**System Test Plan**

**For**

***Smart Nose Surgery***

Team members:

Chris Graziano

Victoria Jordan

Anton Kiselev

Jacob Preseau

Chintan Thakrar

|  |  |  |
| --- | --- | --- |
| Version/Author | Date | Description |
| 0.0.1 / Team 1 | 11/4/2020 | Document created |
| 0.0.2 / Anton Kiselev | 11/4/2020 | Test cases added |
| 0.0.3 / Victoria Jordan | 11/5/2020 | Template Revisions |
| 0.0.4 / Victoria Jordan | 11/6/2020 | Updated section descriptions and requirements. |
| 0.0.5 /Anton Kiselev | 11/9/2020 | Updated traceability matrix,  Updated test cases |
| 0.0.6 / Victoria Jordan | 11/10/2020 | Updated traceability matrix |
| 0.0.7/ Chintan Thakrar | 11/10/2020 | Updated risk and assumption section |
| 0.0.8 / Victoria Jordan | 11/27/2020 | Document revisions based on instructor feedback. |
| 0.0.8 / Chris | 11/29/2020 | Updated Traceability Matrix |

Table of Contents

[1.](#_heading=h.gjdgxs) Introduction 2

[1.1](#_heading=h.30j0zll) Purpose 2

[1.2](#_heading=h.1fob9te) Objectives 2

[2.](#_heading=h.3znysh7) Functional Scope 2

[3.](#_heading=h.2et92p0) Overall Strategy and Approach 2

[3.1](#_heading=h.tyjcwt) Testing Strategy 2

[3.2](#_heading=h.3dy6vkm) System Testing Entrance Criteria 2

[3.3](#_heading=h.1t3h5sf) Testing Types 2

[3.3.1](#_heading=h.4d34og8) Usability Testing 2-3

[3.3.2](#_heading=h.2s8eyo1) Functional Testing 3

[3.4](#_heading=h.17dp8vu) Suspension Criteria and Resumption Requirements 3

[3.4.1](#_heading=h.3rdcrjn) Suspension Criteria 3

[3.4.2](#_heading=h.26in1rg) Resumption Requirements 3

[4.](#_heading=h.lnxbz9) Execution Plan 3

[4.1](#_heading=h.35nkun2) Execution Plan 3-5

[5.](#_heading=h.1ksv4uv) Traceability Matrix & Defect Tracking 5

[5.1](#_heading=h.44sinio) Traceability Matrix 5-8

[5.2](#_heading=h.2jxsxqh) Defect Severity Definitions 8

[6.](#_heading=h.z337ya) Environment 8

[6.1](#_heading=h.3j2qqm3) Environment 8

[7.](#_heading=h.4i7ojhp) Assumptions 9

[8.](#_heading=h.2xcytpi) Risks and Contingencies 9

[9.](#_heading=h.1ci93xb) Appendices 9

[9.1 Test Cases 9](#_heading=h.3whwml4)

# Introduction

## Purpose

This document is a test plan for “Smart Nose Surgery” System Testing produced by Team 1. 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.

## Objectives

* Meets the requirements, specifications and rules of the Business.
* Supports the intended business functions and achieves the required standards.
* Satisfies the Entrance Criteria for User Acceptance Testing.
* Ensure the Application Under Test conforms to functional and nonfunctional requirements
* Ensure the Application Under Test meets the quality specifications defined by the client
* Bugs/issues are identified and fixed before go live
* Adheres to U.S. Department of Health and Human Services (HHS) regulations

# Functional Scope

The modules in the scope of testing for the “Smart Nose Surgery” System Testing are mentioned in the document attached in the following path:

* System requirements
* Operating environment
* Files and database design
* Input formats
* Output layouts
* Human-machine interfaces
* External interfaces

# Overall Strategy and Approach

## Testing Strategy

Smart Nose Surgery System Testing will include testing of all functionalities that are identified as in the scope of the system (Refer Functional Scope Section). System testing activities will include the testing of new functionalities, modified functionalities, screen level validations, work flows, functionality access, testing of internal & external interfaces.

## System Testing Entrance Criteria

In order to start system testing, certain requirements must be met for testing readiness. The readiness can be classified into:

* Representative database of faces
* User’s ability to provide images to the system
* Systems functionality to calculate facial recognition points

## Testing Types

### 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.).

### Functional Testing

The objective of this test is to ensure that each element of the component meets the functional requirements of the business as outlined in the:

* System Requirements Specification Machine Learning Algorithm for Rhinoplasty (M-LAR)
* Before the completion of the project, each functional requirement will be tested with input parameters which can be distributed into 3 categories i.e. boundary, normal (expected inputs) and abnormal.(expected to fail) inputs.

