

Name: Rashik Rahman

ID : 17201012

Sec : A

(1)

#1. What is the difference between a software process and a software process model? Suggest two ways in which a software process model might be helpful in identifying possible process improvements.

Ans:

A software process is a sequence of activities that leads to the production of a software product. These activities are software specification, development, validation & evaluation. System process is also known as SDLC. SDLC gives us concept of how to develop a software product, it is a life cycle of software development. But it doesn't describe how to actually develop a product in real life and doesn't give detailed information, it's just a concept. This is the key difference between software process and software process model. Software process model is the development strategies we use to solve real life problem in making a software. These models are based on SDLC. Ex: Waterfall, RAD, Spiral etc.

Software process model is real life implementation of software process so, it is better in every way. Such two ways are given in next page.

2

In software process models like Agile, ~~Incremental~~ we use certain methods to evaluate the products and process. In these models evaluation can happen at each process (iteration) thus ~~improving~~ the process it can identify in which process there's a scope of improvement.

Another way is in all models we can do requirement analysis ~~this~~. The more this analysis is done well the more we get better outputs from other processes. So this is also a way to identify possible process improvements.

#2. Apart from the challenges of heterogeneity, rapid delivery and trust identify other problems and challenges a software engineer is likely to face in 21st century.

Ans.

Some of the key challenges a software ~~engineers~~ might face in 21st century are given and explained in the ~~next~~ ~~next~~ page.

### a. The legacy challenge:

The legacy challenge is the challenge of maintaining and updating the software in such a way it doesn't become old and excessive costs are avoided to deliver continuous business services.

### b. Competence:

Engineers should not misrepresent their level of competence. They should not knowingly accept work which is outwitting their competence.

### c. The delivery challenge:

The delivery challenge is a challenge of shortening delivery times for large and complex systems without compromising system quality. Hence supporting software must change equally rapidly.

### d. Intellectual property rights:

Engineers should be aware of local laws governing the use of intellectual property such as patents, copyright etc. They should insure to protect intellectual property right and shouldn't violate them. In this vastly growing this is quite challenging to protect these rights.

(7)

#3. Ans:

The ethics for developing such system are given below.

#### a. Confidentiality:

As this software has privacy implications so a software engineer must maintain the confidentiality of both citizens and software, database system. Tracking a citizens and their actions are very private thing if leaked then the consequences will be very upsetting that's why confidentiality needs to be maintained.

#### b) Competence:

This software is a large scale project and will need skilled employees as the main goal is to track citizens and their activities. So as a software engineer you shouldn't misrepresent your level of skill or competence. Cause doing so you may harm the company on the project as the work given to you may be out of your league. So you should only sign up

for work that you know you can finish.

### C. Intellectual property right:

Every person has their own personal right. Though you are developing a software to keep track of citizen still there is a limit to it. You shouldn't develop the software that violates their rights. Like the software only should keep track of their activities and not use look into their gallery or such things. Even you are to keep track of their activities you should stone on use them unless they do something suspicious as this might violate their rights.

### d. Concept Device misuse:

You are to only develop the system to keep track of the citizens. The software shouldn't misuse their & the data it collects and as a software engineer your work is to ensure that. If this system is deployed in OS platform you need to make sure their devices shouldn't be misused.

