1. Question 1

With the help of example, differentiate between functional and non-functional requirements of a software project.

Your Answer

Functional Requirements:

0. A function requirement defines a system or its component.
1. It specifies what should the software system do.
2. A functional requirement is specified by the user.
3. It is mandatory.
4. It is captured in the use case.
5. Defined at a component level.
6. Helps you verify the functionality of the software.
7. Functional Testing like system, integration, end to end, API testing, etc is done.

Example:

8. Authentication of the user whenever he/she logs in to the system.
9. System shutdown in case of a cyber attack.
10. A verification email is sent to the user whenever he/she registers for the first time on some software system.

Non-functional Requirements:

11. A non-functional requirement defines the quality attribute of a software system.
12. It places constraints on how should the software system fulfill the functional requirements.
13. A non-functional requirement is specified by technical people e.g. Architects, Technical Leaders, and software developers.
14. It is not mandatory.
15. it is captured as a quality attribute.
16. Applied to a system as a whole.
17. Helps you to verify the performance of the software.
18. Non-Functional Testing like Performance, Stress, Usability, Security testing, etc is done.
19. Usually more difficult to define.

Example:

20. Emails should be sent with a latency of no greater than 12 hours from such an activity.
21. The processing of each request should be done within 10 seconds
22. The site should load in 3 seconds when the number of simultaneous users are > 10000

2. Question 2

Spiral model is a realistic approach to the development of large-scale systems & software. Justify & explain the model.

Your Answer

There are several advantages of the spiral model that makes it a realistic approach to the development of large-scale systems and software like:

0. The spiral model promotes quality assurance through prototyping at each stage in system development.

1. A spiral model is a realistic approach to the development of large-scale software products because the software evolves as the process progresses. The developer and client better understand and react to risk at each evolutionary level.
2. The model uses prototyping as a risk reduction mechanism and allows for the development of prototypes at any stage of the evolutionary development.
3. It maintains a systematic stepwise approach like the classic life cycle model but incorporates it into an iterative framework that more reflects the real world. If employed correctly this model should reduce risk before they become problematic as consideration of technical risk are considered at all stages Spiral model is also known, as the spiral life cycle model is a system development life cycle model used in information technology. This model of development combines the features of the prototyping model and the waterfall model. The spiral model is used for a large, expensive, and complicated project.
4. Presented by BARRY BOHEM in 1986 incorporates the project risk factor.
5. Designed in order to overcome the disadvantages of the waterfall model.
6. The radial dimension represents the cumulative cost incurred in accomplishing the steps done so far and the angular dimension represents the progress made in completing each cycle of the spiral.
7. Each loop is a development stage. •Balance all the risk elements that are the high-risk element must be lowered.
8. The people who are concerned with the project complete each phase with a review.

Different Phases of spiral model:
1) **Planning:** In this phase, the objectives, alternatives, and constraints of the project are determined or documented. The objectives and other specifications are fixed in order to decide which strategies or approaches to follow during the project life cycle.
2) **Risk analysis:** In this phase, all possible and available alternatives which can help in developing a cost-effective project are analyzed and strategies are decided to use them. This phase has been added especially in order to identify and resolve all the possible risks in the project development.
3) **Development:** In this phase, the actual development of the project is carried out. The output of this phase is passed through all the phases iteratively in order to obtain improvements in the same.
4) **Assessment:** In this phase, the developed product is passed on to the customer in order to receive customer comments and suggestions, which can help in identifying and resolving potential problems or errors in software development.

3. Question 3

SRS is known as the black box specification of system. Discuss in detail.

Your Answer

SRS document is a contract between the development team and the customer. Once the SRS document is approved by the customer, any subsequent controversies are settled by referring to the SRS document. The SRS document is known as black-box specification. Since the system is considered as a black box whose internal details are not known and only its visible external (i.e. input/output) behavior is documented. SRS document concentrates on what needs to be done and carefully avoids the solution ("how to do") aspects. The SRS document serves as a contract between the development team and the customer. SRS should be carefully written. The requirements at the SRS stage are written using end-user terminology. If necessary later a formal requirement specification may be developed from it.

It should only specify what the system should do and refrain from stating how to do these. This means that the SRS document should specify the external behavior of the system and not discuss the implementation issues.
The SRS document should view the system to be developed as a black box, and should specify the externally visible behavior of the system. For this reason, the SRS document is also called the black-box specification of a system.