

21st CSEC – Past Year Paper Solution (2018 – 2019 Semester 1)
CE/CZ 3002 – Advance Software Engineering

1)

- a. The mistake is “Friction between developers and customers”. The primary effect of this friction is poor communication, and the secondary effects of poor communication include poorly understood requirements, poor user-interface design, and, in the worst case, customers' refusing to accept the completed product. On average, friction between customers and software developers is so severe that both parties consider cancelling the project (Jones 1994). Such friction is time-consuming to overcome, and it distracts both customers and developers from the real work of the project.
- b. Quality assessment includes product assessment and process assessment. In quality assessment process, we define the quality assessment mechanism (e.g. review every month), choose the right metrics, collect data during the project execution, analyze the data and conclude assessment results. For each characteristic of the product, we assess the quality at each stage of software development lifecycle using the metrics defined. For example, the number of changes can be used as metric to ensure functionality of the product for requirement specification stage. During the project execution, we keep track of the number of changes and review it at the end of the month. If there are too many changes, then the team may want to review the requirements or consult with the customers again to solidify the requirements and evaluate the flaw in the process (eg. the requirements are not defined properly) for improvement. Hence, quality assessment process is needed to ensure that the quality of the product and procedures.
- c. The factors to be considered are organisation and people, client, product, and technology. Organisation and people factor considers the skill, maturity, and experiences of the people and the company. Client factor refers to the requirements and expectations of the client. Product factor determines the size and complexity of the software product. Technology factors depends on the type of technology used in the development eg. AR/VR. Generally, a web based system does not require new technology and detailed documentation. Hence, I would select Agile approach such as Scrum lifecycle to speed up the development process and remain flexible when requirement changes.
- d. We can consider buying in components and the use of a program generator to reduce the development time. These are avoidance strategies to reduce the risk of underestimated development time. In addition, we should monitor the progress occasionally using risk indicators, such as failure to meet agreed schedule, and decide on appropriate actions to keep the project on track.
- e. The 3 characteristics are Customer involvement, Incremental delivery, and Embrace change.

Characteristic	Description
Customer involvement	Customers should be closely involved throughout the development process. Their role is to provide and prioritise new system requirements and to evaluate the iterations of the system.
Incremental delivery	The software is developed in increments with the customer specifying the requirements to be included in each increment.
Embrace change	Expect the system requirements to change and so design the system to accommodate these changes.

2)

21st CSEC – Past Year Paper Solution (2018 – 2019 Semester 1)
CE/CZ 3002 – Advance Software Engineering

a.

- i. The 5 function points characteristics are inputs, outputs, inquiries, logical files, and interfaces.
 1. Login (stored locally) – 1 inquiry
 2. Register, payment page – 2 input, 1 internal file
 3. Verify member through external DataCentre system, Accounting Services system – 2 interfaces, 2 inquiries
 4. Submit claim – 1 input, 1 internal file
 5. Request to generate Member, Provider, Accounts Payable reports – 3 output

We assume that 3, 4, and 5 have medium complexity while the rest have low complexity

ii.

Characteristics	Low Complexity	Medium Complexity	High Complexity
# Inputs	$2 \times 3 = 6$	$1 \times 4 = 4$	$0 \times 6 = 0$
# Outputs	$0 \times 4 = 0$	$3 \times 5 = 15$	$0 \times 7 = 0$
# Inquiries	$1 \times 3 = 3$	$2 \times 4 = 8$	$0 \times 6 = 0$
# Internal Files	$1 \times 7 = 7$	$1 \times 10 = 10$	$0 \times 15 = 0$
# External Interfaces	$0 \times 5 = 0$	$2 \times 7 = 14$	$0 \times 10 = 0$
TOTAL:	16	51	0

Total Unadjusted Points = 67

- iii. Little distributed processing: 1

Little on-line data entry: 1

Average data communications: 3

High operational ease: 4

Low on-line update: 2

Low processing complexity: 2

Total score = $1 + 1 + 3 + 4 + 2 + 2 = 13$

Influence Multiplier = (Total score) * 0.01 + 0.65

= $13 * 0.01 + 0.65$

= 0.78

Total Adjusted FPs

= Unadjusted FP * Influence multiplier

= $67 * 0.78$

= **52.26**

b.

- i. Since the system has high complexity and the development team has a mix level of experience on such system development, the software project mode is **Semi-detached**.

1 FP = 53 lines of code

lines of code = $53 * 345 = 18285$

$$\begin{aligned}
 E &= a * EAF * (KLOC)^b \\
 &= 3.0 * (0.71 * 1.40 * 1.21) * (18.285)^{1.12} \\
 &= 93.50
 \end{aligned}$$

$$D = c * E^d$$

21st CSEC – Past Year Paper Solution (2018 – 2019 Semester 1)
CE/CZ 3002 – Advance Software Engineering

$$= 2.5 * (93.50)^{0.35}$$

$$= 12.23$$

$$\begin{aligned}\text{Team size} &= E / D \\ &= 93.50 / 12.23 \\ &= 7.63 \\ &\approx 8\end{aligned}$$

- ii. Compressed schedule effort = initial effort / compression factor
= 93.50 / 70%
= 133.57 PM

Initial duration = 12.23 months, Compressed duration = 0.7 * 12.23 = 8.56 months

Hence, new team size = 133.57 / 8.56 = 15.60 = 16 people.

