

University of Reading  
Department of Computer Science

# An AI-assisted decision making system for thyroid nodule classification

Stefanos Stefanou

*Supervisor:* Huizhi Liang

A report submitted in partial fulfilment of the requirements of  
the University of Reading for the degree of  
Bachelor of Science in *Computer Science*

April, 2021

## Declaration

I, Stefanos Stefanou, of the Department of Computer Science, University of Reading, confirm that all the sentences, figures, tables, equations, code snippets, artworks, and illustrations in this report are original and have not been taken from any other person's work, except where the works of others have been explicitly acknowledged, quoted, and referenced. I understand that if failing to do so will be considered a case of plagiarism. Plagiarism is a form of academic misconduct and will be penalised accordingly.

Stefanos Stefanou  
April, 2021

# Abstract

Deep learning has found numerous applications in the health care community. Recently, a massive explosion of research on the relevant field, driven by large amounts of available data, has generated important disease prevention and identification results. Fine Needle Aspiration (FNA) is the dominant procedure for thyroid nodule classification. FNA has associated risks and expenses, and in this project, we will try to reduce both using the recent advancements in Artificial Intelligence and Deep Learning. Our primary goal is to bring closer the radiologists 'on the field' with those complex algorithms and provide value to real patients by providing an interface, in the form of a web application, for probabilistically predicting the severity and the category of a given module.

# Contents

<b>Abstract</b>	<b>ii</b>
<b>1 Introduction</b>	<b>2</b>
<b>2 Literature Review</b>	<b>3</b>
2.1 Introduction . . . . .	3
2.2 Brief Table of books . . . . .	3
2.3 Brief Table of papers . . . . .	3
<b>3 Requirement Analysis</b>	<b>5</b>
3.1 Introduction . . . . .	5
3.2 Functional Requirements . . . . .	5
3.3 Non Functional Requirements . . . . .	6
<b>4 Entity Relation Analysis</b>	<b>7</b>
4.1 Introduction . . . . .	7
4.2 Entities . . . . .	7
4.2.1 Scan . . . . .	7
4.2.2 Patient . . . . .	7
4.2.3 Doctor . . . . .	8
4.2.4 Notification . . . . .	8
4.3 Entity Relations . . . . .	8
<b>5 Users Perspective</b>	<b>9</b>
5.1 Introduction . . . . .	9
5.2 Login and Authentication . . . . .	9
5.3 Home . . . . .	10
5.4 Navigation bar . . . . .	10
5.4.1 Profile . . . . .	11
5.4.2 Notifications . . . . .	11
5.4.3 About . . . . .	12
5.5 Action Bar . . . . .	12
5.5.1 My Patients . . . . .	12
5.5.2 New Patient . . . . .	13
5.5.3 My Scans . . . . .	14
5.5.4 New Scan . . . . .	15

<b>6</b>	<b>System Architecture</b>	<b>17</b>
6.1	Overview . . . . .	17
6.2	Frontend Web App . . . . .	18
6.3	Backend . . . . .	18
6.3.1	Information Backend . . . . .	18
6.3.2	Task Backend . . . . .	19
6.4	Benefits from microservice-inspired architectures . . . . .	19
6.4.1	Separation of concerns . . . . .	19
6.4.2	Enhanced Security . . . . .	19
6.4.3	Flexible scalability factor . . . . .	19
<b>7</b>	<b>The Frontend</b>	<b>20</b>
7.0.1	Technology Stack . . . . .	20
<b>8</b>	<b>The Frontend</b>	<b>22</b>
8.0.1	Technology Stack . . . . .	22
<b>9</b>	<b>The Backend</b>	<b>24</b>
9.1	. . . . .	24
9.2	. . . . .	24
9.2.1	. . . . .	24
9.3	Summary . . . . .	24
<b>10</b>	<b>The Prediction Service</b>	<b>25</b>
10.1	. . . . .	25
10.2	. . . . .	25
10.2.1	. . . . .	25
10.3	Summary . . . . .	25
<b>11</b>	<b>Discussion, Conclusion and Future work</b>	<b>26</b>
11.1	. . . . .	26
11.2	. . . . .	26
11.2.1	. . . . .	26
11.3	Summary . . . . .	26
<b>12</b>	<b>Reflection</b>	<b>27</b>
12.1	. . . . .	27
12.2	. . . . .	27
12.2.1	. . . . .	27
12.3	Summary . . . . .	27

## **Acknowledgments**

Acknowledgments section is optional. You may like to acknowledge the support and help of your supervisor(s), friend(s), or any other person(s), department(s), institute(s).

# Chapter 1

## Introduction

Deep learning has found numerous applications in the health care community. Recently, a massive explosion of research on the relevant field, driven by large amounts of available data, has generated important disease prevention and identification results.[Eun Ju Ha (2021)]

The Dominant process for thyroid nodule identification and classification is called FNA (or Fine Needle Aspiration/Biopsy). FNA is an expensive process requiring expensive lab equipment and specialized personnel.[*Fine Needle Aspiration Biopsy of Thyroid Nodules* (2019)]

There is no way to determine the category of a thyroid nodule apart from performing FNA on a sample. Our vision is to create an application to act as a bridge between the academic community working on theoretical Deep Learning models to predict a nodule's category and the radiologists working with actual patients and accurate data. Our hope is that by establishing a common language(the application) we will improve the research process as experimental models will work on accurate nodule scans, providing instant feedback to the researchers for further analysis.

