POLITECNICO DI MILANO

COMPUTER SCIENCE AND ENGINEERING

MASTER OF SCIENCE

PP

PowerEnjoy

Software engineering II project

Giovanni Agugini, Matteo Foglio, Tommaso Massari

Contents

1 Introduction

1.1 Revision History

Table 1: Revision History

Version	Date	Author(s)	Summary
1.0	20/01/2017	Giovanni Agugini, Matteo Foglio e Tommaso Massari	Initial release

1.2 Purpose and Scope

The purpose of this document is to provide a Project Plan in order to ensure that PowerEnjoy software is delivered on time and on schedule in accordance with context and requirements. Our project team, together with the Project Manager, aims at planning the project by estimating:

- Size: the size of the project is the starting parameter from which we assume the other planning constrains and it depends on Function Points (see later) and the number of lines of code of the software
- Effort: the general effort involved in the software development (generally engagement of people)
- Cost: optimizing human and technology costs is a key factor for the project success
- Time: an estimation of the duration of the project. It comes from the other estimations and it can be subject to changes as software requirements and other factors may change during the development

As it concerns the estimation techniques, in this document we're going to show the algorithmic-cost modeling technique which is based on the COCOMO II formula, even though an experience-based technique has also been used in order to exploit all different skills of the development team and involve each member to participate proactively to the project. We'll then present a schedule of the project based on the results of our COCOMO post-architecture estimation. The schedule constitutes an ever-changing document due to requirements volatility and uncertainty. Nevertheless, it is going to be our planning reference for the entire project. Consequently the next section will show the assignment of the various tasks to each team member. Finally, the last section of the document consists also in defining a formal and structured way to avoid risk related to changes in software requirements, project budget, organizational structures, business feasibility. Thus it will be followed a systematic approach coherent to a proactive fashion, which can be summarized in the next points:

- Identification of known, predictable and unpredictable risks

- Analysis of the risks by producing a probability and an impact for each risk
- Prioritization of the risks based on probability and impact
- Definition of a contingency plan for those risks with high probability and high impact

1.3 List of Definitions, Acronyms and Abbreviations

1.3.1 Definitions

- User: can be both an operator or a client
- PHP: is a script language for the creation of dynamic web pages

1.3.2 Acronyms

- DD: Design Document
- RASD: Requirement Analysis and Specification Document
- FP: Function Points
- LOC: Lines Of Code
- KSLOC: Kilo Source Lines Of Code
- EM: Effort Multipliers
- EI: External Input
- EO: External Output
- EQ: External Inquiry
- DBMS: DataBase Management System
- UFP: Unadjusted Function Points
- UI: User Interface
- EJB: Enterprise Java Beans
- RASD: Requirement Analysis and Specification Document
- DD: Design Document
- ITPD: Integration Test Plan Document
- RESTful: Representational State Transfer
- API: Application Programming Interface

- COCOMO II: COnstructive COst MOdel, version II.2000.0
- PM = Person Months
- PH = Person Hours

1.4 List of Reference Documents

- The Requirements Analysis and Specification Document (RASD.pdf) of the project
- The Design Document (DD.pdf) of the project
- The Integration Test Plan Document (ITPD) of the project
- The project description document: Assignment AA 2016/2017.pdf
- Slides from the course Software Engineering 2 - Professor Elisabetta Di Nitto and Professor Luca Mottola - Politecnico Di Milano
- COCOMO II Definition Manual:
  - http://csse.usc.edu/csse/research/COCOMOII/cocomo2000.0/CII_modelman2000.0.pdf

2 Size Estimation: Function Points

In this section we're going to provide an estimation following the Function Points approach. We assume that only the application layer and the business logic of our software are taken into account and no user interface or graphical features are involved in our inference. The program characteristics that we considered as Function Points are data structures, inputs, outputs and inquiries; then a weight is associated to each Function Point based on its complexity. We will then use FP to compute the number of LOC to be written in PHP. We use as reference the following table from the Albrecht's method.

Figure 1: Albrecht Table

	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 Internal Logic Files (ILF)

Our system relies on the several Internal Logic Files used to store information about:

- User: the majority of the fields are static. Most of them are simple strings and integers which represent username, password and personal information of the user
- Car: these data are mainly stored in a simple table and there are both static (id and plate) and dynamic data (state and last maintenance). Furthermore some data can be asked to the car itself like the number of passengers or the of the battery
- Rental: this table is frequently updated and it stores information about reservation and ride
- Payment: they are data related to the rental and they are usually updated together
- Area: these are static data mainly composed by Position object
- Position: it's composed by two static numbers

- Grid Station: it simply contains the number of available sockets and the position of the station

Table 2: Internal Logic Files

ILF	Complexity	FPs
User	Low	7
Car	Medium	10
Rental	Medium	10
Payment	Medium	10
Area	Low	7
Position	Low	7
Grid Station	Low	7
TOTAL		58

2.2 External Interface Files (EIF)

The only External Interface Files in our system can be considered the Mapping Service provided by Google Maps. The interaction between the system and Google Maps is performed through a RESTful API and data can be returned in JSON or in another metalanguage format. We use the Mapping Service in three different ways:

- Map information given to the user in order to reach his/her reserved car
- Position retrieval of all the cars to provide the list of cars
- GPS navigation to give traffic indications and perform uniform distribution

Therefore, we assigned different complexity to the different sub-function points as there are ones that require a more elaborated algorithmic construction and others that are quite simple to implement.

Table 3: External Interface Files

ILF	Complexity	FPs
Map Information	Medium	7
Position Retrieval	Low	5
GPS navigation for uniform distribution	High	10
TOTAL		22

2.3 External Inputs

In this sub-section we're going to show elementary operations to elaborate data coming from the external environment. We mainly show the interaction between

the users of the Power Enjoy service and the system. External Inputs can be divided into three categories:

1. Client side
   (a) Sign-up and modification of personal data
   (b) Call Center
   (c) Settings and discounts selection on the On-Board Computer
2. Operator side
   (a) Change the state of a car
   (b) Approve the user
3. Both client and operator side
   (a) Login / Logout
   (b) Reservation of a car
   (c) Cancel reservation
   (d) Open a car
   (e) End rental

Table 4: External Inputs

Ext. Inquiry	Complexity	FPs
1a	Low	3
1b	Low	3
1c	Medium	4
2a	Medium	4
2b	Low	3
3a	Low	3
3b	High	6
3c	Low	3
3d	Medium	4
3e	High	6
TOTAL		39

2.4 External Outputs

Following the FPs specification, in this section will be listed all kind of information sent to the user of PowerEnjoy system without an explicit request of the user himself, i.e. all the elementary operations that generates data for the external environment:

- Payment confirmation: it's a message sent to the client when a completed has been completed successfully

- Payment request: it's a message sent to the client when the payment has failed
- Client approval notification: it's a message sent to the client when his account has been approved
- Client blocked notifcation: it's a message sent to the client when his account has been blocked
- Reservation expired: it's a message sent to the client when the resevation time exceeds the time allowed for the reservation (i.e. one hour)

Table 5: External Output

Ext. Output	Complexity	FPs
Payment confirmation	Low	4
Payment request	Low	4
Client approval	Low	4
Client blocked	Low	4
Reservation expired	Low	4
TOTAL		20

2.5 External Inquiry

External inquiries include all the requests sent by an user to the system in order to retrieve information from it. They are all simple requests that don't require particular computations or algorithms. Some inquiries belong only to operations performed by clients, other by operator and finally some others by both of them.

