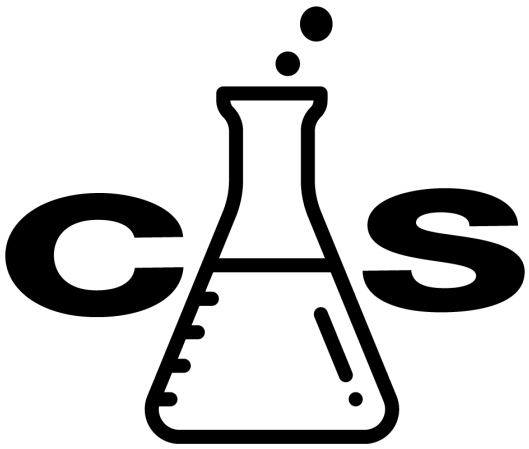
CSLabs Project Extension, Team 1 – 2021–2022

Test Summary Report



CS Labs

Contents

1. Purpose

This document explains and summarizes activities performed for our tests, results of our tests, following recommendations, best practices, and present the exit criteria for the testing of the application.

2. Application Overview

CSLabs is a learning environment that hosts easily accessible virtual machines thru a web interface that run on the data centers at IUS. The application is designed to be an easy learning environment. With a click of a button, a lab is set up in a couple of minutes. The user can be sure everything will work correctly because it is in a reproducible environment. The ability to access the machines from the browser helps people access the material quicker. Users can explore and open free modules and access the collection of labs inside them containing various virtual machines to use for learning.

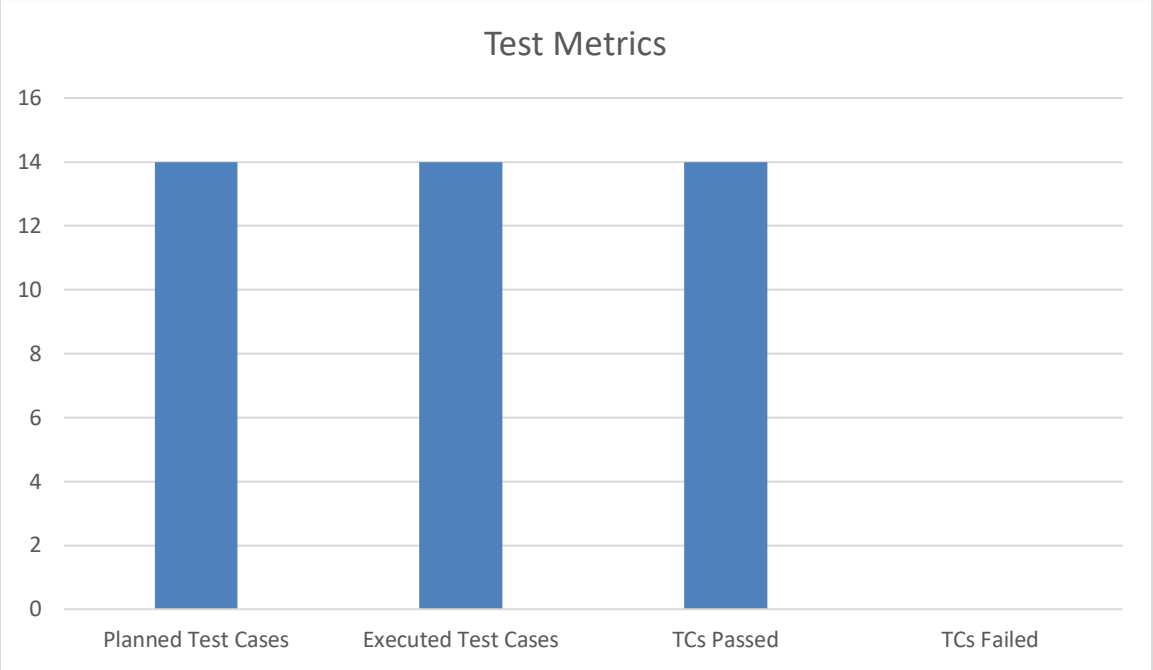
3. Testing Scope

a) In Scope
- Functional Testing for the following is in Scope of Testing
    - User Registration and Authentication
    - Accessing A Module
    - Starting A Lab
    - Module Tags
    - Modules Search
    - Lab Editor GUI

b) Out of Scope
- Outside of beta users Use of a Lab was not tested.
- Stress testing has not been done yet for this application.

c) Elements Not Tested
- Verification of lab completion was not tested due to legal limitations.
- Verifying a lab is complete could imply a grade which our system is notlegally able to store due to FERPA.

4. Metrics



5. Types of Testing Performed

a) Unit Testing

- This testing was performed when a new function was finished before being pulled into the production. This process of unit testing ensured the new function worked correctly with the whole product before being used.

b) System Integration Testing

- System integration testing was performed to ensure that all parts of the product were working together as specified.
- Standard lab usage was tested to be sure that users will be able to make use of the application without hindrance.

c) Regression Testing

- Testing is performed with each new test build before moving it to production.
- This provides security in detecting bugs before users areintroduced to the new features.

- Functionality is also tested when looking at the new features before moving them to production to ensure that they function as required.

d) Usability Testing

- Testing was performed to ensure the application is easily accessible and usable.
- A user can start a lab within a few clicks.
- The structure and location of the pages are self-explanatory for easy navigation.

6. Test Environment & Tools

- Puppeteer - NodeJS Headless chrome End to End testing solution.
- NUnit - .Net testing framework for the backend and CLI automation.
- Manual testing performed in all major browsers including Chrome, Edge, and Firefox.

7. Lessons Learnt

- Progressive unit testing maintains the overall performance of the application.
- E2E testing provides greater confidence of functioning features.

8. Recommendations

- There should be a permanent maintenance team in the CSG group to regularly maintain the application and keep it bug free.
- Regularly add new and attractive content or learning material which respond to student interest.
- Make the application to be more responsive for any screen size such as tablet and phone, however, the virtual machines remain inaccessible on smaller screens.
- More advanced testing tools such HP ALM can be considered in the future to automate the testing process and reduce the time involved in manual testing and improve accuracy.

9. Best Practices

- The application was tested in different browsers and operating systems to ensureit responded the same way and no data or performance was lost due to the environment.

- The application was also tested in a production environment simulating users'action to detect the system weakness and reinforce its security.
- Each piece of code was carefully reviewed by the group and explained by the programmer at the end of each sprint before being merged to the master.

10. Exit Criteria

a) All new functions and functions they interact with will be tested - Yes
b) All issues found during the testing ranging from medium to high severity must be addressed and dealt with or mitigated - Yes
c) Low severity issues - Addressed and planned to resolve
    - Allowable defects contain only cosmetic issues, any issues with functionality should be resolved.

11. Conclusion/Sign Off

All exit criteria have been met and the team has agreed for the application to ‘Go Live’. Initial users have been introduced to the product, and it is currently in preparation to be in production. All new features will still follow the exit criteria before the application is uploaded to production with approval of the team.

12. Definitions, Acronyms, and Abbreviations

- **Regression testing** - ensuring previous working functionality is still working as intended.
- **E2E** - end to end testing is testing the full application together to ensure theproduct functions together as whole.
- **Integration testing** - Testing the integration between multiple services in the application.
- **Unit testing** - Testing individual units of code e.g., classes and methods.

References

Gallavin, Jason et al. " CS Labs – Web Software Requirements Specification." 1 Oct. 2019, https://github.com/ius-csg/CSLabs-Capstone-Documentation/tree/master/cslabs-web-2019-2020/DesignDocs. 18 Feb. 2022