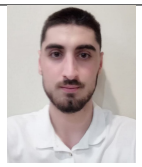
3D Model Binary Vision System

Requirement Specifications Document

Flap_Jacks

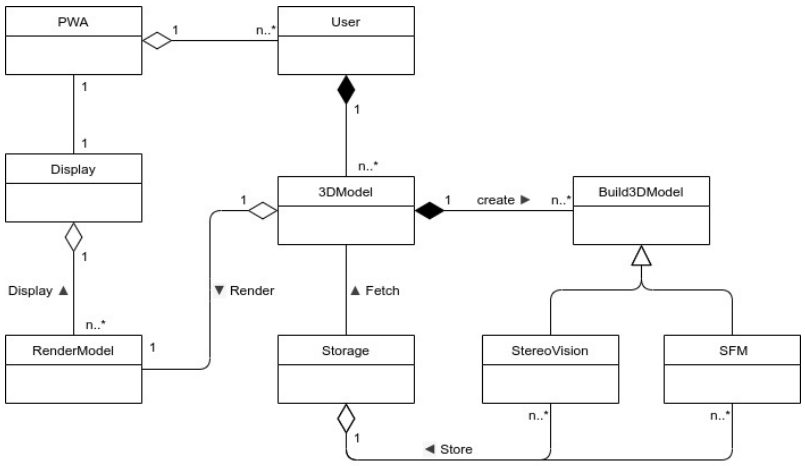
Full Name	Photo
Rani Arraf	
Quinn du Piesanie	
Jacobus Janse van Rensburg	
Steven Visser	
Marcus Werren	

This document will be updated, reviewed and adapted throughout the life cycle of the project.This is in accordance with Agile Principles.

1 Introduction

The vision of this project is to produce a system that allows dental professionals to scan and take photos of their patients teeth using cameras. This media will then in turn be processed and turned into a 3D digital model that can be inspected. The data for the 3D render will be stored and will be made available for retrieval in later stages.

1.1 Domain model



2 User Characteristics

As required by the client, there is one main user in mind when creating this project, however, the users may still expand from this.

2.1 Medical Professionals

These professionals are the main target for this project, focusing mainly on professionals in dental health. These users will make use of all of the subsystems provided by the web application, while focusing on the 3D renders of their patients teeth.

3 Functional Requirements

3.1 Use Cases

U1 Login

U2 Sign-up

U3 Add Patient

U4 View Patient

U5 Edit Patient

U6 Edit Details

U7 Logout

U8 Upload Media

U9 Render Model

U10 Inspect Model

U11 Export Model

U12 Record Media

3.2 Requirements

R1 System must allow users to take their own video input

- **R1.1** The user should be allowed to operate a camera

R2 System must allow users to give prerecorded media as an input

- **R2.1** The user should be allowed to choose what type of media they are providing.
- **R2.2** The user should be allowed to change the media provided.
- **R2.3** The system should upload the selected media to the database.

R3 System must allow users to generate the model

- **R3.1** The system should distinguish the type of media provided.
- **R3.2** The system needs to do all the calculations for the 3D render.
- **R3.3** The system needs to create a mesh from the calculations made.
- **R3.4** The system should send the information of the render back to the web page to be rendered if requested.
- **R3.5** The web page should render the returned data

R4 System should allow the user to inspect a rendered model

- **R4.1** The user should be able to zoom in and out of the render for better perspective.
- **R4.2** The user should be able to rotate the render.
- **R4.3** The user should be able to look around in the rendered view and move to different positions (FPS Mode)

R5 System should be able to store things in the database

- **R5.1** The user should be able to save the render to a database.
- **R5.2** The user should be able to save the media to a database.

R6 System must be able to retrieve information from the database

- **R6.1** The user must be able to retrieve a render data from the database.
- **R6.2** The user should be able to see the media stored with the render data.

