myTaxiService

Project plan document

Belluschi Marco 791878, Cerri Stefano 849945, Di Febbo Francesco 852389

February 2, 2016

Contents

Chapter 1

Introduction

1.1 Purpose and scope

1.1.1 Purpose

This is the *Project Plan* (**PP**) for *myTaxiService*. Its purpose is to evaluate time and resources necessary to develop the entire *myTaxiService* system, including mobile and web applications.
To do so, the **Function Points** measurement and the **COCOMO** model have been used, comparing the results with the system size and the implementation time.

1.1.2 Scope

As more widely explained both in the RASD and in the DD, myTaxiService is a taxi service for a large city. The main goals of the system are:

- simplify the access of passengers to the service;
- guarantee a fair management of taxi queues.

See the two aforementioned documents for further explanations.

1.2 List of definitions and abbreviations

Definitions

- Function Point: unit of measurement that expresses the amount of business functionality an information system (as a product) provides to a user compared to software size; costs (in dollars or hours) are calculated from past projects

- COCOMO Model: algorithmic software cost estimation model that uses a basic regression formula with parameters derived from historical project data and current/future project characteristics
- User: person that uses the service applications
- Passenger: passenger registered to the service
- Taxi driver: taxi driver registered to the service
- Mobile App: *myTaxiService* mobile application
- Web App: *myTaxiService* web application
- System: the union of software and hardware to be developed and implemented

Acronyms

- PP: Project Plan
- RASD: Requirements Analysis and Specification Document
- DD: Design Document
- ITPD: Integration Test Plan Document
- FP: Function Point
- COCOMO: COnstructive COst MOdel
- SLOC: Source Line Of Code
- KSLOC: 1000 SLOCs
- EAF: Effort Adjustement Factor
- ILF: Internal Logic File
- EIF: External Interface File
- EI: External Input
- EO: External Output
- EQ: External Inquiry

1.3 List of reference documents

- Software Engineering 2 Project AA 2015/2016: Project Description And Rules
- Software Engineering 2 Project AA 2015/2016: Assignment 5 - Project Plan
- myTaxiService's Requirement Analysis and Specification Document (RASD)
- myTaxiService's Design Document (DD)
- myTaxiService's Integration Test Plan Document (ITPD)

Chapter 2

Function Points

2.1 Introduction

The Functional Point approach is a technique that allows to evaluate the effort needed for the design and implementation of a project. We have used this technique to evaluate the application dimension basing on the functionalities of the application itself. The functionalities list has been obtained from the RASD document and for each one of them the realization complexity has been evaluated. The functionalities has been groped in:

- Internal Logic File: it represents a set of homogeneous data handled by the system
- External Interface File: it represents a set of homogeneous data used by the application but handled by external application
- External Input: elementary operation that allows input of data in the system
- External Output: elementary operation that creates a bit stream towards the outside of the application
- External Inquiry: elementary operation that involves input and output operations

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 FP Estimation

2.2.1 Internal Logic Files

The Internal Logic Files (ILFs) of the system are the following:

- User : medium complexity
- Passenger : simple complexity
- Taxi Driver : medium complexity
- Ride : simple complexity

2.2.2 External Interface Files

The External Interface File (EIF) of the system is the following:

- Maps : is acquired from Google Maps services. High complexity

2.2.3 External Input

The External Inputs (EIs) of the system are the following:

- User Registration : medium complexity
- User Login : simple complexity
- User Logout : simple complexity
- User Profile Management : simple complexity
- Ride Request : high complexity
- Taxi Availability : simple complexity

2.2.4 External Output

The External Outputs (EOs) of the system are the following:

- Passenger Notification : simple complexity
- Confirmation e-mails : simple complexity
- Taxi Driver Notification : simple complexity

2.2.5 External Inquiry

The External Outputs (EQs) of the system are the following:

- Ride Sharing : high complexity
- Taxi Driver ride request handling : simple complexity

2.2.6 Summary

Function Type	**Complexity-Weight**			**Total Points**
	Simple	Medium	Complex	
Internal Logic File	2×7	2×10	0×15	34
External Interface File	0×5	0×7	1×10	10
External Input	3×3	1×4	2×6	25
External Output	3×4	0×5	0×7	12
External Inquiry	1×3	0×4	1×6	9
Total Points	38	24	28	**90**

Chapter 3

COCOMO

3.1 Introduction

This chapter describes the estimation achieved through COCOMO II: a complex, non linear model that takes in account the characteristics of the product, people and processes.

