University of Asia Pacific

Department of CSE

Mid-Semester Examination, Spring 2020

Name: Rashik Rahman

Reg ID: 17201012

Year: 3rd

Semester: 2nd

Course Code: CSE 321

Course Title: Software Engineering

Date: 28.08.2020

Answer to the Q. No. 1 (a)

Some of the key challenges a software engineer might face in 21st century are given below with their solutions.

i) The legacy challenge:

The legacy challenge is using an old system in this era and maintaining it. It is basically maintaining and updating the software in such a way that it doesn't become old and excessive costs are avoided. This is one of the biggest challenges of a software engineer.

ii) Competence:

Now a days companies have become more competetive. That's why ~~they~~ they hire skilled on potential employees. To get a job ~~so~~ on to get in a project team some engineers tends to misrepresent their skills. Thus creating a fake impression on the company's side. So if he/she gets into the team he/she may harm to project as he ~~isn't sure~~ doesn't have sufficient skills. This also hampers the workflow of other software engineers and have become a challenge for them to ~~it~~ handle.

17201012

iii) The delivery challenge:

The delivery challenge is the challenge of shortening delivery times for large and complex systems without compromising system quality.

To overcome these challenges a software engineer must know how to handle ~~leg~~ and modify legacy systems and should have a profound knowledge of it. Engineers shouldn't misrepresent their skills, they should only take those jobs that they know they can complete. They ~~shoot~~ should learn how to work fast maintaining quality and also companies shouldn't give unrealistic deadlines.

17201012

Answer to the Q. No. 1(b)

The ethics for developing such model is discussed below:

i) Confidentiality:

As this system has privacy implications so a software engineer must maintain the confidentiality of ~~both~~ client ~~and~~, system trade secrets, backend code, ~~&~~ project development process (code part), database, and software. Info about the client and the system secrets shouldn't be leaked as it is a very sensitive issue and a privacy concern.

ii) Competence:

This software is a large project and also involves privacy mattens. So a wrong mistake on the ~~der~~ developer's side ~~we~~ may cause ~~mess~~ much damage to the company or citizen's private life. So a software engineer shouldn't misrepresent his/her skill to join this project cause if he/she does join that may lead to a harmful outcome.

(4)

17201012

iii) Intelectual rights:

Though it's kind of a spying software still ~~a~~ engineer should maintain a limit. Like this can track the activity of the citizen but shouldn't look into their gallery or ~~of~~ shouldn't violate very private spaces. The app shouldn't also violate governing laws.

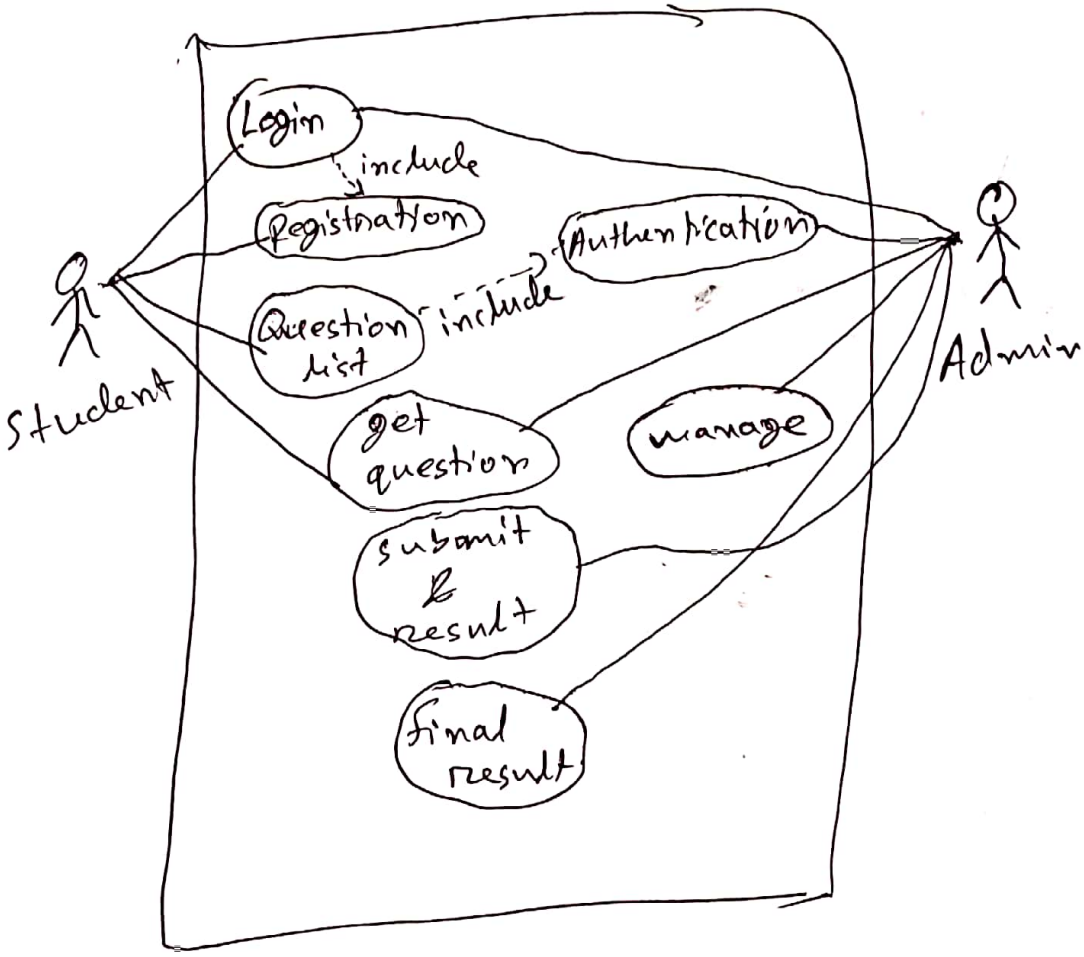
iv) Device misuse:

Engineers should only develop this for ethical purpose. They shouldn't keep a backdoor in this software by which info ~~about~~ and personal stuff can be stolen. So they shouldn't ensure not to misuse ~~their~~ citizen's device or computer.

17201012

(05)
17201012

Answer to the Q. No. 2(a)

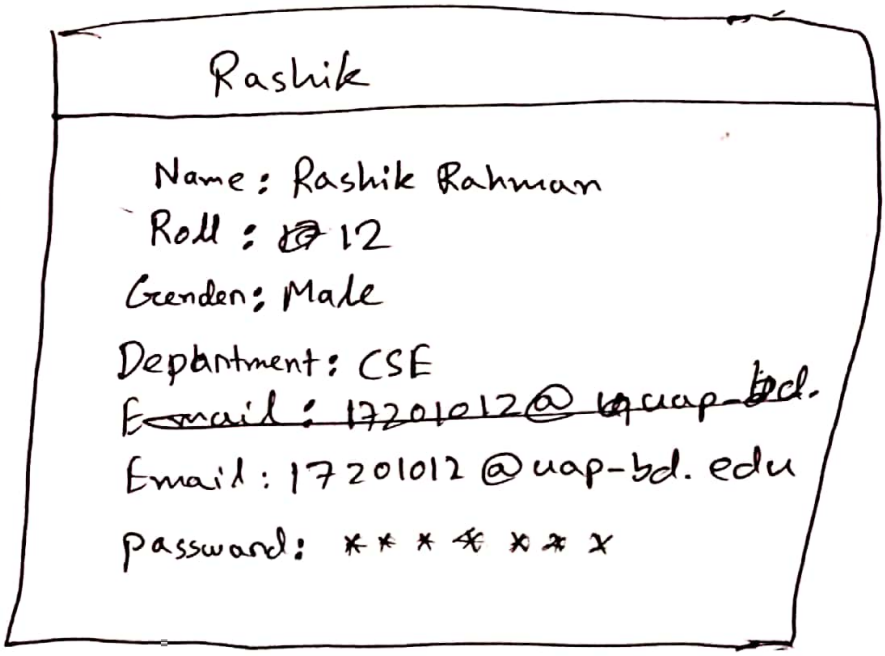


Online exam system of UAP

(6)

17201012

Answer to the Q No. 2(b)



Answer to the Q. No. 3(a)

I would use ~~agile~~ waterfall modeling to develop a ~~vr~~ virtual reality system to support software maintenance. My reasoning given below.

We are to make a support software VR system. ~~Some of it's requirements~~ As we'll develop this for maintenance ~~its~~ so its requirements are very clean. It isn't a big scale project and we would know all the features that are to be added to this. Thus requirement are fixed. VR development now a days has a stable environment. So to develop this I can take a linear approach. This ~~is~~ system doesn't need customer feedback as it will be used by only the selective authenticated maintanence person ~~not~~, it won't be used for the masses. Thus developing this delicate system using waterfall model would be a great approach.

As this is a delicate project so time should be taken in each phase and it

should go through full system testing before deployment to maintain quality.

All the terms to develop this project prefers a linear model like waterfall model. That's why I choose waterfall model.

Answer to the Q. NO. 3(b)

An industry based example of Prototyping Model is Podda Shetu project. My explanation given below.

We use prototyping model when requirement isn't clear or environment unstable and project is large scale. Also there are many other factors. Before Podda-Shetu went into making CSE, CE, EEE engineers assembled and built a prototype of it and simulate all the scenarios the shetu can face as it's a big project and error won't be tolerated.

The engineers didn't knew exact requirement like how the soil, water, & wind would effect the Sheta and will the be any bio-hazard on will the Sheta ~~will~~ face corrosion. Will the Sheta ~~be~~ be able to face natural desasten and calamity.

So they built a prototype of it to gather this kind of info and requirements. ~~And to develop a prototype~~ and to simulate real life scenario. They tried to develop the prototype as realistic as possible. The Sheta is being built based on this prototype model.

So we can conclude that it was the system process model to develop Podda Sheta. This model is used in this kind of scenarios.

—— o ——