CMSC 495
Group- Team Screenshot

Test Plan and User's Guide

Project – Screenshot Capturing System

Instructor: Shanna Kuchenbecker

Group Members

- ➢ Ronald Diggs
- ➢ Evan Troutt
- ➢ Jorge NavedoCabrera
- ➢ Ravinder Miani

Date: 12/05/2021

Contents

Test Plan

Software Test Plan

1.1 Introduction

The purpose of the Screenshot Capturing System project is to develop a program that will implement multiple features to see success at the end of the semester. These features include the ability to select which websites we are capturing screenshots for, how often we are capturing screenshots, sending HTML based emails to multiple users, add/edit/delete users, error logging, and maintenance of history and archives.

1.1.1 Objective

The purpose of the document is to detail the test plan based on requirements and functionality of the Screenshot Capturing System.

1.1.2 Background

The Screenshot Capturing team decided to create a system that will hold websites accountable when changes were implemented on their site. It will also give the user a history log that can than be shared or stored for history purposes.

1.1.3 Testing Strategy

Validate and invalidate the user inputs, edit, save and update functionalities and verify the program output for each scenario as per requirements.

1.1.4 Scope

This test plan covers the required functions of the software:

- Test and validate login information

- Test and validate creating a new Screenshot
- Test and validate updating a Screenshot
- Test and validate utilizing the Capture function
- Test and validate utilizing the History View

1.1.5 Intended Audience

This document is intended to provide the test plan for QA testers and Managers.

1.1.6 Definitions and Acronyms

Acronym / Term	Definition

1.1.7 Reference

Reference Documents	Document Name
Screenshot_Capture_Project Plan_Specification_V1.docx	Screenshot Capturing System Project Plan & Requirements Specifications
IEEE. IEEE Std. 829-1998 IEEE Recommended Practice for Software Test Documentation. IEEE Computer Society, 1998.	Recommended Practice for Software Test Documentation

1.2 Test Items

1.2.1 Program Modules

The modules to be tested:

- Login Module
- Admin Module
- Subscribe/unsubscribe to website
- Send emails to users
- Log errors/exceptions

1.2.2 User Procedures

Tested the usability of the Screenshot Capturing system as outlined in the SRS and BRS

1.3 Features to be tested

- Login Module
- Admin Module
- Subscribe/unsubscribe to website
- Send emails to users
- Log errors/exceptions

1.4 Features not to be tested

Not applicable

1.5 Approach

Test personnel will prepare tests based off of the information provided in the SRS and BRS

- **Conversion testing**
  - ➢ The test plan member will develop input and output for the expected scenarios.
- **Black box Testing**
  - ➢ A method of software testing which seeks to ensure that the program operates as intended through expected outputs; functional
- **White box Testing**
  - ➢ A method of software testing which seeks to execute as many branches of code as possible in order to obtain maximum code coverage; non-functional
- **Component Testing**
  - ➢ The testing of a component of a software program
- **System Testing**
  - ➢ The functional testing of a software program that is fully integrated and complete
- **Job Stream testing**

- ➢ A generated set of expected user and ticket information should be used to test each bound of each module
- • **Interface testing**
  - ➢ Internet connectivity for web application and database access will be tested for system dependability.
- • **Security testing**
  - ➢ Security requirements will be tested to validate the security features of the application.
- • **Recovery testing**
  - ➢ Recovery testing will be done by halting the system and using appropriate recovery procedures
- • **Performance testing**
  - ➢ Performance of the application will be tested for Response time of execution of the application
- • **Regression**
  - ➢ End to end testing of the application for each modules of the application
- • **Comprehensiveness**
  - ➢ Each feature will be compared to the BRS and SRS
- • **Constraints**
  - ➢ Final version will be tested by the Teams

1.6 Pass/Fail criteria

- ➢ Pass/ fail will be determined by whether or not the system sends the appropriate error messages if system constraints are broken.
- ➢ Pass/ fail if the system does not break constraints it will continue to run
- ➢ At the end of each module, the client has the ability to continue or exit based on whether the proper Screenshot was captured

1.7 Suspension and Regression Criteria

- **Suspension Criteria**

  - ➢ At this time the only perceivable suspension will be determined by the team.

- **Resumption Criteria**

  - ➢ If the development is suspended by the team, they will determine resumption.

1.8 Test Deliverables

- **Test documentation**

  - ➢ Test plan
  - ➢ Test cases
  - ➢ Test logs
  - ➢ Test report
  - ➢ Test summary

- **Test data**

  - ➢ Copies of all inputs will be recorded
  - ➢ Copies of all expected and actual outputs will be recorded

User's Guide

User's Guide

1.9 Introduction

The Screenshot Capturing System is a new, self-contained program that will get deployed to user workstations and will interface with a website to capture the screenshot periodically and store information on user workstations. Product users will be able to manage screenshots and websites configuration by other team members.

