**Group 5**

**Test plan for**

**<<WEBMAIL MSU>>**

**Joshua Moore, Rushma Parajuli, Khem Dhami**

*ChangeLog*

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Change Date** | **By** | **Description** |
| version number | Date of Change | Name of person who made changes | Description of the changes made |
| 1.0.0 | 10/19/2021 | Rushma Parajuli | Design/ Scope  This change request calls for creating a UML class diagram to help the programmer to write the codes. |
| 1.0.1 | 10/19/2021 | Khem Dhami | Schedule  This change request calls for managing the meeting schedule to ensure all the group members are updated with the original project plan. |
| 1.0.2 | 10/19/2021 | Joshua Moore | Implementation  Ensuring that all the functinalities of the WebMail is implemented. |
| 1.0.3 | 10/22/2021 | Rushma Parajuli | Verification  Verify if the code runs well both locally and passes all the test case using github actions. |

1 Introduction 2

1.1 Scope 2

1.1.1 In-Scope 2

1.1.2 Out-of-Scope 2

1.2 Quality Objective 2

1.3 Roles and Responsibilities 2

2 Test Methodology 3

2.1 Overview 3

2.2 Test Levels 3

2.3 Bug Triage 3

2.4 Suspension Criteria and Resumption Requirements 4

2.5 Test Completeness 4

3 Test Deliverables 4

4 Resource & Environment Needs 4

4.1 Testing Tools 4

4.2 Test Environment 5

5 Terms/Acronyms 5

# Introduction

The Test Plan for WebMail MSU describes the features to be tested, what tests are being performed and the roles of everyone in performing the testing, resources that are being used.

## Scope

### In-Scope

**The following features are being tested:**

Functional Features for Continuos Integration (CI testing) are: Login, Accessing Emails, Sending Emails, Sending Attachments, Searching, Replying or Forwarding Emails.

Non-functional feautures performance, relaibility, readbility and software quality.

### Out-of-Scope

Out-Of-Scope defines the features, functional or non-functional requirements of the software that **will NOT be** tested which includes interfaces and security.

## Quality Objective

The objective of testing is to ensure functionality of WebMail MSU.

* Ensure the Application Under Test conforms to functional and non-functional requirements
* Ensure the AUT meets the quality specifications defined by the client
* Bugs/issues are identified and fixed before go live

## Roles and Responsibilities

The roles and responsibilities in the testing are:

* Developers- Implements test cases for acceptance and CI testing.
* Installation Team- Install required applications to implement such test cases.
* Test Manager- Manages the quality of the project, define the project direction, secure resources.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Net ID** | **GitHub username** | **Role** |
| Khem Dhami | Kbd167 | Kbd167 | Developer/ Test Manager |
| Joshua Moore | Jjm702 | joshuamoorexyz | Developer/Test Manager |
| Rushma Parajuli | Rp1349 | Rp1349 | Installation/ Test Manager |
|  |  |  |  |

# Test Methodology

## Overview

1. We as the memers of the group followed the waterfall method which allowed us to create a sequential development process that flows like a waterfall through a phase of a project (analysis, design, development, and testing).

Top of Form

## Test Levels

**Test Levels define the Types of Testing (Unit Test, Integration Test, etc…) to be executed on the Application Under Test (AUT**). The Testing Levels primarily depends on the scope of the project, time and budget constraints. Levels of testing performed are: - Requirements - Design - Implementation - Verification – maintenance.

## Bug Triage

The goal of the triage is to

* To define the type of resolution for each bug
* To prioritize bugs and determine a schedule for all “To Be Fixed Bugs’.

## Suspension Criteria and Resumption Requirements

Since, we were following the strict schedule, if one of us reports the test case failure, the whole members resumed or continued to fix the failed cases after it has been suspended.

## Test Completeness

The test will be considered completed when we get a 100% success rate, 100% automated test cases and if all the bugs are fixed.

# Test Deliverables

The test deliverables that will be delivered throughout the testing phases are as follows:

Post-testing

* Before Testing:

Test plan

Test cases

* During Testing:

Testing Tools

Execution Logs

Error

Data Collection

* Post-testing:

Results

Bug Report

Release notes

Customer Sign Off

# Resource & Environment Needs

## Testing Tools

Make a list of Tools like

The tools required to test the project are as follows:

* Web server
* Internet Connection
* Requirements Tracking Tool
* Bug Tracking Tool
* Automation Tools

## Test Environment

It mentions the minimum **hardware** requirements that will be used to test the Application.

The following **software’s** are required in addition to client-specific software.

* Ubuntu-latest or Windows-latest for clients.
* Installation of Python and Flask IDEs with Pytest extensions.

# Terms/Acronyms

Make a mention of any terms or acronyms used in the project

| TERM/ACRONYM | DEFINITION |
| --- | --- |
| API | Application Program Interface |
| AUT | Application Under Test |
| IDE | Integrated Development Environment |