EE4001/IM2001 Tutorial 8 (Web Engineering)

1. Explain briefly in your words: what is Software Engineering? what is Web Engineering?

2. Describe the common Content Architectures of WebApps. Highlight the differences between Content Architecture and WebApp architecture.

3. Describe characteristics of WebApps that must be taken into account by any Web Engineering Process Model.

4. List five important quality dimensions applicable for WebApps, and briefly explain three most important ones to end-users of an online banking system.

5. What are the roles in a Web Engineering Team?

6. What are the four main activities in creating a complete analysis model of WebApps?

EE4001/IM2001 Tutorial 9 (Web Engineering)

1. Discuss the various possible organizational paradigms for software engineering teams.

2. What are the design goals of WebApps?

3. What are the six key elements in creating a complete design model of WebApps?

4. Describe the interface testing strategy for WebApps.

5. What is Stress Testing for a WebApp project? Highlight the goal(s) of the stress testing in a WebApp project.

6. What is the focus of the Compatibility Testing in WebApp, and why it is important for WebApps?

EE4001/ IM2001 Tutorial 10 (Software Project Management)

1. What are metrics? Give a reason why you understand metrics to be important.

2. Describe the primary goals for using metrics in the context of web engineering.

3. What are the four P's in Software Engineering? Briefly describe each.

4. What is the W^5HH principle?

5. Assume that you are the project manager for a company that builds software for household robots. For the purposes, we do a simple functional decomposition. The software system is now composed of 5 subsystems as below:

   user interaction (LOC = 2400)
   sensor monitoring (LOC=1100)
   message display (LOC=850)
   system configuration (LOC=1200)
   system control [activation/deactivation] (LOC=900)

The LOC estimates for each function are noted in parentheses. Assuming that your organization produces 600 LOC/pm with a burdened labor rate of $8000 per person-month, estimate the effort and cost required to build the software using the LOC-base estimation technique described.

6. Application composition usually involves software reuse, and some of the total number of application points in the system may be implemented with reusable components. Consequently, project manager has to adjust the estimate based on the total number of application points to take into account the percentage of reuse expected.

Using the following formula and productivity table, estimate the person-months needed if the project has a highly experienced team and the expected level of reuse code is about 60%. The number of estimated application points is 250. The formula PM = (NAP * (1 - %reuse)) / PROD, where NAP is the total number of application points in the delivered system.

Developer's experience	Very low	Low	Nominal	High	Very high
PROD (NOP/month)	4	7	13	25	50

EE4001/IM2001 Tutorial 11 (Software Project Management)

1. Given the effort-time graph and figures in Figure1,
   i) List two possible reasons on why the cost increases non-linearly when try to accelerate the project from t_d to T_{min}.
   ii) State the cost corresponding to the nominal schedule.
   iii) Briefly explain whether it is possible to reduce the delivery time to six months, when the budget is sufficient to cover the cost.
   iv) How long it takes to develop the project if we aim for the lowest effort cost?
   v) What will happen if we allow the team to deliver the project in 30 months?