1. Client side:
   (a) Retrieve the list of his personal data
   (b) Retrieve the list of the car in a specified area
   (c) Retrieve the way to reach the car after the reservation
   (d) Retrieve the way to reach the destination on the On-Board Computer. This request include the computation of uniform distribution if enabled
   (e) Manage the payment and corresponding dialogue between our system and the payment agency
2. Operator side:
   (a) Retrieve the list of the users that need to be approved
   (b) Retrieve personal data of a particular user

(c) Retrieve the list of cars in a specified area. This operation includes the the analysis of each of the car shown, in order to find the ones the require the intervention of an operators. For instance, the system has to look for vehicles to be moved in order to guarantee an uniform distribution or for vehicles with empty battery

Table 6: External Inquiry

Ext. Inquiry	Complexity	FPs
1a	Low	3
1b	Medium	4
1c	Medium	4
1d	Medium	4
1e	Medium	4
2a	Low	3
2b	Low	3
2c	High	6
TOTAL		31

2.6 Overall Estimation

Table 7: Internal Logic Files

Function Type	FPs
Intenal Logic File	58
External Interface File	22
External Input	30
External Output	20
External Inquiry	31
TOTAL	161

The overall estimation is 161 FP and can be obtained by summing up the result of the single evaluations.

Our size evaluation considers only business logic part of the system, in fact in our estimation we have not included the presentation layer i.e. the use interface

Since our system in base on Apache platform and PHP, the AVC multiplying constant is equal to 67.

Following the calculous of estimated number of LOC:

LOC = AVC * FP = 10787

3 Cost and Effort Estimation: COCOMO II

This section will explain we use COCOMO II to estimate cost and effort required to develop the system of Power Enjoy. The COCOMO formula is PM = $A*Size^{E}* \prod_{1<=i<=17} EM_j$ where PM is Person-Month, A = 2,94 is a constant coefficient, Size is the size of the project in KLOC, E is an exponent computed on scale factors and EM_j are the effort multipliers.

3.1 Scale Drivers

Figure 2: Scale Factors COMOMO II Official Table

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	The estimated Equivalent Process Maturity Level (EPML) or					
	SW-CMM Level 1 Lower	SW-CMM Level 1 Upper	SW-CMM Level 2	SW-CMM Level 3	SW-CMM Level 4	SW-CMM Level 5
SF_j:	7.80	6.24	4.68	3.12	1.56	0.00

The scale factor we are going to use will be:

- Precedentedness: our team is not very experienced but some members have already used some of the tools required for development. Thus we will set this scale factor to low
- Development flexibility: the already defined requirements are not very strict since we are developing a new software from scratch: some functionalities can be modified with good level of flexibility. However some components will be interfaced with external gateways and their flexibility will be more strict. Overall level of flexibility can be set to High
- Risk resolution: can be considered high as a good risk management plan has benn carried out

- Team cohesion: can be considered very high
- Project maturity: can be considered nominal

The following table shows the result of our estimation:

Table 8: Scale Factors PowerEnjoy Evalutation

Scale Driver	Factor	Value
PREC	Low	4.96
FLEX	High	2.03
RESL	High	2.83
TEAM	Very High	1.10
PMAT	Nominal	4.68

The Exponent for the COCOMO formula is calculated with its equation: E = B + 0,01*$\sum_{1<=j<=5} SF_j$ = 0,91 + 0,01* (4,96 + 2,03 + 2,83 + 1,10 + 4,68) = 1,066.

3.2 Cost Drivers

Since we've already designed our architecure we consider the post-architecture cost drivers. Following here are the cost drivers grouped by functionality fields.

Figure 3: Cost Drivers COMOMO II Official Table

Early Design Cost Driver	Counterpart Combined Post-Architecture Cost Drivers
PERS	ACAP, PCAP, PCON
RCPX	RELY, DATA, CPLX, DOCU
RUSE	RUSE
PDIF	TIME, STOR, PVOL
PREX	APEX, PLEX, LTEX
FCIL	TOOL, SITE
SCED	SCED

3.2.1 Personnel Factors

- Analyst Capability (ACAP): Nominal

  After evaluting our ability of analysising ,designing and cooperating we estimate this value to nominal.

- Programmer capability (PCAP): Very High

  It has been evaluated that the team of programmers have already shown a good level of cohesion and cooperation skills.

- Personnel Continuity (PCON): Very High

  The members of the team will always be the same. We expect not to have personnel turnover.

- Application Experience (APEX): Low

  The members of our team have just few previous experiences in small projects

- Language and Tool Experience (LTEX): Low

  Our teams have already used few of the tools required for the development of this project. The level of knowledge of the programming language is eterogeneous among the members but most of them have to approach new tools and languages

- Platform Experience (PLEX): Low

  Our knowledge of database system can be considered Nominal. However we have limited experience with other platoforms such as PhoneGap and Android systems

3.2.2 Product Factors

- Required Software Reliability (RELY): Nominal

  The system functionalities don't include risks for human life. However a low level of reliability would lead to financial losses and the lost of fidelization of our clients

- Database Size (DATA): Low

  Since our system needs to store lots of images provided by clients, such as ids, driving licenses and personal photos we can estimate the value in this way:

  considering about 8 megabytes needed per client and approximately 10 clients in the database (compared to them all other data are negligible, the parameter can be computed in the following way:

  DATA = 8000 bytes * 10 clients / 10787 LOC = 7.42.

- Product Complexity (CPLX): Nominal

  The complexity of our product is the weighted average of the following evalutation:

  - Control Ooperations: Very High

    Our system includes some task synchronization and some recursive coding.

  - Computational Operations: Nominal

    Our system relies on well-known complex algorithm but they don't represent the main functionalities of our service

  - Device-dependent Operations: Low

    Our system relies on high level functions provided by widely used operating systems and software such as Android and Apache

  - Data Management Operations: Low

    Our data structures are quite simple.
  - User Interface Management Operations: Low

    We use simple graphic user interface (GUI) builders.

- Documentation Match To Lifecycle Needs (DOCU): High

  We believe that documentation represents a foundamental part of the work since it will allow us to reduce effort in updating the software later. Moreover it gives a complete overview of the software architecture giving a better understanding of it.

- Developed For Reusability (RUSE)

  The reusability of our software is limited to the software itself. According to COCOMO documentation we'll set this level to Nominal.

3.2.3 Platform Factors

- Execution Time Contraint (TIME): Nominal

  The usage of CPU will be mostly below 50% of availability since the majority of the operations are quite simple and well distribuited in time

- Main Storage Constraint (STOR): Nominal

  Our system does not require a large amount of data to store information about rental, users and payment. Moreover nowadays storage prices are very accessible.

- Platform Volatility (PVOL): Low

  We expect our software to be updated with major changes only on years scale alongside major operating system updates. Minor fixes and changes can happen on monthly scale.

