EECS2311 – Iteration 3: Time to Release

Team Name
Section Z
Group 12

March 31, 2025

Contents

1 Introduction and Overview

This document is prepared for Iteration 3 of our EECS2311 Software Development Project. It follows the instructions laid out in the *Itr3.pdf* document and addresses the process, implementation, testing, refactoring, and documentation requirements for the final release.

2 Revised Planning Process

2.1 Updated User Stories and Priorities

Provide a summary of all user stories (both new and leftover from previous iterations), along with their updated priorities and cost estimates.

2.2 Velocity and Sprint Planning

Discuss your estimated velocity based on the progress from Iteration 2. Explain how you accounted for additional tasks such as refactoring and testing, and how you reallocated your resources for Itr3.

2.3 Discrepancies from Original Plan

Include the original plan (from Itr2) and highlight any changes here. Briefly justify why these changes were necessary.

3 Implementation, Testing, and Bug Fixing

3.1 Implemented Features

List the user stories implemented in this iteration. Each user story should be fully functional with proper unit tests, integration tests, and a working UI.

3.2 Testing Approach

Describe how you tested the new features and how you addressed previously reported bugs. Mention the specific frameworks/tools used (e.g., JUnit, PyTest, etc.). Refer to any relevant test logs or code coverage reports here.

3.3 Bug Fixes

Explain the critical bugs that were identified and fixed. Reference any GitHub Issue IDs, how you resolved them, and whether any remain outstanding.

4 Refactoring and Code Smell Resolutions

4.1 Identified Code Smells / Design Issues

List the code smells or design issues you discovered. For each, provide a short before/after snapshot of your design if possible.

4.2 Refactoring Techniques Applied

For each significant refactoring, describe:

- **What was changed** (e.g., introduced new classes, replaced primitive types with objects)
- **Why it was changed** (e.g., to improve maintainability, readability, testability)
- **How it improves the system** (link to relevant design principles)

4.3 Outstanding Issues

If there are known issues you have chosen not to fix this iteration, explain the rationale and future plan for addressing them.

5 Documentation Updates

5.1 Wiki / README Changes

Summarize any major changes made in your GitHub Wiki or README to reflect Iteration 3 updates. This includes new instructions for how to install, run, and test your application.

5.2 Architecture Diagram

Include or describe any architecture changes here. Label each seam in the diagram with the corresponding integration test classes or modules.

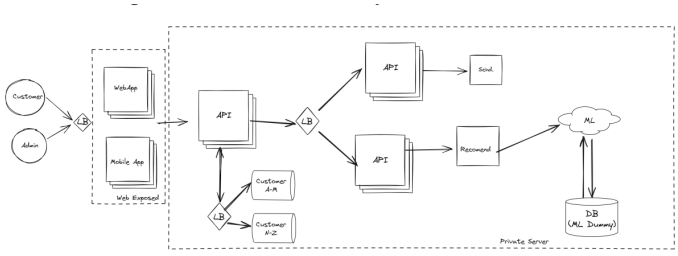

Figure 1: Revised Architecture Diagram (Example)

6 Submission and Evaluation

According to *Itr3.pdf*, the final submission must include:

- Source code with proper structure (`src/`, `test/`, `lib/`, etc.).
- Database folder if your application needs it.
- Documents such as:
  - Old planning document (from Itr2).
  - This updated Iteration 3 plan.
  - A refactoring document or incorporate that content here.
  - Revised architecture diagram.
- Log file containing meeting minutes, rationale for changes, big design decisions, etc.

Include everything in your GitHub repository by the deadline. Ensure it is straightforward to compile, run, and test your code on a fresh machine.

7 Conclusion

In this document, we presented the updated plan, implemented features, refactoring steps, and documentation for Iteration 3. We have accounted for the guidelines in *Itr3.pdf* and prepared the project for final release.