The compression duration is too short to have a try on the project development because research suggests no less than 75% compression factor is possible.

3)

a.

- i. Capability Maturity Model Integration (CMMI)
- ii. There are 5 maturity level in CMMI — Initial, Managed, Defined, Quantitatively Managed, and Optimizing. The level of each organization depends on how well are the processes defined in the organization. At Initial level, processes are unpredictable, poorly controlled and reactive. At Managed level, processes are characterized for projects and is often reactive. At Defined level, processes area characterized for organization and is proactive. At Quantitatively Managed level, processes are measured and controlled. At Optimizing level, organization focus on process improvement.
- iii. To establish an appropriate maturity, the company must define all the key process areas required. Each key process area consists of goals the company wants to achieve. Each goal is organized by common features that address the implementation and contains key practices that describes activities to achieve it. By doing so, we will be able to make the processes more well-defined and reach the desired maturity level.

b.

- i. Version control helps to keep track of changes made and allow developers to restore to previous versions and add more functionalities conveniently.
- ii. SVN. SVN is a free centralized version control system and it is under the Apache License.
- iii. Function 1: Version control systems facilitates developers in tracking the modification history of the project.
Function 2: Version control systems allows multiple developers to work together on the project seamlessly.
- iv. While both CVS and SVN allow developers to track the modification of the project, CVS only tracks modification on a file-by-file basis while SVN tracks the whole commit as a new revision. This makes it easier to follow the history of the project
In addition, SVN allows atomic commit while CVS does not. In CVS, it is possible for the commit to be successful on several files while fail on some other files. This leaves the system in an unfortunate state because part of the commit is missing. As a result, the developer will need to fix the commit as soon as possible before other developers update the project to the broken version. On the other hand, this will not happen in SVN because SVN will either commit everything or the whole commit fails. This makes SVN a better choice compared to CVS.

21st CSEC – Past Year Paper Solution (2018 – 2019 Semester 1)
CE/CZ 3002 – Advance Software Engineering

Sources: <https://softwareengineering.stackexchange.com/questions/163593/what-does-it-mean-by-atomic-commit-for-a-versioning-system>
<https://stackoverflow.com/questions/1261/what-are-the-advantages-of-using-svn-over-cvs>

4)

a.

- i. Assuming that the system is valuable to the company but the quality is low, I agree with the decision of migrating all old systems to a cloud platform.
- ii. Two factors of software-intensive systems to consider is are business value and quality of the system. The business value refers to how important the system is to the company. The quality of the system refers to the current state of the system (low/ high quality). These two factors will determine how we should maintain the system.
- iii. There are 4 options in maintaining a legacy system – Scraping the system completely and modify business processes so that it is no longer required, continue maintain the system, transform the system with re-engineering to improve its maintainability, and replacing the system with a newer system. For a low business value and low quality system, we should scrap the system completely and modify business processes so that it is no longer required. For a low business value and high quality system, we can either scrap the system or continue maintaining the system. For a high business value and low quality system, we should either transform the system with re-engineering to improve its maintainability or replace the system with a newer system. For a high business value and high quality system, we should just continue maintaining the system.

b.

- i. Assuming that a game system is created which contains multiple integrated subsystems including login and user progress. This regression testing will aim to test if the function verifyAccount in the system will allow user to login if the user enter correct username and password. The system contains multiple databases including account database and progress database. The username and password should be searched in account database, not progress database. In the account database, we have created an account, with username “James” and password “james1234”, which should lead to successful login. Initially the developer has implemented the verification function correctly and the whole system works fine. However, the developer then unintentionally changed account database to progress database, introducing a bug to the system as shown in the pseudo code below.

Correct version:

```
function verifyAccount(username, password)
    if match of username and password are found in the account database
        then Allow user to login
    otherwise, unsuccessful user login
```

Wrong version:

```
function verifyAccount(username, password)
    if match of username and password are found in the progress database
        then return true //Allow user to login
    otherwise, return false //unsuccessful user login
```

21st CSEC – Past Year Paper Solution (2018 – 2019 Semester 1)
CE/CZ 3002 – Advance Software Engineering

- ii.

```
Boolean sl = verifyAccount("James", "james1234");  
if (sl)  
    System.out.println("Pass test.");  
else  
    System.out.println("Fail test case, test output " + sl + " instead of true");
```
- iii. Initially, since the developer used account database to verify the login, the username "James" with password "james1234" can be found in the account database and it will return true which is correct. However, later on when the developer modified the login to use progress database, the function verifyAccount will return false because there is not such username and password in the progress database. By rerunning regression testing, the test case in part ii will show that it fail and the developer will be able to detect the problem that he had used the wrong database.

Solver: Chulpaibul Jiraporn (chul0004@e.ntu.edu.sg)