Project Plan Document

Version 1.0

Giorgio Pea(Mat. 853872), Andrea Sessa(Mat. 850082)

2/2/2016

Contents

1 Introduction

1.1 Purpose

The main purpose of this document is to analyze effort and cost for MyTaxiService. The analisys is performed using two different models:

- Function Points: to determine the size and the overall complexity of the project
- COCOMO II: to determine the effort and cost of the project

In the final part of the document are also included a Gantt diagram to visualize thepage general schedule of the project and a resource allocation diagram to show how the team members have been assigned to the variuos tasks.

1.2 Acronyms

- **RASD:** Requirements Analisys and Specification Document
- **DD:** Design Document
- **ITPD:** Integration Test Plan Document
- **AWT:** Approximate Waiting Time

1.3 References

- COCOMO II Specification
- Function Point Specification

2 Function Point Analisys

2.1 Introduction

The function point approach is a tecnique that allows the project manager to extimate the size of the project.
It is based on a combination of the project charateristics:

- **Internal Logic Files:** It represents a homogeneous set of data managed by the application
- **External Logic Files:** It represents a homogeneous set of data used by the application but generated and maintained by other applications
- **External Input:** It represents a set of elementary procedures to eleborate data coming from the external environment

- **External Output:** It represents a set of procedures that generates data for the external environment with a significant eleboration of data from logic files
- **External Inquiry:** It represents a set of input/output operation that do not require a significant eleaboration of logic data files.

The following table outline the number of Functional Point based on functionality and relative complexity:

Function Type	Complexity		
	Simple	Medium	Complex
Internal Logic File	7	10	15
External Interface File	5	7	10
External Input	3	4	6
External Output	4	5	7
External Inquiry	3	4	6

2.2 Internal Logic Files

The system needs to store information about:

User
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Administrator
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Mtaxi driver
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Mtaxi
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

WorkTime Table
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Zone
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Location
This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Ride Request This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Booking Request This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

Queue This data entity consist in a small set of information, for this reason its complexity has been considered **SIMPLE**

$$ILFFunctionPoints = numberOfILF * 7 = 7 * 7 = 49$$

2.3 External Logic Files

The system needs to access data about:

External Traffic data
The structure of this data could be complex and could need a digest process, for this reason its complexity has been considered **MEDIUM**

$$ILFFunctionPoints = numberOfELF * 7 = 1 * 7 = 7$$

2.4 External Input

The system needs to process the following input:

Ride Request creation
This operation requires the user to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

Booking Request creation
This operation requires the user to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

Booking Request editing
This operation requires the user to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

User Login/Logout
This operation requires the user to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

User Registration
This operation requires the user to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

Mtaxi Driver Registration
This operation requires the Mtaxi driver to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

Driver Notification
This operation requires the user to perform few and simple actions and the system(including the MYT device) more complex and numerous procedures, for this reason its complexity has been considered **MEDIUM**

Administrator Operations
This operation requires the administrator to perform few and simple actions and the system to perform straightforward checks and data procedures, for this reason its complexity has been considered **SIMPLE**

$$EIFunctionPoints = numberOfSimpleEI * 3 + numberOfMediumEI * 4 = \\ 7 * 3 + 1 * 4 = 25$$

2.5 External Output

This operation requires the system to perform complex calculations on traffic data and Mtaxi positions, for this reason its complexity has been considered **COMPLEX**

AWT Notification

Zone Change Notification
This operation implies that the system noticed an unbalanced distribution of Mtaxi in city zones; this last process requires complex and numerous calculations and data checks, this operation requires the system to perform complex calculations on traffic data and Mtaxi positions, for this reason its complexity has been considered **COMPLEX**

$$EOFunctionPoints = numberOfEO * 7 = 2 * 7 = 14$$

2.6 External Inquiry

User Profile Visualization
This operation the system to retrieve and elaborate data in a simple way, for this reason its complexity has been considered **SIMPLE**

User Ride Request Visualization
This operation the system to retrieve and elaborate data in a simple way, for this reason its complexity has been considered **SIMPLE**

User Booking Request Visualization
This operation the system to retrieve and elaborate data in a simple way, for this reason its complexity has been considered **SIMPLE**

Mtaxi Notification Visualization
This operation the system to retrieve and elaborate data in a simple way, for this reason its complexity has been considered **SIMPLE**

