

CSCI 4210: Introduction to Software Engineering



University of New Orleans
Department of Computer Science



Capstone Project: Touchless Kiosk System

Dev-Cycle 2: User Feedback & Refinement

Cycle Overview

The User Feedback and Refinement Cycle focuses on gathering user feedback, identifying improvement areas, and refining the MVP based on real-world use cases and user interactions. This cycle emphasizes testing, iterative enhancements, and alignment with user needs.

Key Objectives

- **Collect User Feedback:** Conduct testing with representative users to identify key issues and improvement opportunities.
- **Implement Refinements:** Address feedback by making necessary modifications to enhance usability, performance, and functionality.
- **Enhance Documentation and Training Materials:** Provide clear, updated documentation to guide future users and developers.

Cycle Tasks

Part 1: User Testing and Feedback Collection

Objective: Conduct user testing to gain insights into user experience, identify usability challenges, and gather improvement suggestions.

1. Design User Testing Scenarios:

- Develop scenarios or tasks that represent typical user interactions.
- Ensure testing scenarios cover core functionalities and common use cases.

2. Conduct User Testing Sessions:

- Run testing sessions with representative users.
- Collect qualitative and quantitative feedback on usability, performance, and user satisfaction.

3. Compile and Analyze Feedback:

- Review feedback, identifying trends, recurring issues, and areas needing improvement.
- Prioritize issues based on impact on usability and user experience.

Part 2: Refinement and Iteration

Objective: Based on user feedback, refine the MVP to improve performance, usability, and functionality.

1. Implement High-Priority Changes:

- Address critical issues identified during user testing.
- Make iterative improvements to enhance the user experience.

2. Optimize System Performance:

- Identify and resolve performance bottlenecks.
- Ensure smooth operation and responsiveness.

3. Update User Interfaces:

- Refine UI components based on feedback, focusing on accessibility and ease of use.
- Ensure consistency in design and layout for an intuitive user experience.

Part 3: Documentation and Training Materials

Objective: Update documentation and training materials to reflect refinements and provide clear guidance to future users and developers.

1. Revise API Documentation:

- Ensure API documentation is updated to include any changes or enhancements.
- Add additional explanations for improved clarity where needed.

2. Create User Training Materials:

- Develop user guides or tutorials that cover key functionalities and provide troubleshooting tips.
- Ensure training materials are accessible and easy to understand.

3. Document User Feedback and Resolution Summary:

- Create a report summarizing user feedback and the steps taken to address each issue.
- Include a list of remaining issues or future improvement opportunities.

Deliverables

The following deliverables will be managed using GitHub Issues and GitHub Project management tools, ensuring transparent tracking and organized collaboration:

- **User Testing Report:** This deliverable will be documented as a GitHub Issue. Each testing scenario, along with key feedback and identified issues, will be logged in a structured format. Use the **User Testing** label for categorization and assign contributors responsible for testing each feature.
- **Refined Feature-Ready Components:** For each core feature requiring refinement based on user feedback, create individual GitHub Issues. Each issue should reference related feedback from the **User Testing Report** and include action items and acceptance criteria for completion. Use the **Feature Update** label and assign relevant team members.
- **Updated Documentation:** Documentation updates, including API documentation and user training materials, will be tracked as separate GitHub Issues. Each issue should specify the documentation sections requiring updates, with a checklist for tracking progress. Use the **Documentation Update** label to organize related tasks.
- **User Feedback Summary Report:** A GitHub Issue will document user feedback comprehensively, with sections for listing each piece of feedback, corresponding resolution actions, and status updates (e.g., **Resolved**, **Pending**). Use the **Feedback Summary** label and keep this issue updated as feedback is addressed.

Each deliverable will be tracked within the GitHub Project board, categorized by cycle stages (e.g., **Backlog**, **In Progress**, **Completed**). This setup ensures streamlined communication, accountability, and structured progress tracking across teams.

Grading Rubric for Dev-Cycle 2

This rubric evaluates key aspects of the User Feedback and Refinement Cycle, focusing on user testing, refinement implementation, documentation updates, and project management.

Category	Description	Points
User Testing and Feedback Collection	Design and conduct effective user testing sessions. Compile and analyze feedback to identify key areas of improvement. Demonstrate thorough feedback collection and prioritization of refinements.	25%
Refinement and Iteration	Implement improvements based on user feedback. Prioritize and address high-impact refinements to enhance functionality and user experience, showing clear alignment with user needs.	25%
Documentation and Training Materials	Update documentation to reflect changes and provide clear user guides and API documentation. Ensure comprehensive documentation that aids both end-users and future developers.	20%
Project Management and Collaboration	Track all tasks related to feedback, refinement, and documentation on GitHub Project boards. Maintain transparency, organization, and accountability in progress tracking.	15%
User Feedback Summary Report	Compile a detailed report summarizing user feedback, resolutions, and remaining issues. Clearly document feedback, actions taken, and track the resolution status for future reference.	15%
Total		100%

Detailed Breakdown

- **User Testing and Feedback Collection (25%):** Conduct structured user testing with scenarios that cover key functionalities, gathering valuable insights for further refinement. Prioritize improvements based on the impact on user experience.
- **Refinement and Iteration (25%):** Implement prioritized improvements effectively, focusing on enhancing user experience and addressing usability or performance issues identified during testing.
- **Documentation and Training Materials (20%):** Ensure that all changes are reflected in updated documentation, including API and user guides. Create user-friendly training materials to facilitate easy onboarding and usage.
- **Project Management and Collaboration (15%):** Use GitHub Projects for organized task tracking, reflecting real-time progress on feedback and refinements. Encourage team collaboration and accountability.
- **User Feedback Summary Report (15%):** Compile a well-organized summary report documenting feedback, actions taken, and any remaining or future issues. Ensure the report is accessible and helpful for future improvement cycles.

Appendix: Agile Development Roadmap

This Agile Development Roadmap provides an outline of each development cycle in the project, highlighting the objectives and activities central to each phase.

0. **Discovery and Planning Cycle** – This phase centers on requirements gathering, stakeholder analysis, and initial roadmap creation. It aligns with Agile’s discovery phases, where understanding user and system needs is essential for setting a clear development path.
1. **MVP Development and Integration Readiness Cycle** – This cycle emphasizes initial development, focusing on isolated component building and preparing for integration. The goal is to establish a minimal product foundation that can operate independently but is ready for further integration.
2. **User Feedback and Refinement Cycle** – After the MVP is complete, this phase collects user feedback, allowing for testing insights to guide iterative improvements. This cycle ensures the product evolves to meet real user needs effectively.
3. **Feature Expansion and Optimization Cycle** – Building on the MVP, this phase introduces high-priority features and optimizations. It expands the product’s functionality and enhances performance, creating a more robust and refined system.
4. **Deployment and Postmortem Analysis Cycle** – The final phase focuses on deploying the application, delivering a structured presentation, and conducting a post-mortem to reflect on the project’s journey and lessons learned.

Each cycle represents an integral stage in Agile development, from planning to deployment, ensuring continuous alignment with user needs and system goals.