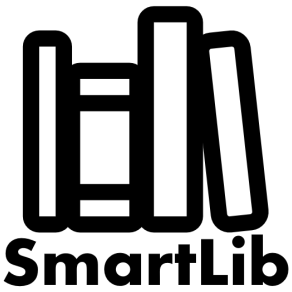

TEST PLAN

Version *1.2*
1/4/2021

Team Xeon

Document Number: SL-11

REVISION HISTORY

Version	Date	Author	Description of Changes	Approved by
1.0	**18/03/2021**	**Boon Shuan**	**Initial Draft**	**Kenny Voo Tze Rung**
1.1	**25/04/2021**	**Wilson**	**Wrote section 1-3**	**Kenny Voo Tze Rung**
1.2	**1/04/2021**	**Kenny**	**Added appendices**	**Kenny Voo Tze Rung**

TABLE OF CONTENTS

1 Test Plan Identifier

The Test Plan Identifier is a unique identifier for the Test related documents. There are 3 types of Test Plan available:

- Master Test Plan - a standalone high-level test plan for a project that combines all other test plans
- Testing Level Specific Test Plan - includes Unit, Integration and System Test Plan
- Testing Type Specific Test Plans - includes Regression Test Plan

2 Introduction

This document outlines the methods and strategies for testing our web application, *SmartLib*. For this test plan, we aim to accomplish the following objectives:

- Identify the features of the software to be tested
- Describe the testing strategies to be implemented and the criteria to determine whether test cases pass or fail
- Identify suspension criteria and resumption requirements
- Identify the required resources and provide an estimate for the test efforts

We will apply testing techniques to 3 levels - Unit Testing, Integration Testing and System Testing. Testing techniques used will be ***Black Box Testing*** and ***White Box Testing.***

3 Test Items

The items to be tested are:

- Software Requirement Specification
- Quality Plan
- Web application, *SmartLib*
    - The following section will identify the functions of *SmartLib* to be tested.

4 Features To Be Tested

Function	Description
Login	1. Users with existing accounts must be able to login successfully.
Add new seat	1. Administrators must be allowed to add new seat into the database.
View seat status	1. Seat status must be updated and retrieve the latest seat status
Image Detection	1. Image detection algorithm must be accurate in detecting 3 different seat statuses: Empty, Occupied and Hogged
Booking System	1. Users must be able to book their seat, selected from those that are empty.

5 Features Not To Be Tested

All features are required to be tested.

6 Approach

6.1 Techniques

6.1.1 White-box/Structural Testing

Control-Flow Testing technique will be used to test the navigation of the web application.
Sets of predefined paths will be determined in the test for the tester to navigate and identify errors found.

We will also use Data Flow testing to determine how test data inputs will flow through the system into the database, and vice versa. We will also ensure that all data requested by the user

are correctly retrieved and displayed.

6.1.2 Regression Testing

Testing will be executed at every change such as bug fixes or feature updates. The tests which identified bugs in the previous execution will be given higher priority to verify if these bugs are fixed. All tests will be executed again to ensure that no features are affected in the new update.

These tests are also to be automated to allow for efficient resource and time utilisation, as they will be executed multiple times during the development lifecycle.

6.2 Objectives

Test objective	Method
Database must be accessed without corruption.	Attempt every function that requires access to the database. After each function call, check the database manually to ensure data is correctly recorded/altered and unrelated data are untouched.
Data displayed must be correct throughout the navigation of the application.	Design sets of navigation tests which covers all of the application functions, and list down the expected results. Execute the test sets and check expected results with the actual output.

7 Item Pass/Fail Criteria

This segment identifies the requirements for each component testing.

Features	Pass Criteria	Fail Criteria
Login	The users can successfully log in with their registered account	The users are unable to log in with their registered account
Add and remove seats	The admin must be able to add and remove seats from the database	The admin is unable to add or remove seats, or the change is not reflected in the UI
View Seat status	Correct details of the seat statuses are retrieved from the database and displayed	Incorrect details of the seat statuses are retrieved from the database and displayed
Image Detection	The camera is able to correctly draw bounding boxes around detected objects and classify them correctly with 90% threshold	The camera is detects objects incorrectly
Booking System	a) The users are able to select only empty seats and reserve them for 15 mins b) The admin must be notified if seats are determined to be 'hogged' for more than 60 mins	a) The users are unable to select or reserve any empty seats b) The admin is not notified after seats have been 'hogged' for more than 60 mins

7.1 Database Access

- Verify accurate writing of data into the database.
- Verify simultaneous access to database.
- Verify accurate retrieval of data from the database.

7.2 Functions Testing

- Ensure that inputs from users are processed correctly.
- Verify that image detection; bounding boxes and classification are calculated accurately.
- Verify that the functions accessing the same type of data have the same format.

