**Indiana University Southeast**

**2021-22 CSCI Capstone Project**

**Workshop Management Web Application**

**Sponsor: Dr. Suranga Hettiarachchi**

**Software Test Plan**

**James Schlesener**

1 Introduction 3

2 Business Background 3

3 Test Objectives 3

4 Scope 3

5 Test types Identified 3

6 Problems Perceived 3

7 Architecture 3

8 Environment 3

9 Assumptions 4

10 Functionality 4

11 Security 4

12 Performance 5

13 Usability 5

14 COMPATIBILITY 6

15 Test Team Organization 6

16 Schedule 6

17 Defects Classification Mechanism 6

18 Configuration Management 7

19 Release Criteria 7

# Introduction

The Workshop Management Web Application consists of 14 front-end components that need to be tested. Manual testing will test each component and how they interact with the back-end server and database.

# Business Background

The Workshop Management Web Application is designed to track workshops, their participants, and the equipment used within them.

# Test Objectives

The tests will ensure that all Angular components are fully functioning along with the server routes and pipe files and that all Express REST APIs are responding correctly.

# Scope

***Inclusions***

Front End: Angular

Back End: Express

***Exclusions***

MySQL

# Test types Identified

Manual integration testing will be done to test all the components.

# Problems Perceived

There aren’t any problems perceived at this time. The manual tests should ensure that the system is working properly.

# Architecture

The manual tests will be performed on each individual component and all their functionality.

# Environment

The client and server will be executed locally on the same computer. Each test will be manually executed.

# Assumptions

If the manual testing passes, then the software is functioning correctly.

# Functionality

***Constraints and Resolutions***

None

***Risk Identified & Mitigation Planned***

The risk of testing manually is human error and the need to perform tests again when updates are made. This is a very time-consuming process. To mitigate these issues, the tests will be executed upon completion of all programming before the annual student conference occurs.

***Test Strategy***

Manual tests will be performed to test the components and their functionality.

***Automation Plans***

The tests will be done manually. Future iterations of software development will include writing unit tests for the components using Jasmine.

***Deliverables***

The software will be fully functional.

# Security

***Constraints and Resolutions***

None

***Risk Identified & Mitigation Planned***

There aren’t any restrictions in the registration process. This is not an issue though, because the ask for this web application is to have a login process for tracking changes, not for security.

***Test Strategy***

Manually testing will be performed to test the login and registration components and their functionality.

***Automation Plans***

The tests will be done manually. Future iterations of software development will include writing unit tests for the components using Jasmine.

***Deliverables***

The user will be able to register, login, and logout and the user will be assigned a JWT so they can access the web application screens.

# Performance

***Constraints and Resolutions***

None

***Risk Identified & Mitigation Planned***

The performance of the software will depend on the client and server used. If either of those fails, the software can be run locally on the user’s computer.

***Test Strategy***

There aren’t any test plans at this time.

***Automation Plans***

There aren’t any plans to automate testing at this time.

***Deliverables***

The software should be reliable and perform as expected.

# Usability

***Constraints and Resolutions***

None

***Risk Identified & Mitigation Planned***

The usability of the software will depend on the availability of the client and server used. If either of those fails, the software can be run locally on the user’s computer.

***Test Strategy***

There aren’t any test plans at this time.

***Automation Plans***

There aren’t any plans to automate testing at this time.

***Deliverables***

The software should be reliable and perform as expected.

# Compatibility

***Constraints and Resolutions***

None

***Risk Identified & Mitigation Planned***

The software was developed on the Windows platform. However, it could also be executed on other platforms. No issues have been identified at this time.

***Test Strategy***

There aren’t any test plans at this time.

***Automation Plans***

There aren’t any plans to automate testing at this time.

***Deliverables***

The software has been developed on the Windows platform, but it shouldn’t have any compatibility issues.

# Test Team Organization

The test team consists of just one person, James Schlesener.

# Schedule

Manually testing will be executed after each component is built and at the end of software development before the annual student conference. This will ensure that the software is fully functioning during development and before demonstrated at the conference.

# Defects Classification Mechanism

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type of Defects | Functionality | Performance | Security | Usability | Compatibility |
| Critical | Software crashes | The server is unresponsive | none | The database is unavailable | Software doesn’t execute |
| Major | Software gives incorrect results | Slow response time | none | Some items do not display | Software doesn’t run on user’s browser |
| Minor | Software slow to respond. | Slight delay in response time | none | Items do not appear where expected | Software runs but visually has some compatibility issues |
| Cosmetics | The UI displays data in an unformatted way | Minor flaws in the UI. | none | The layout is altered and does not seem intuitive | The data does not appear exactly as designed |

***Defects Logging and Status Changing Mechanism***

Any issues will be documented in the individual test cases. These will be addressed as they occur to ensure they are resolved.

***Turn Around Time for defect fixes***

Any defects will be addressed as they occur. The goal will be to have defects fixed within 3 days of detection.

# Configuration Management

A list of tests with details of what to do and what to expect will be compiled in the test cases document. They will be arranged by component and by each feature of the component. Any tests that fail will be addressed within 3 days of detection to ensure the software is functioning properly for further development.

# Release Criteria

The software can be released if the tests all pass.