POLITECNICO DI MILANO

SOFTWARE ENGINEERING 2 (2015 - 2016)

Project Plan Document

myTaxiService V1

Authors:

Sun Chao

Daniel Naveda

Bakti Ariani Melinda Pertiwi

January 29th 2016

Contents

1. Introduction

1.1 Purpose and Scope

1.1.1 Purpose

The purpose of the project plan is to covers the content and enablement portions of the MyTaxiService project. A separate plan covers the tooling component.

1.1.2 Scope

Project plan document estimate the project size and estimate effort and cost. Identify the task for the project and the schedule. Allocate the resource to the various task. Define the risk of the project, the relevance and the associated recovery actions.

2. Function Point

- **Internal Logic Files :** Data maintained by the application. We count the number of the table or entities in Database. The database is comprised of the following tables: Passanger, Driver, and System Manager. All of them can be considered of medium complexity.
  Total 4 medium => 3*10 = 30

- **External Interfaces :** Data used by our application but maintained by another (external) application. In Taxi Service there are no external interface.

- **External Inquiries :** Small interaction with the environment, without much data processing (diagnostic). We count the number of queries.
  View Guidebook, Order history, View history
  Total 3 medium => 3*4 = 12

- **External Inputs :** Actions or data from the environtment to the system.
  User inputs include Registration, Login/Logout, input destination, set available, pick up setting, Profile settings change

Total: 5 Simple, 1 High => 6 + 3*5 = 21

- **External Outputs :** We count the operations that generate Data/Report/Something for the external environtment

  Taxi code notification, Waiting time notifications

  2 Simple => 2* 4 = 8

- **COCOMO Analysis**

  We will derive the required effort using the COCOMO post architecture model, using the actual source line count.

  - **Master formula :**

    PM = 2.94 * [Size]^E * product(effort multipliers)

    Where

    E = 0.91 + 0.01 * sum(scale factors)

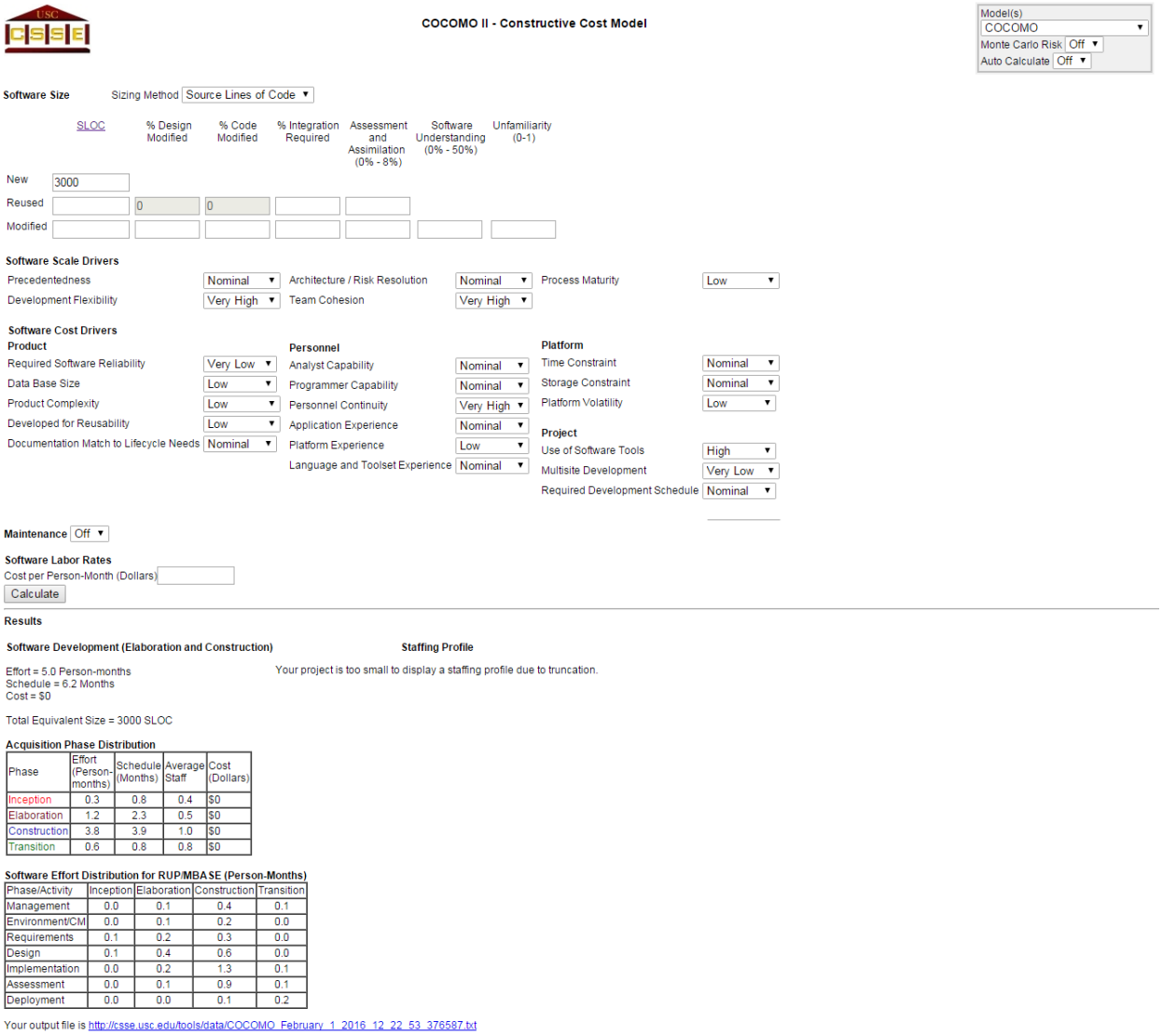
    Size = 3KSLOC (considering 0 adapted SLOC, and 0 breakage factor due to requirements changes)

