

Code Testing Report
SSE Capstone - University of Regina
Safety Toolbox
Winter 2024

Mackenzie Kot
Mikayla Peterson

Introduction

This code testing plan describes the approach we used to test Safety Toolbox. This document covers the testing strategy, execution and management process, and test results.

Code Testing Strategy

Our code testing strategy was to perform testing manually by navigating through the UI ourselves using White Box testing. Our application is CRUD-based and does not perform complex calculations of any kind, which is why we felt our testing approach was appropriate. Backend verification was done by inspecting the tables in the database to ensure entries were appropriately added and modified. Since we have several different levels of access in our application, we did all the tests using the role that has the highest privileges and then confirmed access was adjusted correctly for the other roles.

We used an unautomated version of Test-Driven Development to check for problems while we were developing, and we held a final large testing session in April to test the entire application to ensure nothing was missed. Testing along the way made the final testing session run smoother.

The scope of the tests includes: input validation, access level verification, observing the UI for visual problems, and edge case tests.

Execution and Management Process

Since we performed manual testing, we kept track of all our tests in a spreadsheet and would update the spreadsheet when new tests were made. The spreadsheet has separate sheets for each main page of the application and each sheet has sections for subpages and processes.

Our tests were designed to focus on inputting data and visual bug monitoring. Testing the inputs was done by selecting valid and invalid inputs and edge cases. These tests would confirm that the application would accept valid input and produce error messages for invalid inputs.

In the event a test failed, we would investigate the cause and fix it. The test would then be rerun and this process would repeat until the test would pass.

Test Results

We ended up having a total of 89 test cases during our final testing session in early April. We deemed this appropriate since we were able to cover all the 'happy' paths and all alternative paths a user can take in the application. These tests also covered different display options that can occur for specific pieces of data. Our final spreadsheet is available for viewing in our GitHub.