**System Requirements Specification**

**of**

**Machine Learning Algorithm for Rhinoplasty (M-LAR)**

**SE 450, Fall, 2020**

**Team Name: Nose Surgery 1**

**Team Members:**

Chris Graziano

Victoria Jordan

Anton Kiselev

Jacob Preseau

Chintan Thakrar

|  |  |  |
| --- | --- | --- |
| **Date/Time** | **Author** | **Description** |
| September 17, 3:24 PM | Jacob Preseau | Added Team Name and Team Members |
| September 18, 11:47 PM | Chris Graziano | Added Project Name and Course, Semester, Year; first bullet in Definitions |
| September 18, 11:47 PM | Victoria Jordan | Added System to be Produced |
| September 21, 3:15 PM | Jacob Preseau | Added Assumptions, first bullet in Stakeholders, and second bullet in Definitions |
| September 21, 5:55 PM | Anton Kiselev | Added Use Case Model, DFD Models, and State Chart Model |
| September 21, 7:05 PM | Jacob Preseau | Added second bullet in Stakeholders, Use Case Descriptions |
| September 22, 3:31 PM | Chris Graziano | Added Applicable Standards |
| September 23, 5:44 PM | Anton Kiselev | Updated Use Case Model, model descriptions |
| September 24 | Chris Graziano, Victoria Jordan, Anton Kiselev, Chintan Thakrar | Added Requirements |
| September 25 | Chris Graziano, Victoria Jordan, Anton Kiselev, Jacob Preseau, Chintan Thakrar | Revised Requirements |
| September 25 | Anton Kiselev | Updated models and description |
| October 14, 10:00 PM | Victoria Jordan | Document revisions based on version 1 feedback |
| November 13, 10:00 AM | Anton Kiselev | Updated models |
| November 14, 1:00 PM | Anton Kiselev | Updated requirements |
| November 26, 10:00 PM | Victoria Jordan | Document revisions based on version 2 feedback |
| November 27, 1:30 PM | Victoria Jordan | Updated section requirements to meet SRS template formatting and continued version 2 feedback revisions. |
| November 29, 5:44 PM | Anton Kiselev | Updated models. |

Table of Contents

[1. Introduction 4](#_heading=h.gjdgxs)

[1.1. System to be Produced 4](#_heading=h.30j0zll)

[1.2. Applicable Standards 4](#_heading=h.1fob9te)

[2. Definition, Acronyms, and Abbreviations 4](#_heading=h.3znysh7)

[3. Product Overview 4](#_heading=h.2et92p0)

[3.1. Assumptions 4](#_heading=h.tyjcwt)

[3.2. Stakeholders 4](#_heading=h.3dy6vkm)-5

[3.3. Event Table 5](#_heading=h.1t3h5sf)

[3.4. Use Case Diagram 6](#_heading=h.4d34og8)

[3.5. Use Case Descriptions 6](#_heading=h.2s8eyo1)

[3.6. DFD Diagram 7](#_heading=h.17dp8vu)

[3.6.1. DFD Model level 0 7](#_heading=h.3rdcrjn)

[3.6.2. DFD Model level 1 7](#_heading=h.26in1rg)

[3.6.3. DFD Model level 1 extended state Applying KNN](#_heading=h.lnxbz9) 8

[3.7. DFD Descriptions 8](#_heading=h.35nkun2)

[3.8. State Chart Model](#_heading=h.1ksv4uv) 9

[3.9. State Chart Model Description](#_heading=h.44sinio) 9-10

[4. Specific Requirements](#_heading=h.2jxsxqh) 10

[4.1. Functional Requirements](#_heading=h.z337ya) 10-17

[4.2. Interface Requirements 1](#_heading=h.3j2qqm3)7-18

[4.3. Physical Environment Requirements 1](#_heading=h.1y810tw)8

[4.4. Users and Human Factors Requirements 1](#_heading=h.4i7ojhp)8-20

[4.5.](#_heading=h.2xcytpi) Documentation Requirements 20-21

[4.6. Data Requirements](#_heading=h.1ci93xb) 21-23

[4.7. Resource Requirements](#_heading=h.3whwml4) 23-27

[4.8. Security Requirements](#_heading=h.2bn6wsx) 27

## 

## Introduction

### System to be Produced

The system will provide a patient with representative similar faces with different nose types that align with the measurements of their face, therefore allowing the patient to select which nose is preferred for their rhinoplasty procedure. In addition to that system will display a patient photo with required changes on their nose. This system also complements https://digitized-rhinoplasty.com/app.

