System Test Plan

For

Facial Analyzer for Rhinoplasty Operation

Team members: Priscilla Carbo, Marvin Cazeau, Jared Curtis, Maree Kelly, Alessandra Oo

Version/Author	Date
Version 1 / A.O, P.C, M.K, J.C, M.C	27 February 2022
Version 2 / A.O, P.C, M.K, J.C, M.C	15 April 2022

Table of Contents

1.   Introduction

1.1 Purpose

This document is a test plan for Facial Analyzer for Rhinoplasty Surgery System Testing, produced by the System Testing team. It describes the testing strategy and approach to testing the team will use to verify that the application meets the established requirements of the business prior to release.

1.2 Objectives

- ☐ Meets the requirements, specifications and the Business rules.
- ☐ Supports the intended business functions and achieves the required standards.
- ☐ Satisfies the Entrance Criteria for User Acceptance Testing.

2.   Functional Scope

The Modules in the scope of testing for the Facial Analyzer for Rhinoplasty Surgery System Testing are mentioned in the document attached in the following path :

1. The System Requirement Specification Documents:
   https://docs.google.com/document/d/1By6d6mRB4j_fJBMGeEJjDnlBKS7ZbIJNQxqU2LrMRk4/edit?usp=sharing

2. User Manuals:
   https://github.com/jcurtis664/FacialAnalyzer/tree/main/Deliverables/CS491-Deliverables

3. Section 3.1: Testing Strategy

3.   Overall Strategy and Approach

3.1 Testing Strategy

Facial Analyzer for Rhinoplasty Surgery System Testing will include testing of all functionalities that are in scope (Section 2) identified. System testing activities will include the testing of new functionalities, modified functionalities, functionality access, testing of internal & external interfaces.

3.1.1   Function Testing

Test Objective: Confirm the functionality of clustering algorithms and use FaceGen modeling software.

Technique: Execute multiple use cases to confirm that the provided output matches the expected output and run the FaceGen software through an evaluation

program.

Completion Criteria: All cases for clustering algorithm match expected output and The FaceGen modeling software can pass evaluation.

Special Consideration: Access to the Digitized Rhinoplasty website and System Requirement Specifications are necessary.

3.1.2 Database Testing

Test Objective: Confirm the functionality of the database for generating face clusters with the new FaceGen generated faces.

Technique: Upload multiple new models and test their functionality with the clustering algorithms.

Completion Criteria: The database functions properly when utilized by all the face clustering algorithms.

Special Consideration: Access to the Digitized Rhinoplasty website and System Requirement Specifications are necessary.

3.1.3 Performance Testing

Test Objective: Compare accuracies of various face clustering methods to determine the optimal algorithm.

Technique: Execute testing code for each clustering algorithm to generate graphs comparing average accuracies.

Completion Criteria: Graphs have been generated comparing the accuracies.

Special Consideration: Access to the Digitized Rhinoplasty website and System Requirement Specifications are necessary.

3.2 System Testing Entrance Criteria

In order to start system testing, certain requirements must be met for testing readiness. The readiness can be classified by: access to file structure from GitHub (https://github.com/jcurtis664/FacialAnalyzer), access to the internet, access to digitized rhinoplasty web application, and access to a computer with Windows 10 or Mac OSX, and Python 3.0 or later.

3.3 Testing Types

3.3.1 Usability Testing

   User interface attributes, cosmetic presentation and content will be tested for accuracy and general usability. The goal of Usability Testing is to ensure that the User Interface is comfortable to use and provides the user with consistent and appropriate access and navigation through the functions of the application (e.g., access keys, consistent tab order, readable fonts etc.)

3.3.2 Functional Testing

   The objective of this test is to ensure that each element of the component meets the functional requirements as outlined in the:
   - Functional Requirements
   - Business rules or conditions
   - Other functional documents produced during the course of the project i.e. resolution to issues/change requests/feedback

3.4 Suspension Criteria and Resumption Requirements

   This section will specify the criteria that will be used to suspend all or a portion of the testing activities on the items associated with this test plan.

3.4.1 Suspension Criteria

   If the user uploads the wrong format of files, the program will suspend.

3.4.2 Resumption Requirements

   The program will continue once the user has uploaded the correct files.

4.   Execution Plan

4.1 Execution Plan

   The execution plan will detail the test cases to be executed. The execution plan will be put together to ensure that all the requirements are covered. The execution plan will be designed to accommodate some changes if necessary, if testing is incomplete on any day. All the test cases of the projects under test in this release are arranged in a logical order depending upon their inter dependency.

