Table of Contents

1. Test Plan Identifier

Identifier	TP_2023_001
Name	Shop Surveillance System
Version	1.0
Date	May 23, 2023
Authors	Muhammad Azfar Iqbal
Contact Information	abc@gmail.com
Revision History	Version 1.0 Initial Version

Shop Surveillance System Test Plan V_1.0

2. Introduction

This is the Master Test Plan for our Final Year Project. Its purpose is to mention the tools and the techniques that we are going to use throughout the testing phase of the software that is currently in development. This document will be helpful in identifying the resources required for testing the software and also the time required to completely test the software.

The objectives of this document include:

- To mention the tools and techniques required for testing.
- To mention the features to be tested and not to be tested.
- To define the environmental constraints and needs.
- To mention the responsibilities and risks included while conducting testing.

2.1 References

Documents that support this test plan include:

- Scope for Shop Surveillance System.
- Software Requirement Specification (SRS) Document for Shop Surveillance System.
- Software Design Specification (SDS) Document for Shop Surveillance System.

2.2 Intended Audience

This test plan document is intended for the development team, quality assurance team, project stakeholders such as customers, employees, and shop owner. The development team will use this document as a guideline for testing and ensuring the quality of the system. The quality assurance team will execute the test cases and report any issues. Project stakeholders will review the test plan to gain confidence in the product's quality. Potential users will benefit from a thoroughly tested and bug-free surveillance system experience.

3. Executive Summary

The shop surveillance system is developed to minimize any robbery or theft at the shop. Nowadays many surveillance systems have been developed, some have good functionalities but bad design. Some have good user interface design but bad functionalities. The system is going to resolve the problem of detecting theft or robbery occurring outside the shop at any time of the day. The system will immediately send an alert if it detects some suspicious activity. The main objective of the system is to automatically detect activities related to robbery, shoplifting, or theft and generate notifications. Another is to provide real time surveillance through monitoring instead of a person keeping an eye on the footage. The system will provide real time surveillance so that no one can escape after any suspicious activity occurs.

Requirements Specifications

The Shop Surveillance System will undergo thorough testing to verify that all the functional requirements outlined in the SRS document have been implemented correctly. Test cases will be designed to cover all specified features and ensure their proper functioning.

Integration Testing

After the individual modules of the Shop Surveillance System have been tested, integration testing will be conducted to verify that they work seamlessly together without any defects or issues. This will ensure that the overall system functions are as expected and delivers a cohesive surveillance environment.

4. Test Items

The items to be tested include the frontend that is intended for the user and the backend that is concerned with the proper working of the frontend. It will also include the testing of the accuracy of trained models of Weapon detection as well as Intruder Trespassing Detection.

The overall testing will be conducted on Web.

5. Features to be Tested

No.	Feature	Risk Level
1.	As a user, connecting the camera to the system.	L
2.	As a user, disconnecting the camera from the application	L
3.	As a user, having a good hardware system to load the model.	H
4.	As a user, having a good internet connectivity.	H
5.	As a user, having a good lighting in which trespassing intruder can be easily detected.	H
6.	As a user, checking for possibilities of intruder trespassing detection that are trained on the dataset.	M
7.	As a user, checking for all the weapons that are trained on the dataset.	M
8.	As a user, checking notification is sent in case of weapon detection.	M
9.	As a user, checking whether portal notifications are working or not.	L
10.	As a user, viewing their respective dashboards.	L
11.	Checking the connectivity of the camera with the application.	H
12.	Checking the signal strength of the internet.	M

6. Features not to be Tested

This document includes no features of software that will not be tested within the scope of this document. It is expected to test all the features of the application.

7. Approach

All the above mentioned items will be tested using automatic testing tools. The backend of the software will be tested mainly using testing tools known as PyTest and PyCharm. The frontend functionalities will be tested using KARMA and JEST.

The metrics to determine the test status will be PASS, FAIL or IN PROGRESS. In case of FAIL, the tester will mention the possible reason as to why the test failed. In case of IN PROGRESS, the tester will have to provide an explanation for not completing the test.

After all the manual and automated tests are completed up to a certain acceptable level, a test report will be generated which will be then approved by the supervisor and the committee.