### Applicable Standards

The program will adhere to all U.S. Department of Health and Human Services (HHS) regulations on medical standards.

## Definition, Acronyms, and Abbreviations

* KNN - "*k*-nearest neighbors", a machine learning algorithm that, when given a test point represented by a point on the coordinate plane, will find the *k* points in the data set, also in the coordinate plane, that are nearest to the test point.
* "Patient" and "User" are used as interchangeable terms.

## Product Overview

### Assumptions

* The product will be delivered through a standalone Python application.
* Shape predictor works with error less than 1 percentage
* All photos are taken from the same distance such way we can perform scaling
* Not every time the user will be able to see all 5 nose clusters due to face shape.
* For different clients, the system will get the different number of available clusters.

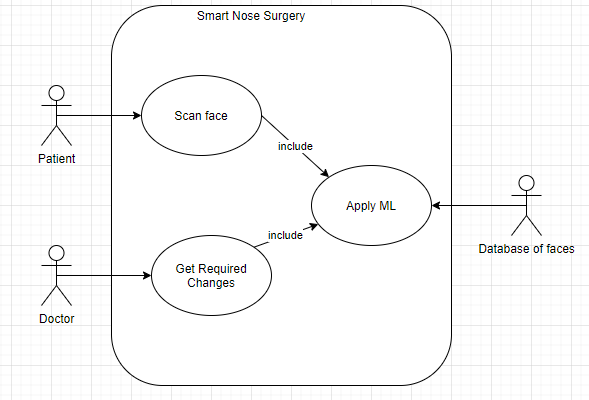
### Stakeholders

* User - The user's interest is in being able to use the product to select their preferred nose for their procedure.
* Doctor - The doctor performing the user's rhinoplasty procedure would benefit from knowing exactly what changes are needed to perform the procedure.
* Cosmetic Investors - Provide cosmetic products or discounts when users complete the rhinoplasty process.
* Model - add face to database.
* Consultant - walk user through the application, can give secondary suggestion
* IT support - help to update database of faces
* Hospital - use the system for patients that need immediate surgery on their nose due to a severe injury.
* Clinic - use for various locations that can perform rhinoplasty of customers
* Health Care Provider - determine if the use of this system will be covered under a users’ insurance plan.
* Pharmaceutical company - use the system to determine what type of medicine could be provided for the recovery process.

### Event Table

|  |  |  |  |
| --- | --- | --- | --- |
| Event Name | External Stimuli | External Responses | Internal data and state |
| Doctor scans patient face | Doctor | None | App transform photo into coordinate file |
| Low quality photo | None | None | App notifies that photo is not appropriate |
| High quality photo | User | None | App transform photo into coordinate file |