   The test plan for the system is as follows:

   4.1.1 Function Testing (See 3.1.1)

   4.1.2 Database Testing (See 3.1.2)

   4.1.3 Performance Testing (See 3.1.3)

Requirement (From SRS)	Test Case Identifier	Input	Expected Behavior	Pass/Fail
3.3.1 The system must generate new landmarks for the patient's nose.	1.0	Insert patient's facial landmarks	Return new x,y,z coordinates for patient's nose	Fail
3.3.2 The system must display the 3D model of the patient's face with a new nose.	1.1	Open patient's scan info in Blender	Blender generates a 3D image of the patient's face with new nose landmarks	Pass
3.3.3 The surgeon must mark points on the patient's face upon upload using the Digitized Rhinoplasty web application.	1.2	Open patient's scan info on digitizedrhinoplasty.com and manually choose the significant landmarks	Digitized rhinoplasty web application allows the output of a JSON file containing the landmarks	Pass
3.3.4 The system must display multiple similar face options for the patient to choose from.	1.3	Run the 'match_faces.py' file with patient's ratios	The program will fit the patient into an existing cluster	Pass
3.3.5 The system must generate more similar face suggestions based on the patient's request.	1.4	Run the 'match_faces.py' file with patient's ratios	The program will fit the patient into an existing cluster for each different clustering algorithm	Pass

5.   Traceability Matrix & Defect Tracking

5.1 Traceability Matrix

List of requirement, corresponding test cases

Requirement CRITICAL: Function Testing, 3.1.1: "Confirm the functionality of clustering algorithms and GAN model."

Test Cases: Execute multiple use cases to confirm that the provided output matches the expected output and run the GAN model through an evaluation program.

***Requirement CRITICAL*:** Database Testing, 3.1.2: "Confirm the functionality of the database for generating face clusters with the new FaceGEN generated faces."

Test Cases: Upload multiple new models and test their functionality with the clustering algorithms.

Requirement CRITICAL: Performance Testing, 3.1.3: "Compare accuracies of various face clustering methods to determine the optimal algorithm."

Test Cases: Compare accuracies of various face clustering methods to determine the optimal algorithm.

Requirement	Requirement Dependencies	Test Case	Test Case Dependencies	Severity	Responsibility	Result
3.3.1 The system must generate new landmarks for the patient's nose.	3.3.2 3.3.3	1.0	1.1 1.2	Critical	P.C. M.C. J.C. M.K. A.O.	Fail
3.3.2 The system must display the 3D model of the patient's face with a new nose.	3.3.1 3.3.3	1.1	1.0 1.2	Critical	P.C. M.C. J.C. M.K. A.O.	Pass
3.3.3 The surgeon must mark points on the patient's face upon upload using the Digitized Rhinoplasty web application.	3.3.2 3.3.1 3.3.5	1.2	1.1 1.0 1.4	Critical	P.C. M.C. J.C. M.K. A.O.	Pass
3.3.4 The system must display multiple similar face options for the patient to choose from.	3.3.3 3.3.5	1.3	1.2 1.4	Medium	P.C. M.C. J.C. M.K. A.O.	Pass
3.3.5 The system must generate	3.3.3 3.3.4	1.4	1.2 1.3	Medium	P.C. M.C. J.C.	Pass

					M.K. A.O.	
more similar face suggestions based on the patient's request.						

5.2 Defect Severity Definitions

Critical: A critical severity defect can be classified as a defect that severely or significantly impairs the overall functionality of the software. This class of defect is fixed with a significant amount of effort in the functionality of the system.

Medium: A medium severity defect can be classified as a defect that does not impair the overall functionality of the software, but impairs certain aspects. This class of defect can be fixed with a medium amount of effort.

Low: A low severity defect can be classified as a defect that does not impair the functionality of the software. This class of defect can be fixed with a low amount of effort.

6.   Environment

6.1 Environment

The System Testing Environment will be used for System Testing. In order to conduct the testing the tester needs to have the following installed into their computer:
- Python v3.0 or later
- Access to database of faces
- Access to database of facial ratios

7.   Assumptions

This section lists the assumptions for this project.
- User is on a machine with the necessary software
- User is on a machine that has access to internet
- All files have the correct format
- Clients will submit pictures with no facial expressions to render a 3D image
- FaceGEN dataset creation is able to generate faces sufficient to use for our digitized rhinoplasty landmark software

8.   Risks and Contingencies

Risk #	Risk	Impact	Contingency Plan
1	Unsuitable Suggestion	High	Making sure different scenarios are tested rigorously to ensure this happens as infrequently as possible.
2	Dataset Creation Errors	High	To ensure our dataset is accurate, the faces created using FaceGen must look realistic and believable.
3	Facial Landmarks Pointing Errors	High	To ensure the facial points are marked correctly, the marked points must be checked thoroughly and the measurements must be in reasonable range.