017	07/01/2017	7g
4	Component interfaces definition	01/01/2017	07/01/2017	7g
5	Meeting with stakeholders	08/01/2017	08/01/2017	1g
6	High level component view refinements	09/01/2017	15/01/2017	7g
7	Low level component view refinements	09/01/2017	15/01/2017	7g
8	Component interfaces refinements	09/01/2017	15/01/2017	7g
9	Deployment view definition	16/01/2017	22/01/2017	7g
10	Runtime sequence diagrams definition	17/01/2017	23/01/2017	7g
11	Meeting with stakeholders	24/01/2017	24/01/2017	1g
12	Component views refinements	25/01/2017	31/01/2017	7g
13	Component interfaces refinements	25/01/2017	31/01/2017	7g
14	Deployment view refinements	25/01/2017	31/01/2017	7g
15	Runtime sequence diagrams refinements	25/01/2017	31/01/2017	7g
16	Algorithms design	01/02/2017	07/02/2017	7g
17	UX and mockups design	01/02/2017	07/02/2017	7g
18	Meeting with stakeholders	08/02/2017	08/02/2017	1g
19	Component views finalization	09/02/2017	23/02/2017	15g
20	Component interfaces finalization	09/02/2017	23/02/2017	15g
21	Deployment view finalization	09/02/2017	23/02/2017	15g
22	Runtime sequence diagrams finalization	09/02/2017	23/02/2017	15g
23	Algorithms finalization	09/02/2017	23/02/2017	15g
24	UX and mockups finalization	09/02/2017	23/02/2017	15g
25	Meeting with stakeholders	24/02/2017	24/02/2017	1g
26	Final refinements	25/02/2017	06/03/2017	10g



3.4 IT

ID	*Nome attività*	*Inizio*	*Fine*	*Durata*	*mar 2017*				*apr 2017*
					5/3	*12/3*	*19/3*	*26/3*	*2/4*
1	Integration strategies overview	07/03/2017	09/03/2017	3g					
2	Integration strategy definition	10/03/2017	19/03/2017	10g					
3	Individual tests identification	13/03/2017	27/03/2017	15g					
4	Testing tools and equipment definition	28/03/2017	30/03/2017	3g					
5	Final refinements	31/03/2017	06/04/2017	7g					

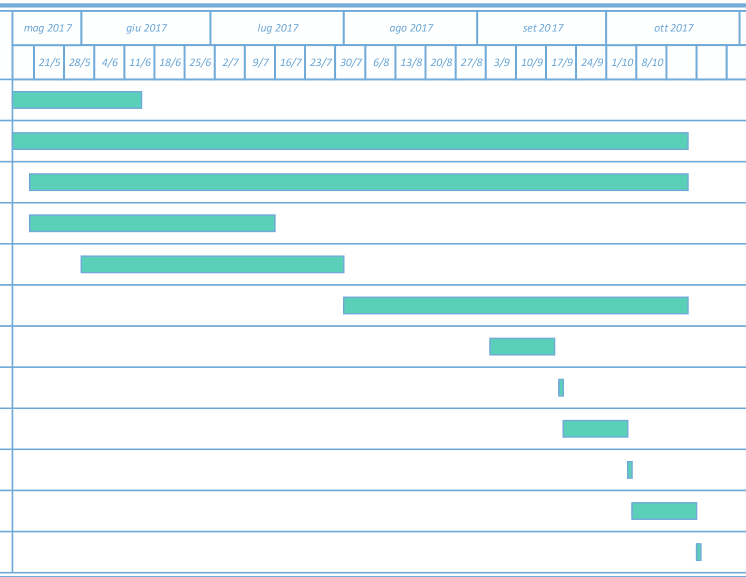


ID	Nome attività	Inizio	Fine	Durata	apr 2017				mag 2017		
					9/4	16/4	23/4	30/4	7/5	14/5	
1	Initial project size, cost and effort estimation	07/04/2017	13/04/2017	7g							
2	Initial schedule definition	11/04/2017	13/04/2017	3g							
3	Meeting with stakeholders	14/04/2017	14/04/2017	1g							
4	Project size, cost and effort refinements	15/04/2017	21/04/2017	7g							
5	Schedule refinements	15/04/2017	21/04/2017	7g							
6	Resource allocation definition	22/04/2017	24/04/2017	3g							
7	Risk management definition	23/04/2017	29/04/2017	7g							
8	Meeting with stakeholders	30/04/2017	30/04/2017	1g							
9	Final schedule	01/05/2017	07/05/2017	7g							
10	Final Resource allocation	01/05/2017	07/05/2017	7g							
11	Final risk management	01/05/2017	07/05/2017	7g							
12	Meeting with stakeholders	08/05/2017	08/05/2017	1g							
13	Final refinements	09/05/2017	15/05/2017	7g							

