TED UNIVERSITY

TEST PLAN REPORT

CMPE 492

SENIOR DESIGN PROJECT 2

Begüm AKDENİZ 16669037806

Umut YILDIRIM 10822781650

Ceren BÜYÜKGÜLLÜ 27142510204

İrem TAMAY 59701366816

İçindekiler Tablosu

1. Introduction

Our "School Entrance with Facial Recognition System" project, which provides solutions to the security needs in educational institutions, is a technological initiative focused on ensuring the safety of students and staff. By replacing traditional access control methods with a modern approach, we aim to increase the security of our school and optimize operational efficiency. In this project, a system that can detect student faces from entrance cameras was designed. If the faces match the passports obtained through the database, the system approves and allows passage through the turnstile, and also displays the student's information on the success screen (for example, the student's department). In order for the camera to read the face, it is necessary to look at the camera from a certain distance. Our system also offers the ability to record student check-in/check-out times. Our main goal is to improve the entrance process of our school with modern technology, to accelerate the physical identity verification process by recognizing the faces of our students and staff, and to maximize student and staff security by quickly detecting potential security threats with the facial recognition system.

1.1 Purpose of the Test Plan Report

The purpose of this report is to provide an outline for managing the testing processes of the project and achieving the project goals. This makes it simple to see the project's overall going. The report was created to guide the team and ensure successful completion of the project.

1.2 Scope

The precise scope of the developing "FaceGuard" project is as follows: Accurate face recognition compared to photo. Recognition accuracy ensures that it meets the rate determined by the system. The camera can recognize the face if it is an accessory that does not cover the face too much. The turnstile opens after successful face recognition, and if the face is not recognized, the turnstile does not open. The turnstile can open the passage quickly according to the success or failure signal sent.

1.3 Features to be Tested

Face Recognition with Camera
Ability to monitor
User Interface
Security Features
Performance Tests

1.4 Testing Methodology

Unit Testing

Unit testing constitutes the most basic testing phase in our face recognition project. It is critical to verify that each unit is operating as expected. Unit tests performed for the face recognition algorithm include detection accuracy and matching accuracy. To evaluate detection accuracy, tests are performed in low light conditions and on faces captured from different angles.

System Testing

In the system testing phase of the face recognition project, the interaction between different modules is first tested. It requires a seamless interaction between modules such as facial recognition system, camera integration, database operations, access control and user interface. At this stage, it is examined in detail whether the modules work in harmony with each other. System testing also evaluates the security features of the project. Our facial recognition project must store student information securely and only be accessible by authorized users. During the system testing process, the effectiveness of these security measures is tested and the project is examined for potential security risks.

Integration Testing

First of all, integration tests between the face recognition module and the database integration module are performed. These tests check whether the facial recognition system and student information in the database have been successfully integrated. Likewise, the integration between the user interface module and other modules is also examined in detail. The user interface must work in harmony with other modules such as facial recognition system and database integration.

Performance Testing

Performance testing is a comprehensive testing process to evaluate the overall performance of the face recognition project and understand how it behaves under load on the system. This phase aims to optimize the user experience by testing the speed, response time, durability and overall processing capacity of the project. The project's capacity to handle heavy usage situations and a large number of users at the same time is also evaluated within the scope of performance testing. During busy times at school entrances, the system's response time, accuracy rate and overall performance are tested. During the performance testing process of the project, resource usage and processor performance under the system are also monitored. This is important to check whether the project works in accordance with the specified system requirements and uses hardware resources effectively.

1.5 Test Schedule

Tests	Weeks
Unit Testing	4 Weeks
System Testing	2 Weeks
Integration Testing	4 Weeks
Performance Testing	6 Weeks

1.6 Roles and Responsibilities

Responsibilities/Roles	Umut Yıldırım	İrem Tamay	Ceren Büyükgüllü	Begüm Akdeniz
Unit Testing	✓		✓	✓
System Testing		✓	✓	
Integration Testing	✓			
Performance Testing	✓	✓		✓

2. Unit Testing

2.1 Objectives

The purposes of unit testing are to ensure that the design goals of the system are met and that the project's methods work correctly in line with these goals.

These are the following as the main objectives of this test.

- Testing each module independently of each other.
- Verifying that modules react correctly to scenarios.
- Testing error conditions and limits of the project.
- Checking the accuracy of the functions and algorithms within the module.

2.2 Test Cases and Scenarios

2.2.1 Success – Face Recognition Algorithm

Test Case 1: Simple Face Recognition

Scenario: When the face of a defined person is shown to the system, the recognition algorithm should recognize it successfully and this should be considered successful entry and the turnstile should be opened.

Expected Result: Opening of the turnstile due to successful facial recognition.

Test Case 2: Error Status - Face Not Found

Scenario: A person who is not defined in the system introduces his/her face

Expected Result: The turnstiles do not open, the camera continues scanning

2.2.2 Nearest Turnstile – Turnstile Control

Test Case 3: Selection of Suitable Turnstile

Scenario: After the system finds the registered face and finds successful entry, it should assign the person to the nearest available turnstile and open the door.

Expected Result: The available and closest turnstile opens to the recognized person.

Test Case 4: All Turnstiles Occupied - No Available Turnstiles Found

Scenario: The face registered in the system logs in successfully, but there is no suitable turnstile.