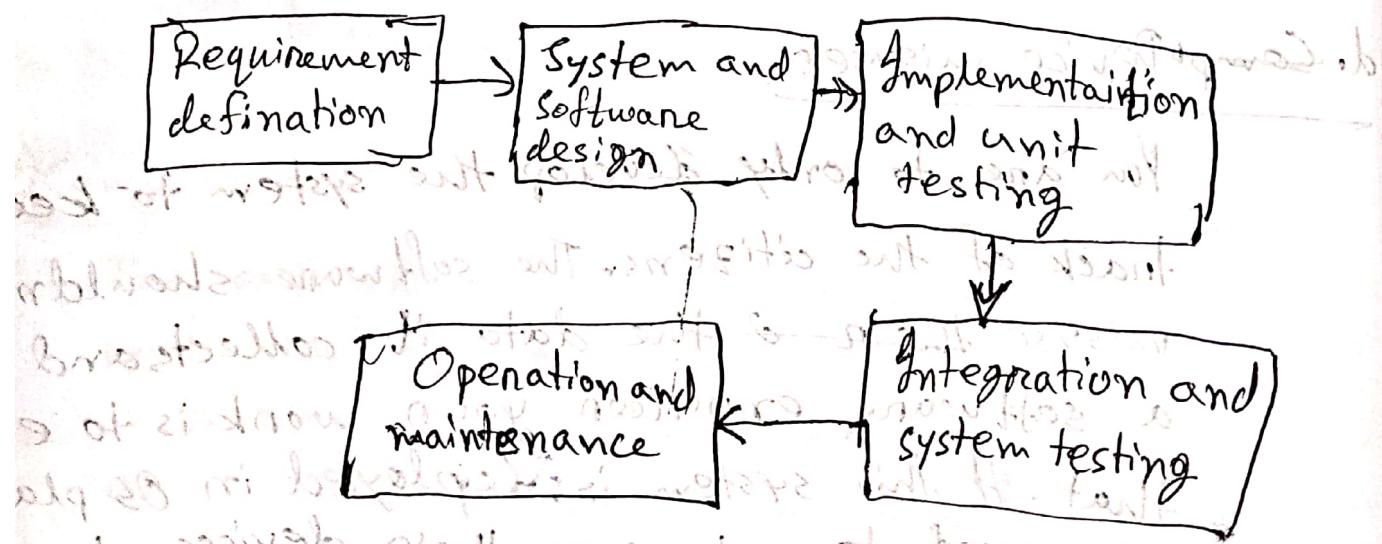
6

#4. Ans <sup>so</sup> very word very half show not

From the question we can see that we need to develop a home security system.

A home security system is a very sensitive aspect of every household. This kind of system can be developed following many models. But the most optimal model would be a linear model like waterfall model.

As it's a ~~less~~ sensitive aspect, so we don't have to develop it rapidly or very iteratively so the according to me the best way is to take a linear approach. Block diagram given below.