3.2.4 Project Factors

- Use of Software Tool (TOOL): High

  We are planning to use realiable integrated software tools with management of software lifecycles and releases

- Multisite Development (SITE): Extra High

  Since we have just one site the communication are instant.

- Required Development Schedule (SCED)

  It has been evaluated that no stretch-out or acceleration are needed. We espect to follow the nominal schedule for the project and thus we estimate the level of SCED to nominal.

Table 9: Cost Drivers Power Enjoy

Cost Driver	Factor	Effort Multiplier
ACAP - Analyst Capability	Nominal	1.00
PCAP - Programmer Capability	Very High	0.76
PCON - Personnel Continuity	Very High	0.81
RELY - Required Software Reliability	Nominal	1.00
DATA - Database Size	Low	0.90
CPLX - Product Complexity	Nominal	1.00
DOCU - Documentation Match to Life-Cycle Needs	High	1.11
RUSE - Development	Nominal	1.00
TIME - Execution Time Constraint	Nominal	1.00
STOR - Main Storage	Nominal	1.00
PVOL - Platform Volatility	Low	0.87
APEX - Application Experience	Low	1.10
LTEX - Language and Tool Experience	Low	1.09
PLEX - Platform Experience	Low	1.09
TOOL - Use of Software Tools	High	0.90
SITE - Multisite Development	Extra High	0.80
SCED - Required Development Schedule	Nominal	1.00

3.2.5 From Cost Drivers to Effort Multipliers

The following table shows the values of the effort multipiers based on the cost drivers analysis. The final effort multiplier can be computed by multiplying all the single ones. The result is $\prod EM_i$ = 0,5034.

3.3 Summary

After having computed the Size, the Exponent and the Effort Multiplier we put all together to compute the COCOMO II formula:

$$\text{PM} = A * Size^{E} * \prod_{1<=i<=17} EM_j = 2,94 * 10,787^{1,066} * 0,5034 = 18,67$$

For our organization PM = 152 PH. So PH = 18,67 * 152 = 2837,84 Person Hours.

4 Schedule

In this section we propose a high-level schedule estimation for the project using the COCOMO formula results available in the previous section.

4.1 Schedule Estimation

The initial baseline schedule equation for the COCOMO II Post-Architecture stages is:

$$TDEV = \left[C * (PM_{NS})^{(D+0{,}2*(E-B))}\right] * \frac{SCED\%}{100} = 14{,}7 \text{ months}$$

where C = 3,67, D = 0,28, B = 0,91 and PM_{NS} is the PM without the SCED effort multiplier that in our case is 1.0, so PM_{NS}= PM.