7.1 Testing Levels

7.1.1 Unit Testing

Unit testing will be applied to every individual module and component in the system. The objective is to test all the functions or pieces of code that can be tested independently of other functionalities or code components.

7.1.2 Integration Testing

Integration testing will be applied to those modules of the system that interact or communicate with each other. It focuses on testing modules that exchange data with one another.

7.1.3 System Testing

System testing will be applied to the complete and fully integrated system of the Web based Shop Surveillance System.

7.1.4 Acceptance Testing

Acceptance testing will focus on testing the overall functioning and quality of the Web based Shop Surveillance System. It aims to verify that the system conforms to the specified requirements.

7.2 Testing Types

7.2.1 Functional Testing

Functional testing will verify that all the core functions and features of the Web based Shop Surveillance System are working as intended.

7.2.2 Usability Testing

Usability testing will evaluate the user interface, user experience, and ease of use of the Web based Shop Surveillance System. This involves testing the application's navigation, workflows, input validation, and overall user interaction to ensure an excellent and usable system.

7.2.3 Performance Testing

Performance testing will measure the performance and responsiveness of the Web based Shop Surveillance System under different loads and stress conditions. This includes testing its ability to handle maximum resource utilization.

7.2.4 Integration Testing

Integration testing will verify the seamless integration of the Web based Shop Surveillance System with their associative components and technologies, and any other necessary third-party services. This ensures the proper functioning of interoperability.

7.2.5 Cross-Platform Testing

Cross-platform testing will involve testing the Web based Shop Surveillance System on different platforms and browsers to ensure compatibility and consistent functionality across various environments.

7.3 Testing Technique

For black box testing, equivalence class partitioning technique will be used to ensure comprehensive coverage of different input scenarios. For white box testing, decision coverage technique will be employed to test different decision outcomes and code paths within the system.

7.4 Meetings

The test team will meet once every two weeks to evaluate progress, identify trends, and address any issues or challenges that arise during the testing process. The test team leader will also meet with the project manager once every two weeks to provide updates and align testing activities with the project's overall timeline and objectives. Additional meetings can be scheduled as required for any emergency situations or critical decision-making needs.

7.5 Metrics Collection

The testing team will collect the following information during all testing phases:

- ✓ Defects by module and severity, to track and prioritize the resolution of identified issues.
- ✓ Defect origin (requirement, design, code), to identify the root causes of defects and take appropriate corrective actions.
- ✓ Time spent on defect investigation, to assess the overall efficiency of defect resolution processes and identify areas for improvement.
- ✓ Number of times a program is submitted to test as ready, to track the progress and readiness of the system for testing.
- ✓ Defects located at higher levels that should have been caught at lower levels, to identify any gaps in the testing process and take preventive measures.

8. Items Pass/Fail Criteria

The passing and failing of a test item depends on what risk level is associated with it. For a test item that has risk level as high, the passing criteria must be 100 percent as it is the core functionality of the project. For a test item that has a risk level as medium, the passing criteria must be 80 percent as it is an important feature but not as important as a test item having risk level as high. For a test item that has a risk level as low, the passing criteria must be 65 percent as it is not an important feature but only to make the look and working of the software more advanced.

All core functionality of the systems should function as expected and outlined in the individual test cases. There must be no critical defects found. 90 percent of all test cases should pass and no failed cases should be crucial to the end-user’s ability to use the application.

9. Suspension Criteria and Resumption Requirements

The suspension of testing in our system at any stage will be carried out whenever the hardware faces any issues regarding the connectivity with the application.

Moreover, the testing might be suspended when the environment required for our application to execute at its fullest is not suitable and not provided during testing. Another reason to suspend the testing will be when 20 percent of test cases will be failed.

In case of suspension due to any of the above mentioned issues, the testing will have to be carried out again from the point where the issues occurred. In case of test case failures, the test cases will be rewritten according to new conditions and specifications.

10. Test Deliverables

The following test documents will be delivered after successful completion and evaluation of this document:

- Test Plan Document.
- Test Traceability Matrix.
- Test Case Specifications Document.
- Test Case Data.
- Test Logs.
- Test Incident Reports.
- Test Summary Reports.
- Test Requirements Document.
- Software Manual.