### Use Case Model

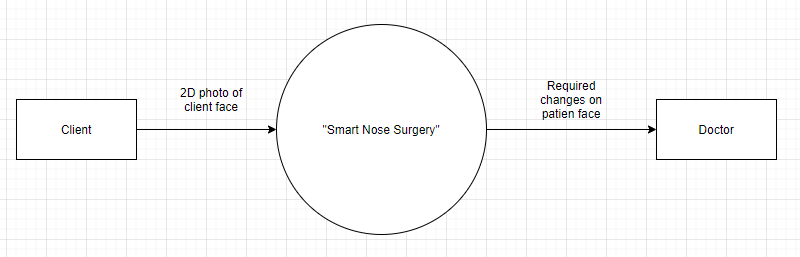


### Use Case Model Descriptions

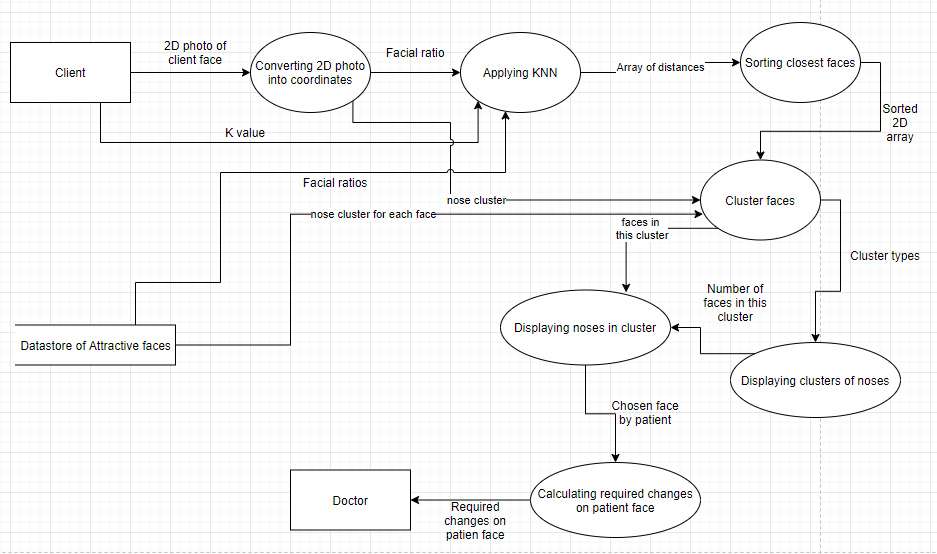
* **Scan face** - System asks the patient to provide an image of their face. The patient uploads an image of their face to the system.
* **Apply ML** - Included use case. The system gets images from the database to apply the KNN algorithm.
* **Get Required Changes** - Use case includes “Apply ML” use case. After applying KNN and getting the closest faces, the doctor receives required changes on the patient face from the system.

### DFD Model

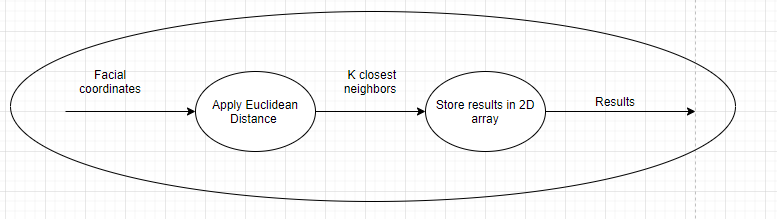
### DFD Model Level 0



### DFD Model Level 1



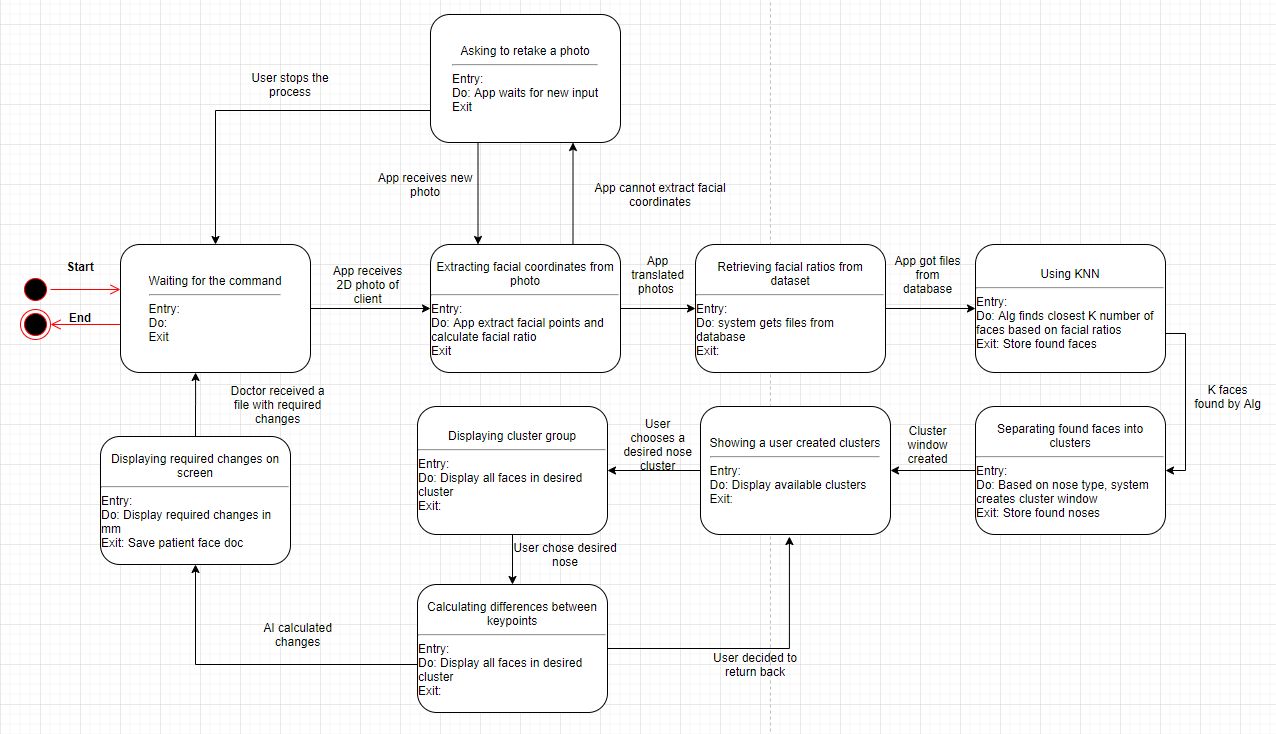
### DFD Model Level 1 extended state Applying KNN



