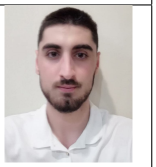
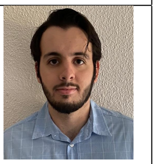
3D Model Binary Vision System

Requirement Specifications

Flap_Jacks

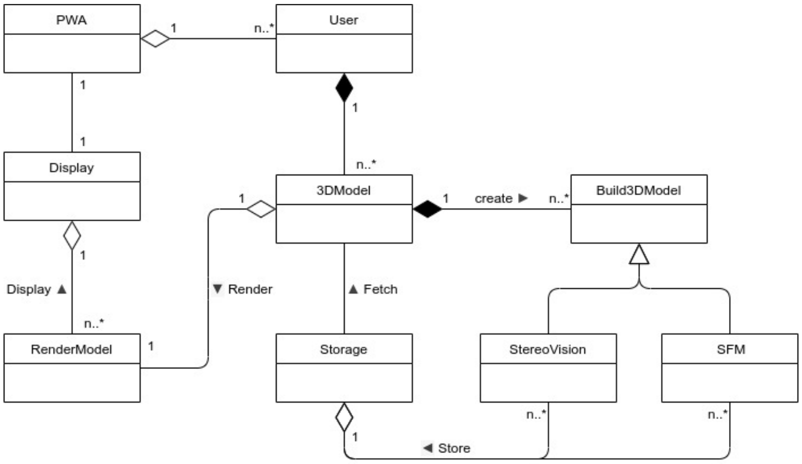
Full Name	Photo
Rani Arraf	
Quinn du Piesanie	
Jacobus Janse van Rensburg	
Steven Visser	
Marcus Werren	

This file needs to be updated and adapted throughout the project life cycle since we are following the agile principles. (Spec says we need not have a full SRS document for demo 1 but needs the sections already stated in the document)

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 investigated. The data for the 3D render will be stored and will be available for retrieval in later stages.

1.1 Domain model



2 User Characteristics

Due to the nature and the expertise needed to use the system it limits the users to a single field.

2.1 Medical Professionals

Our system will be used by professionals in Dental health and will use the system to create 3D renders of the tooth layout that the professional can look at and inspect to understand the nature of the dental problem. They will also record all the data of their patients.

3 Functional Requirements

3.1 Use Cases

UC1 User must be able to take there own video/Stereo input

- 1.1 The user should be allowed to operate the camera

UC2 User must to be able to give media as an input

- 2.1 The user should be allowed to choose what type of media they are providing.
- 2.2 The user should be allowed to change the media provided.
- 2.3 The system should upload the selected media to the server.

UC3 User must be able to render the

- 3.1 The system should distinguish the type of media provided.
- 3.2 The system needs to do all the calculations for the 3D render.
- 3.3 The system needs to create a mesh from the calculations made.
- 3.4 The system should send the information of the render back to the web page to be rendered if requested.
- 3.5 The web page should render the returned data

UC4 User must be able to inspect the render

- 4.1 The user should be able to zoom in and out of the render for better perspective.
- 4.2 The user should be able to rotate the render.
- 4.3 The user should be able to look around in the rendered view and move to different positions (FPS Mode)

UC5 User must be able to store the results

- 5.1 The user should be able to save the render to a database.
- 5.2 The user should be able to save the media to a database.

UC6 User must be able to retrieve stored results

- 6.1 The user must be able to retrieve a render data from the database.
- 6.2 The user should be able to see the media stored with the render data.

UC7 User must be able to export the rendered model

- 7.1 The user must be able to export the rendered model of the patients mouth/teeth so that the model can be 3-D printed.

UC8 User must be able to create an account

8.1 The user must be able to create an account so that all the functionalities of the software can be made available to the user.