Official manual at: http://csse.usc.edu/csse/research/COCOMOII/cocomo2000.0/CII_modelman2000.0.pdf

In the previous chapter we have estimated a Function Point value. Following the table described in http://www.qsm.com/resources/function-point-languages-table we estimate a conversion factor of 46.

$$90 \text{ FPs} \times 46 = 4140 \text{ SLOC}$$

3.2 Scale Drivers

Detailed descriptions of scale drivers are provided by COCOMO manual, mentioned before.

- **Precedentedness (PREC):** our team lacks of a previous experience using these development methodologies, thus the value will be *very low*.
- **Development flexibility (FLEX):** this project requires general specifications without going deep in details, thus the value will be *nominal*.
- **Architecture/Risk resolution (RESL):** our architecture provides security mechanisms and risk preventions, thus the value will be *high*.
- **Team cohesion (TEAM):** our team is composed by members who had a previous working experience together, thus the value will be *very high*.

- **Process Maturity (PMAT):** reflects the process maturity of the organization. Our CMM level corresponds to *high* value.

Scale driver	Factor	Value
PREC	*Very low*	6.20
FLEX	*Nominal*	3.04
RESL	*High*	2.83
TEAM	*Very high*	2.19
PMAT	*High*	3.12
Total points		17.38

3.3 Cost Drivers

Detailed descriptions of cost drivers are provided by COCOMO manual, mentioned before.

- **Required Software Reliability (RELY):** a software failure in this project may lead to delays on the deliveries thus value will be *nominal*.
- **Data Base Size (DATA):** our system requires a reasonable database size, necessary to modules' tests, thus the value will be *nominal*.
- **Product Complexity (CPLX):** according to the new COCOMO II, this value will be *high*.
- **Developed for Reusability (RUSE):** this project requires the software components to be highly modular and reusable so the value will be *high*.
- **Documentation Match to Life-Cycle Needs (DOCU):** this project, cause of the modularity and reusability of software components, requires a *nominal* level of documentation.
- **Execution Time Constraint (TIME):** this project requires less than 50% of use of available execution time, thus the value will be *nominal*.
- **Main Storage Constraint (STOR):** this project requires less than 50% of use of available storage, thus the value will be *nominal*.
- **Platform Volatility (PVOL):** the software platform should not change too often so the value will be *low*.
- **Analyst Capability (ACAP):** this project requires an high analysis and design ability, thus the value will be *high*.
- **Programmer Capability (PCAP):** our team members show high ability to communicate and cooperate together, thus the value will be *high*.

- **Personnel Continuity (PCON):** this project requires less than half a year of available time, thus the value will be *very low*.
- **Applications Experience (APEX):** our team members don't have a previous experience on this type of project, thus the value will be *very low*.
- **Platform Experience (PLEX):** our team doesn't have sufficient knowledge on platforms used in this project, thus the value will be *very low*.
- **Language and Tool Experience (LTEX):** our team has a minimal knowledge of languages and tools, thus the value will be *low*.
- **Use of Software Tools (TOOL):** our team uses a minimum set of tools for this project, thus the value will be *low*.
- **Multisite Development (SITE):** out team members reside in the same metro area, thus the value will be *high*.
- **Required Development Schedule (SCED):** this project requires many schedule constraints, thus the value will be *high*.

Cost driver	**Factor**	**Value**
RELY	*Nominal*	1.00
DATA	*Nominal*	1.00
CPLX	*High*	1.17
RUSE	*High*	1.07
DOCU	*Nominal*	1.00
TIME	*Nominal*	1.00
STOR	*Nominal*	1.00
PVOL	*Low*	0.87
ACAP	*High*	0.85
PCAP	*High*	0.88
PCON	*Very low*	1.29
APEX	*Very low*	1.22
PLEX	*Very low*	1.19
LTEX	*Low*	1.09
TOOL	*Low*	1.09
SITE	*High*	0.93
SCED	*High*	1.00
Product		1.686

3.4 Schedule Estimation

$Effort : A \times EAF \times KSLOC^{\mathrm{E}} =$ **23.11 person/months**
Where:

- $A = 2.94$
- $EAF = 1.686 \rightarrow product\ of\ all\ the\ cost\ drivers$
- $E = 1.08 \rightarrow 0.91 + 0.01 \times \sum SF : exponent\ derived\ from\ scale\ drivers.$
- $KSLOC = 4.140 \rightarrow estimated\ lines\ of\ code$

$Duration : 3.67 \times Effort^{\mathrm{F}} =$ **9.84 months**
Where:

- $F = 0.314 \rightarrow 0.28 \times 0.2 \times (E - 0.91)$

The estimation of required number of people in this project is:
$P : Effort/Duration =$ **2.35**

COCOMO II - Constructive Cost Model

Model(s): COCOMO ▼
Monte Carlo Risk: Off ▼
Auto Calculate: Off ▼

Software Size Sizing Method: Source Lines of Code ▼

	SLOC	% Design Modified	% Code Modified	% Integration Required	Assessment and Assimilation (0% - 8%)	Software Understanding (0% - 50%)	Unfamiliarity (0-1)
New	4140						
Reused		0	0				
Modified							

Software Scale Drivers

Precedentedness	Very Low ▼	Architecture / Risk Resolution	High ▼	Process Maturity	High ▼
Development Flexibility	Nominal ▼	Team Cohesion	Very High ▼		

Software Cost Drivers

Product		Personnel		Platform	
Required Software Reliability	Nominal ▼	Analyst Capability	High ▼	Time Constraint	Nominal ▼
Data Base Size	Nominal ▼	Programmer Capability	High ▼	Storage Constraint	Nominal ▼
Product Complexity	High ▼	Personnel Continuity	Very Low ▼	Platform Volatility	Low ▼
Developed for Reusability	High ▼	Application Experience	Very Low ▼	**Project**	
Documentation Match to Lifecycle Needs	Nominal ▼	Platform Experience	Very Low ▼	Use of Software Tools	Low ▼
		Language and Toolset Experience	Low ▼	Multisite Development	High ▼
				Required Development Schedule	High ▼

Maintenance Off ▼

Software Labor Rates
Cost per Person-Month (Dollars)

Calculate

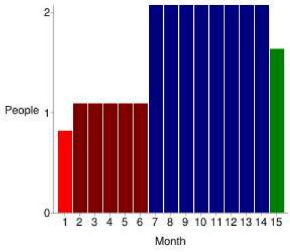
Results

Software Development (Elaboration and Construction)

Effort = 22.8 Person-months
Schedule = 13.4 Months
Cost = $0

Total Equivalent Size = 4140 SLOC

Staffing Profile



Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	1.4	1.7	0.8	$0
Elaboration	5.5	5.0	1.1	$0
Construction	17.3	8.4	2.1	$0
Transition	2.7	1.7	1.6	$0

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

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.2	0.7	1.7	0.4
Environment/CM	0.1	0.4	0.9	0.1
Requirements	0.5	1.0	1.4	0.1
Design	0.3	2.0	2.8	0.1
Implementation	0.1	0.7	5.9	0.5
Assessment	0.1	0.5	4.2	0.7
Deployment	0.0	0.2	0.5	0.8

Your output file is http://csse.usc.edu/tools/data/COCOMO_January_31_2016_06_46_45_543169.txt

Created by Ray Madachy at the Naval Postgraduate School. For more information contact him at rjmadach@nps.edu

COCOMO tool available on http://csse.usc.edu/tools/COCOMOII.php

Chapter 4

Tasks and Schedule

4.1 Tasks

The tasks of the project are:

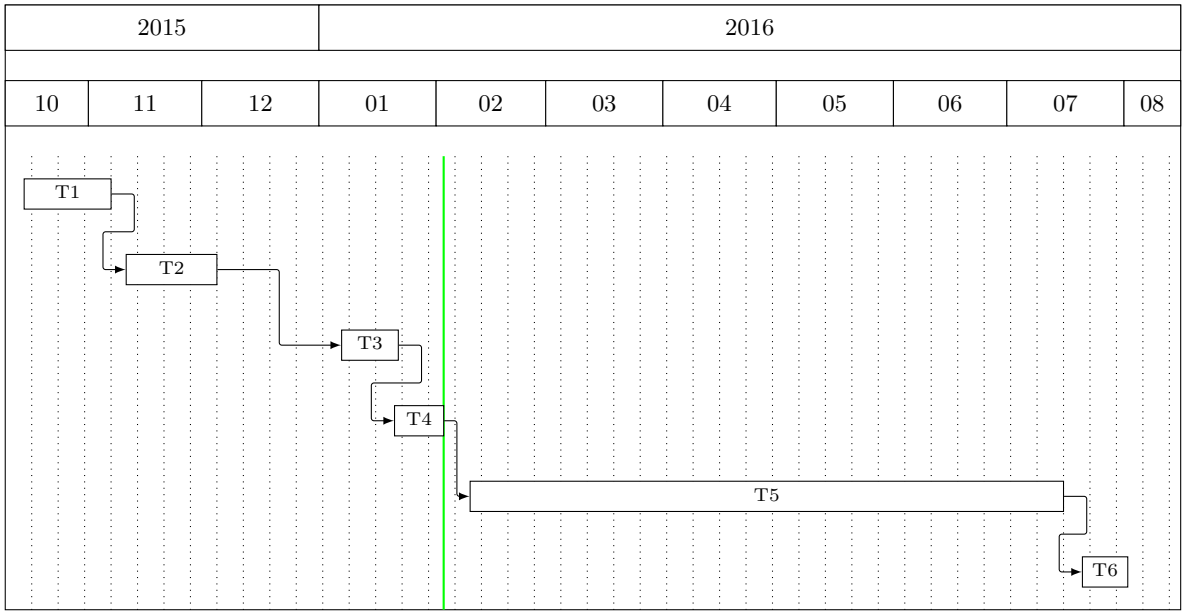
- **T1** Requirement Analysis and Specification Document (RASD) :
- **T2** Design Document (DD)
- **T3** Integration Testing Plan Document (ITPD)
- **T4** Project Plan (PP)
- **T5** Implementation
- **T6** Testing

4.2 Schedule

The schedule of the project is shown in the following table:

Task	Start Date	Deadline
T1	2015-10-15	2015-11-06
T2	2015-11-11	2015-12-04
T3	2016-01-07	2016-01-21
T4	2016-01-21	2016-02-02
T5	2016-02-10	2016-07-15
T6	2016-07-21	2016-08-01

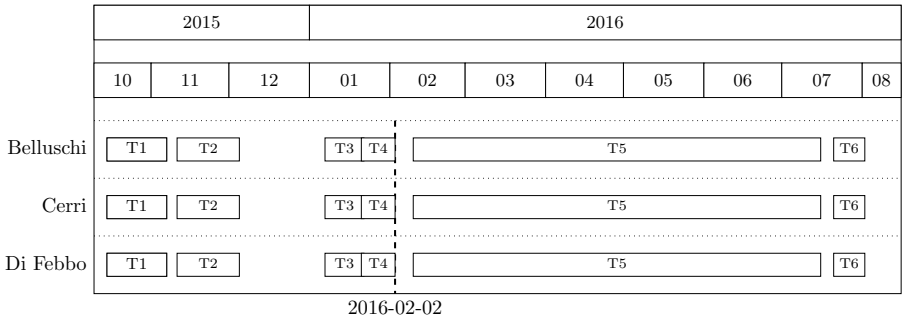
The Gantt chart is:



Chapter 5

Resources

This section shows the allocation of team members to each task.



In this project there aren't parallel tasks, thus each member will work on the same task. Furthermore our team consists of three members, whereas COCOMO estimation suggests two people, so this allocation can respect the schedule.

Chapter 6

Risks

The project is subject to a series of risks with different levels of danger.
Main problems such as developers' lack of experience in programming and in project planning and management could potentially slow down the overall project development, thus leading to the risk of an extension of some deadlines, or at least to some incomplete, minor tasks at the time of delivery. In particular, the team could get into various technical issues, mainly deployment difficulties of myTaxiService in pre-existing, large-city, taxi-management systems (deliberately not considered in the development of myTaxiService). Besides, scalability issues and subsequent server downtime and data loss could pose some major threats. Finally, noteworthy legal and economical problems such as bankruptcy, market competition and major changes in taxi management policy ought to be taken into account: to avoid them, a good feasibility study is recommended.

Appendix A

Appendix

A.1 Software and tool used

- TeXstudio http://www.texstudio.org/: to redact and to format this document

A.2 Working hours

This is the time spent for redact the document

- Belluschi Marco: 6 hours
- Cerri Stefano: 6 hours
- Di Febbo Francesco: 6 hours