Mtaxi Accident Reports Visualization
This operation the system to retrieve and elaborate data in a simple way, for this reason its complexity has been considered **SIMPLE**

Mtaxi Bad Behavior Reports Visualization
This operation the system to retrieve and elaborate data in a simple way, for this reason its complexity has been considered **SIMPLE**

$$EIFunctionPoints = numberOfEI * 3 = 6 * 3 = 18$$

2.7 Summary

$$TotalFunctionPoints(UFP) = 49 + 7 + 25 + 14 + 18 = 113$$

3 COCOMO II Analisys

3.1 Introduction

The COCOMO II calculations are based on estimates of a project's size in Source Lines of Code (SLOC). In this case the previous Function Points analysis is used to estimate the SLOC number.
Assuming that the programming language used for the will be Java EE, the conversion factor between the total function point counts(UFP) and the SLOC is 46.

$$SLOCs = conversionFactor * UFP = 113 * 46 = 5198 \tag{1}$$

The SLOC number represents the entry point, the analysis proceeds by defining:

- **Cost Drivers:** COCOMO II has 17 cost drivers, i.e. multiplicative factors that determine the effort required to complete the software project
- **Scale Drivers:** COCOMO II has 5 scale driver that defines the most important factors to determine cost and duration of the project

The method defines the effort equation:

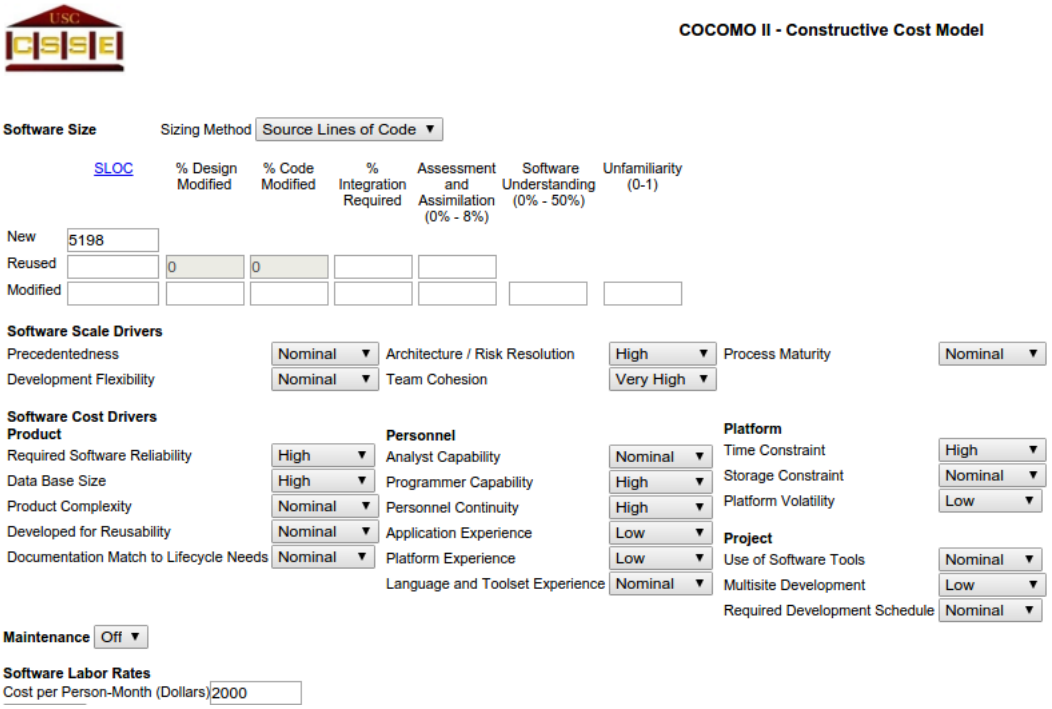
$$effor = 2.94 * EAF * (KSLOC)^E \tag{2}$$

where EAF(Effort Adjustment Factor) depends on the Cost Drivers and E is derived from the Scale Driver.
The duration of the project is defined as:

$$duration = 3.67 * (effort)^E \tag{3}$$

3.2 Analisys

In this section is included the final COCOMO analisys performed using a tool available at:
http://csse.usc.edu/tools/COCOMOII.php