UC9 The user must be allwed to log into

9.1 The user should be able to provide authentication data

UC10 The user must be allwed to logout of their account

10.1 The user should be able to suspend their active session with the software.

UC11 The user should be able to create a new patient

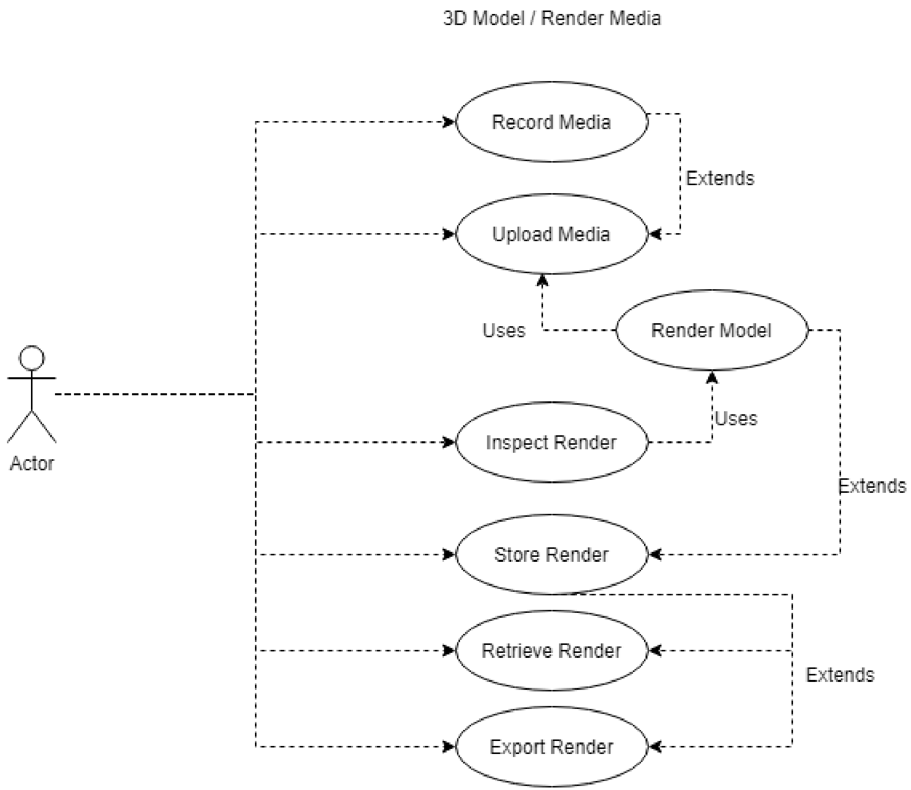
11.1 The user must be allowed to create the user with the data associated with the user.

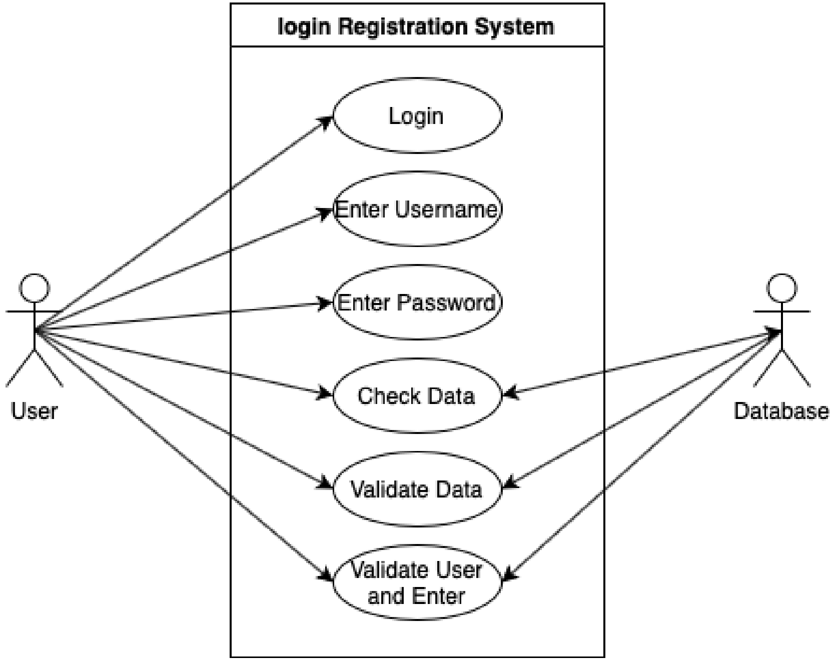
UC12 The user should be able to retrieve a patients information

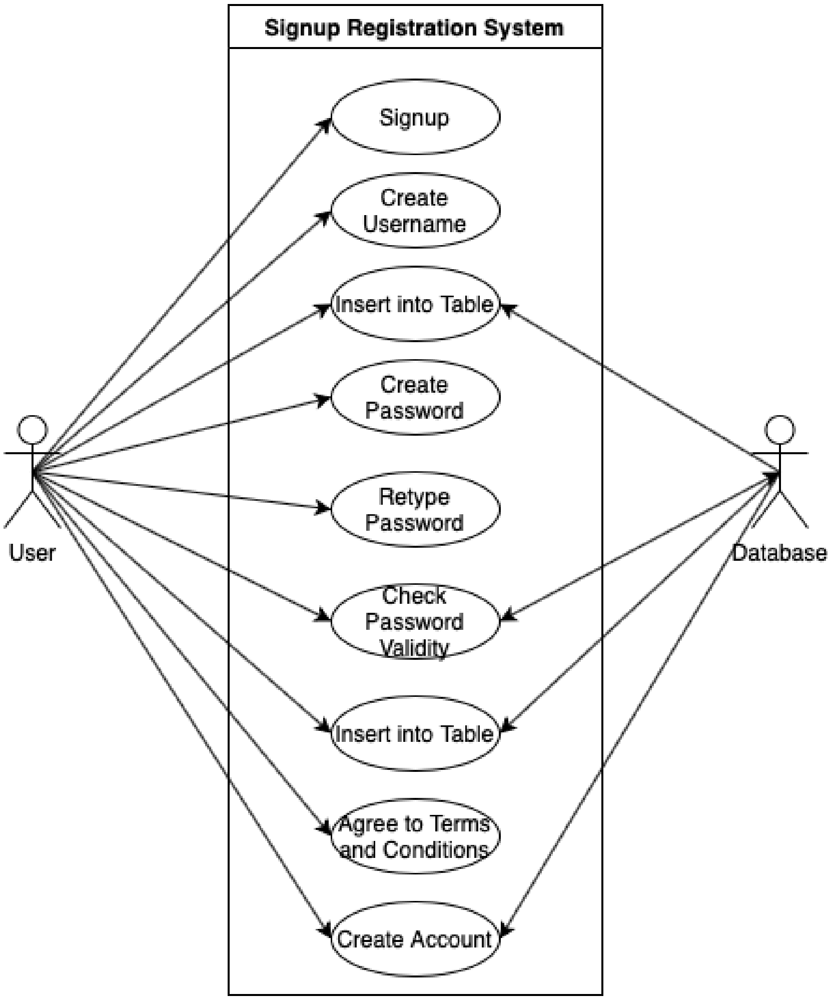
12.1 The user must be allowed to fetch relevant information about a patient such as pre-stored renders of the patients mouth/teeth.