Our system needs to be as generic as possible to support any prediction model, reliable, easy to maintain, and expand. It needs to be optimized to handle the Deep Learning models and finally needs to be as secure as possible because it will eventually work with actual patients on sensitive data.

### Abbreviations

FNA(Fine Needle Aspiration), AI(Artificial Intelligence), DP(Deep Learning)

### Keywords

FNA, AI, DP

## Chapter 2

# Literature Review

### 2.1 Introduction

This section will note the essential sources needed to be studied and to be revised to complete this project. The sources are carefully selected to include theoretical, practical, and best practices knowledge in order to cover the wide variety of topics needed to fulfill the requirements of this project.

### 2.2 Brief Table of books

ISBN	Name	Type
N/A	ST1PS-18-9A: Probability and Statistics (2018/19)	Module Lectures
9780030105678	Linear Algebra and Its Applications	Book
9780131687288	Digital Image Processing	Book
9780262035613	Deep Learning	Book
9780128104088	Deep Learning for Medical Image Analysis	Book
9781491962244	Hands-on machine learning with scikit-learn and tensorflow	Book

### 2.3 Brief Table of papers

- Ye, H., Hang, J., Chen, X. et al. An intelligent platform for ultrasound diagnosis of thyroid nodules. Sci Rep 10, 13223 (2020). <https://doi.org/10.1038/s41598-020-70159-y>
- Nguyen DT, Pham TD, Batchuluun G, Yoon HS, Park KR. Artificial Intelligence-Based Thyroid Nodule Classification Using Information from Spatial and Frequency Domains. J Clin Med. 2019;8(11):1976. Published 2019 Nov 14. doi:10.3390/jcm8111976
- Manivannan T, Ayyappan N. Classification of thyroid nodules using ultrasound images. Bioinformation. 2020;16(2):145-148. Published 2020 Feb 29. doi:10.6026/97320630016145
- Nguyen DT, Kang JK, Pham TD, Batchuluun G, Park KR. Ultrasound Image-Based Diagnosis of Malignant Thyroid Nodule Using Artificial Intelligence. Sensors (Basel). 2020;20(7):1822. Published 2020 Mar 25. doi:10.3390/s20071822
- Chen J, You H, Li K. A review of thyroid gland segmentation and thyroid nodule segmentation methods for medical ultrasound images. Comput Methods Programs



Biomed. 2020 Mar;185:105329. doi: 10.1016/j.cmpb.2020.105329. Epub 2020 Jan 9. PMID: 31955006.

- Ha EJ, Baek JH. Applications of machine learning and deep learning to thyroid imaging: where do we stand? Ultrasonography. 2021 Jan;40(1):23-29. doi: 10.14366/usg.20068. Epub 2020 Jul 3. PMID: 32660203; PMCID: PMC7758100.

## Chapter 3

# Requirement Analysis

### 3.1 Introduction

Before we even start exploring this project and its features, it is essential to define the requirements that need to be fulfilled strictly and this project's scope. Failing to perform a requirement analysis beforehand puts additional and unnecessary risks to the project due to the project's unspecified and volatile scope and target set.