R7 System must be able to export the rendered model

- **R7.1** Generate a 3D Printable structure
- **R7.2** Save the structure to the user's device

R8 System should allow users to create an account

- **R8.1** Allow Users to choose a username
- **R8.1** Allow Users to choose a password

R9 Allow users to login

- **R9.1** System should allow a user to log in with valid credentials
- **R9.1** System should allow a user to logout

R10 System should allow users to update patients

- **R10.1** Allow a user to create a patient
- **R10.2** The user must be allowed to fetch relevant information about a patient such as stored renders of the patients mouth/teeth.
- **R10.3** The user must be allowed edit and update a patients information for example adding new models of their mouth/teeth.

3.3 Subsystems

3.3.1 Media System

Subsystem is comprised of the camera that will be used to record media, recording of the media and uploading this or prerecorded media to the database. This media will be input to the render algorithm.

3.3.2 Web Page

This is the platform that the user will interact with the overall system. It encompasses all UI as well as display functionality.

3.3.3 Storage

This subsystem controls all storage and retrieval of information from databases stored on the server.

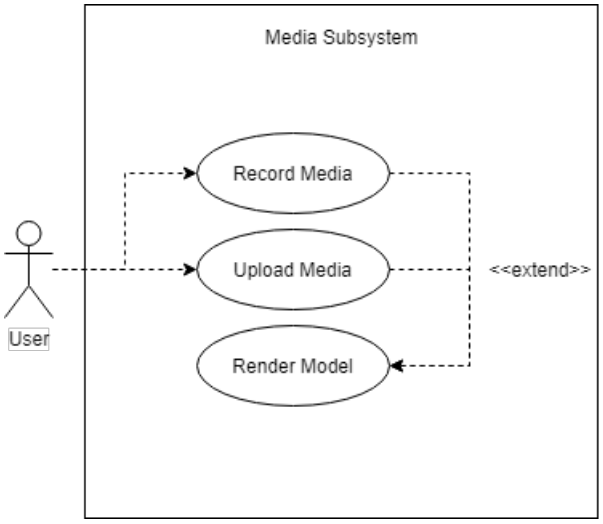
3.3.4 User Management

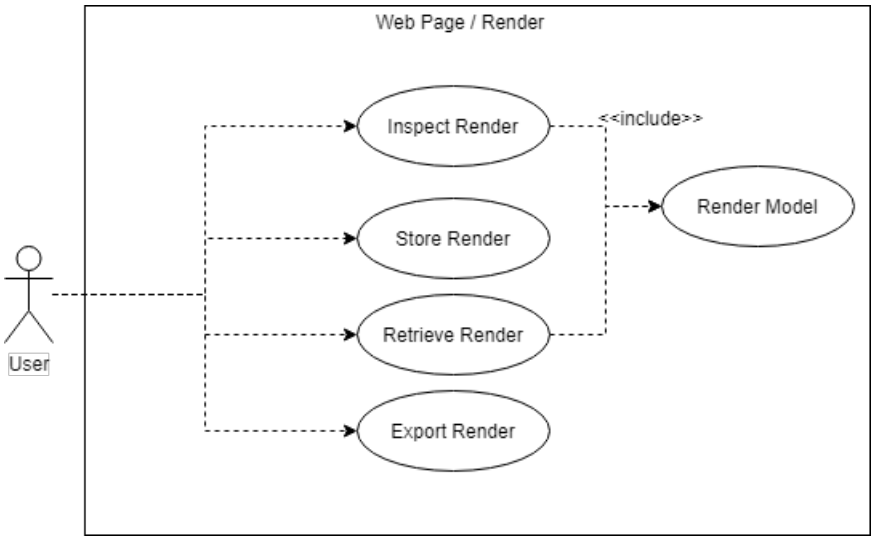
This subsystem encompasses all that is involved with creating users, logging into the system.