## Suspension Criteria and Resumption Requirements

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

### Suspension Criteria

Testing will be suspended if the incidents found will not allow further testing of the system/application under-test. If testing is halted, and changes are made to the hardware, software or database, it is up to the Smart Nose Surgery team to determine whether the test plan will be re-executed or part of the plan will be re-executed.

### Resumption Requirements

Resumption of testing will be possible when the functionality that caused the suspension of testing has been retested successfully.

# Execution Plan

## Execution Plan

The execution plan for this system will cover the test cases listed in the table below. Each test case has specific inputs needed for the system to run appropriately. A test case results in a pass if the actual output aligns with the expected output. Each requirement can be satisfied by a single test cases, however additional test cases can be created if deemed necessary by the smart nose surgery team.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case (TC) Number | Action | Requirement number | Inputs | Expected output | Actual Output | Result  (Pass/Fail) | Test Comments |
| TC1 | Upload Image | FR 4.1.1 | Webcam capture | Image path | Image path | Pass | None |
| TC2 | Extract facial coordinates | FR 4.1.2 | Image path | 2D array of facial coordinates | 2D array of facial coordinates | Pass | None |
| TC3 | Access csv files | FR 4.1.5 | None | None | None | Pass | None |
| TC4 | System uses database ratios in KNN | FR 4.1.6 | Data from csv file | None | None | Pass | None |
| TC5 | System uses user’s data as a test point in KNN | FR 4.1.6.1 | 2D array of facial coordinates | None | None | Pass | None |
| TC6 | System applies KNN | FR 4.1.6 | 2D array of user facial points and ratios from csv file | Array of nearest neighbors | Array of nearest neighbors | Pass | None |
| TC7 | System displays results of KNN | FR 4.1.7 | Array of nearest neighbors | Images displayed on window | Images displayed on window | Pass | None |
| TC8 | Display 5 clusters in app window | FR 4.1.8 | Results of KNN alg | 5 icons of 5 different nose types |  |  |  |
| TC9 | Allow user to choose desired nose group | FR 4.1.9.1 | Mouse click | None |  |  |  |
| TC10 | System displays all faces in desired cluster | FR 4.1.9.2 | None | Window with all faces in chosen cluster |  |  |  |
| TC11 | System allows user chose desired nose | FR 4.1.9.3 | Mouse click | None |  |  |  |
| TC12 | System calculates required changes | FR 4.1.10.1 | User nose points and chosen face nose points | Array of required changes |  |  |  |
| TC13 | System displays required changes on user photo | FR 4.1.10.2 | User photo and required changes | User photo with required changes |  |  |  |
| TC14 | System saves user’s photo with changes as a separate file | FR 4.1.10.3 | User photo with required changes | Separate file on the computer with users photo and changes |  |  |  |
| TC15 | System allows doctor to access the final photo | FR 4.1.11 | None | System displays saved photo |  |  |  |
| TC16 | The system shall accept photo as an input | IR 4.2.1 | Photo | None |  |  |  |
| TC17 | The system shall receive 1 photo at the time | IR 4.2.2 | 1 Photo | Notification of successful uploading |  |  |  |
| TC18 | User uploads incorrect photo | FR 4.1.1 | User’s photo | Error message |  |  |  |
| TC19 | The system shall have login security feature | SR 4.8.1 | None | None |  |  |  |

