

Name : Danyal Ahmad

Id : 13838

Class: BSSE

Course Code: 102007052

Course Title: Software Engineering

Instructor: Engr. Ghassan Husnain

Summer Semester 2020

Program: BS CS (Software Engineering)

Online Final – Term Examination

Q.1: Explain why the rapid delivery and deployment of new systems is often more important to businesses than the detailed functionality of these systems. (6 marks)

Answer(1):

Now a days many businesses were willing to trade off software quality and compromise on requirements to achieve faster deployment of the software. In this way, Rapid development and delivery is the most critical

requirement for software system. Because these business are operating in a changing environment. They have to respond to new opportunities and markets, changing economic conditions and the emergence of competing products and services. So, Rapid delivery focuses on the delivery of the system. It is good for a system that is required to show the result of the system.

And, it is good for business since the system can be used early if the essential functionality is available and be later improved as the user requirements change. Rapid delivery can make profit swiftly.

Rapid software development processes are designed to produce useful software quickly. At the same time, it is not good for critical-system development since the delivery of the system needs to be perfect and without failures.

Q.2: Explain how the principles underlying agile methods lead to the accelerated development and deployment of software.

Answer(2):

The principles of agile methods has contributed to the accelerated development and deployment of software in the following ways:

1.Incremental delivery: In this process, the software is delivered in small increments to the customer. Depends on the customer feedback and requirements, the developers made the increments in the system. In every increment, the new functionalities are developed and deployed into the system.

2.Customer involvement: The customers are involved in the development process of the system. The agile methods involve regular discussions with the customer. As the development of the system is done in small increments, the customer must involve in the development process to provide the requirements of the new features in the system.

3.People, not process: One of the important principle to develop efficient software is to recognize the technical skill of the people working in the

development process of the system. It is necessary to implement and deliver the product quickly with high quality.

4.Embrace change: As technology is developing, the changes must be done to the existing system. The system must be designed as per the requirements of the changes to deploy the new features in the system.

5.Maintain simplicity: As the changes in the system done frequently in small increments, the code used in the system must be simple. If changes are required for the system, the existing code must be restructured as per the requirements of the changes. The simplicity of the code and the development process must be maintained without any complexity in the system.

Q.3: Extreme programming expresses user requirements as stories, with each story written on a card. Discuss the advantages and disadvantages of this approach to requirements description.

Answer (3):

The use of user stories is not unique to Extreme Programming user stories are widely-used in Agile

especially Scrum. They are not necessarily written down on cards for example, many times, an online Agile Project Management tool is used to store user stories online.

A user story is considered a “placeholder for conversation” - it is intended only to be a short and succinct description of the user need and it is expected that the developer(s) responsible for implementing the user story will further elaborate the details based on direct communication with the user.

Advantages and Disadvantages of Extreme Programming user requirements:

Advantages:

1. Scenarios cope with most of common operation. It is easy to identify what type of operation that is required in the user's stories.
2. Customer focus in the scenario card increase the chance that the software produced will actually meet the needs of the users

Disadvantages:

1. Using scenarios on a card can bring to a function overlooked or omission which can be a time-consuming process to complete the system.
2. Two different scenarios can lead to the same function as it will be conflicted each other. Crossing out redundant scenarios can be a cumbersome task.

Q.4: To reduce costs and the environmental impact of commuting, your company decides to close a number of offices and to provide support for staff to work from home. However, the senior management who introduced the policy are unaware that software is developed using agile methods, which rely on close team working and pair programming. Discuss the difficulties that this new policy might cause and how you might get around these problems.

Answer (4):

The difficulties that may arise with this policy if making employees work from home are:

- The benefits obtained through agile methods will be less effective.

- Communication gap between the members of a team
- The benefit of error detection and evaluation through pair programming is lost.
- Pair programming is not possible.
- Due to sudden changes in the teams, the project development may be slowed down.

The measures that can be taken to get around such difficulties are:

- Rather than completely closing some offices and asking people to work from home, employees can be moved to some offices and accommodated.
- The best way would be to have daily coordination calls between various project managers so that every phase of the project is in sync. The project manager in turn should also have daily calls with his team members to get the status on the project.
- Information regarding project should be shared and communicated.

Q.5: Identify and briefly describe four types of requirement that may be defined for a computer-based system.

Answer (5):

Types of requirements for a computer based system:
Generally, system requirements are included to communicate the functions that the system should provide. And every computer based systems consists of many requirements. They are

1. User requirements.
2. System requirements
3. Functional requirements
4. Non-functional requirements.

Description of requirements:

1. User requirements: The requirements are the statements in a natural language plus diagrams of the services the system provides and its operational constraints.

