My Taxi Service:

PROJECT PLAN

Julián David Gallego García

Politecnico di Milano january 28, 2016


POLITECNICO
MILANO

CONTENTS

1. Introduction

1.1. Purpose and scope

The goal of this project is to optimize the taxi service in a large city through a web application and a mobile app. These applications allow the passengers to request a taxi in an easy way and the taxi drivers to have a fair management of taxi queues. This document has the goal to present the principal aspects to start the activities related to the develop of the project, that aspects are, effort, how many people are need it, time that will be expended, the cost, the schedule and the risks.

This document is the Integration Test Plan Document for the system My Taxi Service project. This document is intended for the development team and stakeholder, highlighting the effort need it to develop the target, the time need it and the distribution of task and the risks that will affront the project.

1.2. List of Definitions and Abbreviations

COCOMO: Constructive Cost Model

RASD: Requirements analysis and specification document.

DD: Design Document.

ILF: Internal Logic File.

ELF: External Interface File

1.3. List of Reference Documents

RASD: Julian Gallego, Requirements analysis and specification document for My taxi service.

DD: Julian Gallego, Design document for My taxi service.

CID: Julian Gallego, Code inspection document.

ITPD: Julian Gallego, Integration Test Plan document for My taxi service.

Specification document: Software Engineering 2 Project, Structure of the project plan document.

1.4. Overview

The document is organized in:

- Section 2.1-2.3: Present an approximation of the effort need to accomplish the develop of the system, the quantity of persons required and the cost, all this using a tool call COCOMO II and based on the function points.
- Section 2.4: It will be enunciated all the tasks that will be need it to be done to create the system, then the schedule of this tasks will be presented a comparison if the develop team have more persons.
- Section 2.5: The possible risks that the team will have to solve in the program development and deployment of the system.

2. Project Plan

2.1. Function points

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

2.1.1. Internal Logic File:

The application stores the information about

- Users
- Queues
- Position
- Requests.

Each of these entities has a simple structure as it is composed of a small number of fields. Thus, we can decide to adopt for all the four the simple weight.

Thus, 4 x 7 = 28 FPs concerning ILFs.

2.1.2. External Interface File

It represents a set of homogeneous data used by the application but handled by external application. In the Taxi Service application there are no such functions 0 FPs concerning ELFs.

2.1.3. External Input

The application interacts with the users as follows:

Users registration: this operation involves one entity, the users. It can still be considered simple, so, its adopted the simple weight. 1 x 3 = 3 FPs

Login/logout: these are simple operations, so its can adopt the simple weight for them.
2 x 3 = 6 FPs

Manage personal information: this operation involves one entity, the users. It can be considered medium, so, its adopted the weight. 1 x 4 = 4 FPs

Change taxi drivers' status: This operation involves three entities, the users, the queues and the position. It can be considered medium complexity. 1 x 4 = 4 FPs

Request a Taxi: This operation involves two entities, the request and the position. It can be considered medium complexity. 1 x 4 = 4 FPs

Accept/decline a taxi request(driver): This operation involves one entity, the Queues. It can be considered simple complexity. 1 x 3 = 3 FPs.

2.1.4. External Output

After Registration, the application will send an email validation to the creator of the user this operation involves two entities, user and notification; this can be considered medium cost. 1 x 5 = 5 FPs.

2.1.5. External Inquiry

The application allows customers to request information about:

- Their personal information.
- Notification about the ride.

Are two different external inquiries that can be consider of medium complexity. 2 x 4 = 8 FPs.

2.1.6. Total FP numbers

At the end, taking into account all the the point for every aspect related to the function points its arrived to a total of 65 function points, this way:

ILFs: 28
ELFs: 0
External Inputs: 24
External Outputs: 5
External Inquiries: 8
Total Estimate: 65.

2.2. COCOMO Approach

To pass from FP to SLOC its used an average conversion factor of 46 as described at http://www.qsm.com/resources/function-point-languages-table.

$$65FPs * 46 = 2990\ SLOC$$

A first estimation is based on FPs approach converted to SLOC. We Consider a project with all "Nominal" Cost Drivers and Scale Drivers would have an EAF of 1.00 and exponent E of 1.0997. Following this formula

$$effort = 2.94 * EAF * (KSLOC)^E$$

where:
EAF: Effort Adjustment Factor derived from Cost Drivers.
Exponent derived from Scale Drivers.

Its obtain

$$effort = 2.94 * (1.0) * (2990)^{1.0997} = 9.8\ Person/months$$

Now its try to calculate the schedule (duration) of project in month with the following formula

$$Duration = 3.67 * effort^E$$

We consider an exponent E of 0.3179

$$Duration = 3.67 * 9.8^{0.3179} = 7.58\ months$$

Now its can be estimate the number of people needed to complete the project with the following formula

$$Npeople = effort/duration$$

$$Npeople = 9.8/7.58\ = 1\ person$$

The result of the Npeople is exactly 1.2 people, knowing that you can't have that quantity of persons and the fractional is very small its chosen 1 as result, that is a good result taking into a count that the project was mean to be for 1 person.

