202312014
Kishan Vaghamashi

1) **Write a summary in your own words for SDLC.**

Answer:-

A) **Requirement:**
We understand what features the software needs, what problem this software needs to solve and who will use it.

B) **Specification:**
Now, We define the base (data, functions) and steps (processes) in detail.

C) **Design:**
This is where things get visual. We create blueprints for the software's architecture, user interface, and data flow.

D) **Coding:**
The developers bring the design to display by writing lines of code.

E) **Verification and Validation:**
Once the code is ready, we test it thoroughly to make sure it works as intended and meets all the requirements and ask validation from our client.

F) **Implementation/Installation:**
The finished software is deployed to its actual environment, whether it's a website, app, or something else.

G) **Maintenance:**
The maintenance phase involves fixing bugs, adding new features, and adapting to changing needs.

2) Identify different tools used in each phase of SDLC. Explore TWO tools for each phase of the
SDLC and write a 500 words paragraph on what are your findings about each of that tool.
Answer:

1. Requirements Phase:
Jira: Jira is widely used for requirements management. It allows teams to create, track, and manage user stories, epics, and tasks.

Confluence: Confluence complements Jira by providing a platform for documentation. Teams use Confluence to create detailed requirement documents, user manuals, and other project-related documentation collaboratively.

2. Specification Phase:
Lucidchart: Lucidchart is a versatile tool for creating visual representations of system architectures, workflows, and specifications.

Microsoft Word/Google Docs: Traditional word processing tools like Microsoft Word or Google Docs are still widely used for creating detailed specification documents.

202312014
Kishan Vaghamashi

3. Design Phase:
Sketch: Sketch is a powerful design tool that allows UI/UX designers to create high-fidelity prototypes and designs. It streamlines the design process and facilitates collaboration among designers and developers.

Adobe XD: Adobe XD is another popular tool for designing and prototyping user interfaces. It enables designers to create interactive prototypes, ensuring a more comprehensive understanding of the final product.

4. Coding Phase:
Visual Studio Code: Visual Studio Code is a lightweight, yet powerful, source code editor. It supports various programming languages and provides features like debugging and Git integration, enhancing the coding experience.

Eclipse: Eclipse is an integrated development environment (IDE) widely used for Java development.

5. Verification and Validation Phase:
Selenium: Selenium is a popular automated testing tool for web applications. It allows testers to write scripts in multiple programming languages for testing the functionality of web applications.

JUnit: JUnit is a widely used testing framework for Java applications. It provides annotations and assertions to streamline the process of writing and executing tests, ensuring code reliability.

6. Implementation and Installation Phase:
Docker: Docker is a containerization platform that simplifies the deployment process. It allows developers to package applications and their dependencies into containers, ensuring consistency across different environments.

Jenkins: Jenkins is an automation server that facilitates continuous integration and continuous delivery (CI/CD). It automates building, testing, and deploying code changes, ensuring a smooth implementation process.

7. Maintenance Phase:
Git: Git is a version control system widely used for tracking changes in source code during maintenance. It helps teams manage and collaborate on code changes effectively.

Bugzilla: Bugzilla is a bug tracking tool that aids in managing and prioritizing reported issues. It provides a centralized platform for developers and testers to communicate and resolve bugs during maintenance.

202312014
Kishan Vaghamashi

3) Prepare a report on software development projects that you have developed till now.
Answer:

Flight Ticket Booking App - Software Development Project Report

1. Introduction:

This report details the development and features of the "Flight Ticket Booking App," a project completed as part of B.Sc. IT. This mobile application allows users to conveniently search, book, and manage their flight travel from one user-friendly platform.

2. Project Objectives:

- Develop a web application for booking flight tickets across various airlines.
- Provide real-time flight information and schedule search functionality.
- Include features for trip management and flight status updates.
- Design a user-friendly interface for intuitive navigation and interaction.

3. System Design and Architecture:

- **Client Layer:** The mobile app built using PHP handles user interaction and displays information.
- **Business Logic Layer:** The server-side application processes user requests, interacts with airline APIs, and performs ticket booking and management functions.
- **Data Layer:** A secure database stores user information, flight schedules, booking details, and other relevant data.

4. Key Features:

- **Flight Search:** Search for flights based on origin, destination, dates, and travel preferences.
- **Comparison and Filtering:** Compare flight options and filter based on price, airline, duration, and stopovers.
- **Real-time Flight Information:** View live flight status, including arrival/departure times, delays, and gate information.
- **Booking and Payment:** Book flights securely using integrated payment gateways with various payment options.
- **Trip Management:** View and manage booked flights, including itinerary details, passenger information, and seat selection.

5. Development Tools and Technologies:

- PHP, VSCode, GitHub, Postman

6. Challenges and Solutions:

- Integrating with multiple airline APIs presented data compatibility issues, which were addressed through data normalization and transformation techniques.
- Ensuring secure payment processing required rigorous testing and implementation of industry-standard security protocols.
- Optimizing app performance for various mobile devices required careful memory management and efficient code design.