11. Test Tasks

The following activities must be completed:

- Test plan prepared.
- Functional specifications written and delivered to the testing team.
- Environment should be ready for testing (test data etc.).
- Perform the tests.
- Prepare the test summary report.

12. Environmental Needs

No.	Environmental Need	Version	Purpose
1.	Mocha	10.2.0	For Node.js programs, featuring browser support, asynchronous testing
2.	PyTest	6.2.5	For machine learning testing
3.	PyCharm	2021.3	For machine learning testing
4.	JEST	27.4.5	For white-box testing
5.	KARMA	6.3.9	For white-box testing
6.	MS Word	2016	Documenting the testing phase

13. Responsibilities

Tasks	Azfar	Sarmad	Project Supervisor	FYP Committee
Test Plan Documentation		✓	✓	✓
Test Case Documentation	✓		✓	✓
Test Execution Document	✓	✓	✓	✓
Test Logs	✓		✓	✓
Error Report	✓	✓	✓	✓
Manual Creation	✓		✓	✓
Test Case Specification Documentation		✓	✓	✓

14. Testing and Evaluation

14.1 Unit Testing

Unit Testing 1: Verify creation of user type

Testing Objective: To ensure that correct user type is created

Test Case Id: TC_01

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify that the user type is created successfully.	User type = "Shop Owner"	User type "Shop Owner" is created successfully	Pass
2	Verify that the user type is created successfully.	User type = "Employee"	User type "Employee" is created successfully	Pass

Unit Testing 2: User account creation

Testing Objective: To ensure that correct user account is created.

Test Case Id: TC_02

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify that a user account is created successfully.	Full Name = "Muhammad Azfar", Email = "azfar1@gmail.com", User Type = "Employee", Password = "abc123"	User account for Muhammad Azfar is created successfully.	Pass

2	Verify that an invalid email address is not accepted.	Full Name = "Syed Danish", Email = "Danish", User Type = "Shop Owner", Password = "abc123"	Error message is displayed indicating an invalid email address.	Pass

Unit Testing 3: View User Account

Testing Objective: To ensure that correct user account information is displayed.

Test Case Id: TC_03

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify that the user accounts are displayed correctly.	User Account Data (Full Name, Email, User Type) = [Muhammad Azfar, azfar1@gmail.com, Employee], [Syed Danish, danish@gmail.com, Shop Owner]	User accounts for Muhammad Azfar and Syed Danish are displayed correctly.	Pass

Unit Testing 4: Edit User Account

Testing Objective: To ensure that edited data is properly saved and executed.

Test Case Id: TC_04

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify that the user account details are pre-filled correctly.	(Full Name, Email, User Type, Password) = [Muhammad Azfar, azfar1@gmail.com, Employee, abc123]	The Full Name, Email, User Type, and password fields are pre-filled with the corresponding user account details.	Pass
2	Verify that the user account is updated successfully.	(Full Name, Email, User Type, Password) = [Muhammad Danish, danish@gmail.com, ShopOwner, abc123]	User account details for Syed Danish are updated to Muhammad Danish successfully	Pass

14.2 Functional Testing

Functional Testing 1: Check deletion and editing of user type

Objective: To ensure that the deletion and editing user type functionality works

Test Case Id: TC_01

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Verify that a user type can be deleted.	User type = "Shop Owner"	User type "Shop Owner" is deleted successfully.	User type "Shop Owner" is deleted successfully.	Pass
2.	Verify that a user type can be edited.	User type = "Employee"	User type "Employee" is updated successfully.	User type "Employee" is updated successfully.	Pass
3.	Verify that an invalid user type cannot be created.	User type = "Invalid User Type"	Error message is displayed indicating invalid user type.	Error message is displayed indicating invalid user type.	Pass
4.	Verify that deleting a user type removes associated user records.	User type = "Shop Owner"	All user records associated with the "Shop Owner" user type are deleted.	All user records associated with the "Shop Owner" user type are deleted.	Pass

Functional Testing 2: Verify successful account creation.