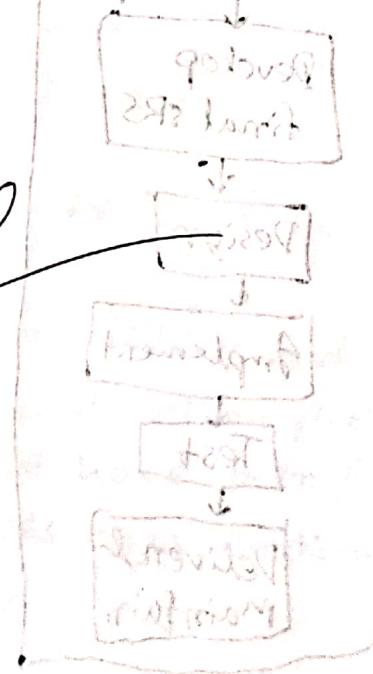


~~#. S. Ans~~

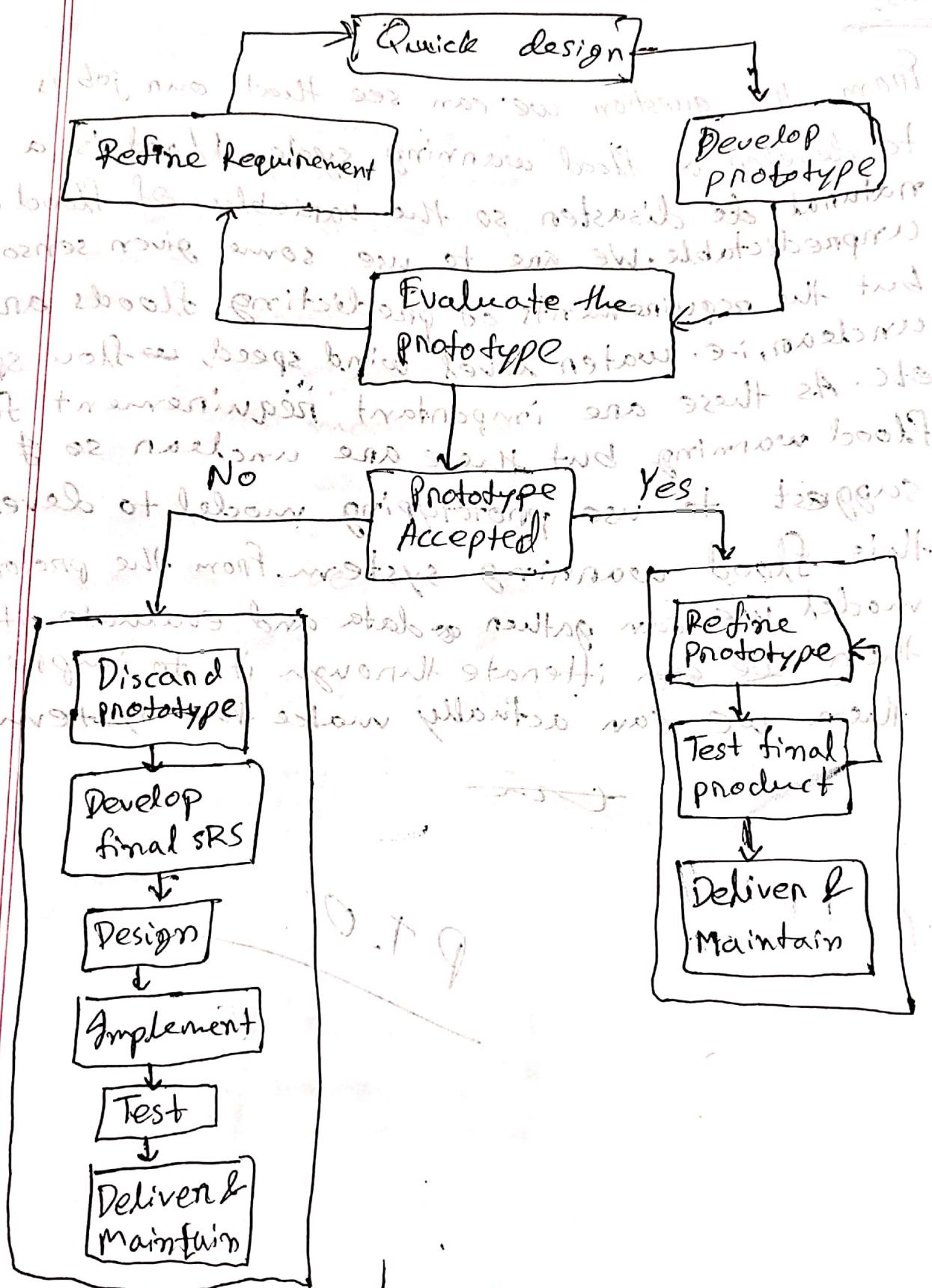
### Floods warning

From the question we can see that our job is to develop a flood warning system. Flood is a natural disaster so the variables of flood are unpredictable. We are to use some given sensors but the requirements of predicting floods are unclear, i.e. water level, wind speed, flow speed etc. As these are important requirement for flood warning but these are unclear so I suggest to use prototyping model to develop this flood warning system. From the prototype model we can gather data and evaluate, if then we can iterate through it to improve it then we can actually make the system.

P.T.O



8



## #6. Ans.

~~Ques. What are legacy systems?~~

~~Ans. Legacy systems are old systems.~~

When a system is based on old data or method or technology then the system is called legacy system. It is basically an outdated system. This type of system can cause difficulties, and these difficulties are explained below.

Despite of being expensive, some companies still use it as such. Systems have become fundamental to the way the organization functions. But with the increase of usage time, such system may cause loss to the company. Causes are:

- i) Legacy system often represent a huge risk when it comes to security, cause they are old according to current security standards.
- ii) Legacy system can often be inefficient and unstable.
- iii) Many legacy systems are standalone and were never designed to integrate with other piece of software. While that might have been fine once but now it drastically reduces organizations flexibility and scalability.

(10)

- iv) When hiring new employee and introducing them with legacy system this having the organizations' ~~re~~ perception's ~~perception~~ perception in employees mind. ~~so staff working beliefs are addressed.~~
- v) These ~~eg~~ types of system ~~eg~~ are prone to failure.
- vi) Modern software is unparalleled when it comes to gaining deep data insights and supporting important business decisions. But legacy software wasn't built for deriving data insights causing data loss of potential profit of the company.

#7. Ans.

~~Given sc~~

In the given scenario my action is explained below.

Being a system engineer it's my job to design the system and use ~~no~~ a model to develop it. If there's no team to do it. So I don't know coding part me to modify this system in the given scenario. Professionally it is my responsibility to see through the matter that I deliver a perfect working system. So I would stop the installation, apologize to the client and send the system for re-evaluation and re-development. Cause I don't have proper access but the ~~dev~~ development team have that as they developed it. If I wait until things go bad in the client end that would be bad for my company. ~~That's~~

That's why I would take these necessary steps mentioned above.

#8 Ans:

- a) ISP system: The most critical dependability attribute for this system is security. Cause this ISP system connects thousands of customers and provides internet. If the system isn't secure then hacker can hack into the system and may steal customers' personal data or harm them.
- b) Computer controlled scalpel: The most critical dependability attribute is reliability. Though safety is also critical but the most critical is reliability. Cause if the system fails the scalpel will stop operating but if it is not reliable it might cut other parts of the body. So it should be reliable to deliver services as specified.
- c) Directional control system: The most critical dependability attribute for this system is reliability. Cause if the system is reliable then it can deliver services as specified.

