|  |
| --- |
|  |
| Testing Documentation |
| Men’s Shed Web Application - 2023 |

|  |
| --- |
| SOC09109 2022-3 TR2 001 - Group Project |

Contents

[Testing Overview 2](#_Toc127736134)

[Functionality Testing 3](#_Toc127736135)

[User Experience Testing 4](#_Toc127736136)

[Standard Member 4](#_Toc127736137)

[Heightened-Risk Member 4](#_Toc127736138)

[Administrator 4](#_Toc127736139)

[Security Testing 5](#_Toc127736140)

[Checklist 5](#_Toc127736141)

[Vulnerability Scan 5](#_Toc127736142)

[Dynamic Testing 5](#_Toc127736143)

[Remediation Log 6](#_Toc127736144)

# Testing Overview

The Men’s Shed web application will need to undergo thorough testing approached from several different angles in order to ensure that it provides the requested functionality, performs as expected and satisfies the requested requirements before being presented to the client. Testing the application from different angles will maximise the chance of spotting errors/flaws that can be addressed so that the final application is of good quality.

The testing phase will therefore contain the following objectives:

* **Functionality/Stress Testing** **–** Does the application provide the desired/requested functionality as agreed with the client? How will the application handle unexpected input/user behaviour?
* **User Experience Testing** **–** From the perspective of a user, how does the application perform/will the application be suitable for the desired target user? This will include testing the application as a standard user and from the administrator’s point of view as they use the application.
* **Security Testing** **–** From the perspective of a threat actor, are there any obvious security issues that stand out? Does the product contain the necessary safeguards to protect user personal information? Can the application be exploited or used as a platform to deliver exploits?
* **Remediation Phase –** Any issues or bugs that are identified during the testing will be addressed and removed/fixed.

# Functionality Testing

***{This section can be described with a standard test table against each of the requirements made by the clients}***

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Expected Result** | **Actual Result** | **Actions Taken** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# User Experience Testing

***{This section will be more suited to a written, point-of-view experience as it is being tested due to being a user-experience testing run}***

## Standard Member

## Heightened-Risk Member

## Administrator

# Security Testing

## Checklist

***{This is a basic must-have checklist from the considerations that will be checked off first, there will be a more dynamic evaluation in this category when the product is operational}***

|  |  |
| --- | --- |
| The application enforces HTTPS |  |
| Passwords are stored hashed with acceptable algorithm |  |
| Input validation in-play |  |
| Member information stored is encrypted |  |
| Administrator has access to user log data |  |
| User password policy is enforced |  |

## Vulnerability Scan

***{This will be a short write up with screenshots of a vulnerability scan run on the finished application}***

## Dynamic Testing

***{I will document my actions and findings in this section as I take a pass at the finished application from an average threat-actor perspective}***

|  |  |  |  |
| --- | --- | --- | --- |
| **Test** | **Expected Result** | **Actual Result** | **Actions Taken** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Remediation Log

***{Any bugs/issues found and addressed during testing will be documented here alongside actions taken to address them}***