4.2 Gantt Diagrams

Figure 4: General View

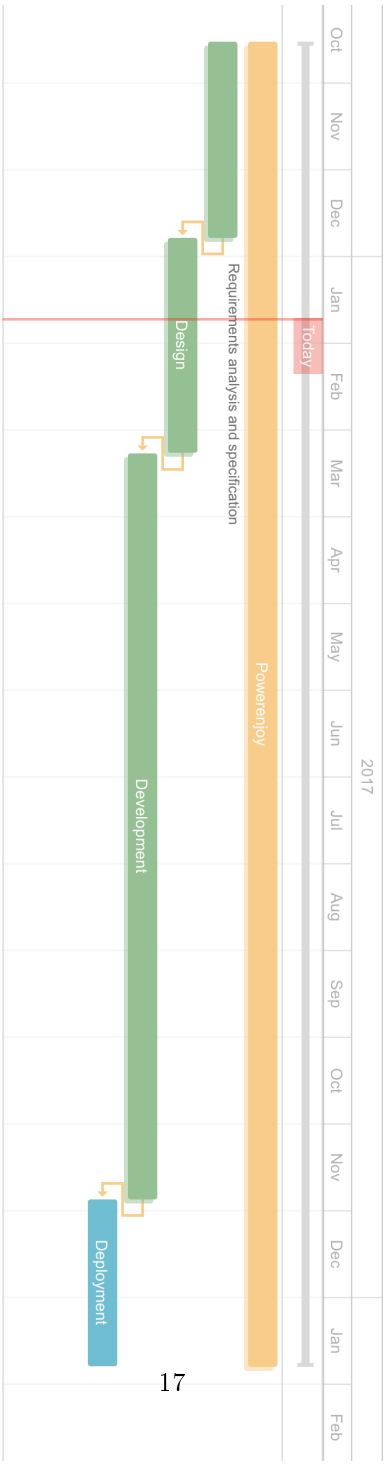


Figure 5: Requirement Analysis and Specification Schedule

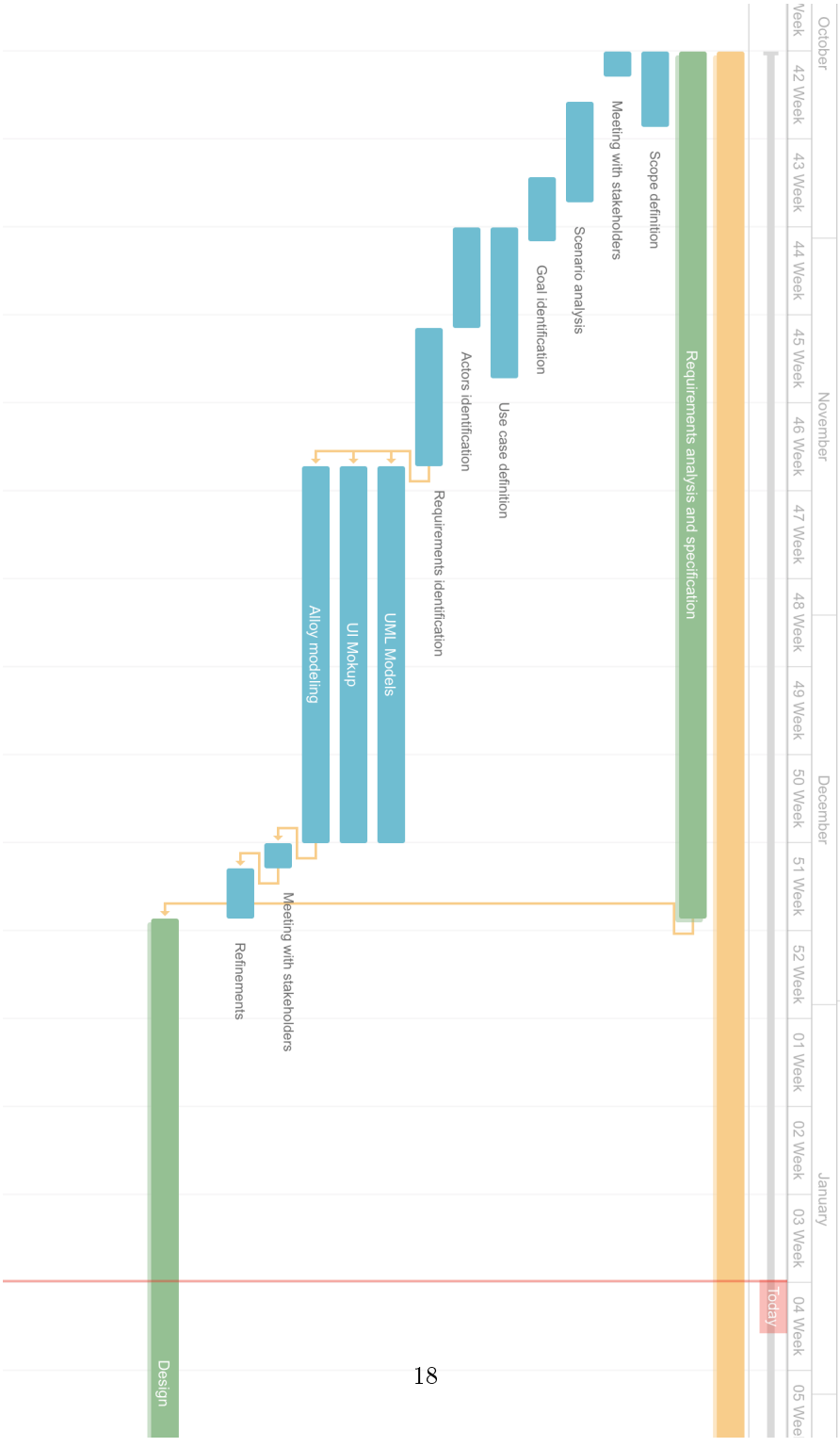
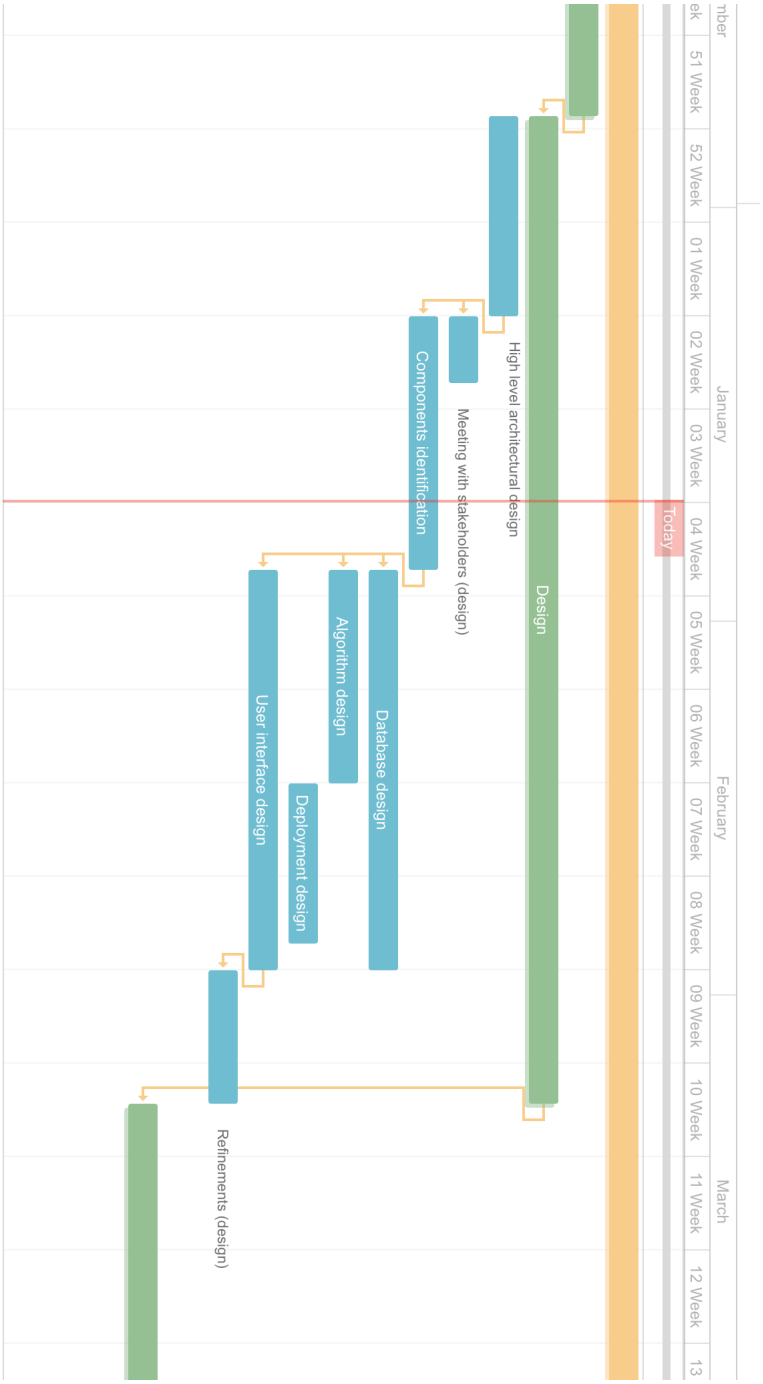


Figure 6: Design Schedule



5 Resource Allocation

In this section we show how the different tasks scheduled in the previous section are assigned to each team member. Some tasks will be executed by the whole team in order to mantain a general overview of the project and to evaluate fundamental decisions together.
To Giovanni Agugini the following tasks have been assigned:

- Model: User, Client, Operator
- Client Controller
- Operator Controller
- Router

To Matteo Foglio the following tasks have been assigned:

- Model: Rental, Reservation, Ride, Payment
- Rental Controller
- Notification Controller
- Payment Controller

To Tommaso Massari the following tasks have been assigned:

- Model: Car, Area, Position, Grid Station
- Car Controller
- Area Controller

Figure 7: Giovanni Agugini Requirements

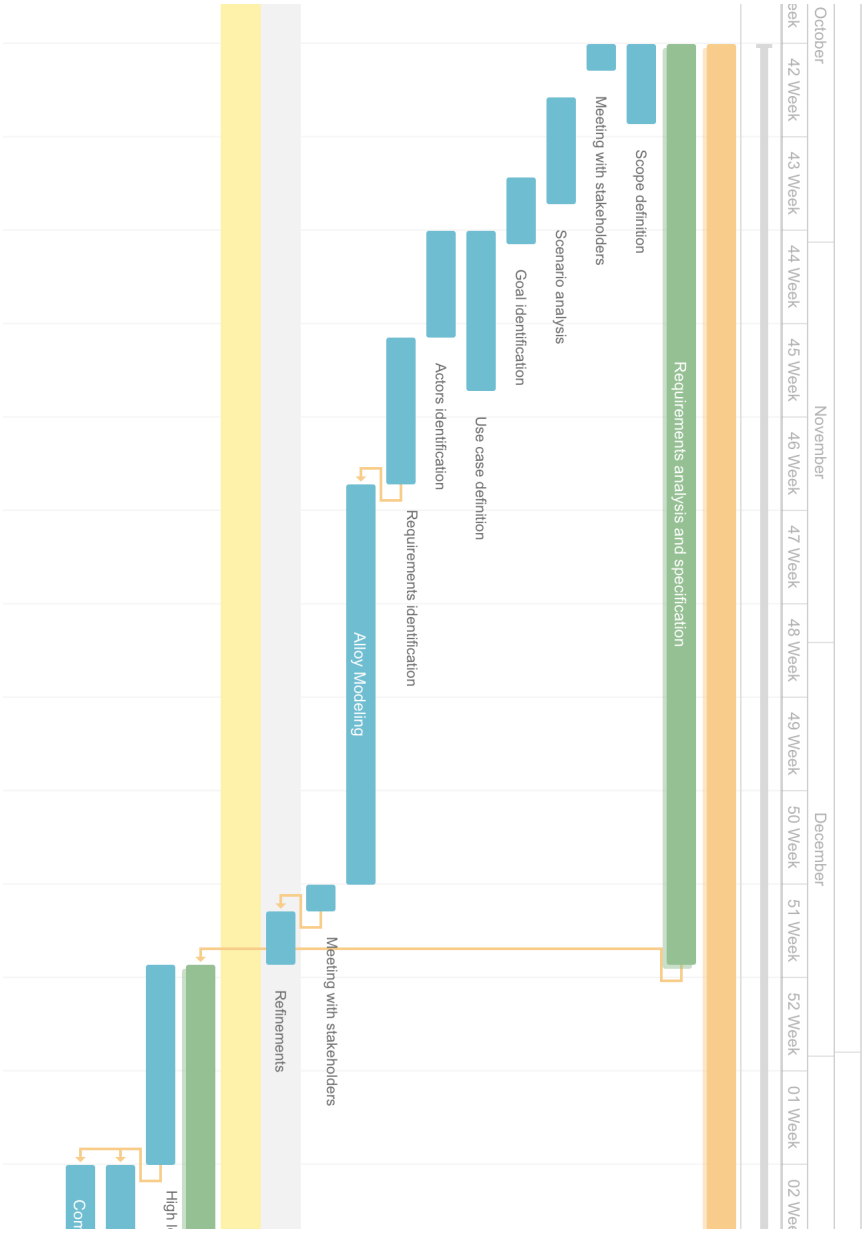


Figure 8: Giovanni Agugini Design

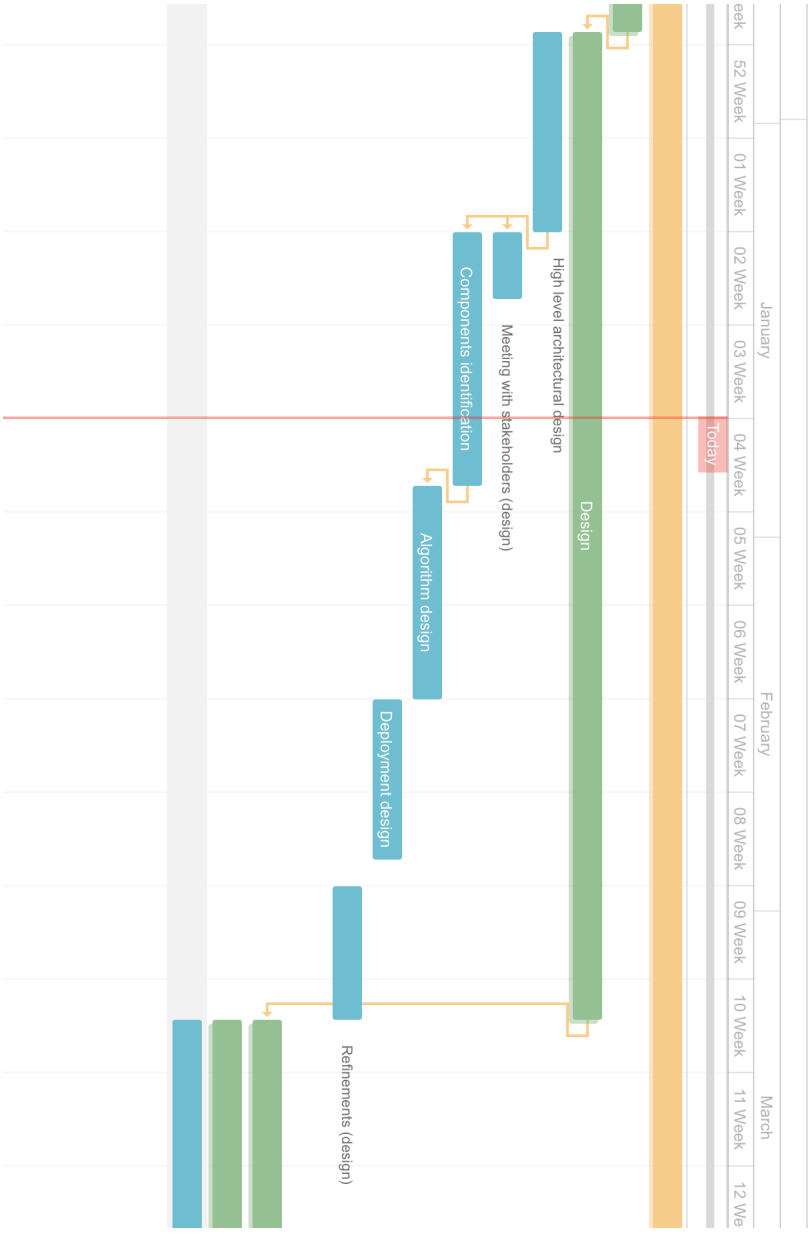