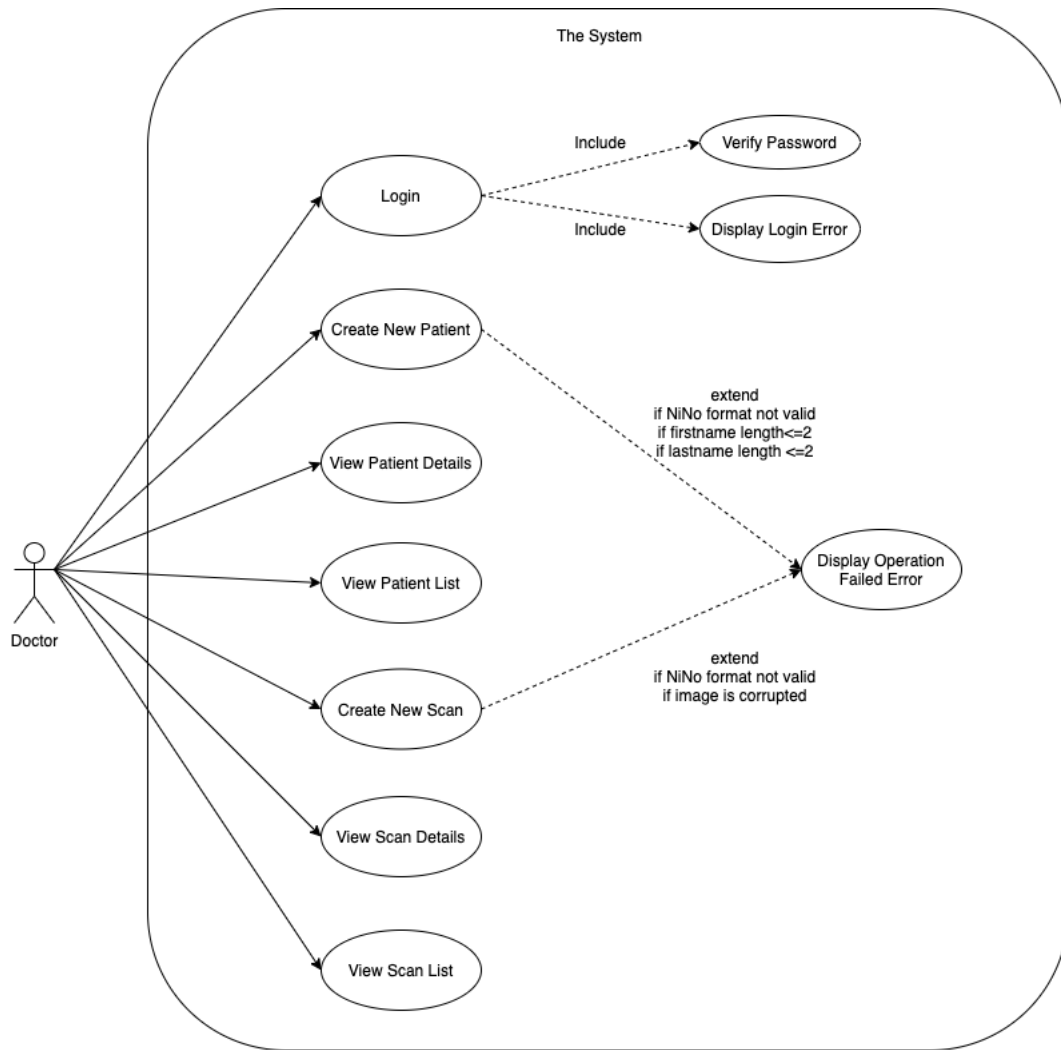
### 3.2 Functional Requirements

Functional requirements define the basic system behavior. We define the functional requirements as follows.

- User needs to log in with a personal password.
- User needs to be able to create a new patient record
- User needs to be able to upload a new ultrasound scan image associated with a given patient
- User needs to be able to see its associated patients
- User needs to be able to see its uploaded ultrasound images
- User needs to be able to search for a specific patient
- User needs to be able to see a list of all ultrasound images for a specific patient
- User needs to be able to see the details of a specific patient
- User needs to be able to see the details of a specific submitted scan, as well as the prediction results if available.
- User should be notified if the prediction results are ready

Those requirements can be easily visualized in a Use Case Diagram, given below.

Figure 3.1: Use Case Diagram



### 3.3 Non Functional Requirements

Nonfunctional requirements are the properties of the system; an comprehensive list of the agreed nonfunctional requirements is given below

- The system must be secure, as it handles the personal information of the patients
- The system should be reliable, as downtimes are affecting the hospital's performance
- The system should be able to complete a prediction scan in a reasonable amount of time(1-10 mins)

## Chapter 4

# Entity Relation Analysis

### 4.1 Introduction

After the requirements have been set. We need to translate them into workable relational entities in order to be able to modeled through a classical relational database system (RDBMS).

### 4.2 Entities

We will start our exploration by defining our entities for this project.

#### 4.2.1 Scan

A scan is the result of an ultrasound scan performed in a specific patient(see [4.2.2]). A scan entity has certain attributes

Image	The image produced by the ultrasound scan. 360x560 pixels
Prediction	The result of the prediction algorithm. Acceptable Values = Malignant, Benign
Results	The logs of the algorithm performed the prediction, Optional
Algorithm	The algorithm used to perform the prediction. Acceptable Values = SVC,RES
Token	The scan identifier across the application services. token type is UUID [Leach (2005)]

#### 4.2.2 Patient

A patient is a physical person that is suspected to have a thuroid nodule. A person may have 0 up to n scans, where n is the theoretical maximum number of records(no limit is enforced by the database or the application). A patient has characteristics explained below

First Name	The first name of the patient.
Last Name	The last name of the patient
NiNo	The National Insurance Number(NiNo) of the patient
Enrolled Date	The Date that the patient was registered in the system
Ascosiate Doctor	The Doctor identification number, handling the case of the patient(see 4.2.3)
Comments	The Doctors(see 4.2.3) comments for the particular patient

### 4.2.3 Doctor

A doctor is a physical person with access on the system. Is the end-user of the system and has rights of uploading ultrasound image scans and retrieve predictions for those scans. It can also provide feedback to the system for a given prediction to be used for further research and developement. A doctor has specific characteristics presented below.

Username	Plain-text username
Password	MD5 Hashed[Rivest (1992)] and salted[Manber (1996)] password
First Name	National Insurance Number[ <i>National Insurance Manual</i> (2021)]
Last Name	The date that the patient was registered in the system
Title	The title of the doctor.
Enrolled Date	The date and time of the user enrolled to the system
Last Seen	The date and time of last login of the user
Online Status	The status of the user, acceptable values are Connected,Not Connected
Tasks	The number of scans uploaded by the user

### 4.2.4 Notification

A notification is a short message from the system to the end-user(The doctor). Its sole purpose is to inform the user about various events that may interest the end-user. An example of this may be that the scan results for a given scan task are ready to view. A notification has specific characteristics witch are displayed and explained below.