Objective: To ensure that the deletion and editing user type functionality works

Test Case Id: TC_02

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Verify that the "Create User Account" button is disabled until all required fields are filled.	Full Name = "", Email = "azfar1@gmail.com", User Type = "Employee", Password = "abc123"	The "Create User Account" button remains disabled until a Full name is entered.	The "Create User Account" button remains disabled until a Full name is entered.	Pass
2.	Verify that the user account creation is successful with all valid input data.	Full Name = "Muhammad Azfar", Email = "azfar1@gmail.com", User Type = "Employee", Password = "abc123"	User account for Muhammad Azfar is created successfully.	User account for Muhammad Azfar is created successfully.	Pass
3.	Verify that the password meets the required complexity criteria.	Full Name = "Muhammad Azfar", Email = "azfar1@gmail.com", User Type = "Employee", Password = "aaaaaa"	Error message is displayed indicating that the password does not meet the complexity requirements	Error message is displayed indicating that the password does not meet the complexity requirements	Pass

Functional Testing 3: Verify delete and edit user account functions.

Objective: To ensure that the delete and edit functions are valid.

Test Case Id: TC_03

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Verify that a user account can be deleted	User Account Data = [Muhammad Azfar, azfar1@gmail.com, Employee]	User account for Muhammad Azfar is deleted successfully	User account for Muhammad Azfar is deleted successfully	Pass
2.	Verify that a user account can be edited.	User Account Data = [Syed Danish, danish@gmail.com, Shop Owner]	User account details for Syed Danish is updated successfully.	User account details for Syed Danish is updated successfully.	Pass
3.	Verify that deleting a user account prompts for confirmation.	User Account Data = [Muhammad Azfar, azfar1@gmail.com, Employee]	Confirmation dialog is displayed before deleting the user account	Confirmation dialog is displayed before deleting the user account	Pass
4.	Verify that editing a user account redirects to the "Edit User Account" screen.	User Account Data = [Syed Danish, danish@gmail.com, Shop Owner]	Clicking on the "Edit" action for Syed Danish's user account opens the "Edit User Account" screen with pre-filled data.	Clicking on the "Edit" action for Syed Danish's user account opens the "Edit User Account" screen with pre-filled data.	Pass

Functional Testing 4: Verify edit account details.

Objective: To ensure that the account is edited successfully.

Test Case Id: TC_04

No.	Test case/Test script	Attribute and value	Expected result	Actual result	Result
1.	Verify that the user account can be updated with a new password.	(Full Name, Email, User Type, Password) = [Muhammad Azfar, azfar1@gmail.com, Employee, xyz123]	User account details for Muhammad Azfar, including the password, are updated successfully.	User account details for Muhammad Azfar, including the password, are updated successfully.	Pass
2.	Verify that an error message is displayed when attempting to update the user account with incomplete or invalid data.	(Full Name, Email, User Type, Password) = [Syed Danish, “ ”, ShopOwner, abc123]	Error message is displayed indicating that a required field (Email) is missing or invalid, preventing the update of the user account.	Error message is displayed indicating that a required field (Email) is missing or invalid, preventing the update of the user account.	Pass
3.	Verify that the "Update" button is enabled and functional when all required fields are filled with valid data.	(Full Name, Email, User Type, Password) = [Syed Danish, “danish11@gmail.com”, ShopOwner, abc123]	The "Update" button is enabled and clicking it successfully updates the user account with the modified details.	The "Update" button is enabled and clicking it successfully updates the user account with the modified details.	Pass

14.3 Integration Testing

Integration Testing 1: Verify the integration between Admin Management and User Management Modules.

Testing Objective: To ensure correct user type assignment.

Test Case Id: TC_01

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify the integration between Admin Management and User Management Modules for user type assignment.	User Account Data (Full Name, Email, User Type) = [Muhammad Azfar, azfar1@gmail.com, Employee], [Syed Danish, danish@gmail.com, Shop Owner]	The assigned user role is reflected in the User Management Module, granting appropriate access and permissions to the user.	Pass
2	Verify the integration between Admin Management and User Management Modules for user account deletion.	User Account Data (Full Name, Email, User Type) = [Muhammad Azfar, azfar1@gmail.com, Employee]	The deleted user account is removed from the User Management Module, ensuring the user can no longer access the system.	Pass

Integration Testing 2: User Management and Notifications Management Modules

Testing Objective: Verify the integration between User Management and Notifications Management Modules for user notification settings.