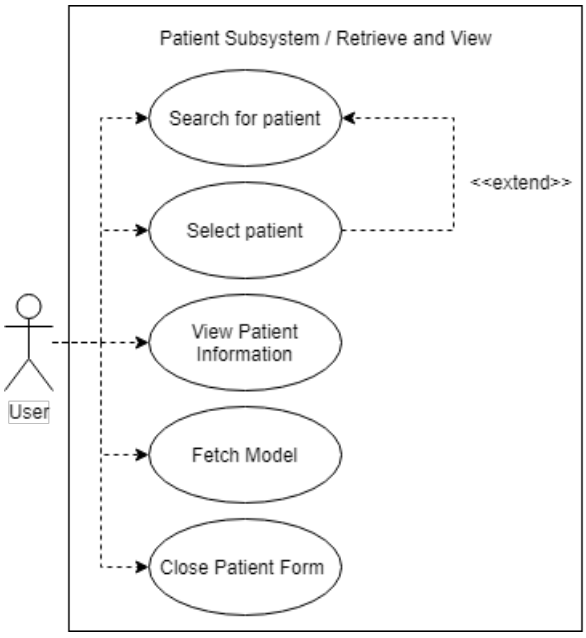
3.3.5 Patient Management

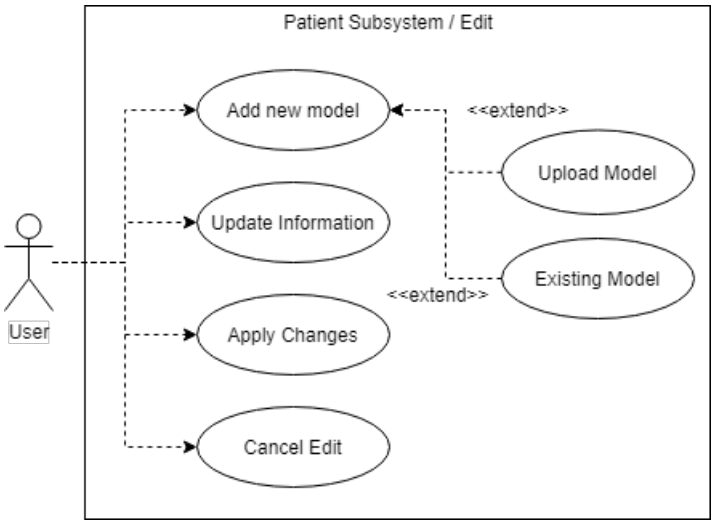
This subsystem encompasses all that is involved with creating patients and editing patient information.