Message	The message in question
Ascociated Doctor	The receipient doctor identification number
Created Date	The Date and Time where the event in question where happened

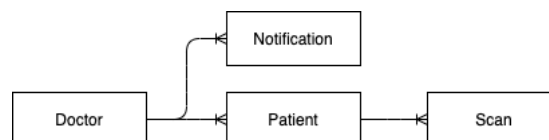
## 4.3 Entity Relations

The aforementioned entities have well defined relations. An exhaustive list is given below

- A doctor has many patients ( $1 - \infty$ )
- A doctor has many notifications( $1 - \infty$ )
- A Patient has many Scans( $1 - \infty$ )

A above relations can be summarized in the following E-R<sup>1</sup> Diagram

Figure 4.1: Entity-Relation Diagram



<sup>1</sup>Entity-Relation

## Chapter 5

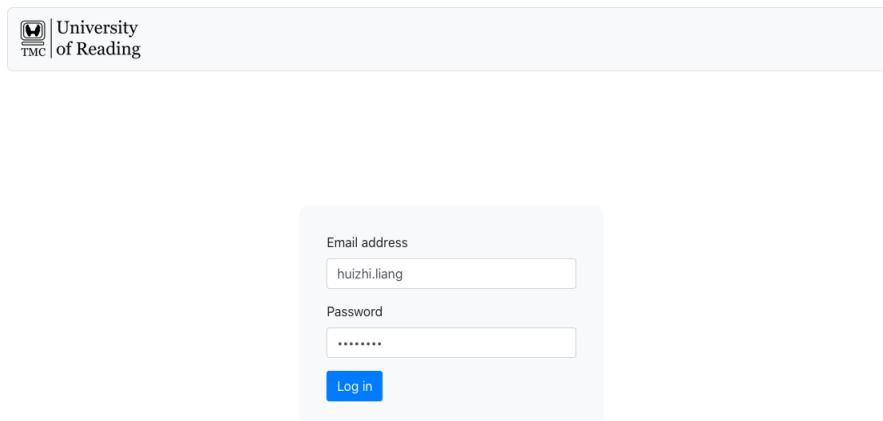
# Users Perspective

### 5.1 Introduction

In this section, we will start our exploration of the application and its features. As the nature of the requirements of this the system is complicated. Unavoidably the system will be complex as well. Taking this into consideration, we will follow a natural top-to-bottom approach explaining its internals, starting as end-users and seeing the system as a black box. In this section, we will analyze its functionality from the user's perspective. This section may also serve as an instruction manual for the end-user as it contains everything needed for an inexperienced user to start working with the software.

### 5.2 Login and Authentication

Figure 5.1: Login Screen



The screenshot shows a web-based login interface. At the top, there is a header bar with the University of Reading logo and name. Below the header, the login form is centered. It includes a label 'Email address' above a text input field containing 'huizhi.liang'. Below that is a label 'Password' above a password input field filled with asterisks. A blue 'Log in' button is positioned at the bottom of the form.

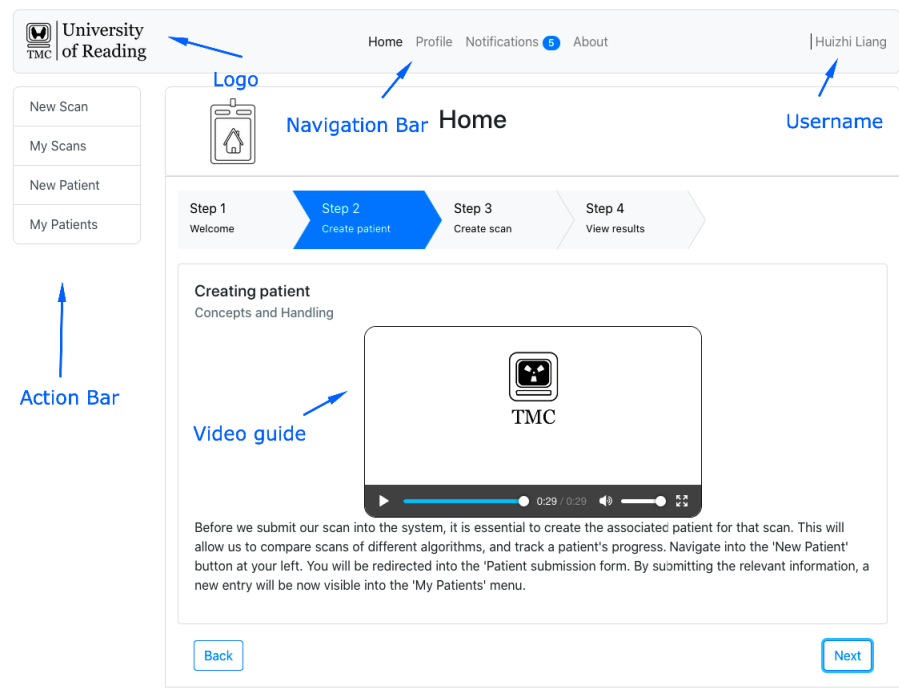
The login screen is the first screen that our end-users will encounter. Here a username and a password is required to be given by the user to log in. The Credentials of the user remain encrypted during the process of login, as the system utilizes an HTTPS[Rescorla (2000)] protocol for its connection, this is essential for the first non-functional requirement about security (see 3.3). The username and the password may be requested by the system administrator or the NOC<sup>1</sup> of the hospital.

---

<sup>1</sup>Network Operations Center

5.3 Home

Figure 5.2: Home Screen



After the login process is completed. The user encounters the 'home screen. From here, it is possible to navigate to the features of the software as well as learn about how the software can be utilized through detailed guides and videos. The UI/UX<sup>2</sup> has been designed to be as user-friendly as possible. Some areas of interest are given below.