# Traceability Matrix & Defect Tracking

## Traceability Matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Requirement Number | Requirement description | Test cases numbers  (from test case table) | Status |
| FR 4.1.1 | The system shall allow a user to scan their face with a camera. | TC1 | TC1.Pass |
| FR 4.1.2 | The system shall be able to transform the picture of the user's face into a set of coordinates representing facial landmark measurements. | TC2 | TC2.Pass |
| FR 4.1.3 | The system shall contain coordinate files with an array of 67 variables that can be used to uniquely characterize each face in the database and that of the user. | TC2 | TC2.Pass |
| FR 4.1.4 | The system shall ask a user to retake the photo in case coordinate detection fails. | TC18 | TC18.Pass/Fail |
| FR 4.1.5 | The system shall be able to get the files representing faces from the database. | TC3 | TC3.Pass |
| FR 4.1.6 | The system shall be able to apply the KNN algorithm on the files in the database. | TC4 | TC4.Pass |
| FR 4.1.6.1 | The system shall be able to apply KNN with the user's face as the test point, using all landmarks other than those relating to the nose. | TC5 | TC5.Pass |
| FR 4.1.6.2 | The system shall be able to save the closest 40 faces from the dataset into an array. | TC6 | TC6.Pass |
| FR 4.1.6.3 | The system shall be able to apply KNN on 40 faces and the user’s face, using only the landmarks relating to the nose. | TC6 | TC6.Pass |
| FR 4.1.6.4 | The system shall save the 5 closest faces into an array. | TC6 | TC6.Pass |
| FR 4.1.7 | The system shall be able to display faces stored in the array | TC7 | TC7.Pass |
| FR 4.1.8 | The system shall display to the user 5 different noses. | TC7 | TC7.Pass |
| FR 4.1.9 | The system shall allow the user to choose a nose from the 5 presented. | TC9 | TC7.Pass/Fail |
| FR 4.1.10 | The system shall be able to calculate required changes on the user’s face. | TC12 | TC12.Pass |
| FR 4.1.10.1 | The system shall be able to calculate distances between the nose points of the user face and the chosen face. | TC12 | TC12.Pass |
| FR 4.1.10.2 | The system shall be able to display changes on the user’s photo. | TC13 | TC13.Pass/Fail |
| FR 4.1.10.3 | The system shall be able to save the photo with final changes. | TC14 | TC14.Pass/Fail |
| FR 4.1.11 | The system shall allow the doctor to access the final photo with text explaining changes to be made. | TC15 | TC15.Pass/Fail |
| IR 4.2.1 | The system shall have a webcam capture as an input | TC1 | TC1.Pass |
| IR 4.2.2 | The system shall receive one user’s face at a time | TC17 | TC1.Pass |
| IR 4.2.3 | The system shall display five clusters | TC8 | TC8.Pass |
| IR 4.2.4 | The system shall show every face from a cluster if the user asks | TC9  TC10 | TC9.Pass  TC10.Pass |
| IR 4.2.5 | The login screen moves to the start screen if the user inputs the correct password. | TC19 | TC19.Pass |
| PR 4.3.1. | The system shall operate on any computer operating system (i.e. Linux, Windows) that supports Python 3. | TC1 | TC1.Pass |
| UHFR 4.4.2 | The system shall provide instructions for use. | N/A | README File |
| DR 4.6.1 | The system shall select the initial 40 faces with the KNN machine learning algorithm, using Euclidean distance as the distance metric. | TC6 | TC6.Pass |
| DR 4.6.2 | The system shall select the final 5 noses with the KNN machine learning algorithm, using Euclidean distance as the distance metric. | TC6 | TC6.Pass |
| DR 4.6.3 | The system shall have a represented user approval rating of at least 80% for the measure of precision of the algorithm. |  |  |
| DR 4.6.4 | The system shall obtain data from trials based on whether or not the user successfully found a nose they approved of. |  |  |
| DR 4.6.5 | The system shall contain a face library that will exist as a part of the app, and not on a network. | TC3 | TC3.Pass |

## Defect Severity Definitions

The table below describes some examples of actions of the system that would be considered critical, medium, and low defects.

|  |  |
| --- | --- |
| **Critical** | * System can not access user photo * System can not calculate KNN algorithm * System does not display the user multiple face options * System application can not access the image database |
| **Medium** | * Facial points are not displaying * Some CSV files are missing |
| **Low** | * UI font is incorrect * UI button is in an incorrect position |

# Environment

## 6.1 Environment

## This section describes how the environment for the system to run successfully.

* The System Testing Environment will be used for System Testing.
* Python 3.4 with the dlib, pandas and Opencv libraries installed are required for the system to run.
* The system configuration does not involve any hardware.
* The system configuration consists of a python to run the system and perform test cases.

# Assumptions

Define test plan assumptions.

For the purpose of the project, we make the following assumptions:

* It is assumed that the system is not expected to produce precise results primarily because the system uses a 2D projection of a face on the frontal view. This prevents the extraction of sufficient nasal landmarks to categorize a nose to a large amount of categories.
* It is assumed that the surgeon is the primary user of the application since it presents the statistical points and the appropriate manipulations that need to be conducted to achieve the user selected face.

Regarding our test plan, we make the following assumptions:

* A test case passes if the actual output aligns with the expected output.
* DRs are satisfied after running the system more than 15 times.
* The overall execution of the system is successful if the user can take a photo, the photo is processed and returns with facial coordinates, the system displays five nose clusters, the user can selected their desired nose, and the system returns calculations for the required changes the doctor needs to make on the users’ face to provide a quality rhinoplasty.

# Risks and Contingencies

* The system is expected to identify user faces when available, it does not have the capabilities to recognise a situation where a user face is not present when the photo is being clicked.
* The orientation of the user face is not embedded into the system. The user is expected to use the system and click a photo from the frontal view and perpendicular to the direction of the camera.

# Appendices

* 1. Glossary

|  |  |
| --- | --- |
| TERM/ACRONYM | DEFINITION |
| API | Application Program Interface |
| AUT | Application Under Test |
| HHS | U.S. Department of Health and Human Services |
| FR | Functional Requirements |
| IR | Interface Requirements |
| PR | Physical Requirements |
| UHFR | Users and Human Factors Requirements |
| DR | Data Requirements |
| SR | Security Requirements |