ADDIS ABABA SCIENCE AND TECHNOLOGY UNIVERSITY

College of Engineering

Agile Methodology for mPOS Application Development

Software Component Design

Group Members	ID
1. Abdi Getachew	ETS 0003/13
2. Ahmed Abdulhakim	ETS 0091/13
3. Akrem Ahmed	ETS 0097/13
4. Ali Salahadin	ETS 0105/13
5. Mahfouz Teyib	ETS 0786/13

Submitted to: Lecturer Gizatie D.

Submission date: December 18, 2024

Table of Contents

1. Introduction

Purpose

The adoption of the Agile methodology for developing the mPOS application ensures a structured yet flexible approach to the project. Agile emphasizes iterative and incremental development, allowing the team to adapt quickly to changing requirements and priorities. This approach fosters close collaboration among stakeholders, ensuring their input is incorporated at every stage of development. By breaking down the project into manageable sprints, Agile facilitates continuous improvement and delivers a high-quality mPOS application tailored to the needs of merchants and administrators.

Scope

In line with the Agile model, the development of the mPOS application will be divided into iterative phases or sprints, each focusing on delivering specific features incrementally. This ensures that core functionalities are available early in the project lifecycle, with subsequent iterations refining and expanding on them. Key areas of focus include:

- **Inventory Management**: Delivered in early sprints to enable stock viewing, adding, and editing capabilities.
- **Payment Processing**: Implemented incrementally with seamless Telebirr integration to ensure smooth transaction handling.
- **Admin Dashboard**: Introduced in later sprints to provide advanced functionalities like user management, transaction reports, and business analytics.

This Agile approach ensures that each phase of the project is driven by stakeholder feedback, fostering collaboration and delivering a product that evolves alongside user needs.

2. Agile Methodology Overview

Core Values

The Agile methodology is guided by the following core values, which ensure that the development process remains flexible, collaborative, and outcome-oriented:

1. **Focus on Individuals and Interactions Over Processes and Tools**
    - Prioritizing team collaboration and effective communication ensures smooth project execution, fostering innovation and problem-solving.
2. **Deliver Working Software Over Comprehensive Documentation**
    - The primary goal is to create a functional product that delivers value to users, with documentation serving as a supporting tool rather than the main focus.
3. **Foster Customer Collaboration Over Contract Negotiation**
    - Actively involving stakeholders throughout the project ensures their requirements and feedback shape the development process, resulting in a product that meets their expectations.
4. **Respond to Change Over Following a Fixed Plan**
    - Agile embraces changes in requirements, even late in the development cycle, to deliver a product that adapts to evolving user needs and market conditions.

Key Principles

The following principles underpin Agile practices, ensuring a streamlined and dynamic development process:

1. **Iterative Development**
    - The project is divided into short, time-boxed sprints, with each iteration focusing on delivering specific features incrementally.
2. **Regular Feedback Loops**
    - Continuous engagement with stakeholders provides opportunities to validate progress and incorporate feedback, improving the product with each iteration.
3. **Continuous Delivery of Valuable Functionality**
    - Agile ensures that every sprint delivers working features, prioritizing the most critical functionalities to deliver value early and consistently.

4. **Embrace Change to Meet Evolving Needs**
    - Flexibility to adapt to changing requirements ensures the final product aligns with user expectations and remains relevant in a dynamic environment.

3. Agile Frameworks and Tools

Scrum Framework

Scrum will serve as the primary Agile framework to manage the mPOS project, ensuring a structured approach through iterative development cycles called Sprints.

1. **How Scrum Will Be Used**
    - The project will be broken into time-boxed Sprints, each lasting 1–2 weeks.
    - At the start of each Sprint, the team will prioritize tasks from the Product Backlog and commit to delivering them by the end of the Sprint.
    - Progress will be reviewed during Sprint ceremonies, fostering collaboration and continuous improvement.
2. **Key Roles**:
    - **Product Owner**: Represents stakeholders and defines the Product Backlog items. Ensures features like inventory management, payment processing, and admin dashboards align with business goals.
    - **Scrum Master**: Facilitates the Scrum process, removes obstacles, and ensures the team adheres to Agile principles.
    - **Development Team**: Focuses on delivering the committed Sprint goals, including Flutter-based front-end features and Firebase integration for the backend.
3. **Artifacts**:
    - **Product Backlog**: A dynamic list of features and tasks (e.g., login functionality, payment processing) prioritized based on business needs.
    - **Sprint Backlog**: A subset of Product Backlog items chosen for development during a specific Sprint.
    - **Increment**: A working product increment delivered at the end of each Sprint, demonstrating completed functionality.
4. **Ceremonies**:

- **Sprint Planning**: The team defines the Sprint goal and selects tasks from the Product Backlog to complete during the Sprint.
- **Daily Standups**: 15-minute meetings to discuss progress, challenges, and plans for the day.
- **Sprint Review**: A demonstration of the completed Increment to stakeholders for feedback.
- **Sprint Retrospective**: Reflect on the Sprint to identify areas for improvement in the next iteration.

Kanban

Kanban will complement Scrum by providing a visual representation of workflows, enabling the team to monitor progress and identify bottlenecks.