Action bar	The Actions that can be performed using the software can be accesed from here. General information and notification bar.
Navigation bar	

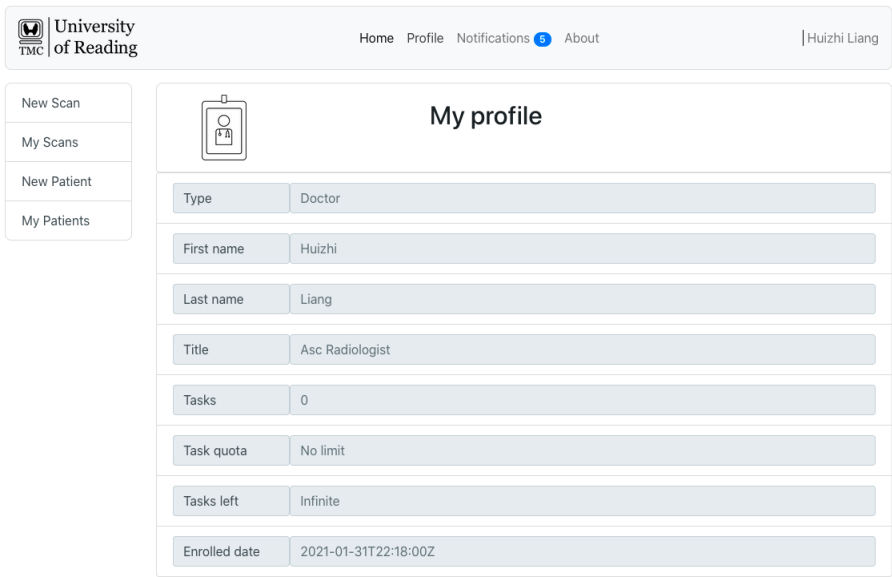
5.4 Navigation bar

In this section, we will briefly look at the options under the Navigation bar.

<sup>2</sup>User Interface-User Expieriance

5.4.1 Profile

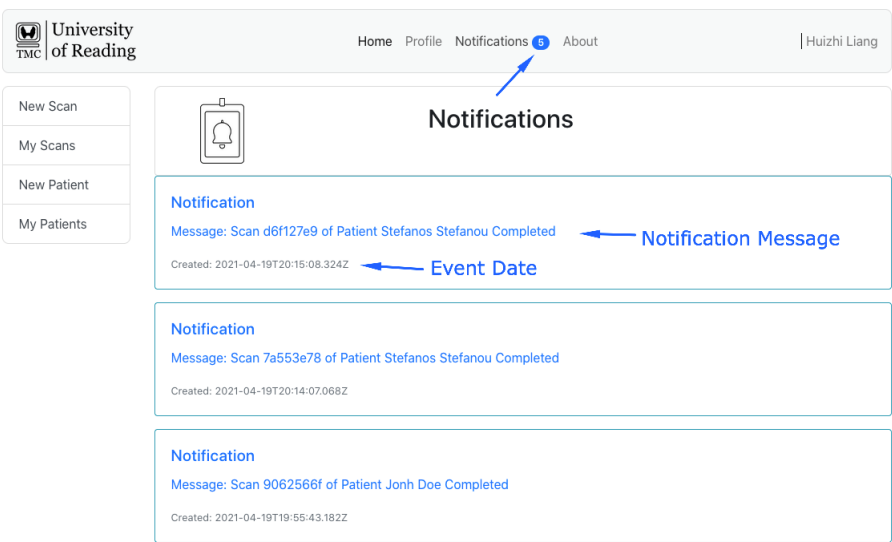
Figure 5.3: Profile



In the profile section, the user can see its associated information, saved on the registration date. The information for security reasons cannot be altered by the user itself, but only after a request to the system administrator or NOC<sup>3</sup>.

5.4.2 Notifications

Figure 5.4: Notifications



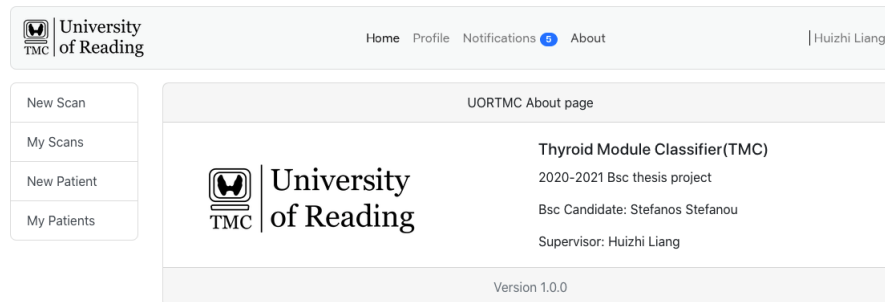
In the notification section, helpful information about events that may interest the end-user can be found, such as the fact that uploaded scan results are ready to view.

<sup>3</sup>Network Operations Center



### 5.4.3 About

Figure 5.5: About



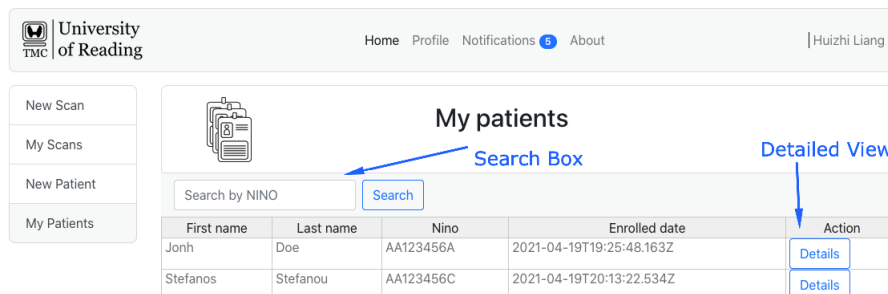
From here, a user may find helpful information about the software, such as the current version.