Figure 9: Giovanni Agugini Development

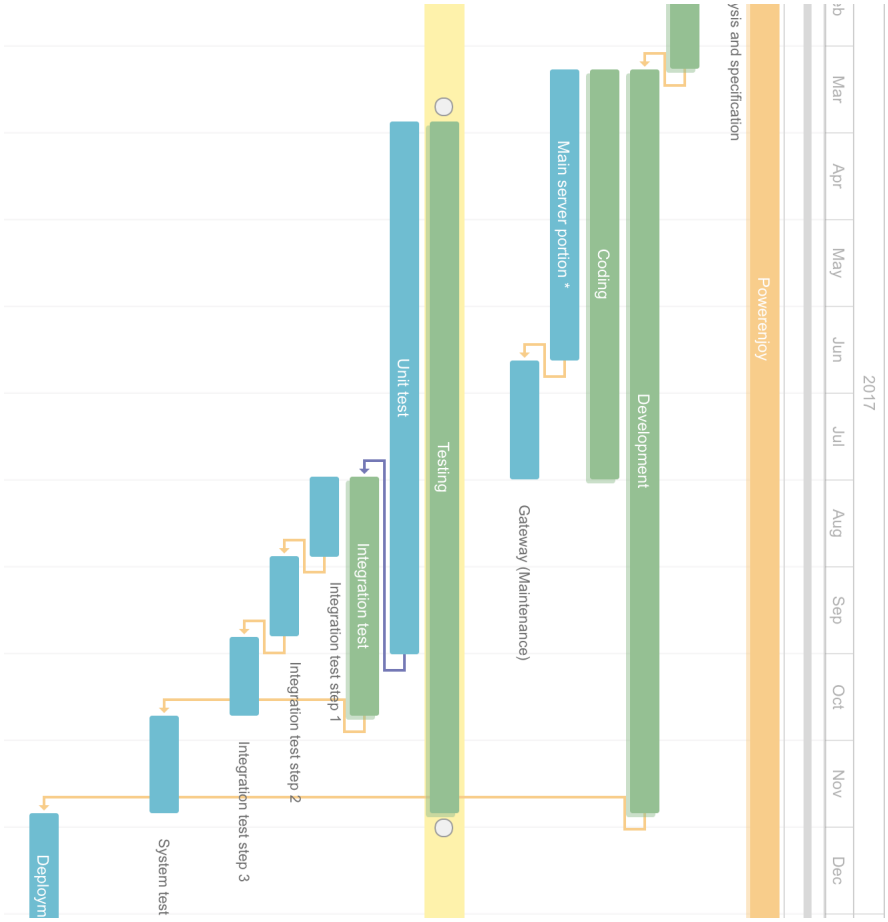


Figure 10: Matteo Foglio Requirements

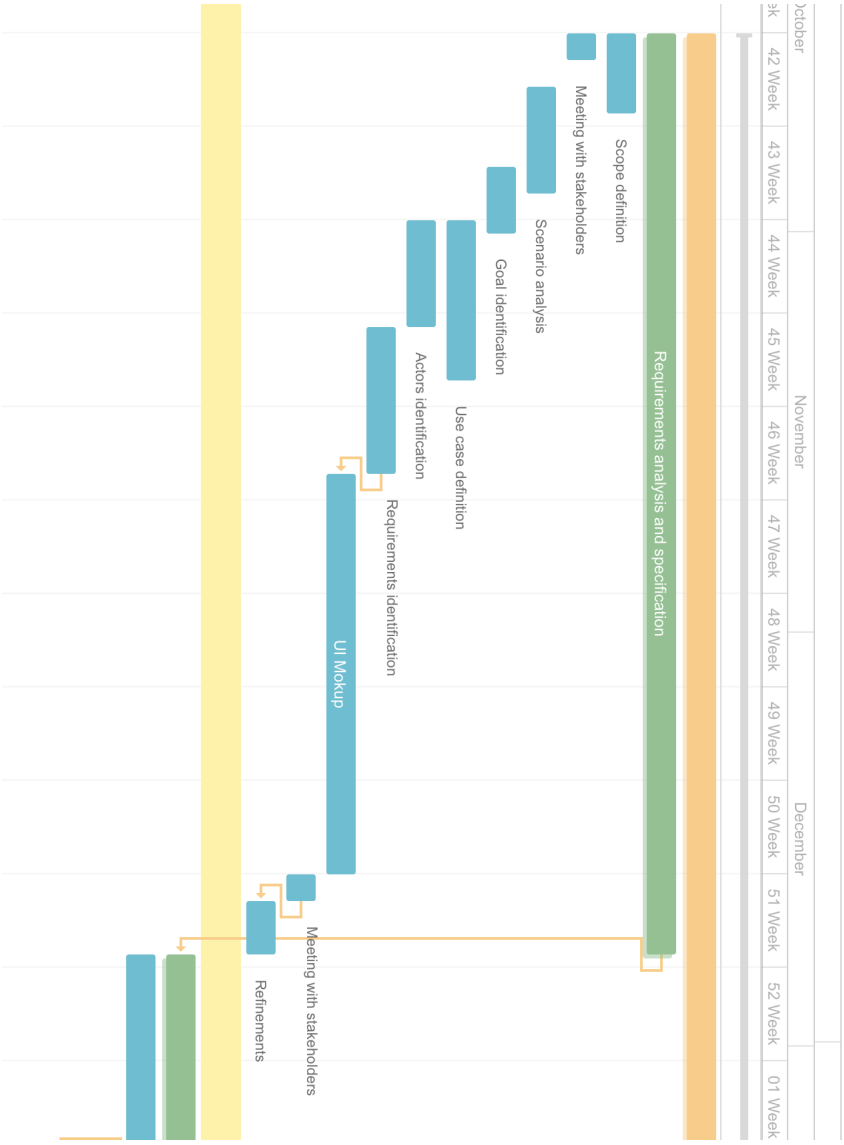


Figure 11: Matteo Foglio Design

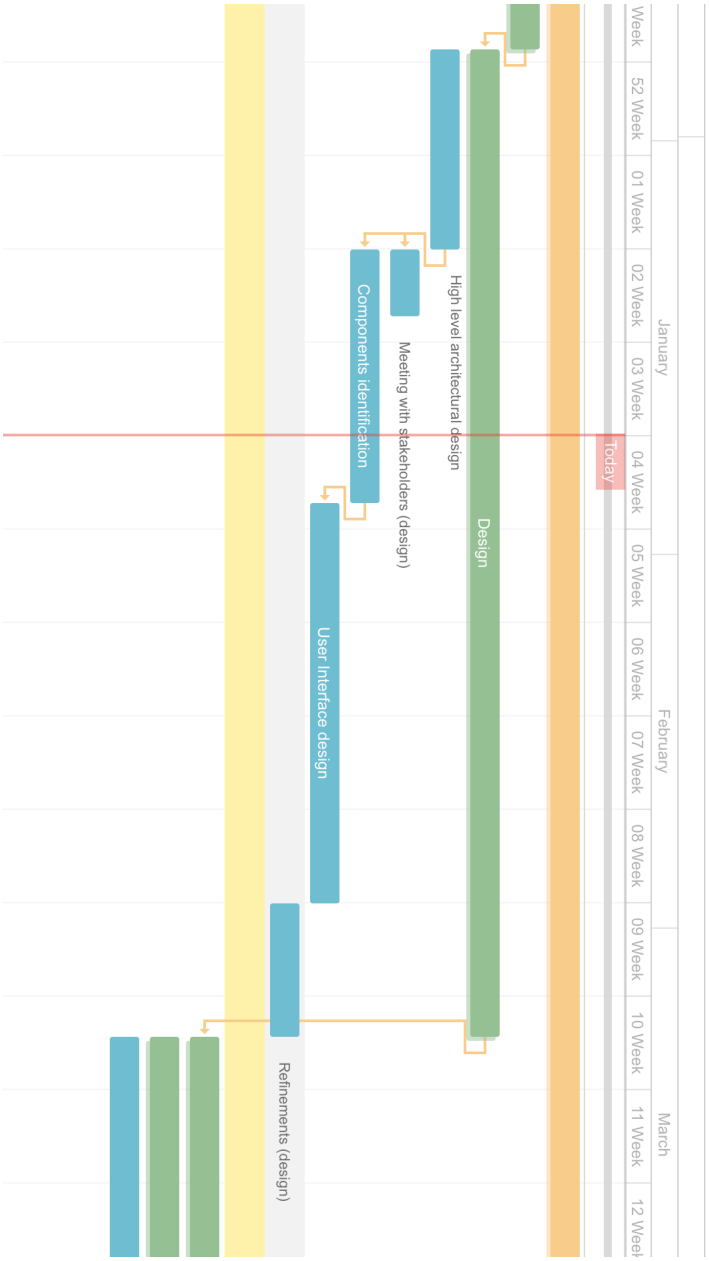


Figure 12: Matteo Foglio Development