### DFD Descriptions

* DFD Model Level 0 - in this model, we can see three main components: a source of data - Client, main process - “Smart Nose Surgery” and sink for data - doctor.
* DFD Model Level 1 - in this model we expanded the main process into several smaller processes to get a better idea of what is behind it. First of all the system converts a patient photo into facial coordinates and retrieves facial coordinates of different people from the data store. After that, the process “Applying KNN” which is described below level 1 model, will do calculations. In the extended state, the model shows that the algorithm applies Euclidean distance check and stores results for future access. After applying KNN for faces, the next process will display the results of KNN with noses and let the client choose the desired face. The last process will calculate the required changes for the client's face and the doctor will get the data of it.

### State Chart Model



### State Chart Model Description

* State Chart allows seeing what states and conditions will be in the system. This state chart represents the basic cycle of the process.
  + The system waits for the command.
  + After receiving photos, the system will transform it into a 2D array with facial coordinates.
    - Possible error: photo taken incorrectly, system will ask a user to retake photo.
    - User may cancel the process and system will get into initial state
  + After a successful transformation, the system will be obtaining files from the data store.
  + After that, the system will apply KNN to find K closest faces.
  + K value will be specified by the user, based how many faces the user wants to look at.
  + Based on the results of the KNN algorithm, the system will separate faces into 5 different nose clusters based on nasal indexes.
  + The system will display window with available clusters from given KNN results
  + Each cluster will contain a number of faces which will be different for each client face.
  + After the client chooses the desired face, the system will be calculating the required changes on nose and display them to the doctor.
    - Client has an option to return and choose a different cluster and different nose.
  + After that doctor receives the final results and the system will be ready for a new cycle.

## Specific Requirements

### Functional Requirements

|  |
| --- |
| No: 4.1.1 |
| Statement: The system shall allow a user to scan their face with a camera. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Use Case Model, DFD Model (Level 0 and Level 1) |
| Evaluation Method: The system does not require the user to retake the photo. (4.1.4) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.2 |
| Statement: The system shall be able to transform the picture of the user's face into a set of coordinates representing facial landmark measurements. |
| Source: User |
| Dependency: 4.1.1 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model, Figure 5 and 6 (Smart Nose Surgery Team 1 SDS) |
| Evaluation Method: The users image is processed and displays 67 coordinate points outlined on their face. |
| Revision History: Internal Design Team, September 2020, Decided to implement 67 facial point predictor to apply to KNN algorithm. |