## 5.5 Action Bar

In this section, we will briefly look at the options under the Action Bar.

### 5.5.1 My Patients

Figure 5.6: Patients List



This page will show us a list of the currently registered patients. Each end-user(doctor) may only see its patients and not others. The end-user can search the list based on NiNo[*National Insurance Manual* (2021)] of the given patient for convenience. The end-user can also view the details of a given patient and record various notes/comments for that patient by clicking the 'Details' button on his selected patient, as seen below. Finally, clicking the button 'View Scans' can see the specific patient history of uploaded scans.

Figure 5.7: Patient Details

University of Reading TMC | Home Profile Notifications 5 About | Huizhi Liang

New Scan  
My Scans  
New Patient  
My Patients

### Patient details

Type Patient

First name Jonh

Last name Doe

Nino AA123456A

Enrolled date 2021-04-19T19:25:48.163Z

Comments Not Set

Scans of the patient →

Save changes View scans

### 5.5.2 New Patient

Figure 5.8: New Patient

University of Reading TMC | Home Profile Notifications 5 About | Huizhi Liang

New Scan  
My Scans  
New Patient  
My Patients

### Patient submission form

Type Patient

First name

Last name

Nino AA123456C

Enrolled date Sun Apr 25 2021

Create new patient

By clicking the 'New Patient' action on Action Bar, the user can register a new patient on the system. The following conditions need to be met for the operation to be successful.

- First name length should be more than 2 characters, encoded as UTF-8[Yergeau (2003)]
- Last name length should be more than 2 characters, encoded as UTF-8[Yergeau (2003)]
- NiNo should be at standard format [*National Insurance Manual* (2021)], encoded as UTF-8[Yergeau (2003)]

Failing to fulfill these constraints should lead to an error, as shown below.

Figure 5.9: New Patient Error

The screenshot shows the 'New Patient' form in the 'University of Reading' system. The form is titled 'Patient submission form'. It contains fields for 'Type' (set to 'Patient'), 'First name' (J), 'Last name' (Doe), 'Nino' (AA123), and 'Enrolled date' (Sun Apr 25 2021). A 'Create new patient' button is at the bottom. An error message box is displayed at the top right, stating 'System Operation Failed: Please insert valid values on the respective inputs'. A blue arrow points to the error message box.

**System** just now ✕  
Operation Failed: Please insert valid values on the respective inputs

**Error Message**

### 5.5.3 My Scans

Figure 5.10: My Scans

The screenshot shows the 'My Scans' page in the 'University of Reading' system. It features a search box labeled 'Search by NINO' and a 'Search' button. Below the search box is a table of scans. A blue arrow points to the search box, and another points to the 'Details' button in the 'Action' column. The table has columns for 'First name', 'Last name', 'Nino', 'Created date', 'Status', 'Identifier', and 'Action'.

**My scans**

**Search Box**

Search by NINO

**Scan Details**

First name	Last name	Nino	Created date	Status	Identifier	Action
Jonh	Doe	AA123456A	2021-04-19T19:29:28.607Z	COMPLETED	c5fcd9c8	<a href="#">Details</a>
Jonh	Doe	AA123456A	2021-04-19T19:54:47.063Z	COMPLETED	f12cc82e	<a href="#">Details</a>
Jonh	Doe	AA123456A	2021-04-19T19:55:42.686Z	COMPLETED	9062566f	<a href="#">Details</a>
Stefanos	Stefanou	AA123456C	2021-04-19T20:13:57.502Z	COMPLETED	7a553e78	<a href="#">Details</a>
Stefanos	Stefanou	AA123456C	2021-04-19T20:15:07.579Z	COMPLETED	d6f127e9	<a href="#">Details</a>

'MyScans' are a complete list with all submitted scans for a given end-user. The user can search for the scans of a specific patient by using the search box and viewing the scan results (if a given scan is complete) by clicking the 'Details' button of the scan in question.

Figure 5.11: Scan results

The screenshot shows the 'Scan details' page. On the left is a sidebar with links: 'New Scan', 'My Scans', 'New Patient', and 'My Patients'. The main content area has a header 'Scan details' and two tabs: 'General Information' and 'Results'. The 'Results' tab is active, displaying an ultrasound image. Below the image, there is a 'Classification' section with a 'Malignant' label, a 'Prediction' label with a blue arrow pointing to it, an 'Agree?' section with a 'No' button, and a 'Comments' section with a 'Not Set' label and a 'Doctor Feedback' label with a blue arrow pointing to it. A 'Save Changes' button is at the bottom.

### 5.5.4 New Scan

Figure 5.12: New Scan

The screenshot shows the 'Scan submission form'. On the left is a sidebar with links: 'New Scan', 'My Scans', 'New Patient', and 'My Patients'. The main content area has a header 'Scan submission form' and a form with the following fields: 'Type' (Scan), 'Created date' (Sun Apr 25 2021), 'Patient's nino' (AA123456C), 'Algorithm' (a dropdown menu with '✓ (SVC) Simple C-Support Vector Machine v1' and '(RES) Residual neural network V18' options), and 'Scan image' (a 'Browse...' button and 'No file selected.' text). A 'Submit new scan' button is at the bottom.