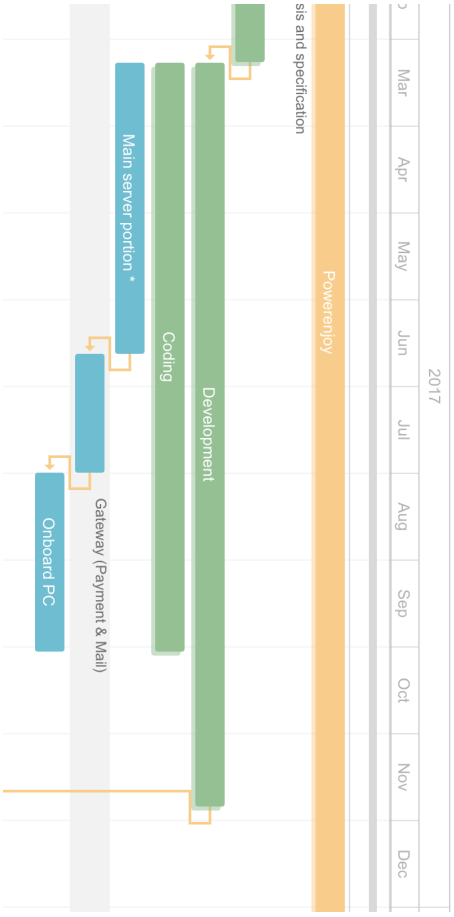


Figure 13: Tommaso Massari Requirements

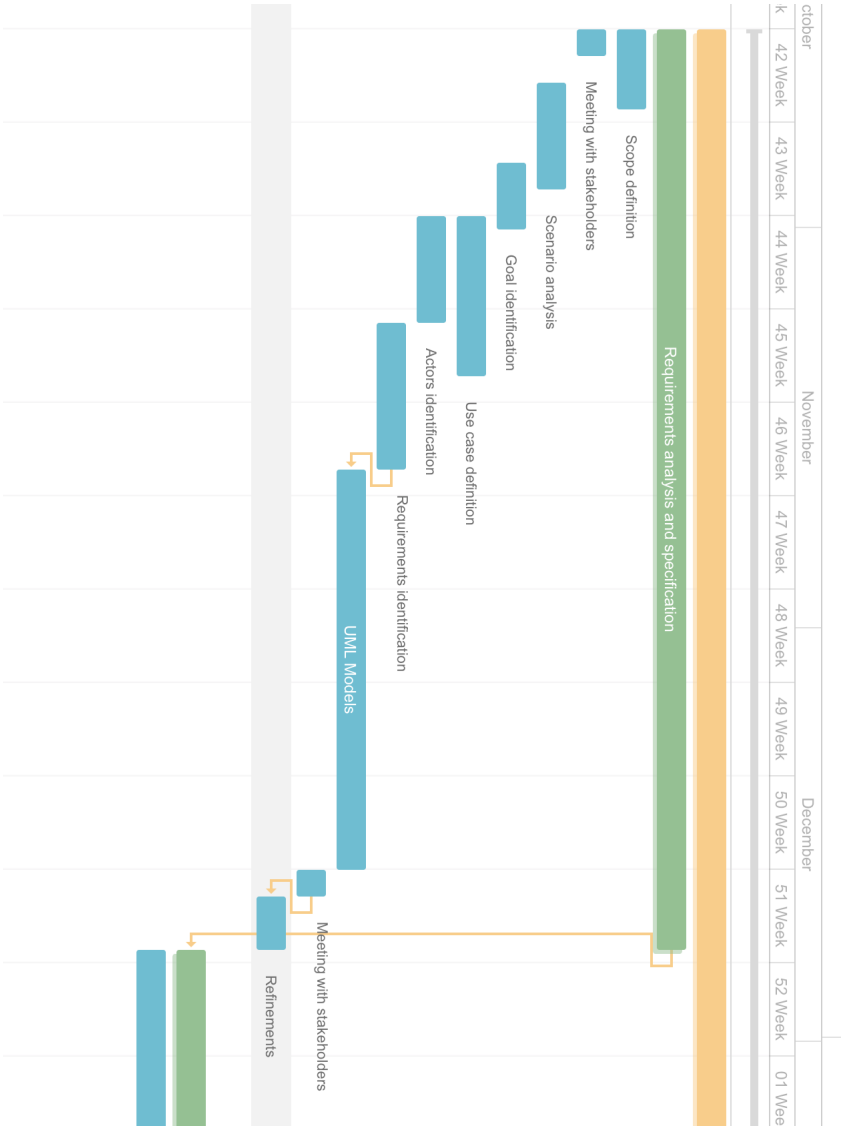


Figure 14: Tommaso Massari Design

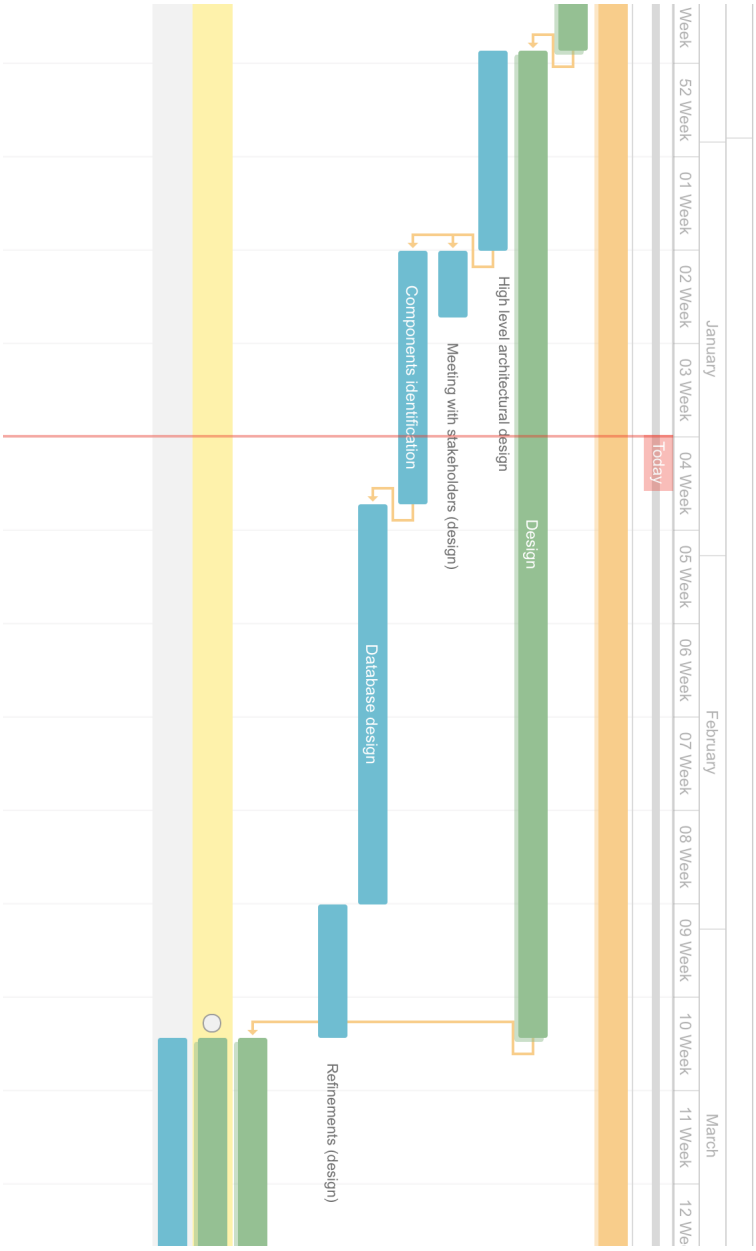
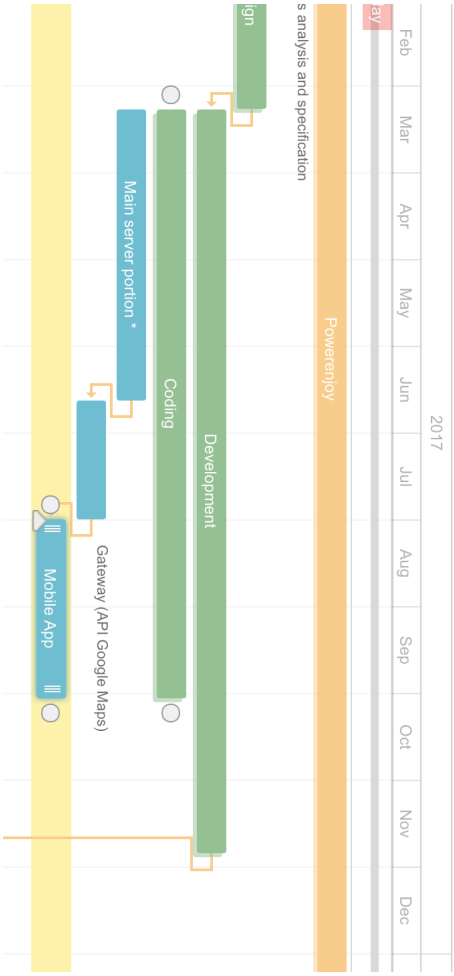