UC13 The user should be able to edit patient information

13.1 The user must be allowed edit and update a patients information for example adding new models of their mouth/teeth.


3D Model / Render Media
Record Media
Extends
Upload Media
Uses
Render Model
Actor
Inspect Render
Uses
Extends
Store Render
Retrieve Render
Extends
Export Render


login Registration System
Login
Enter Username
Enter Password
User
Database
Check Data
Validate Data
Validate User and Enter



3.2 Requirements

R1 Build 3-D Model

R2 Render Model

R3 Store Render

R4 Display Model

R5 Export Model

R6 Create User

R7 Create/Edit Patient

3.3 Subsystems

///need to add these things to the use case diagrams as well with ¡¡extends¿¿ and ¡¡includes¿

This project will be split into a numerous amount of subsystems. These subsystems are labelled and explained below.

Media System: This system is comprised of the cameras that will is used to capture the media(video , stereo) that is to be used as input to create the render.

Web Page: The web page is the platform on which the user will be able to upload (use) the media that was captured. The web page will also be used to be a platform on which the render will take place.

Server: The server will host the database, API's and the program to determine the properties of the render.

Database: The database will be able to store the media that you have captured, the render info for that media, a date it was uploaded and a doctor the entry belongs to.

API: not so sure what to say here yet

4 Quality Requirements

Performance:

The preformance 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.

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.

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.

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.

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.

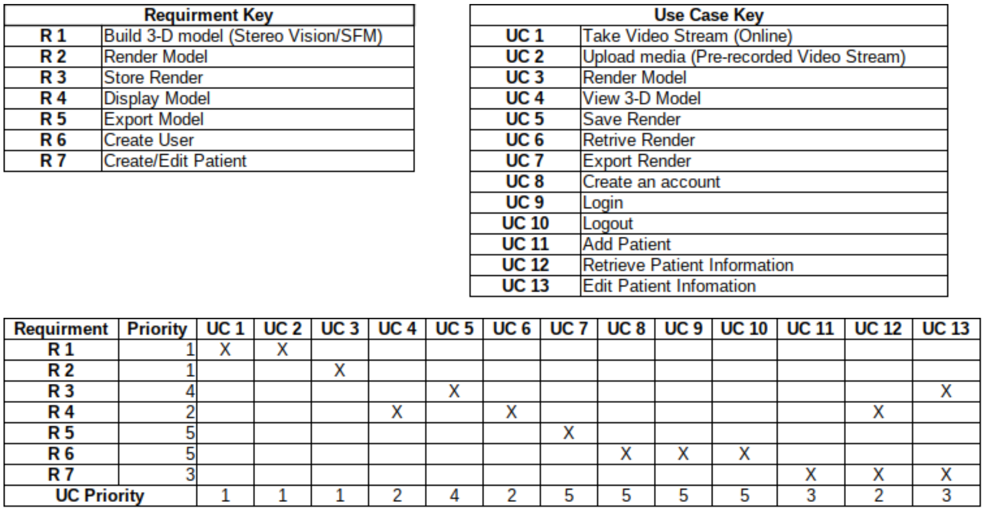
Maintainability:

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.

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



Requirment Key	
R 1	Build 3-D model (Stereo Vision/SFM)
R 2	Render Model
R 3	Store Render
R 4	Display Model
R 5	Export Model
R 6	Create User
R 7	Create/Edit Patient

Use Case Key	
UC 1	Take Video Stream (Online)
UC 2	Upload media (Pre-recorded Video Stream)
UC 3	Render Model
UC 4	View 3-D Model
UC 5	Save Render
UC 6	Retrive Render
UC 7	Export Render
UC 8	Create an account
UC 9	Login
UC 10	Logout
UC 11	Add Patient
UC 12	Retrieve Patient Information
UC 13	Edit Patient Infomation

Requirment	Priority	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC 10	UC 11	UC 12	UC 13
R 1	1	X	X											
R 2	1			X										
R 3	4					X								X
R 4	2				X		X						X	
R 5	5							X						
R 6	5								X	X	X			
R 7	3											X	X	X
UC Priority		1	1	1	2	4	2	5	5	5	5	3	2	3