2.3. COCOMO computation

2.3.1. Evaluation using COCOMO II tool

To give a more precise estimation adjusting some Scale Driver. To evaluate the COCOMO II and determine the effort required to complete the software project we also use an online tool that helps us to do some calculus (http://csse.usc.edu/tools/COCOMOII.php). Its added the choices made about the Scale Driver to adjust the parameters to the project characteristics and the result report of that site.

COCOMO II - Constructive Cost Model

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	2990						
Reused		0	0				
Modified							

Software Scale Drivers

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

Software Cost Drivers

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

Maintenance Off

Software Labor Rates

Cost per Person-Month (Dollars) 2000

Calculate

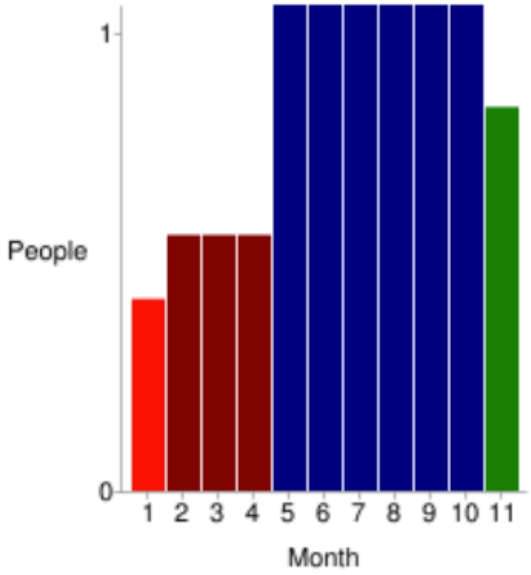
2.3.2. Results of COCOMO II tool

Software Development (Elaboration and Construction)

Effort = 8.4 Person-months
Schedule = 9.6 Months
Cost = $16793

Total Equivalent Size = 2990 SLOC

Staffing Profile



Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	0.5	1.2	0.4	$1008
Elaboration	2.0	3.6	0.6	$4030
Construction	6.4	6.0	1.1	$12763
Transition	1.0	1.2	0.8	$2015

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

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.1	0.2	0.6	0.1
Environment/CM	0.1	0.2	0.3	0.1
Requirements	0.2	0.4	0.5	0.0
Design	0.1	0.7	1.0	0.0
Implementation	0.0	0.3	2.2	0.2
Assessment	0.0	0.2	1.5	0.2
Deployment	0.0	0.1	0.2	0.3

2.4. Schedule

2.4.1. Tasks:

- M1 Requirements analysis and specification document:

T1 Define Goals, assumptions, constraints and dependencies for the systems.
T2 Define System attributes, functional requirements, software functions.
T3 Define Software interface requirements.
T4 Do the alloy model of the system.
T5 Define the scenarios, use cases and class diagram.
T6 Writing of the document with the correct format parting from all the work made so far.

- M2 Design Document:

T7 Define architectural style and patterns
T8 Define High levels components, description and deployment view.
T9 Define Run time view and sequence diagrams.
T10 Define Algorithm design, experience diagram and requirements traceability.
T11 Writing of the document with the correct format parting from all the work made so far.

- M3 Code inspection document:

T12 Review Naming conventions, Indention, Braces, Consistent Allman style, File and Organization on the code.
T13 Review Wrapping lines, Comments, Java source files, Package and import statements and Class and interface declarations on the code.
T14 Review Initialization and declarations, Method calls, Arrays, Object comparison ad Output Format on the code.
T15 Review Computation, comparisons and assignments Exceptions, Flow of control and Files on the code.
T16 Writing of the document with the correct format parting from all the work made so far.

- M4 Integration Plan Document:

T17 Define the Integration strategy, entry criteria, elements to be integrated
T18 Define integration sequence, individual steps and test description
T19 Define Test Procedures
T20 Describe tools, Test Equipment, program stubs and test data require.
T21 Writing of the document with the correct format parting from all the work made so far.

- M5 Project Plan Document:

T22 Identify the function points then using COCOMO estimate the effort and cost of the system.
T23Identify the tasks of the project and their schedule.
T24 Assign the tasks to the available members of the group.
T25 Define the risks for the project, their relevance and recovery actions
T26 Writing of the document with the correct format parting from all the work made so far.

2.4.2. Tasks duration and dependencies

Timetable of work made during the semester for the project.