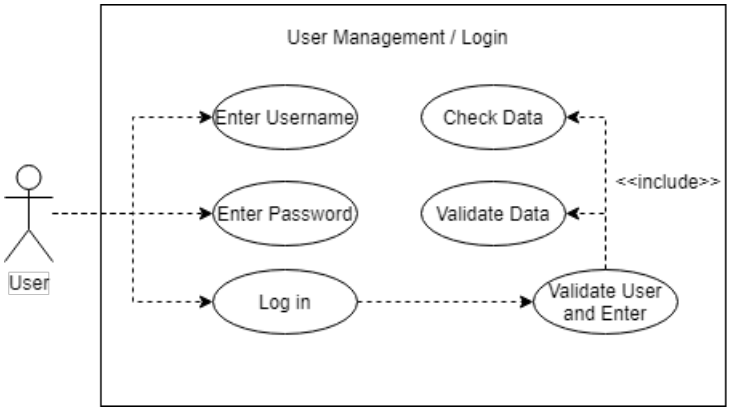
3.4 Use Case Diagrams

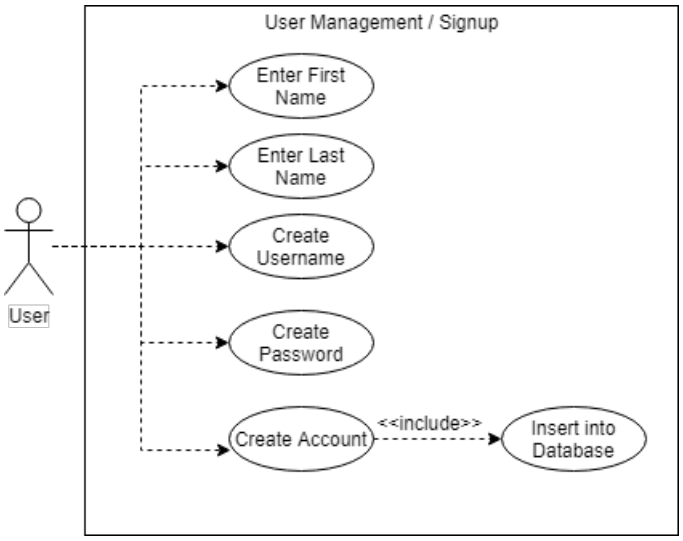



Web Page / Render
Inspect Render
<<include>>
Render Model
Store Render
Retrieve Render
Export Render
User


Patient Subsystem / Retrieve and View
Search for patient
<<extend>>
Select patient
View Patient Information
Fetch Model
Close Patient Form
User


Patient Subsystem / Edit
Add new model
<<extend>>
Upload Model
Update Information
Existing Model
<<extend>>
Apply Changes
Cancel Edit
User


User Management / Login
Enter Username
Check Data
<<include>>
Enter Password
Validate Data
Log in
Validate User and Enter
User



4 Quality Requirements

1 Performance:

The performance is very important in any aspect of the system we are developing. In the case of this project our performance will be the accuracy and the speed at which the render is computed and produced. We ideally would like to minimize the time taken to create the render and maximize the accuracy of the render. An important aspect to keep in mind is the camera used to take the media photo and the media type. Since the resolution of a camera can greatly impact the speed and accuracy of making the render.

2 Reliability:

One of the biggest frustrations that a user can experience is if an application freezes, crashes or gives incorrect results after you spent your precious time waiting for it to do its computations. We aim to prevent the application from breaking as much as we possibly can and to ensure that our results are accurate most of the time.To do this we intend to employ many "corner Case" tests to ensure that the user is using the system as intended.

3 Scalability:

Scalability refers to the ability of the system to expand. Our system was intended to be able to be integrated into the clients system that he is developing. Therefore our system is intended to be a little scalable but will in itself be an extension of another project.

4 Security:

To demonstrate how the doctors will use the system the user will be required to log in. The user will therefore require a username and a password in order to use the system. This login information will be used when storing the media and the render info. This will ensure that we know which Doctor stored which information and prevent other doctors to be able to retrieve information that does not belong to them which will be a violation of doctor-patient-confidentiality.

5 Flexibility:

The flexibility of the project refers to the system to work as it intended but also as the simplest version possible if some error is to occur. For our project we will implement the flexibility that the render information is computed on the server and not on the users personal computer in order to prevent their systems from crashing.

6 Maintainability:

Maintainability refers to the ability of the system to be easily repaired. The system will be divided into smaller subsystems that will each be able to only do their individual tasks. Therefore if something breaks it will be easy to find the error in its subsystem. We will employ appropriate error messages so that the developer knows where and why the error occurred. With proper unit/integration tests it will be easy to monitor the working state of the system.

7 Usability:

This requirement is related to how long it takes a user to becomes familiar with the system. For our project this is a special case since our system is intended to be just an extension of a much larger system and that our client wants to employ their own UI. For our sake we need to show them the basics of how the system will work and operate. It is due to this that the UI design will be very simplistic and we will ensure that making the media and interaction with the system is as simplistic as possible. For navigating the render we will use methods that are popular such as a first person shooter type of camera and allow the render to rotate about different axis' with pressing buttons.

5 Trace-ability Matrix

	SS1	SS2	SS3	SS4	SS5
R1	x				
R2	x				
R3	x	x	x		
R4		x	x		
R5	x		x		
R6	x	x	x		
R7			x		
R8			x	x	
R9			x	x	
R10			x		x
QR1	x	x			
QR2	x	x	x	x	x
QR3	x	x	x		
QR4			x	x	
QR5	x	x			
QR6	x	x	x	x	x
QR7	x	x		x	x