|  |
| --- |
| No: 4.1.3 |
| Statement: 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. |
| Source: Internal Design Team |
| Dependency: 4.1.2, 4.1.2 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model, Figure 5 and 6 (Smart Nose Surgery Team 1 SDS) |
| Evaluation Method: The users image is processed and displays 67 coordinate points outlined on their face. |
| Revision History: Internal Design Team, September 2020, Decided to implement 67 facial point predictor to apply to KNN algorithm. |

|  |
| --- |
| No: 4.1.4 |
| Statement: The system shall ask a user to retake the photo in case coordinate detection fails. |
| Source: Internal Design Team, User |
| Dependency: 4.1.1 |
| Conflicts: None |
| Supporting Materials: Use Case Model, DFD Model (Level 0 and Level 1) |
| Evaluation Method: The users image is processed and displays 67 coordinate points outlined on their face. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.5 |
| Statement: The system shall be able to get the files representing faces from the database. |
| Source: Internal Design Team |
| Dependency: 4.1.2 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.6 |
| Statement: The system shall be able to apply the KNN algorithm on the files in the database. |
| Source: System Design - Implemented algorithm in source code |
| Dependency: 4.1.1, 4.1.2, 4.1.3, 4.1.5 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system saves the closest 40 faces from the dataset into an array (4.1.6.2) |
| Revision History: Internal Design Team, September 2020, Decided to implement 67 facial point predictor to apply to KNN algorithm. |

|  |
| --- |
| No: 4.1.6.1 |
| Statement: 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. |
| Source: System Design - Implemented algorithm in source code |
| Dependency: 4.1.1, 4.1.2, 4.1.3, 4.1.5, 4.1.6 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: Internal Design Team, September 2020, Decided to implement 67 facial point predictor to apply to KNN algorithm. |

|  |
| --- |
| No: 4.1.6.2 |
| Statement: The system shall be able to save the closest 40 faces from the dataset into an array. |
| Source: System Design - Implemented algorithm in source code |
| Dependency: 4.1.1, 4.1.2, 4.1.3, 4.1.5, 4.1.6, 4.1.6.1 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.6.3 |
| Statement: The system shall be able to apply KNN on 40 faces and the user’s face, using only the landmarks relating to the nose. |
| Source: System Design - Implemented algorithm in source code |
| Dependency: 4.1.1, 4.1.2, 4.1.3, 4.1.5, 4.1.6, 4.1.6.1, 4.1.6.2 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.6.4 |
| Statement: The system shall save the 5 closest faces into an array. |
| Source: System Design - Implemented algorithm in source code |
| Dependency: 4.1.1, 4.1.2, 4.1.3, 4.1.5, 4.1.6, 4.1.6.1, 4.1.6.2, 4.1.6.3 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.7 |
| Statement: The system shall be able to display faces stored in the array. |
| Source: System Database |
| Dependency: 4.1.6, 4.1.6.1, 4.1.6.2, 4.1.6.3, 4.1.6.4 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces with 5 different noses stored in the array on the user interface. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.8 |
| Statement: The system shall display to the user 5 different noses. |
| Source: System Database |
| Dependency: 4.1.6, 4.1.6.1, 4.1.6.2, 4.1.6.3, 4.1.6.4, 4.1.7 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces with 5 different noses stored in the array on the user interface. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.9 |
| Statement: The system shall allow the user to choose a nose from the 5 presented. |
| Source: System Database |
| Dependency: 4.1.6, 4.1.6.1, 4.1.6.2, 4.1.6.3, 4.1.6.4, 4.1.7, 4.1.8 |
| Conflicts: None |
| Supporting Materials:DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces with 5 different noses stored in the array on the user interface. The user can highlight one specific image. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.10 |
| Statement: The system shall be able to calculate required changes on the user’s face. |
| Source: System Design - Implemented algorithm in source code, User, Database |
| Dependency: .1.6, 4.1.6.1, 4.1.6.2, 4.1.6.3, 4.1.6.4, 4.1.7, 4.1.8, 4.1.9 |
| Conflicts: None |
| Supporting Materials: State Chart Model |
| Evaluation Method: The user is able to select a specific image from the set and the system returns calculations of the required changes. |
| Revision History: Internal Design Team, September 2020, adjusted requirement to account for differences between the users’ submitted and selected images. |