2. System requirements: A structured document setting out detailed description of the system's functions, services and operational constraints. Define what should be implemented. It may be part of a contract between client and contactor.

3. Functional requirements: These are the statement of the services the system should provide, how the system should react to particular input and how the system should behave in particular situation.

4. Non-functional Requirements: Constraints on the services or functions offered by the system such as timing constraints, constraints on the development process, standards, etc. often these are applied to the system as a whole rather than individual features or services.

Q.6: Using your knowledge of how an ATM is used, develop a set of use cases that could serve as a basis for understanding the requirements for an ATM system.

Answer(6):

Given the variety of systems used, there can't be one unified set of use cases for all ATMs, but many share common functionality. A few examples are:

1. Withdraw Funds :

Actors – user, ATM, user account/s

Input – user card, user PIN, account select, withdrawal amount

Output – user card, transaction receipt, account/s balance

Exceptions:

Card invalid – user advised to seek assistance; card retained

Invalid PIN – transaction rejected after predetermined number of attempts; card retained; user advised to seek assistance

Insufficient funds – transaction canceled; user advised to seek assistance

Use case:

User inserts card into card reader. ATM prompts user to enter PIN. User enters PIN. If PIN is correct, ATM prompts user to select transaction type. User selects withdrawal. ATM prompts user to select account for

withdrawal. User selects account. ATM prompts user to enter amount of funds for withdrawal. User enters amount for withdrawal. If sufficient funds exist for withdrawal, funds are dispensed. ATM prompt asks user if they would like a receipt. User makes selection. If yes, receipt is printed. Card is returned to user.

2. Deposit Funds:

Actors – user, ATM, user account/s

Input – user card, user PIN, account select, cash/check for deposit

Output – user card, transaction receipt, account/s balance

Exceptions:

Card invalid – user advised to seek assistance; card retained

Invalid PIN – transaction rejected after predetermined number of attempts; card retained; user advised to seek assistance

Deposit failure – transaction cancelled if user fails to

input funds for deposit within predetermined amount of time

Use Case:

User inserts card into card reader. ATM prompts user to enter PIN. User enters PIN. If PIN is correct, ATM prompts user to select transaction type. User selects deposit. ATM prompts user to select account for deposit. User selects account. ATM prompts user to select cash or check deposit. User makes selection. If cash, ATM prompts user to insert cash into cash receptacle. ATM reads bills as they enter and counts total. ATM displays amount and asks user to confirm total. If check, ATM reads scans check and reads total. ATM displays check scan and total for deposit. ATM asks user to confirm total. User can confirm total or reject total. If user rejects total, ATM asks user to enter amount for deposit and confirm. ATM displays total deposit and new account balance. ATM prompt asks user if they would like a receipt. User makes selection. If yes, receipt is printed displaying account balance. Card is returned to user.

Q.7: Suggest how an engineer responsible for drawing

up a system requirements specification might keep track of the relationships between functional and non-functional requirements.

Answer(7):

While drawing up a system requirements specification, an engineer might keep track of the functional and non-functional requirements by ensuring the following:

- The requirements needed to design meets the requirements such as compatibility, portability etc.
- Design the system so that it ensures the safety and security.
- Implementing the system in an efficient manner.
- The cost and time required for the development should not affect the design and implementation of the system

Here, the non-functional requirement defines what are the expectation to get out and the user requirements.

The functional requirement defines the use of the developer knowledge.

It does not conflict with each other.

The first step is to make the Systems Requirement Document.

It is engineer responsible to prepare documents to each functional and non-functional requirement.

- The engineer needs to prepare the document depending on this; Non-functional requirements need the natural language and functional requirements need the structured language to understand better.
- It gives the matrix that shows each requirement related to each other.
- It is very difficult to manage because the functional and non-functional requirements put efforts with each other on track of relationships.
- Non-functional requirements linked with functional requirements to list, identify the system levels that have related each other.
- The engineer needs to prepare the way to link the functional to non-functional to implement it.

- The functional requirements enforce the non-functional requirements that shall be recorded and tracked.

For example:

The user needs to search for the candidate list for the interview.

- It is a functional requirement.

That the search should return all the list of candidates who are attending the interview.

- It is a non-functional requirement.

Therefore, it helps the engineer to avoid overlap and that relates to each other.

And it keeps track the relationships between functional and non-functional requirements.

The End