Cost Estimation

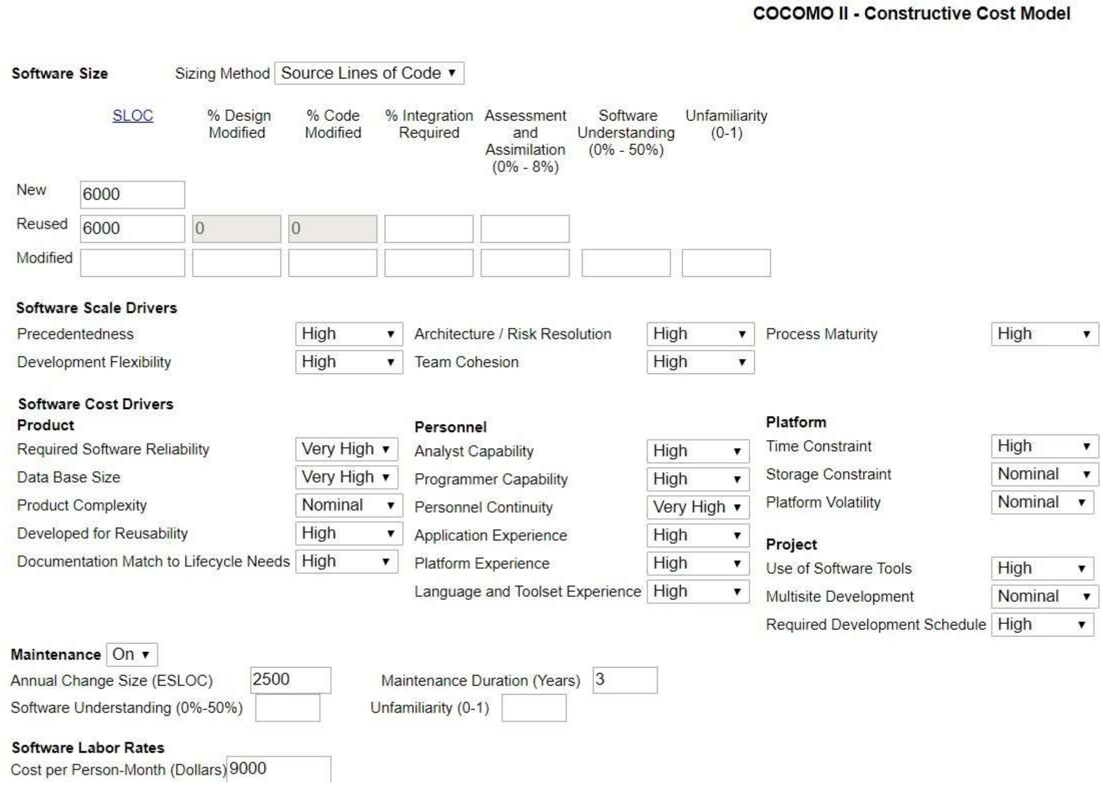
Specification and justification for cost modelling technique

The COCOMO II (Constructive Cost Model II) is a software cost estimation model used to estimate the cost, effort and schedule for a new software development activity. The model consists of three sub models that offer increasingly detailed accuracy. These include Application Composition used for early prototyping efforts and Early Design and Post-Architectural models used further into the development life cycle. The COCOMO II model derives its estimations based on estimated project size, several scale drivers and software cost drivers that assess the project, development environment, and project team.

The COCOMO II model is very well suited to our cost estimation requirements. COCOMO II's use of industry data was very helpful in terms of getting an approximate starting figure. The generated estimations will allow the project management team to have a very clear and in depth understanding of the budget breakdown, how to determine a development team size and meet project deadlines.

The project cost estimation will also give the team an idea of how to approach investors along with determining how many investors will be required to finance the project. The estimated costs can be weighed against the estimated profits in efforts to further persuade investors about the feasibility and profitability of the project.

Project cost (breakdown)



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	6000						
Reused	6000	0	0				
Modified							

Software Scale Drivers

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

Software Cost Drivers

Product

Required Software Reliability	Very High
Data Base Size	Very High
Product Complexity	Nominal
Developed for Reusability	High
Documentation Match to Lifecycle Needs	High

Personnel

Analyst Capability	High
Programmer Capability	High
Personnel Continuity	Very High
Application Experience	High
Platform Experience	High
Language and Toolset Experience	High

Platform

Time Constraint	High
Storage Constraint	Nominal
Platform Volatility	Nominal

Project

Use of Software Tools	High
Multisite Development	Nominal
Required Development Schedule	High

Maintenance On

Annual Change Size (ESLOC)	2500	Maintenance Duration (Years)	3
Software Understanding (0%-50%)		Unfamiliarity (0-1)	

Software Labor Rates

Cost per Person-Month (Dollars) 9000

Results

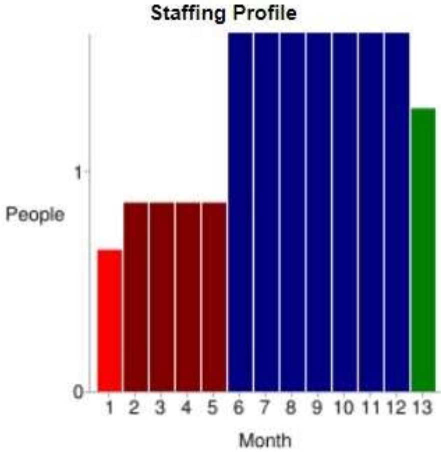
Software Development (Elaboration and Construction)

Effort = 15.9 Person-months
Schedule = 11.9 Months
Cost = $143201

Total Equivalent Size = 6000 SLOC

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	1.0	1.5	0.6	$8592
Elaboration	3.8	4.5	0.9	$34368
Construction	12.1	7.4	1.6	$108833
Transition	1.9	1.5	1.3	$17184



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

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.1	0.5	1.2	0.3
Environment/CM	0.1	0.3	0.6	0.1
Requirements	0.4	0.7	1.0	0.1
Design	0.2	1.4	1.9	0.1
Implementation	0.1	0.5	4.1	0.4
Assessment	0.1	0.4	2.9	0.5
Deployment	0.0	0.1	0.4	0.6

Maintenance

Annual Maintenance Effort = 6.4 Person-Months
Annual Maintenance Cost = $57790
Total Maintenance Cost = $173372