

Md Joynel Abedin Joy
IT- 21058

Q-1: Scrum Based E-commerce Project :

→ product backlog (Breaking user stories into tasks)

User Story: ① As a user I want to login securely so that I can access my account.

Tasks:

- I) Design the login page
- II) set up the database to store dataset
- III) Develop the backend for the login system
- IV) Encrypt password.
- V) valid inputs (check if username and password are correct)
- VI) Add a forget password option.

User Story-2: As a user. I want to search for products by category to find items easily

Task:

- I) Design the searchbar.

2. Create a database to store product information
3. Develop the backend for the search feature
4. Optimize search for large database.
- 5) Implement category based filtering
6. Test the feature
7. Conduct user testing for the feature

Sprint planning :

- login feature (User story 1)
→ highest priority because user need secure access to their accounts.
- search feature (User story 2):
→ second priority because user can manually browse products & the search feature is not ready.

D. How to scrum board works:

- To Do : Task that haven't started yet
- In Progress : Tasks that are currently being worked on .

Q-2:

For a project with high risks and changing requirement, choosing the right methodology is important to manage risks and adopt to change effectively. Let's look at how each methodology helps.

1) Spiral Methodology:

- Risk management: The spiral model focuses on managing risk by constantly reviewing them throughout the project. It allows for quick identification of risks and fixes them early with testing and prototyping in each phase.

2) Agile Methodology:

- Risk management: Agile works in small steps called sprints, where the team checks and deals with risk can be address early and issues are found quickly.

3) Extreme Programming (XP):

Risk management : XP uses continuous testing and regular release, which helps identify problems immediately. The team also work in pairs which helps reduce errors and risk in code.

- Adaptability: XP is highly adaptable with short cycle and constant client feedback so it can quickly respond to changing needs.

For this project Agile is the best choice

while Agile?

→ Risk management

- Adaptability.

- Cost effectiveness.

Q-3

To chose the best development methodology for each project we need to consider the characteristics of both projects.

- Project A: well defined requirements uncertain at strict deadline.

- Project B: Evolving requirements uncertain timelines and continuous customer feedback.

let's compare the methodologies based on those characteristics

i) waterfall Model:

- predictability: The waterfall model is linear and sequential. It works project with well-defined requirements.

- Customer coll

SCANNED BY CamScanner

2. Agile Model:

- Predictability :- Agile is iterative delivering small pieces of the product regularly. It is less predictable but more adaptable to changes.
- Customer collaboration: continuous customer involvement and feedback is key in Agile. The product evolves based on customer needs throughout the process.

QSN-9:

Principal of software Engineering ethics:

- 1) Public Interest : Always prioritize the well-being of society, users and public over personal or company benefits
- 2) Quality of works: Deliver high quality software that meets user needs work efficiently and avoid harm.

Role of ACM/IEEE code Ethics:

The ACM/IEEE code of Ethics provides guidelines to help software engineer make ethical decision.

1) Act in public interest: Always prioritize safety fairness and social good in software development.

2) Be honest and trustworthy: Avoid misleading stakeholders on users about software capabilities or limitations.

3) Respect privacy: Protect user data and ensure it's used responsibly

4) Continuous learning :- stay updated with the latest knowledge and best practices.

5) work with integrity: Avoid bias, unfair treatment or harm in software processes.

[QSN-5]

1) Flight booking :- User must be able to search for flight, select one and book tickets.

• Contribution: Provides the core functional of the system and enhances user satisfaction.

2) User registration and login : The system should allow user to create account and login securely.

• Contribution: Ensures personalized services and protect user information.

3) Payment processing :- The system should support multiple payment method like credit card, debit card and digital wallets.

• Contribution: Simplifies the transaction process improving the user experience.

[Q5N-6]

Definition: The V-model (verification and validation model) is a software development model that emphasizes the relationships between development phase and corresponding testing phase and . The V shape visually represents the sequence of activities where the left side involves testing

*Key features of the V-model:

1) Development phase: (left side of the V)

• Requirement Analysis: understanding what the customer want.

Test: Acceptance testing ensures the final product meets user needs

Architectural Design: Designing the modules components of the software

Test: Integration testing checks how modules work together.

2) Testing phases (right side of the V)

- Each development phase has a corresponding test phase to validate it
- Testing starts early with plans and progresses alongside development.

[QN-7]

prototype development is a process used in software engineering to create a working model of software

It helps refine requirement and ensures that the final product meets user expectations.

1) Requirement Gathering:

- Collect initial requirement from users or clients.
- Focus on understanding the key functionalities.

2) Quick Design:

- Create a simple and rough design of the software including basic screens, navigation and essential feature.

3) Prototype Building:

- Develop the prototype with the necessary features.
- Ensure it is functional but not the final product.

The process improvement cycle helps make software development better by finding problem, fixing them and making the process smoother and more efficient :-

1) Measure the process :-

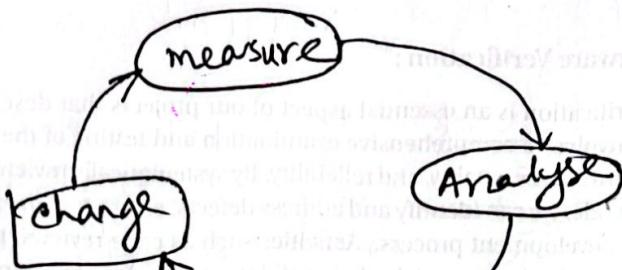
collect data about how tasks are being done :-

Example: check how long it takes to finish coding or how many bugs are found.

2) Analyse the data :-

Look at the data to find problems

Example: if coding takes too long figure out why.



1) Time taken:

- Measures how long tasks take, like coding or testing.

2) Defect count:

- tracks the number of bugs

3) Maintenance

• maintenance means fixing bugs in the software after it has been released to customers. It involves identifying and fixing errors in the existing codebase to ensure the software continues to function correctly and efficiently.