Assignment 1

Task 1:

We are using the Agile methodology for the Banking Management System, as Agile focuses on customer satisfaction and the continuous delivery of valuable software by dividing the project into small, manageable units called sprints. For this, we are going to use Jira software, in which the sprint will go on, and the project will be divided into small tasks. These tasks will then be assigned to team members according to their duties in the team. This approach allows for flexible planning, development, testing, and implementation stages, making it easier to adapt to changes in project requirements or market conditions.

Reason for using Agile Method:

1. Focus on Customer Satisfaction: This is crucial for banking software, where user satisfaction can significantly impact adoption and usage rates.

2. Dividing the Project into Sprints: Sprints are short, time-boxed periods during which specific work has to be completed and made ready for review. This structure allows for regular reassessment of project priorities and the flexibility to adapt to changes.

3. Use of Jira Software: Jira is an agile project management tool that facilitates sprint planning, task assignment, and progress tracking. Its use underscores a commitment to maintaining organizational transparency and enhancing team coordination,

enabling more efficient workflow management and problem resolution.

5. Assignment of Tasks According to Team Duties: Agile encourages the distribution of tasks based on team members' roles and expertise, promoting efficiency and quality in task execution. This tailored assignment ensures that each component of the banking system is developed by individuals best suited to address its specific requirements and complexities.

6. Flexible Planning and Development: Agile's iterative approach allows for ongoing planning and development adjustments based on feedback and testing results. This adaptability is vital for banking applications where functional and security requirements must meet the highest standards.

Task 2:

The 5 functional and 3 non-functional requirements related to our project is given below.

Functional Requirements:

1. User Authentication and Authorization: The system must allow users to register (sign-up) and log in using secure credentials.

2. Transaction Processing: Users should be able to perform banking transactions such as bill payments, deposits, withdrawals, and online purchases. The system must accurately process these transactions in real-time, updating account balances and transaction histories accordingly.

3. Account and Card Management: The system must provide functionality for users to view and manage their account and card

details, including the ability to freeze and unfreeze their cards. Users should be able to review their transaction history, current balance, and other relevant account information.

4. Beneficiaries Management: The system must enable users to easily add, modify, and delete beneficiaries for transactions such as wire transfers and bill payments. This feature facilitates efficient and safe money transfers between accounts, both within the same bank and to external institutions, enhancing the user experience by simplifying recurring transactions.

5. Budget Planning Tool: The system should include a budget planner that allows users to set budget goals, track their spending against these goals, and receive notifications or advice based on their financial behavior and patterns.

Non-Functional Requirements:

1. Usability: The web application should be user-friendly, with an intuitive interface that allows users to easily navigate and perform banking tasks without extensive training. It should be accessible on various devices, including desktops, tablets, and smartphones.

2. Performance and Scalability: The system must be able to handle high volumes of transactions and user interactions simultaneously without significant delays. It should be scalable, allowing for future expansion in terms of users, transactions, and product offerings without degradation in performance.

3. Security: The system must implement strict security measures, including encryption, secure data storage, and protection against common web vulnerabilities. It should ensure the confidentiality, integrity, and availability of customer data and financial records.

Task 3:

Team Duties:

For the Banking Management System project, each group member will take on specific roles that leverage their skills and expertise to ensure the project's success. Here are the roles assigned to each member:

1. Abdullah Awan – Scrum Master, Backend Developer, Tester

Scrum Master: Oversees project progress, ensures milestones are met, and maintains communication among team members.

Developer: Focuses on server-side logic, and integration of the application's frontend.

Unit and Integration Testing: Responsible for testing individual units of code for functionality and integration points between system components to ensure they work together seamlessly.

2. Ifra Ejaz – Designer and Developer

Designer: Creates the user interface and user experience designs, ensuring the application is intuitive and user-friendly.

Developer: Implements the front-end logic based on the designs, working closely with the backend team to integrate the application's user-facing elements.

3. Syed Ali Hassan Zaidi - Developer, Tester

Developer: Contributes to both front-end and back-end development, ensuring smooth data flow and functionality.

Unit and Integration Testing: Tests both individual components and their interactions within the system to identify and fix issues early in the development process.

4. Sultan Sheikh - Database Administrator and Developer

Database Administrator: Manages the database architecture, ensuring data consistency, security, and availability.

Developer: Implements the front-end logic based on the designs, working closely with the backend team to integrate the application's user-facing elements.