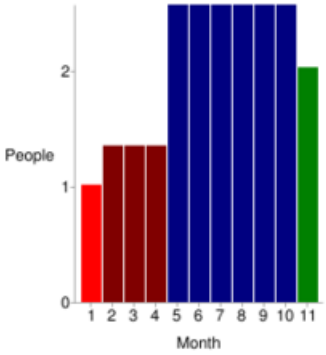
Results

Software Development (Elaboration and Construction)

Effort = 21.3 Person-months
Schedule = 10.1 Months
Cost = $42552

Total Equivalent Size = 5198 SLOC

Staffing Profile



Acquisition Phase Distribution

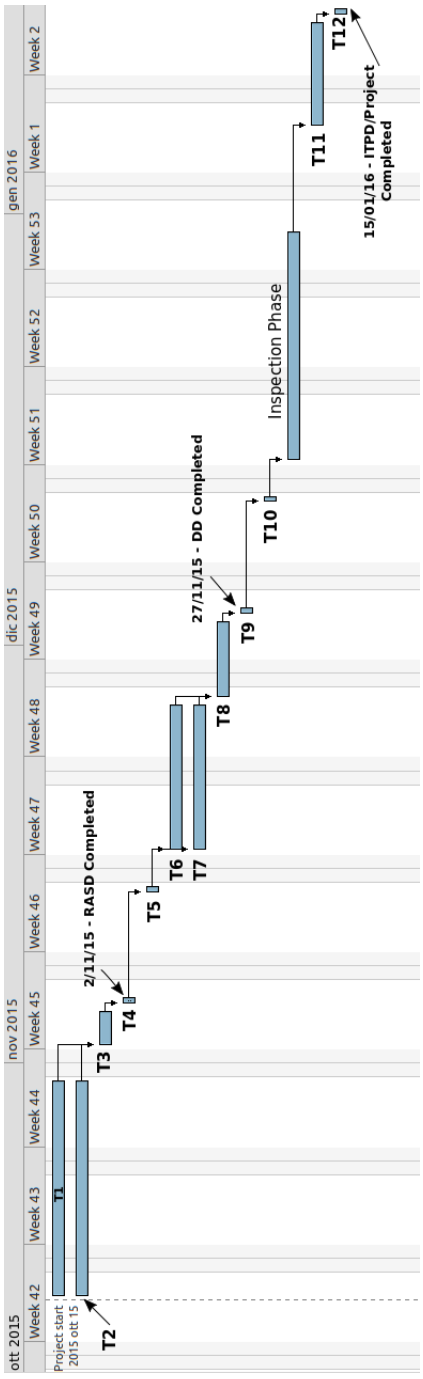
Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	1.3	1.3	1.0	$2553
Elaboration	5.1	3.8	1.4	$10213
Construction	16.2	6.3	2.6	$32340
Transition	2.6	1.3	2.0	$5106

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

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.2	0.6	1.6	0.4
Environment/CM	0.1	0.4	0.8	0.1
Requirements	0.5	0.9	1.3	0.1
Design	0.2	1.8	2.6	0.1
Implementation	0.1	0.7	5.5	0.5
Assessment	0.1	0.5	3.9	0.6
Deployment	0.0	0.2	0.5	0.8

4 Task Gantt Diagram

In this section is included a gantt diagram that represents the tasks in which the project is divided.



In the following paragraph is included an explanation of each task and of its duration in terms of work

- **T1:** Requirements Specification - Duration: 29h
- **T2:** RASD Diagrams Specification - Duration: 29h
- **T3:** Alloy Model Definition - Duration: 4h
- **T4:** RASD Revision - Duration: 2h
- **T5:** RASD Post-Presentation Revision - Duration: 2h
- **T6:** Architecture Specification - Duration: 18h
- **T7:** DD Diagrams Specification - Duration: 18h
- **T8:** Algorithms Definition - Duration: 2h
- **T9:** DD Revision - Duration: 2h
- **T10** DD Post-Presentation Revision - Duration: 2h
- **T11:** Integration Test Plan Definition - Duration: 8h
- **T12:** ITPD Revision: Duration: 1h

5 Resource Allocation Diagram

6 Risk Management

7 Appendix

7.1 Tools

- **Planner:** to draw gantt diagram and resource
- LaTeX/ **Atom:** to redact this document

7.2 Hours

- Andrea Sessa: xxx hours
- Giorgio Pea: xxx hours