|  |
| --- |
| No: 4.1.10.1 |
| Statement: The system shall be able to calculate distances between the nose points of the user face and the chosen face. |
| Source: System Design - Implemented algorithm in source code, User, Database |
| Dependency: 4.1.7, 4.1.8, 4.1.9, 4.1.10 |
| Conflicts: None |
| Supporting Materials: State Chart Model |
| Evaluation Method: Method: The user is able to select a specific image from the set and the system returns calculations of the required changes. |
| Revision History: Internal Design Team, September 2020, adjusted requirement to account for differences between the users’ submitted and selected images. |

|  |
| --- |
| No: 4.1.10.2 |
| Statement: The system shall be able to display changes on the user’s photo by drawing a line to the required final point and specifying distance in mm. |
| Source: System Database |
| Dependency: 4.1.7, 4.1.8, 4.1.9, 4.1.10, 4.1.10.1 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system returns the users’ original image with detailed lines and calculations. |
| Revision History: Internal Design Team, November 2020, adjusted requirement to account for display to stakeholders of the differences between the users’ submitted and selected images. |

|  |
| --- |
| No: 4.1.10.3 |
| Statement: The system shall be able to save the photo with final changes. |
| Source: System Database |
| Dependency: 4.1.7, 4.1.8, 4.1.9, 4.1.10, 4.1.10.1, 4.1.10.2 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1 |
| Evaluation Method: The user interface displays “Image saved” |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.1.11 |
| Statement: The system shall allow the doctor to access the final photo with text explaining changes to be made. |
| Source: System Database |
| Dependency: 4.1.10, 4.1.10.1, 4.1.10.2, 4.1.10 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system returns the users’ original image with detailed lines and calculations. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

### Interface Requirements

|  |
| --- |
| No: 4.2.1 |
| Statement: The system shall have photo files as an input. |
| Source: User |
| Dependency: 4.1.1, 4.1.4 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The photo is a .jpg format |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.2.2 |
| Statement: The system shall receive one user’s face at a time |
| Source: User |
| Dependency: 4.1.1, 4.1.4 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model, Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The photo is a .jpg format. Then the users’ image is processed and displays 67 coordinate points outlined on their face. |
| Revision History: Internal Design Team, September 2020, Determined that the system can only handle one photo at a time. If the user has to retake their photo the original one is discarded. |

### Physical Environment Requirements

|  |
| --- |
| No: 4.3.1 |
| Statement: The system shall operate on any computer operating system (i.e. Linux, Windows) that supports Python 3. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system runs on a users’ machine. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

### Users and Human Factors Requirements

|  |
| --- |
| No: 4.4.1 |
| Statement: The systems’ users shall be the patients and the doctor. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The user has a successful login attempt. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.4.2 |
| Statement: The system shall provide instructions for use. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The user interface displays buttons for the user to select for the next step of the process. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.4.3 |
| Statement: The system shall provide each user a brief description of the output. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The user interface displays a description of each process. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.4.4 |
| Statement: The systems’ user shall not require any special accommodations. |
| Source: Internal Design Team, User |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Applies one the user has a successful login attempt. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.4.5 |
| Statement: The system shall be able to prompt the user to retake their photo in case of coordinate detection failure. |
| Source: Internal Design Team |
| Dependency: 4.1.4 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Occurs when requirement 4.1.4 needs to be performed. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

### Documentation Requirements

|  |
| --- |
| No: 4.5.1 |
| Statement: All shareholders shall be provided with a user manual |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Provided to the user prior to running the system. Located in the README file. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.5.2 |
| Statement: The user manual shall specifically describe how to use the system. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials:Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Provided to the user prior to running the system. Located in the README file. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.5.3 |
| Statement: The user manual shall specifically describe the purpose of the system. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Provided to the user prior to running the system. Located in the README file. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