By clicking the 'New Scan' action, the user is redirected into the scan submission form. Here it is possible to submit a new ultrasound image for a given patient. The user can also select the algorithm for performing the classification (see 10). The operation to be completed should meet the following criteria.

- Patients Nino should be in standard format[*National Insurance Manual* (2021)] and encoded as UTF-8[Yergeau (2003)]

- Scan Image should be a ISO/IEC 10918-1/JPEG[International Organization for Standardization (1994)] format with 360x560 resolution. The image name should be encoded as UTF-8[Yergeau (2003)]

Failing to fulfill these constraints should lead to an error, as shown below.

Figure 5.13: Invalid image format (PNG) error

The screenshot shows a web application interface for a 'Scan submission form'. The header includes the 'University of Reading' logo and navigation links: Home, Profile, Notifications (5), and About. The user 'Huizhi Liang' is logged in. On the left, there is a sidebar with links: New Scan, My Scans, New Patient, and My Patients. The main form area is titled 'Scan submission form' and contains the following fields:

- Type: Scan
- Created date: Sun Apr 25 2021
- Patient's nino: AA123456A
- Algorithm: (SVC) Simple C-Support Vector Machine v1
- Scan image: Browse... Screenshot 2021-04-24 at 16.04.22.png

A blue button labeled 'Submit new scan' is at the bottom of the form. An error message is displayed in a system notification box, stating: 'Operation Failed: Given image base64 string wasn't decoded successfully, possibly corrupt data or invalid data format? (Note that jpeg files are supported only in this version)'. A blue arrow points from the 'Error Message' text to the error message box.

## Chapter 6

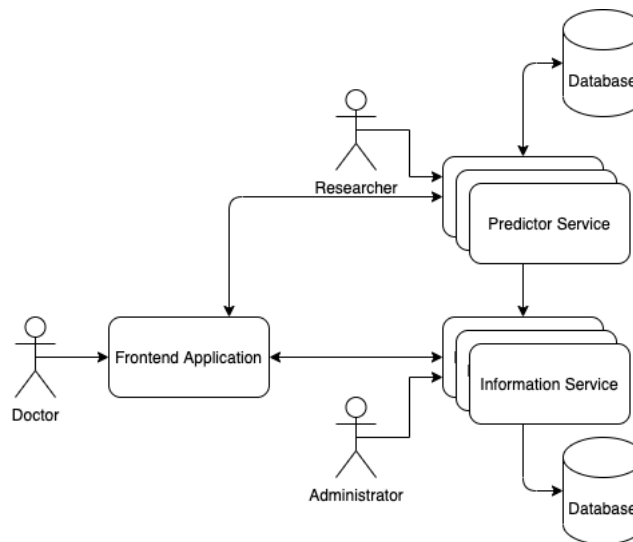
# System Architecture

In this chapter, we will introduce the architecture of our system, explaining the essential elements that is composed of, and their interactions.

### 6.1 Overview

In the section 3.3 we disussed the non-functional requirements of this application. Two of the most important ones were security and performance. This requirements heavily influenced the design desisions of this project, leading to the microservice-inspired architecture [Newman (2020)]shown below

Figure 6.1: Simplified Architectural Diagram

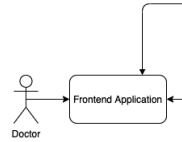


Microservice pattern tries to decrease complexity and increase safety by splitting the internal logic of a system into several components called 'Microservices'. Each microservice is essentially a server that handles a small portion of the systems logic. As opposed to the monolithic services, microservices have a number of advantages that made them ideal for the requirements of this project(see 6.4)

## 6.2 Frontend Web App

The Frontend component has the responsibility of being the edge in our system.

Figure 6.2: Frontend Application



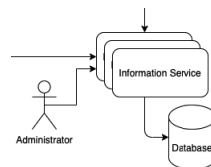
Every action from our end-users, will be channeled through the frontend application. Our frontend is a web-based application (for more information please see 7) and handles the application infrastructure via a well designed stateful [Barth (2011)], Json-based [T. Bray (2014)] Https [Rescorla (2000)] RESTFull [R. Fielding (2014)] protocol (for more information please see 8).

## 6.3 Backend

The backend services are the backbone of our application. They handle all the logic behind the application, from the saving and retrieval of patients, scans, images and notifications, to prediction and classification of the ultrasound images. There are 2 services with distinct areas of interest and different purposes, the Information Backend (also known as 'Information Service' to the simplified diagrams) and Task Backend (Also known as Predictor Service)

### 6.3.1 Information Backend

Figure 6.3: Information Service



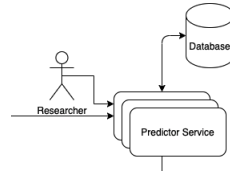
The Information Backend has the responsibility to perform all the logic apart from the prediction itself. It includes the functionality of keeping the associations of Patients, Scans as well as the information composing those entities. It also includes the notification system and the authentication services. Finally it includes a mini application for use by the NOC<sup>1</sup> for administrator purposes.