Figure 15: Tommaso Massari Development



6 Risk Management

Risks are always present in every software project, but not always their impact is well-assessed in order to avoid several threats to the project. In this section we show possible risks emerging from the development of the project, their impact in the organizational businesses and some contingency plans to be put in place in case of emergence.We assessed potential risks following a mixed approach: in fact we took into account results from the COCOMO evaluation and opinions from our experienced risk management team. In order to minimize the risk probability to become a real damage, we are strongly motivated to join a proactive approach, meaning that every time a functional feature of the project is completed we're going to refine our schedule plan and almost to assess new risks which may come across. All risks we evaluated and assessed can be grouped by their domain: there are project risks, technical risks and business risks. Their impact could be economical, normative or related to reputation. Following here is a reference table which gives us the definition of the four parameter given to the impact, and then the reference for the probability definition.

6.1 Impact Reference

Table 10: Impact Handbook 1

Impact Type	Economical
Low	Loss between 0,05% and 0,1% of the sales volume
Moderate	Loss between 0,1% and 1% of the sales volume
Elevated	Loss between 1% and 5% of the sales volume
Critical	Loss greater than 5% of the sales volume

Table 11: Impact Handbook 2

Impact Type	Normative
Low	Fines which don't impact on the business
Moderate	Fines and penalties which can be a damage for the business
Elevated	Administrative and civil penalties which can threat the normal ongoing of the project
Critical	Serious criminal and penal offenses which can be followed by a termination of the project

Table 12: Impact Handbook 3

Impact Type	Reputation
Low	Loss of good reputation in a small case from a part of the clientele
Moderate	Loss of good reputation on the entire organization from a part of the clientele
Elevated	Loss of good reputation in a small case from the whole clientele
Critical	Loss of good reputation on the entire organization from the whole clientele

6.2 Probability Reference

Table 13: Probability Reference

Probability Type	Description
Low	Between 0,1% and 10%
Medium	Between 10% and 20%
High	More than 20%

6.3 Project Risks

The table is composed for each risk identified by a risk ID, the description, the risk probability and its impacts in economical, normative and reputational terms.

Table 14: Project Risks

ID	Description	Probability	Economical	Normative	Reputation
1	It is impossible to recruit staff with the skills required for the project	Low	Elevated	n/a	Low
2	Gold plating	Medium	Moderate	n/a	Moderate
3	Misunderstanding of the requirements	High	Moderate	Low	Elevated

6.4 Technical Risks

The table is composed for each risk identified by a risk ID, the description, the risk probability and its impacts in economical, normative and reputational terms.

Table 15: Technical Risks

ID	Description	Probability	Economical	Normative	Reputation
4	Developing the wrong software functions	Medium	Elevated	n/a	Elevated
5	High level of technical complexity	Medium	Elevated	n/a	Moderate
6	The database used in the system cannot process as many transaction per second as expected	High	Critical	n/a	Critical

6.5 Business Risks

The table is composed for each risk identified by a risk ID, the description, the risk probability and its impacts in economical, normative and reputational terms.

Table 16: Business Risks

ID	Description	Probability	Economical	Normative	Reputation
7	Organizational financial problems force reductions in the project budget	Medium	Critical	n/a	Critical
8	The organization is restructured so that different management are responsible for the project	Low	Critical	Low	Crticial
9	Building an excellent system that no one really wants	Medium	Critical	n/a	Critical

6.6 Solutions and Contingency Plans

The risks listed above are the ones that are more likely to happen in the development of our system. Our focus and our attention will be completely put on the processes involved in those risks and as already said we'll follow a proactive approach in order to avoid the worst consequences in case of emergency. As con-

cerns project risks the principal risk is to loss time efficiency due to a too much effort on useless software features (ID 2), but our organization agree that every week each team member is going to be accountable for what he has developed and in which way. Another relevant risk is related to a misunderstanding of the requirements (ID3). To solve this problem, we decided to constantly involve our stakeholders to participate actively in the analysis of the functionalities already developed. In technical risks we found that the most relevant one is to have a database inefficient (ID 6), a risk which is not acceptable in our case. For what it concerns the solution to this problem, we developed a contingency plan which has to be put into practice as soon as we realize that the risk is going to happen. Business risks are the most important ones not only because they are related to the biggest financial losses but also because they seriously threaten the project life. To avoid such risks, we carried out a contingency plan which consists in the fullfilment of the following points:

- Business Impact Analysis: identification of the processes in-scope and definition of their impact on the business
- Risk Assessment: evaluation of the risks in quantitative and qualitative terms
- Business Continuity Plan (Disaster Recovery): development of a plan in order to recover the vital functions of the organization and let the business continue

7 Effort Spent

- Giovanni Agugini:
  - 16/01/2017: 5h
  - 18/01/2017: 3h
  - 20/01/2017: 5h
  - 22/01/2017: 3h
- Matteo Foglio:
  - 16/01/2017: 5h
  - 18/01/2017: 4h
  - 20/01/2017: 4h
  - 22/01/2017: 3h
- Tommaso Massari:
  - 16/01/2017: 4h
  - 18/01/2017: 3h
  - 20/01/2017: 5h
  - 22/01/2017: 4h