Expected Result: The person's facial recognition is successful, the person is kept waiting, and the first available turnstile is opened for successful entry.

3. Integration Testing

3.1 Objectives

The purpose of integration testing is to test the creation of a complete system when the different modules included in the project are brought together and the interactions between them.

The objectives at this test,

- Checking the accuracy of data transitions between the modules that make up the system.
- Verifying that the system operates as intended when modules are put together.
- Checking that all integration points in the system are working correctly

3.2 Test Cases and Scenarios

3.2.1 Face Recognition and Turnstile Integration

Test Case 5: Face Recognition and Turnstile Control Integration

Scenario: The face of the person registered in the system must be scanned and a face recognized by the algorithm must be successfully transmitted to the control module to find an available turnstile.

Expected Result: Opening of the available and closest turnstile

Test Case 6: Error Status - Integration Error

Test Case 6: Error Status - Integration Error

Scenario: There is an error in the face recognition system algorithm and the incoming face is not recognized

Expected Result: The recognition system does not open the turnstile because the correct data is not received from the algorithm.

4. Performance Testing

4.1 Objectives

Performance tests are a form of testing used to measure and evaluate the performance of a software or system. These tests are used to evaluate speed, durability and efficiency.

4.2 Test Cases And Scenarios

Test Case 1: Face Recognition Speed

Scenario: Student approaches the line to pass through the turnstile.

Expected Result: Our system should produce a result in a maximum of 1.5 seconds to recognize a face.

Test Case 2: Accuracy Rate

Scenario: The face is recognized and then this recognition request is processed.

Expected Result: During the recognition process, users should be recognized 97% correctly. In a positive result, the patient should be directed to the tourniquet, and in a negative result, there should be no change.

Test Case 3: Simultaneous Login Request

Scenario: 8 people approach the turnstile during the busy entrance systems. The facial recognition system is initialized.

Expected Result: The system should recognize all 8 people and open the doors. After a person is assigned to a turnstile, the turnstile must wait 10 seconds and all users must pass gradually.

Test Case 4: Database Activity

Scenario: 8000 student entries are added to the database and a user recognition request is sent.

Expected Result: The system should recognize the data and match the user in a maximum of 2 seconds.

Test Case 5: System Reliability
Scenario: The system is operated for 48 hours without ever being shut down.
Expected Result: The system should be able to withstand long-term operation. No performance degradation or errors should be observed.

Test Case 6: Security Testing
Scenario: The person tried to pass by using a mask or showing a photo.
Expected Result: The system should not recognize the face and open the door. The security guards present should initiate legal proceedings with the person.

5. System Testing

5.1 Objectives

System tests are generally tests performed to evaluate the overall functioning of a system.

5.2 Test Cases And Scenarios

Test Case 1: User Registration and Database Integration
Scenario: The new student is enrolled in the school and added to the system.
Expected Result: The database is successfully added. After adding a student, it should be easily integrated into the facial recognition system and an easy transition to school should be provided.

Test Case 2: Facial Recognition and Turnstile Integration
Scenario: Student approaches the line to pass through the door.
Expected Result: The facial recognition system should be implemented successfully. If the face is recognized, the door must be opened with the signal sent from the main server.

Test Case 3: Turnstile Pass Request from Different Doors Simultaneously
Scenario: Facial recognition is performed by one student from door A and one student from door C.
Expected Result: Since the doors are different, the system should detect these doors, respond correctly to passage requests and direct each student to the turnstiles at different doors.

Test Case 4: Integration Tests
Scenario: Do the system and other dependencies work in an integrated manner?
Expected Result: The system should not cause a malfunction and work harmoniously. The internet connection should not be disconnected and should work together with a minimum error level.

6. Control Procedures

6.1 Version Control Software

Git have been used by us to oversee development-related changes and make sure that every code modification has accurate documentation. Git was our choice due to a some of the benefits it offers over other version control systems. Codes developed thanks to Gi-t can be viewed in a common area. Because of its lightweight branches, developers may easily build, switch between, and merge branches. Because of this, it's ideal for concurrent development, feature branching, and maintaining several codebase versions.

Because Git is a distributed version control system (DVCS), each developer has a local copy of the repository that is fully operational on their workstation. This makes it possible for developers to work independently and offline before syncing their changes with the primary repository. It offers faster operations and more flexibility in remote or dispersed development environments.

In addition to Git, each og us uploaded the project they developed to a shared drive file after completing their work. Thus, we kept our project and files safer against possible problems that may occur in Git.

7. Conclusion

The stages of testing performed on the facial recognition system are critically important in terms of system security, performance, accuracy and sustainability. Unit testing evaluates the basic functions of the facial recognition algorithm in detail. Integration testing, on the other hand, tests the compatibility processes of the integrated components. The Performance testing section includes performance-based tests such as the performance speed, durability and reliability of the system. System tests are tests carried out to check and ensure the compatible operation of the system. Thanks to these tests, integration within the system is carried out in the most error-free manner. These tests are supported by scenarios and the best performance is expected. As a result, these tests we carry out on the facial recognition system increase the quality of the system. It enables us to keep user satisfaction at high levels. It is also important to detect and correct possible errors early and get the best performance.