Test Case Id: TC_02

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify the integration between User Management and Notifications Management Modules for user notification settings.	User Account Data (Full Name, Email, Select notifications	The updated notification preferences are reflected in the Notifications Management Module, ensuring the user receives notifications according to their selected settings.	Pass

Integration Testing 3: Trespassing Detection and Notifications Management Modules

Testing Objective: Verify the integration between Trespassing Detection and Notifications Management Modules for real-time alert notifications.

Test Case Id: TC_03

No.	Test case/Test script	Attribute and value	Expected result	Result
1	Verify the integration between Trespassing Detection and Notifications Management Modules for real-time alert notifications.	Detect Trespassing	The Notifications Management Module receives the alert from the Trespassing Detection Module and sends real-time notifications to the designated users or administrators.	Pass

15. Staffing and Training Needs

Critical training requirements and concerns for the Shop Surveillance System project include:

15.1 Training on the product:

Understanding system architecture, including frontend, backend, and trained models for weapon detection and intruder trespassing detection.

Familiarization with system functionalities, such as camera connectivity, hardware requirements, internet connectivity, lighting conditions, dashboard viewing, and notification mechanisms.

Knowledge of user roles and workflows to create relevant test cases.

- ✓ Integration testing will be performed on the completion of 40% and 60% of the system.
- ✓ System testing will be performed once the whole system is fully built and integration testing has been performed.
- ✓ System testing will be executed by the developers along with the project supervisor.
- ✓ The responsibility for acceptance testing will be undertaken by the project evaluation committee.

15.2 Training for test tools:

- ✓ PyTest and PyCharm training for backend testing.
- ✓ KARMA and JEST training for white-box testing of the frontend.

15.3 Concerns:

- ✓ Availability of training resources.
- ✓ Time allocation for training.
- ✓ Ongoing support and guidance.

Addressing these requirements and concerns will ensure that the testing team is proficient in testing the Shop Surveillance System effectively.

16. Schedule

Milestones	Start Date	End Date	Testing
Module 5	1/03/23	15/03/23	Manual
Module 1	16/03/23	25/03/23	Unit Testing
Module 4	26/03/23	07/04/23	Integration Testing
Module 2	08/04/23	16/04/23	Unit Testing
Module 7	17/04/23	25/04/23	Unit Testing
Module 3	26/04/22	13/05/23	Unit Testing
Module 6	14/05/23	15/05/23	Unit Testing
System	16/05/23	20/05/23	System Testing
Evaluation by Supervisor	21/05/23	22/05/23	Acceptance Testing
Evaluation by FYP Committee	24/05/23	28/05/23	Acceptance Testing

17. Risks and Contingencies

Risk Id	Risk Description	Probability	Impact	Area of Impact
RI-01	Quality camera not available in the market	High	High	Cost & Schedule
RI-02	Dependencies issue in production	Low	Low	Schedule
RI-03	Test data not available	Medium	High	Schedule
RI-04	Change of requirement during development	Medium	Low	Scope
RI-05	Modification of scope during development	Low	High	Scope

18. Approvals

The undersigned acknowledge that they have reviewed the **Shop Surveillance System Test Plan** document and agree with the approach it presents. Any changes to its requirements definition will be coordinated with and approved by the undersigned or their designated representatives.

Name:	Ms. Saneeha Amir	**Date:**	23/05/2023
Role:	Project Supervisor		
Signature:			

Name:	Mr. Inayat	**Date:**	23/05/2023
Role:	Evaluation Committee Member		
Signature:			

Name:	Ms. Najmun Nisa	**Date:**	23/05/2023
Role:	Evaluation Committee Member		
Signature:			

Name:	Dr. Nusrat Shaheen	**Date:**	23/05/2023
Role:	Evaluation Committee Member		
Signature:			