Software Project Management: Graphical Password Strategy

SOFE: Software Project Management

Lab 3: Cost estimation, Activity Diagram, Risk

Hamza Farhat Ali- 100657374

Mohammad Minhal Syed - 100618744

Marvin Vuu - 100658622

Calculating Effort:

Since we are going to be using waterfall software development for the graphical strategy password we believe that is best to use COCOMO I. Our system is an Embedded System as it deals with a high level of complexity, creativity and experience to produce our final product. It requires a large team size and our development team needs to be sufficiently experienced and creative to develop complex models. To calculate the effort this system will take to implement, we will be using the COCOMO model. This uses the number of Lines of Code. From this model we will be able to calculate effort and the duration of the project. The basic model for COCOMO is able to give us a rough estimation of Software Costs quickly.

This is shown as:

$$E = a(KLOC)^b$$

Project Type	**a**	**b**
Organic	3.2	1.05
Semi Detached	3.0	1.12
Embedded	2.8	1.20

The constants for different project types to calculate project months.

The effort is then given to us in person months. We are using an estimate of 2000 lines of code due to the JavaScript functionality being added to any general web application.

$$E = 2.8(2)^{1.20}$$

E=6.4 person months

We can then find the COCOMO duration estimation from constants:

Project Type	**a**	**b**
Organic	2.5	0.38
Semi Detached	2.5	0.35
Embedded	2.5	0.32

$$D = a * E^{b}$$

Suppose the overall size of the embedded project type is estimated to be 2000

$$D = 2.5(2)^{0.32}$$

Duration = 3.12 Months.

The duration of this project is 3.12 months and 6.4 person-months. We believe this is reasonable as the JS functionality that will be added to any of our customers already existing web applications.

Function Points:

Next, to get an estimate of the functionality of this graphical password application to get an indication of the size. The following steps will occur in the application's function point analysis:

1. Count the number of functions of each proposed type.
2. Compute the Unadjusted Function Points(UFP).
3. Find Total Degree of Influence(TDI).
4. Compute Value Adjustment Factor(VAF).
5. Find the Function Point Count(FPC).

Function Type	Simple	Average	Complex
External Inputs: The related data entering the systems are all the user inputs and how many images they choose to set as their password, the maximum being 100 images as referred to in the project objectives.	1	25	100
External Outputs: This is the same amount of data to enter the system to check if the password is correct.	1	25	100
External Inquiries: There is only one correct password to allow the user to login	1	2	3
Internal Logical Files: This includes any files that save the images if it may be in hash tables	5	10	20
External Interface Files: This includes the logical files being used by the application to receive the password, php files, to retrieve and check from database	5	7	10

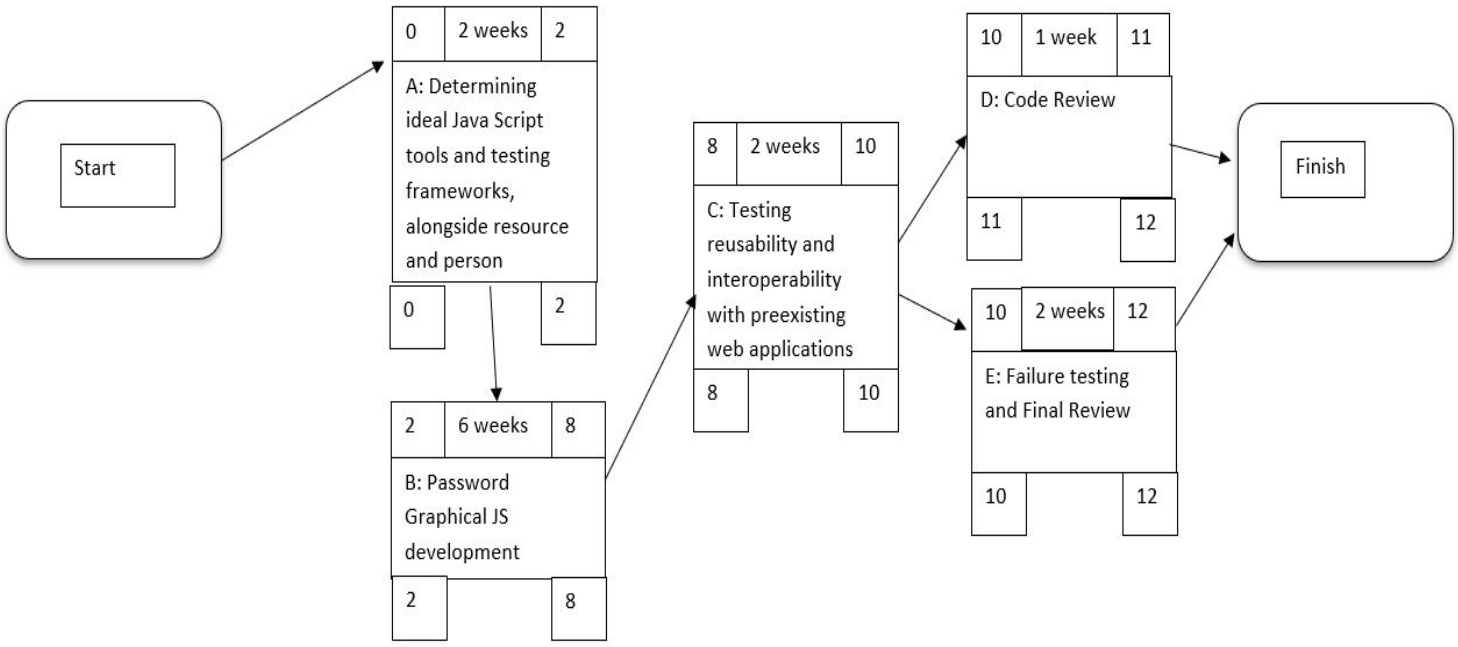
The total number of functions and the unadjusted function points are calculated above. The total degree of influence is determined by the images the user chooses. This leads to why the most complex function points is determined by the max number of images in the system, 100 images. That is why the external outputs and inputs are the same. Due to this the total degree of influence is anywhere from a minimum of 1 image to 100, as user password. The TDI is the max number of images the user can choose as their password.

Compute Value Adjustment Factor(VAF): VAF = (TDI * 0.01) + 0.65 = 100*0.01+0.65=1.65
Find the Function Point Count: FPC = UFP * VAF= 100 * 1.65= 176 Function points Count

Activity Diagram:

Since the Duration of the project is 3.12 months as listed above, the total number of weeks spent on this project is 3.12 * 4 weeks = 12.48. The start will be from week 0 and end at week 12.

The activities are determined by the project objective as mentioned in the earlier deliverable:



Risk Analysis:

These are 4 different types of risks that may be identified during the project.

Risk 1: Employees are not motivated to work/slacking off.
Countermeasure: Hire with a strict criteria.
Risk 2: The amount of time needed to complete each task is underestimated.
Countermeasure: Set multiple deadlines for each activity to check progress.
Risk 3: Requirements change during the development phase.
Countermeasure: Hold stakeholder meetings during the development phase to re-outline what requirements are needed.
Risk 4: Workload is too heavy for employees.
Countermeasure: Allocate resources properly to make sure each activity will have enough employees.

Risk Analysis and Prioritization:

	Likelihood	Impact	Risk
Risk 1	7	7	49
Risk 2	5	8	40
Risk 3	5	5	25
Risk 4	3	5	15