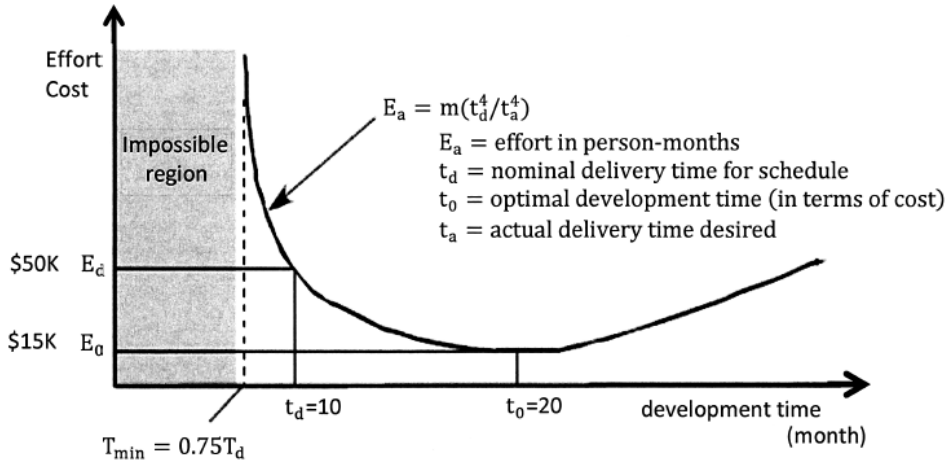


Figure 1. Effort vs. Development Time

2. List the "best practices" that should be applied to build high quality WebApps.

3. Compared to Gantt Chart, highlight the major advantages of using Precedence Activity N-etwork for scheduling a software project.

4. For code that is automatically generated, the reuse model estimates the number of person months required to integrate the code:

PM = (ASLOC x AT) / ATPROD

Where AT is the percentage of adapted code that is automatically generated and ATPROD is the productivity of engineers in integrating such code. Boehm et al. have measured ATPROD to be about 2,400 source statements per month. *ASLOC* denotes the Adapted Source Lines of Code.
If there are a total of 20,000 lines of white-box reused code in a system and 30% of this is automatically generated, compute the effort required to integrate this generated code.

5. Discuss the problems associated with estimation using use-cases.

6. A software project manager is faced with the following options to make/buy a piece of software. Draw a decision tree to support the decision process and make your recommendation when the following criteria are taken into consideration: 1) training team members; 2) cost.
Note: The percentage chance of the various possibilities and the cost outlay is indicated in the following table.

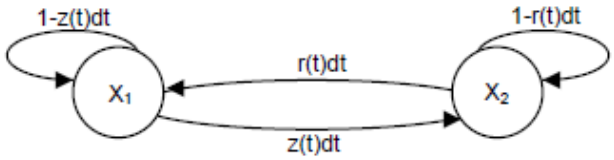
Option 1: To Build		Option 2: Reuse			Option 3: To Buy		Option 4: To Outsource	
Simple	Difficult	Minor changes	Major changes		Minor changes	Major changes	Without changes	With changes
30%	70%	40%	60%		70%	30%	60%	40%
$380,000	$450,000	$275,000	Simple	Difficult	$210,000	$400,000	$350,000	$500,000
			20%	80%				
			$310,000	$490,000				

EE4001/IM2001 Tutorial 12 (Software Project Management)

1. The following Table gives the activities details identified in a software project. (i) Create a precedence activity network using the Table. (ii) Calculate the earliest and latest start and end dates, and the float associated with each activity. (iii) Identify the critical path.

Activity	Depends on	Duration (days)
A		5
B	A	7
C	B	6
D	A	5
E	D	10
F	B	15
G	B	8
H	G	8
I	C	4
J	G	4
K	E,F	5
L	I,H	3

2. The following figure shows the state transition diagram of a dual identical-components system(X1 and X2) for air traffic control. One of the systems runs as front-end and the other is redundant. In times of failure of the front-end system, the redundant system will have to take over the control. Due to reliability requirement, a maximum of five seconds is all that is needed for the switchover. Given the mean-time-to-failure of either system is 8816 hours. Compute the availability of such a system. State your assumption.



r(t) = Repair hazard
z(t) = Failure hazard

3. What is risk analysis and management in the context of software engineering? Describe key elements of the risk management.

4. Discuss the various options for using Reusable Software Resources, and briefly explain the risk level for each of the options.

5. Suppose the information below shows the financial status of a 20 weeks project at the end of the 12th week: Earned Value = S$25, 000 and Actual Cost = S$ 30,000 The budget at completion is S$50,000. Compute (i) the Planned Value as of the reporting date; (ii) the Cost Performance Index (CPI).

6. Describe a few risk reduction techniques for software project.