Exam Week-1

Date: 28th December,2023 - 3rd January, 2024

Assignment-1

Submitted by

Apple Mahmood Oni

1. What is SDLC?

SDLC stands for Software Development Life Cycle. It is a process that is used by the software industry to design, develop and test high-quality software. It has several phases. Outcome of a phase serves as input of another phase. Every phase has its own process and deliverables. The aim of SDLC is to produce high quality software that meets the customer expectations, deliver the software within time and estimate the cost. Following SDLC process an application is developed in a systematic and disciplined manner.

2. What is Waterfall Model?

Waterfall Model is the first SDLC model that was introduced. It is also called linear-sequential life cycle model. In this approach, each phase must be complete before the next phase begin. There is no overlapping between the phases. It is simple and easy. If the project is short, requirements are known and doesn't change frequently, technology is known and stable not dynamic , then waterfall is a good choice. But in these days, Waterfall model is not used that much as the requirements of the project or product changes frequently.

3. What is Agile Model?

Agile model is a combination of iterative and incremental process. Each iteration of agile model take a time interval of 1-4 weeks. This model is aligned to deliver the changing business requirements. In real world, requirements changes frequently. So,

agile model is very popular to follow. On the other hand, in agile model the testing begins from the very beginning of the project like when the requirements are specified. It leads to early detection of bugs saving time and cost if it was detected later on the project. There are four values and twelve principles of agile model. One of the most important values of agile model is customer collaboration. Agile model has different kinds of frameworks such as Scrum, Kanban, Lean Software Development etc.

4. What is a user story?

A user story is a informal, natural language description of a feature of software system. A user story is a part of agile approach which is used to capture a description of software features from the end-user perspective. High priority user stories tend to more detailed and low priority user stories tend to less detailed.

Format: *As a [**type of user**], I want [**an action**] so that [**a benefit/a value**]*

5. What are the characteristics of a good user story?

Characteristics of a good user story is **"INVEST".**

Independent, Negotiable, Valuable, Estimable, Small and Testable.

- Independent – they can be developed in any sequence and changes to one User Story don't affect the others.
- Negotiable – it's up for the team to decide how to implement them; there is no rigidly fixed workflow.
- Valuable – each User Story delivers a detached unit of value to end users.
- Estimable – it's quite easy to guess how much time the development of a User Story will take.
- Small – it should go through the whole cycle (designing, coding, testing) during one sprint.
- Testable – there should be clear acceptance criteria to check whether a User Story is implemented appropriately.

6. Define Who, What, and Why?

Who- Customer or types of users, known as user persona.

What- This is something (feature or functionality) that the user wants to have in the application or system to be implemented.

Why- This is the reason for which the user wants that feature or functionality.

7. How to Write a User Story. Give an example?

There are **5 steps** to write a user story.

Step-1: Outline the acceptance criteria

The user story has a end-state. When the user able to complete the task or achieve the goal described that is called end-state. This end-state should be defined in the user stories so that the rest of the team knows that the development work is done.

Step-2: Outline task or subtask

The story needs to be broken down into numerous tasks to make it more manageable. If the requirement is complex, then a task can be broken into many subtasks.

Step-3: User personas

If there are multiple users then there should be multiple user stories.

Step-4: Map stories

User stories need to be mapped to structure work in a large process. In this case, stories will take the form of ordered steps.

Step-5: Seek user feedback

Speak to users and potential customers to know what problems they have or what more they want. Asking them about existing product and what new features they want. Only after gathering and analyzing feedback, user stories should be written.

Step-6: Time

User stories that take longer than a single sprint should be broken into smaller stories. This way, development team gets a sense of completion in each sprint, because they're able to complete some new functionality each time.

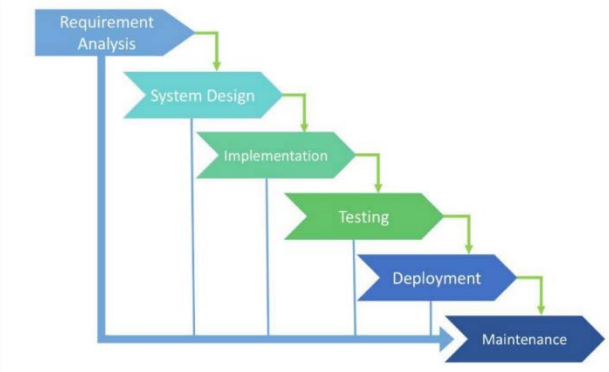
Example of a user story:

Student Education Management Application

As a student, I want to access study resources and materials, so that I can improve in my study.

As a teacher, I want to generate progress report of the students, so that I can inform their parents.

8. Write down the explanation about this diagram.



The above diagram represents the phases of the Software Development Life Cycle.

Explanation:

Requirement Analysis:

This is the first phase of the software development life cycle. In this phase, a dedicated team communicates with the stakeholders to understand their needs and document them in a SRS (Software Requirement Specification). Generally, Business

analyst, Requirement analyst works in this phase. The product owner or project manager also participate.

System Design:

In this phase, based on the SRS the system is designed by the system architect and Ui/Ux designer. There are generally two kinds of design types. High Level Design and Low Level Design. Interface relationships and dependencies of the modules, Database table identifying, Functional logic of the module etc are the work that is done in this phase.

Implementation:

After the system is designed, the development team start to build the system. There are many stack available to build a system such as MERN, MEVN, PHP/Laravel, .Net MVC Core etc. The team choose any stack according to their comfort and project need and build the system. There is front-end development team and back-end development team.

Testing:

After the development team develop the system partially, the testing team start to test the system. It is the testing team who ensures that the product is defect free and meets the user requirements. If problems found then the team member inform the developer to fix the problem. QA engineer, QC engineer or software tester work in this phase. Though QA engineer works throughout all phases, QC engineer and tester works only in this phase.

Deployment:

After the product or project is build fully and the testing team ensures the product or project is ready to release, then the product is released by the deployment team. Generally, the DevOps team deploy or release the product in operation. The feedback of project manager is mandatory to release the product.

Maintenance:

When the product is in operation and the customer is using it, there are mainly 3 kind of things that happens. Bug fixing, Upgrading to newer version and Adding new feature or functionality. Support team works in this phase to enquire customer problem

and collaborate with the other teams to solve customer problem. As long as , the product is in operation the maintenance of the product is needed.