- **Kanban Boards**:
  - Tasks will be categorized into columns like "To Do," "In Progress," "Review," and "Done."
  - Ensures transparency and helps the team track task status at any point during the Sprint.

Tools

1. **Project Management Tools**:
   - **Jira or Trello**: Used to manage the Product Backlog, Sprint Backlogs, and Kanban boards, providing visibility into task assignments and progress.
2. **Development Tools**:
   - **Firebase**: Backend services for authentication, data storage, and real-time updates.
   - **Flutter**: Cross-platform framework for building the app's user interface and functionalities.

These frameworks and tools will ensure the project remains organized, transparent, and adaptable to changes while delivering high-quality functionality.

4. Agile in mPOS Development

Planning Phase

The development process begins with a thorough planning phase where stakeholders collaborate to define the **Product Backlog**. This backlog contains all the desired features of the mPOS application, prioritized by importance and feasibility. Key features include:

- **User Authentication**: Secure login and role-based access.
- **Payment Processing**: Seamless integration with Telebirr.
- **Analytics**: Tools for viewing sales trends, stock reports, and transaction summaries.

Stakeholder meetings ensure all requirements are captured, clarified, and prioritized to set the foundation for iterative development.

Sprint Execution

The project will be divided into **2-week sprints**, each focusing on delivering specific functionalities:

- **Sprint 1**: Develop the Login Page and implement User Authentication using Firebase.
- **Sprint 2**: Build Inventory Management features, including adding, editing, and viewing stock.
- **Sprint 3**: Integrate the Telebirr Payment Gateway to enable smooth payment processing.
- **Sprint 4**: Create the Admin Dashboard and Reporting Tools for analytics and user management.

Each sprint concludes with a working increment of the application, ensuring steady progress and early feedback.

Daily Standups

The team will hold 15-minute daily meetings to discuss:

1. Progress made since the last standup.
2. Challenges or roadblocks encountered.
3. Plans for the day.

These standups ensure alignment, transparency, and quick resolution of issues.

Sprint Reviews and Retrospectives

- **Sprint Reviews**: At the end of each sprint, deliverables are demonstrated to stakeholders for feedback and validation.
- **Retrospectives**: The team reflects on the sprint to identify successes and areas for improvement, applying these insights to subsequent sprints.

5. Key Agile Practices for mPOS Development

Continuous Integration and Deployment (CI/CD)

Automate testing and deployment processes to ensure:

- Code changes are integrated frequently.
- Each update is tested for functionality and deployed incrementally.

Test-Driven Development (TDD)

Adopt a **TDD approach** to maintain high code quality by:

1. Writing tests before implementing features.
2. Ensuring all tests pass before proceeding.
   This ensures robust, error-free functionality.

Incremental Delivery

Deliver the application in small, functional modules to:

- Gain early feedback from stakeholders.
- Allow stakeholders to use and validate features such as user authentication and payment processing as soon as they are ready.

Collaborative Development

Promote teamwork through practices such as:

- **Pair Programming**: Two developers working together to enhance code quality.
- **Code Reviews**: Reviewing code to ensure consistency, maintainability, and adherence to project standards.

These Agile practices ensure that the mPOS application is developed efficiently, with continuous improvement and stakeholder collaboration throughout the lifecycle.

6. Benefits of Agile for mPOS

Flexibility

Agile allows the development team to adapt quickly to stakeholder feedback and changing requirements. Features like inventory management and payment processing can be adjusted or refined in response to evolving business needs, ensuring the application remains relevant and useful.

Transparency

The use of tools like Kanban boards and Scrum ceremonies, such as daily standups and sprint reviews, provides real-time visibility into project progress. This transparency fosters trust and keeps all stakeholders informed about the status and direction of the project.

Faster Time-to-Market

By delivering core functionalities (e.g., login page, user authentication) early in the project lifecycle, Agile ensures that stakeholders can begin using the product sooner. Advanced features can be iteratively added without delaying the initial release.

Risk Reduction

Regular feedback loops, through sprint reviews and retrospectives, help identify and resolve issues early in the development process. This minimizes the risk of significant problems going unnoticed until later stages, ensuring a smoother development journey.

7. Challenges in Agile Implementation

Managing Stakeholder Expectations

Agile encourages flexibility, but frequent changes in requirements can disrupt sprints. Balancing evolving stakeholder expectations with the need to complete planned tasks requires careful management and communication.

Team Collaboration

Effective communication is essential for Agile to succeed. Misalignment among team members can lead to delays or inconsistencies in the deliverables, especially in distributed or cross-functional teams.

Prioritization

Agile relies on prioritizing tasks in the backlog, but balancing urgent, short-term requirements with long-term goals can be challenging. Clear prioritization is crucial to avoid neglecting critical but non-urgent tasks.

8. Conclusion

Agile methodology plays a central role in enabling the efficient and adaptive development of the mPOS application. By focusing on iterative delivery, stakeholder collaboration, and continuous improvement, Agile ensures the project remains aligned with business goals and user needs.

The expected outcomes of adopting Agile include a high-quality product delivered on time, satisfied stakeholders who actively participate in the development process, and a culture of continuous learning and adaptation that ensures the mPOS application evolves effectively over time.