(8)

d) Internet based

d) finance management system:

The most critical dependency

attribute for this system is

security, cause if the system

isn't secure customer's financial

financial credential can be

leaked and this may harm them

financial state because it'll be

easy to transfer money from

to their account.

It'll be easier for the bad guys

and black-hat hackers to steal info thus

as consequences of attack of the day

will be huge loss for the business

Business team will waste time dealing with

it and not suitable to find a proper

solution of waste of time). Solution

to solve needs of not fit

with the business

so we have to make a proper solution

# 9. Ans.

Reliability and safety are related but distinct dependability attribute. Reliability is concerned with delivering specified services and safety is concerned with ~~safety~~ the ability of the system to operate ~~without~~ without catastrophic failure. A system can be reliable and unsafe or safe but unreliable. We can take the example of computer operated scalpel. It can be reliable but unsafe. Cause it may deliver precise cutting as specified but may shutdown in the middle of operation.

On the other hand if can be safe but unreliable. The system may not fail catastrophically but it may be unreliable cause it may not cut with precision. On cuts somewhere else where it shouldn't happen. Straying in precision. Wrong targets are sent as mistake instead of waste have effects that are few waste have soft grid and egg along a ribbon spirals down.

15

### #10. Ans

No it's not ethical for an engineer to agree to deliver a software product with known faults to customers. Yes it makes differences if the customer was not told of the known existence of these faults in advance. Still it won't be reasonable to claim the system isn't reliable cause you know it isn't.

#11. Ans

a) Anti-lock system: It has benefits with no disadvantages. But for your prototype soft will use prototype model to develop this system. Cause the requirements are not clear in such system as there are different factors that effects such system. So before making the real system we can make, evaluate, modify a prototype

to get the most optimal system for all tenains and then make the ~~not~~ real life system.

b) virtual reality systems: Generally VR systems

requirements are clearly specified. And we can use linear model like waterfall to develop this model. This system is not large waterfall model easily.

c) University accounting system: We use agile model

to develop this system, cause agile model is combination of iterative and incremental model. There may be many features in such system so with each iteration we can improve certain feature and with each increment we can add new features.

Submitted with care

d) Train schedule system: This is not very complicated ~~or~~ system and doesn't have many feature and have requirement specified clearly. So I'll use waterfall model to develop this system.

#12. Ans:

Evolutionary model is also called prototyping model. Software developed with this model are difficult to maintain cause of the following reasons:

- i) Practically this methodology may increase the complexity and it will make it more difficult to maintain.
- ii) There are no specific requirements on requirements are unclear in such model. We gather requirement via try & error. So it can be bias to testing methods. Thus maintaining such system is very difficult.

#13. Ans:

The waterfall model is accommodated when there is a low specification risk and no need for prototyping after risk resolution. The activities in the 2nd quadrant of the spiral model are skipped. The prototyping model is accommodated when the specification phase is limited and the risk resolution phase predominates. The activities in the 3rd quadrant of the spiral model are skipped or reduced in scope.

#14. Ans:

A classification scheme can be helpful for procurement because it helps identifying gaps in the CASE tools coverage in an organization. Procurement may be aimed at filling these gaps.

(19)

Q1

#15.1, Ans<sub>e</sub>

For a ~~fresh~~ student using agile model for the final project and will be helpful for the following reasons:

- i) They can get used to frequent delivery and thus time management.
- ii) It will improve their face to face communication skills with client.
- iii) It will help them to do efficient design and will teach them to ~~fully~~ fulfil the business requirements.
- iv) They will be introduced to frequent changes and this will learn to accept anytime changes.
- v) They'll be able to learn & ~~reduce~~ how to reduce total development time.

#15.2, Ans:-

Agile methods may not work well in organizations that have teams with a wide range of skills and diff abilities and well-established process for the following reasons.

- i) Lack of clarity: Large enterprises often include big teams that may consist of remote members. Under agile ~~strategies~~ strategies it's important to have everybody involved in the project collaborate, so if remote employee aren't able to easily communicate, it will be difficult to contribute.
- ii) Inadequate experience with agile: Possibly the biggest reason why agile project fails in large enterprises is the fact that people just don't have experience with methodology or how to integrate it.
- iii) Lack of collaboration: In this context sometimes we can see in large companies that a team isn't really a team or they lack same goals so they can't collaborate with each other. But collaboration is very important in agile modeling. Thus if we use agile in large organization, it may not benefit the organization.

#### iv) Lack of testing strategy:

The role of the tester changes and so does the required skill to fulfil this ~~roll~~ role. As the role and skill changes frequently it results in lack of testing strategy.