    All terms regarding adapted sources have been set to zero because we're developing a product from scratch.

  - **Scale and Cost Drivers**

    According to the Cocomo specifications, for each of the cost and scale drivers, a value between very low and very high is chosen. Each value is mapped to a weight (that can be found on the tables in the linked manual) and computed in the formulas above.

    To perform these calculation conveniently we used an online calculator. In the screenshot below, the values we chosen for every driver can be seen, as well as the result of the calculation: 5 man-month.



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	3000						
Reused		0	0				
Modified							

Software Scale Drivers

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

Software Cost Drivers

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

Maintenance Off

Software Labor Rates
Cost per Person-Month (Dollars)
Calculate

Results

Software Development (Elaboration and Construction)

Effort = 5.0 Person-months
Schedule = 6.2 Months
Cost = $0

Total Equivalent Size = 3000 SLOC

Staffing Profile

Your project is too small to display a staffing profile due to truncation.

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	0.3	0.8	0.4	$0
Elaboration	1.2	2.3	0.5	$0
Construction	3.8	3.9	1.0	$0
Transition	0.6	0.8	0.8	$0

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

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.0	0.1	0.4	0.1
Environment/CM	0.0	0.1	0.2	0.0
Requirements	0.1	0.2	0.3	0.0
Design	0.1	0.4	0.6	0.0
Implementation	0.0	0.2	1.3	0.1
Assessment	0.0	0.1	0.9	0.1
Deployment	0.0	0.0	0.1	0.2

Your output file is http://csse.usc.edu/tools/data/COCOMO_February_1_2016_12_22_53_376587.txt

3. Allocation the Resource

During various phases of the project it was impossible to keep an exact count of the time, as a lot of our time went into thinking about the possible architecture of the system. The provided count is thus a very rough estimate, provided mainly to give a general idea and compare our results with the estimation effort provided by Cocomo.

Requirements document	About 38 hours each
Design document	About 30 hours each
Implementation and testing	About 50 hours each
Acceptation testing	About 8 hours each

With an approximate total of 378 hours of work. We can approximately say that this value amounts for less than four months. There can be several reasons for the discrepancy between the actual and expected effort, the first being the different nature of the project: MyTaxiService project, isn't indeed a commercial product, but a prototype one, and it has still many rough edges.

4. Risk

The following risks have been identified and the appropriate action identified to mitigate their impact on the project. The impact (or severity) of the risk is based on how the project would be affected if the risk was triggered. The trigger is what milestone or event would cause the risk to become an issue to be dealt with.

No.	Risk	Impact	Trigger	Mitigation Plan
1.	Changes to the functionality may negate the tests already written and we may loose test cases already written.	High – to schedule and quality	Loss of all test cases	Export data prior to any upgrade, massage as necessary and re-import after upgrade.
2.				