3.5 PP

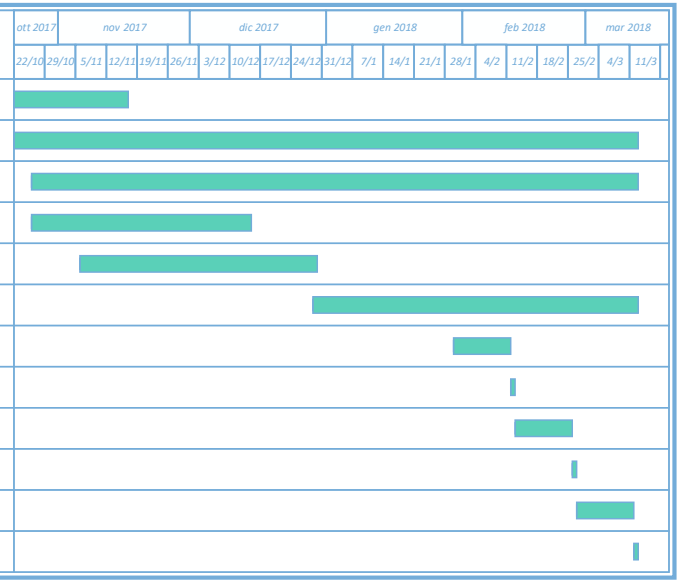
3.6 First release

ID	*Nome attività*	*Inizio*	*Fine*	*Durata*
1	External components acquisition and study	16/05/2017	14/06/2017	30g
2	Components development	16/05/2017	19/10/2017	157g
3	Code inspection	20/05/2017	19/10/2017	153g
4	Unit tests	20/05/2017	15/07/2017	57g
5	Integration testing	01/06/2017	31/07/2017	61g
6	System testing	01/08/2017	19/10/2017	80g
7	Refinements	04/09/2017	18/09/2017	15g
8	Presentation to stakeholders	20/09/2017	20/09/2017	1g
9	Refinements	21/09/2017	05/10/2017	15g
10	Final presentation to stakeholders	06/10/2017	06/10/2017	1g
11	Final refinements	07/10/2017	21/10/2017	15g
12	Release	22/10/2017	22/10/2017	1g



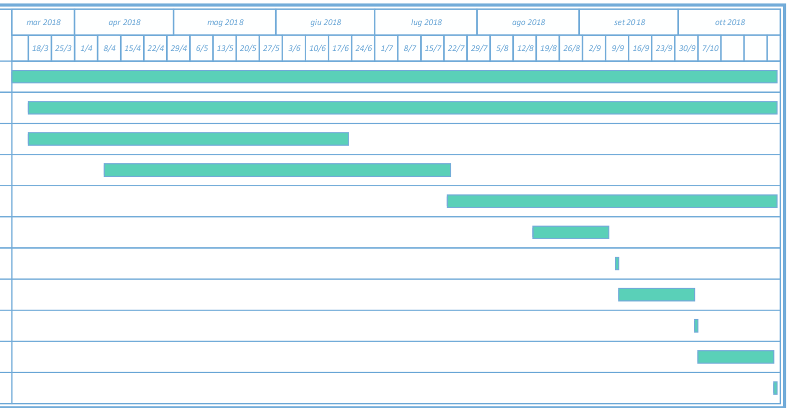
3.7 Second release

ID	Nome attività	Inizio	Fine	Durata
1	External components acquisition and study	22/10/2017	16/11/2017	26g
2	Components development	22/10/2017	12/03/2018	142g
3	Code inspection	26/10/2017	12/03/2018	138g
4	Unit tests	26/10/2017	14/12/2017	50g
5	Integration testing	06/11/2017	29/12/2017	54g
6	System testing	29/12/2017	12/03/2018	74g
7	Refinements	30/01/2018	11/02/2018	13g
8	Presentation to stakeholders	12/02/2018	12/02/2018	1g
9	Refinements	13/02/2018	25/02/2018	13g
10	Final presentation to stakeholders	26/02/2018	26/02/2018	1g
11	Final refinements	27/02/2018	11/03/2018	13g
12	Release	12/03/2018	12/03/2018	1g