### Data Requirements

|  |
| --- |
| No: 4.6.1 |
| Statement: The system shall select the initial 40 faces with the KNN machine learning algorithm, using Euclidean distance as the distance metric. |
| Source: Internal Design Team - randomly selected number of faces. |
| Dependency: 4.1.6.2, 4.1.6.3 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.6.2 |
| Statement: The system shall have 5 different nose clusters based on nasal index. |
| Source: Internal Design Team - randomly selected number of nose clusters |
| Dependency: 4.1.6.4 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.6.3 |
| Statement: The system shall have a represented user approval rating of at least 80% for the measure of precision of the algorithm. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system provides the doctor with calculations of the required changes needing to be performed on the user. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.6.4 |
| Statement: The system shall obtain data from trials based on whether or not the user successfully found a nose they approved of. |
| Source: User, Internal Design Team, System Database |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system provides the doctor with calculations of the required changes needing to be performed on the user. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.6.5 |
| Statement: The system shall contain a face library that will exist as a part of the app, and not on a network. |
| Source: System Database |
| Dependency: 4.1.5 |
| Conflicts: None |
| Supporting Materials: DFD Model Level 1, State Chart Model |
| Evaluation Method: The system displays faces from the database (4.1.7) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

### Resource Requirements

|  |
| --- |
| No: 4.7.1 |
| Statement: The system shall require an experienced team in Python and Machine learning |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Develop system in python supported platforms. |
| Revision History: Internal Design Team, November 2020, to be more precise with language in this document. |

|  |
| --- |
| No: 4.7.2 |
| Statement: The system shall be operable by anyone familiar with a computer. |
| Source: Internal Design Team, User |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Use Case Model |
| Evaluation Method: The user has read the provided user manual (4.5.2) |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.7.3 |
| Statement: The system shall require minimal resources upon launch. |
| Source: Internal Design Team |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system performs the desired actions in the application. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.7.3.1 |
| Statement: The system shall not require a skilled personnel to maintain the system. |
| Source: Internal Design Team |
| Dependency: 4.7.3 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system performs the desired actions in the application. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.7.3.2 |
| Statement: The system shall run on, and be managed during operation, by the user’s machine. |
| Source: Internal Design Team |
| Dependency: 4.7.3, 4.7.3.1 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system performs the desired actions in the application only. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.7.3.3 |
| Statement: The system shall be ready on December 1st, 2020. |
| Source: Internal Design Team |
| Dependency: 4.7.3, 4.7.3.1, 4.7.3.2 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Determined by Internal Design Team satisfaction with the application. |
| Revision History: Internal Design Team, September 2020, updated based on course deadlines timeline. |

|  |
| --- |
| No: 4.7.3.4 |
| Statement: The system shall require no additional funding. |
| Source: Internal Design Team |
| Dependency: 4.7.3, 4.7.3.1, 4.7.3.2, 4.7.3.3 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system performs the desired actions in the application only. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.7.3.5 |
| Statement: The system shall require no additional hardware. |
| Source: Internal Design Team |
| Dependency: 4.7.3, 4.7.3.1, 4.7.3.2, 4.7.3.3, 4.7.3.4 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system performs the desired actions in the application only. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.7.3.6 |
| Statement: The system shall utilize software tools: GitHub, ZenHub, Anaconda 3 (and all of its included tools), as well as the cv2 and dlib Python libraries. |
| Source: Internal Design Team |
| Dependency: 4.7.3, 4.7.3.1, 4.7.3.2, 4.7.3.3, 4.7.3.4, 4.7.3.5 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: Platforms are updated during the development of this system. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

### Security Requirements

|  |
| --- |
| No: 4.8.1 |
| Statement: The system shall verify login credentials through the API associated with the user. |
| Source: User |
| Dependency: None |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system allows the user to take a photo. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |

|  |
| --- |
| No: 4.8.2 |
| Statement: The system shall lock users out for 1 minute if the verification process fails 3 times consecutively. |
| Source: Internal Design Team |
| Dependency: 4.8.1 |
| Conflicts: None |
| Supporting Materials: Smart Nose Surgery Team 1 SDS |
| Evaluation Method: The system denies the user access and does not allow the user to take a photo. |
| Revision History: None - Original requirement by the Internal Design Team, September 2020 |