1.9.1 Purpose

The purpose of the document is to detail the test plan based on requirements and functionality of the Screenshot Capturing Software Application. The purpose of this User's Guide is to provide general information and instructions to users regarding the use of the Screenshot Capturing Software. This guide contains all essential information for the user to make full functional use of the Screenshot Capturing Software.

1.9.2 Objectives

The objective of this User's Guide is to present the material user will need to access and use the application. This guide includes a description of the system functions and capabilities,

contingencies and step-by-step instructions to perform specific tasks

1.9.3 Background

This Screenshot Capture team has decided to create the Screenshot Capturing system to help audit websites when changes are made.

1.9.4 Scope

This User's Guide covers the following functions of the software:

- Instructions and Procedures to login in the application
- Instructions and Procedures to create a new Screenshot
- Instructions and Procedures to update a Screenshot
- Instructions and Procedures to utilize the Capture software
- Instructions and Procedures to utilize the History log
- Instructions and Procedures to sync changes from Offline Mode to Online Mode

1.9.5 Intended Audience

This document is intended to provide the User's Guide for application users and clients.

1.9.6 Definitions and Acronyms

Acronym / Term	Definition

1.9.7 Reference

Reference Documents	Document Name
Screenshot_Capture_Business_Requirement_Specification_V1	Screenshot Capture Business Requirement Specification
Screenshot_Capture_Software_Requirement_Specification_V1	Screenshot Capture Software Requirement Specification
Screenshot_Capture_Software_Test_Plan_V1	Screenshot Capture Software Test Plan
IEEE. IEEE Std. 829-1998 IEEE Recommended Practice for User's Guide Documentation. IEEE Computer Society, 1998.	Recommended Practice for User's Guide Documentation

1.10   System Requirements

1.10.1   Hardware

➢ Window / Mac / Linux computer

1.10.2   Software

- ➢ **Operating system**
  - • Windows 10, Mac OS, or Linux
- ➢ **Communication**
  - • Internet connection
- ➢ **Java Runtime Environment 11**
- ➢ **Java SE Development Kit 11 which can be download from the following link:** http://www.oracle.com/technetwork/java/javase/downloads/index.html
- ➢ **Maven**
- ➢ **MongoDB**
- ➢ **Chrome web browser**

1.11   Instructions and Procedures

1.11.1 How to Set up Environment for the Software

1. Install Maven.
2. Install MongoDB.
3. Download Chromedriver for Selenium. Make sure you check the version of Chrome you have installed, and get that same version of Chromedriver.
4. Add Chromedriver to your PATH, this will vary by what OS you use.
5. Assuming you're using gmail, google how to set up an application password for it. This is necessary for sending email from the application.
6. Update src/main/resources/application.yml:
   - admin should be your email address

   - password can be anything (these are for logging in as an admin in the application)

   - smtp:address should be your email address(has to be gmail)

   - smtp:password should be the gmail application password you set up

   - spring:resources:static-locations should be: file:/PROJECTPATH/src/main/resources, where PROJECTPATH is the absolute path of the project

7. Make a folder named mongo at the top level of the project.
8. Make a folder named captures in the src/main/resources directory.
9. open command line/terminal, run: mongod --dbpath /PROJECTPATH/mongo
10. In another window or tab, run: mvn spring-boot:run

1.11.2 How to log in to the software

1. Point your web browser to localhost:8080.

2. Create a new user, using the button.

3. Log in using those user credentials.

1.11.2 Screen Capturing

1. After creating a user and subscribing to a site, log out.

2. Log back in as admin, using the credentials you defined earlier in application.yml.

3. The application will start automatically opening a browser window and taking screenshots of the subscribed sites. It will do this every 1 minute.

Change Log

Date	Version	Description of Change	Item(s) Changed	Requested By
11/01/2021	V0	Created the baseline and Initial draft	Baseline and Initial Draft	Group
11/01/2021	V1	Modified the format and added the cover page for uniformity	Format Change and Added Cover page	Group
11/07/2021	V2	Modified the format of Table of content	Format Change for table of content	Group
	V3	Minor formatting changes	Sections 5 and 6	Group
	V4	Added Phase 1 Report	Phase 1 Report	Group
12/05/2021	V5	Added Phase 3 Report	Scope, Program Modules, Features to be Tested, System Requirements, Software, Instructions and Procedures	Evan Troutt