3.8 Third release

ID	Nome attività	Inizio	Fine	Durata
1	Components development	13/03/2018	30/10/2018	232g
2	Code inspection	18/03/2018	30/10/2018	227g
3	Unit tests	18/03/201818	22/06/2018	97g
4	Integration testing	10/04/2018	23/07/2018	105g
5	System testing	23/07/2018	30/10/2018	100g
6	Refinements	18/08/2018	09/09/2018	23g
7	Presentation to stakeholders	12/09/2018	12/09/2018	1g
8	Refinements	13/09/2018	05/10/2018	23g
9	Final presentation to stakeholders	06/10/2018	06/10/2018	1g
10	Final refinements	07/10/2018	29/10/2018	23g
11	Release	30/10/2018	30/10/2018	1g



4 Resource allocation

In the early analysis and design part of the project we worked together on the various tasks in order to keep a shared view of the product. On the other hand we need additional people to develop the project by deadlines, this requires a form of task assignment to different team members.

We adopt a hierarchical approach to accomplish tasks' assignment which consist in people being subdivided into 2 teams, lead by one of the original member. The team leader is assigned an area of the project, and assign tasks to team members. One of the original project member act as project manager, and daily meets with the team leaders, keeping a global perspective on the project.

The teams and areas we identified are:

- **Project manager**
- **Developers team 1**: Code implementation
  (team leader + 3 members)
- **Developers team 2**: Testing
  (team leader + 2 members)

5 Risk management

We are going to summarize in tabular format the main risks we think can threat the project and the corresponding mitigating actions we planned. We adopt a proactive strategy against most serious risks related to business and a reactive strategy (for which we have allocated additional time in project schedule) for unpredictable threats.

5.1 Business risks

Can jeopardize the entire project and menace the business viability of the project.

Risk	**Probability**	**Impact**
Due to the high number of competitors in the car sharing panorama, the product can not be noticed if not properly advertised	High	Catastrophical
Unintuitive user interface or wrong set of functionalities drastically reduce the user's adoption of the service	Medium	Catastrophical
Lost of interest in the product by market users / new competitors introduction in the market	Low	Significant
Unintuitive user interface reduce the efficiency of the staff	Low	Minor

Risk	**Mitigation plan**
Not properly advertised	Designers, advertisers and sales personnel actively take part to the early stages of UX designing, in order to have a shared vision of the project through all the involved personnel.
Unintuitive user interface / wrong set of functionalities	Surveys and user group to fit the service experience to the average market user. Analysis of other car-sharing services experience to offer a fast learning curve to their users and emphatize our strength points.
Lost of interest / new competitors	Occasional in-app quality assessment of the user's service experience in exchange of car sharing credit.
Low staff enthusiasm / efficiency	Interaction with staff exponent during the early phase of the user interface design through the presentation / assessment of mockups. In-house usability test for UI prototypes during the early phases of implementation.

5.2 Project risks

Threaten the project schedule and can increase the project cost.

Risk	Probability	Impact
'Goldplating' software features results in a delayed schedule	Low	Significant
Personnel shortfall in proximity of milestones / meeting can delay the release date	High	Minor

Risk	Mitigation plan
'Goldplating' software features results in a delayed schedule	Planning of 3 features-incremental releases that can be adjusted in feature-richness if schedule problems arise.
Personnel shortfall	Introduction of additional time in the schedule between predicted milestone date and milestone release date in order to mitigate personnel absence or for fix eventual problems. Encourage developer motivation through involvement in early analysis phases and in following meetings, thus incentive a shared product vision that lead to better teamwork.

5.3 Technical risks

Threaten the quality and timeliness of the project, leading to a more difficult implementation.

Risk	Probability	Impact
Integration test fails require a redesign of part of the components	Medium	Significant
Unstable external software components / services lead to service partial or full crashes	Low	Significant

Risk	**Mitigation plan**
Integration test fails	Adoption of critical modules test strategy to be able to deal with this kind of issue as soon as possible.
Unstable external software	Selection of widely adopted services (such as Google Map services). Exploration / planning of possible external services' alternatives in case of downtime or discontinuation of those services.

6 Effort Spent

- Alessandro Erba ≈ 14h
- Filippo Leveni ≈ 14h
- Luca Lodi ≈ 14h