Document	Tasks	Start date	Finish date	Duration	Dependencies
Requirements analysis and specification document	T1	22/10/15	23/10/15	2 days	
	T2	24/10/15	25/10/15	2 days	
	T3	26/10/15	29/10/15	3 days	T1,T2
	T4	30/10/15	02/11/15	4 days	
	T5	03/11/15	03/11/15	1 day	T2
	T6	04/11/15	06/11/15	3 days	T1,T2,T3,T3,T5
Design document for My taxi service	T7	12/11/15	13/11/15	2 days	M1
	T8	14/11/15	14/11/15	1 day	T7
	T9	15/11/15	19/11/15	5 days	T5,T8,M1
	T10	22/11/15	24/11/15	3 days	T9, M1
	T11	01/12/15	04/12/15	4 days	T7,T8,T9,T10
Code Inspection document	T12	16/12/15	19/12/15	4 days	
	T13	16/12/15	19/12/15	----	
	T14	16/12/15	19/12/15	----	
	T15	16/12/15	19/12/15	----	
	T16	02/01/16	04/01/16	3 days	T12,T13,T14,T15
Integration Plan document	T17	16/01/16	16/01/16	1 day	T8
	T18	17/01/16	17/01/16	1 day	T17,M1,M2
	T19	18/01/16	18/01/16	1 day	T18,M1,M2
	T20	19/01/16	19/01/16	1 day	T17,T18,T19,M2
	T21	20/01/16	21/01/16	1 day	T17,T18,T19,T20
Project Plan document	T22	23/01/16	23/01/16	1 day	M1,M2,M3,M4
	T23	23/01/16	23/01/16	----	
	T24	24/01/16	24/01/16	1 day	T23
	T25	24/01/16	24/01/16	----	
	T26	25/01/16	25/01/16	1 day	T22,T23,T24,T25
Total Time				45 days	

2.4.3. Allocation of the resources for the project

Timetable of work made during the semester for the project is the work was made with a partner.

	Responsible
	Integrant 1
	Integrant 2
	All

Document	Tasks	Start date	Finish date	Duration	Dependencies	
Requirements analysis and specification document	T1	22/10/15	23/10/15	2 days		
	T2	22/10/15	23/10/15	2 days		
	T3	24/10/15	26/10/15	3 days	T1,T2	
	T4	24/10/15	27/10/15	4 days		
	T5	27/10/15	27/10/15	1 day	T2	
	T6	28/10/15	29/10/15	2 days	T1,T2,T3,T3,T5	
Design document for My taxi service	T7	12/11/15	12/11/15	1 day	M1	
	T8	13/11/15	13/11/15	1 day	T7	
	T9	14/11/15	16/11/15	3 days	T5,T8,M1	
	T10	17/11/15	18/11/15	2 days	T9, M1	
	T11	01/12/15	02/12/15	2 days	T7,T8,T9,T10	
Code Inspection document	T12	16/12/15	17/12/15	2 days		
	T13	16/12/15	17/12/15	----		
	T14	16/12/15	17/12/15	----		
	T15	16/12/15	17/12/15	----		
	T16	02/01/16	02/01/16	1 day	T12,T13,T14,T15	
Integration Plan document	T17	16/01/16	16/01/16	1 day	T8	
	T18	17/01/16	17/01/16	1 day	T17,M1,M2	
	T19	18/01/16	18/01/16	1 day	T18,M1,M2	
	T20	19/01/16	19/01/16	1 day	T17,T18,T19,M2	
	T21	20/01/16	20/01/16	1 day	T17,T18,T19,T20	
Project Plan document	T22	23/01/16	23/01/16	1 day	M1,M2,M3,M4	
	T23	23/01/16	23/01/16	----		
	T24	24/01/16	24/01/16	1 day	T23	
	T25	24/01/16	24/01/16	----		
	T26	25/01/16	25/01/16	1 day	T22,T23,T24,T25	
Total Time				28 days		

Looking the total time from the two tables, working 2 persons on the project the time that its need need it to accomplish is reduced to the half and the most important the workload is reduced considerably.

2.5. Risks analysis

2.5.1. Project risks

Underestimated development time: underestimation of the time necessary for accomplish a task, the relevance of this problem on the project is very high and the probability to this to happen is high, because it can slip the project schedule and increase the costs. The recovery action for this problem is finish the task in the shortest time possible and reschedule trying to recovery the time lost cutting time in the tasks that are less critical or easy to accomplish.

Requirements changes: changes on the goals or the functionalities of the project, the relevance is also very high and the probability to this to happen is moderate, because it will slip the project schedule and increase the costs. The recovery action for this problem is first make the changes that can be implement immediately then have a pre delivery of the product on the programed date without the new features that couldn't be implemented if them don't change dramatically the implementation of the final product or reschedule and postpose the delivery of the final project if it does.

2.5.2. Technical risks

The database used in the system cannot process as many transactions per second as expected: the relevance is moderate and the probability to this to happen is moderate, this will increase the costs. The recovery action for this if the flow of transactions grow so much will be the change of the database to a one that can handle it.

Problems with the communication network: the relevance is high and the probability to this to happen is low, this will have terrible repercussions on the service. The recovery action from this will be contact the enterprise in charge on the service and ask them for solutions, in the mean time have everything ready to restart the service when the communication will be ok again.

Staff illness: The relevance is moderate and the probability to this to happen is moderate, this will have repercussions on the workload of the other workers and will slow down some process. The recovery action for this will be the reorganize of the work team so that there is more overlap of work and people therefore understand each other's jobs.

Staff conflicts: The relevance is moderate and the probability to this to happen is moderate, this will have repercussions on the workflow and slow down some process. The recovery action for this will call a meeting with the persons involved in the problem and the human resources manager to deal with the problem, if the problem persists, reorganize the work team so the conflicts part not will longer be working together.