7.3 User Interface Testing

- Ensure ease of navigation through all of the screens.
- Verify that system is user-friendly for the targeted audience of the application.
- Ensure that the user-interface is intuitive

7.4 Performance Testing

- Verify the response time of retrieval of data from the database is below 1 second.
- Verify the response time of writing of data into the database is below 1 second.

8 Suspension Criteria And Resumption Requirements

8.1 Suspension Criteria

If a particular test case fails, testing for dependent features on SmartLib will be suspended. The failed test case will be logged into the test log, with initial findings and conditions that cause the failure.

8.2 Resumption Requirements

Testing for the remaining features will resume after the root cause of the failed test has been identified and resolved. Test cases testing non-dependent features will be executed in parallel during the debugging process of the failed test.

9 Test Deliverables

Deliverables	By	Deadline
Test plan	Teo Boon Shuan	30 March 2021
Test environment	Kong Hou Jing	31 March 2021
Test data set	Wilson Tai	1 April 2021
Test script	Teo Boon Shuan	1 April 2021
Test logs	Kenny Voo Tze Rung Zhou ZeYu Muhammad Irsyad Bin Redzuan	2 April 2021
Test evaluation report	Kenny Voo Tze Rung Zhou ZeYu Muhammad Irsyad Bin Redzuan	2 April 2021

10 Testing Tasks

10.1 Plan Test

a) Identify requirements for tests
b) Assess risks
c) Develop test strategy
d) Identify test resources
e) Create schedule
f) Generate test plan

10.2 Design Test

a) Workload Analysis
b) Identify and describe test cases
c) Identify and structure test scripts
d) Identify test completion criteria
e) Review and access test coverage

10.3 Implement Test

a) Setup Test Environment
b) Record or Program Test Scripts
c) Identify Test-Specific functionality in the design and implementation model
d) Establish External Data sets

10.4 Execute Test

a) Execute Test Scripts
b) Evaluate Execution of Test
c) Recover from Halted Test
d) Verify the results
e) Investigate Unexpected Results
f) Log Defects

10.5 Evaluate Test

a) Evaluate Test-case coverage
b) Evaluate code coverage
c) Analyze defects
d) Determine if Test Completion Criteria and Success Criteria have been achieved
e) Create Test Evaluation Report

11 Environmental Needs

11.1 System And Hardware

- Personal computer with Javascript-compatible Internet browser
- Firebase server
- Access to internet connection

11.2 Documentation

- Reference documents
- Test set designs
- Bug reporting tools

11.3 Personnel

- System admins
- Testers
- Developers

11.4 Setup

To test that the application works, various tablets with different screen sizes are set up. The tablets are required to have the following browsers installed:

- Chrome
- Firefox
- Safari (iOS)
- Android Browser(older Android models)

12 Responsibilities

Role	**Name of personnel**	**Tasks**
Test Manager	Teo Boon Shuan	- Provides management oversight - Sets goals and approach - Acquire required resources
Test Designer	Kenny Voo	- Design testing procedures - Generate test plan - Evaluate effectiveness of test effort
Database tester	Wilson Tai Zhou Zeyu	- Execute tests on database - Log results - Provide rectifying measures if required
Function tester	Kong Hou Jing Muhammad Irsyad Bin Redzuan	- Execute tests on functionality - Log results - Provide rectifying measures if required
Database administrator	Wilson Tai	- Ensure integrity of database during testing - Administer test data

13 Staffing And Training Needs

The following staffing is expect for the test plan

- 1x Quality Assurance Manager
- 1x Quality Assurance Engineer
- 1x Developer

The Quality Assurance Manager and Quality Assurance Engineer needs to be familiar with both the *Test Plan* Documentation and *Test Case and Requirements Test Coverage* Report. The Developer.

The Developer has to be Familiar with the Software System and detailed knowledge of individual components of SmartLib.

14 Schedule

Task	Estimated time taken	Deadline
Design tests	2 day	30/3/2020
Implement tests	1 day	1/4/2020
Execute tests	2-3 days	2/4/2020
Evaluate tests/Fix bugs	6 days	5/4/2020
Re-run tests	2 days	10/4/2020
Documentation	2 days	12/4/2020

15 Risks And Contingencies

15.1 Insufficient Resources

Lack of resources or absence of personnel during the testing phase.

Contingency:

- The team will work overtime if necessary.
- Request for an extension of deadline.
- Resources will be optimized during testing.
- Testing will be outsourced to external personnel.

15.2 Changes To Product

Emergency changes to designs or requirements of the product may affect test designs.

Contingency:

- Number of acceptable defects will be increased.
- Scope of the plan will change to fit the new requirements.

15.3 Insufficient Time

There may not be enough time to clear all tests implemented.

Contingency:

- Number of tests performed will be reduced.
- Functions with higher importance will be tested with higher priority.

16 Approvals

Project Manager is in charge of approving the tests designed and implemented by Quality Manager. Once the tests are approved, test implementation is conducted by the Quality Engineers and the developers. They will have to report to the Quality Manager with documentation on the tests. The Project Manager is then updated with the progress of the testing.