---

<sup>1</sup>Network operations center

### 6.3.2 Task Backend

Figure 6.4: Predictor Service



The Task Backend has the responsibility of performing the predictions based on the received ultrasound scan images that it receives. After a completion of a prediction, Task Backend should communicate with Information Backend to inform the user about the completion of the scan.

## 6.4 Benefits from microservice-inspired architectures

- Highly maintainable and testable
- Loosely coupled
- Independently deployable
- No Single Point of Failure

different scalability. separation of concerns. enhanced security, as breach on one does not imply breach on the other.

### 6.4.1 Separation of concerns

By using a microservice-inspired architecture, we tend to have simpler, and more easily maintainable system instead of a complex monolith that attempts to perform all the various tasks at once.

### 6.4.2 Enhanced Security

Multiple independent microservices enhances security. This is because in an event of a malicious computer attack, the attacker needs to bypass multiple services to gain access to the entirety of the data. In other words, breaking one microservice does not imply breaking the whole system.

### 6.4.3 Flexible scalability factor

splitting the services into multiple simpler ones provide the benefit of flexible scalability factor.



## Chapter 7

# The Frontend

### 7.0.1 Technology Stack

In the construction of our system, we will need a number of open source technologies, libraries and standards to support our development. An exhaustive list is given below

- HTML5
- CSS3
- Javascript
- React.Js
- Bootstrap

#### HTML5

HTML5 is a markup language mainly used for structuring content on the World Wide Web. The its last major version(version 5.0) it is recommended by the World Wide Web Consortium (W3C). The responsible organisation WHATWG (Web Hypertext Application Technology Working Group) is a consortium of the major browser vendors(Apple, Google, Mozilla, and Microsoft)?.

#### CSS3

CSS stands for Cascading Style Sheets with an emphasis placed on “Style.” While HTML is used to structure a web document, CSS comes through and specifies your document’s style—page layouts, colors, and fonts are all determined with CSS?. We will use CSS, version 3, to make our frontend application aesthecaly pleasing and easy-to-use for our end-users.

#### Javascript

longside HTML and CSS, JavaScript is one of the major technologies of the World Wide Web. JavaScript makes possible interactive web pages and is an integral part of web applications.

#### React

React (also known as React.js or ReactJS) is an open-source, front end, JavaScript library[3] for building user interfaces or UI components. It is maintained by Facebook and a community

of individual developers and companies.[4][5][6] React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing.[7][8] React Router[9] is an example of such a library.

### **Bootstrap**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components. Bootstrap is among the most starred projects on GitHub, with more than 142,000 stars, behind freeCodeCamp (almost 312,000 stars) and marginally behind Vue.js framework.[2]

## Chapter 8

# The Frontend

### 8.0.1 Technology Stack

In the construction of our system, we will need a number of open source technologies, libraries and standards to support our development. An exhaustive list is given below

- HTML5
- CSS3
- Javascript
- React.Js
- Bootstrap

#### HTML5

HTML5 is a markup language mainly used for structuring content on the World Wide Web. The its last major version(version 5.0) it is recommended by the World Wide Web Consortium (W3C). The responsible organisation WHATWG (Web Hypertext Application Technology Working Group) is a consortium of the major browser vendors(Apple, Google, Mozilla, and Microsoft)?.

#### CSS3

CSS stands for Cascading Style Sheets with an emphasis placed on “Style.” While HTML is used to structure a web document, CSS comes through and specifies your document’s style—page layouts, colors, and fonts are all determined with CSS?. We will use CSS, version 3, to make our frontend application aesthecaly pleasing and easy-to-use for our end-users.

#### Javascript

longside HTML and CSS, JavaScript is one of the major technologies of the World Wide Web. JavaScript makes possible interactive web pages and is an integral part of web applications.

#### React

React (also known as React.js or ReactJS) is an open-source, front end, JavaScript library[3] for building user interfaces or UI components. It is maintained by Facebook and a community

of individual developers and companies.[4][5][6] React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing.[7][8] React Router[9] is an example of such a library.

### **Bootstrap**

Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components. Bootstrap is among the most starred projects on GitHub, with more than 142,000 stars, behind freeCodeCamp (almost 312,000 stars) and marginally behind Vue.js framework.[2]

## Chapter 9

# The Backend

...

### 9.1 ...

....

### 9.2 ...

...

#### 9.2.1 ...

### 9.3 Summary

...

## **Chapter 10**

# **The Prediction Service**

...

**10.1 ...**

....

**10.2 ...**

...

**10.2.1 ...**

**10.3 Summary**

...

## **Chapter 11**

# **Discussion, Conclusion and Future work**

...

**11.1 ...**

....

**11.2 ...**

...

**11.2.1 ...**

**11.3 Summary**

...

## Chapter 12

# Reflection

...

**12.1** ...

....

**12.2** ...

...

**12.2.1** ...

**12.3** **Summary**

...



# References

- Barth, A. (2011), 'HTTP State Management Mechanism', Internet Requests for Comments.  
**URL:** <https://tools.ietf.org/html/rfc6265>
- Eun Ju Ha, J. H. B. (2021), 'Ha ej, baek jh. applications of machine learning and deep learning to thyroid imaging: where do we stand? ultrasonography. 2021 jan;40(1):23-29. doi: 10.14366/usg.20068. epub 2020 jul 3. pmid: 32660203; pmcid: Pmc7758100.'.   
**URL:** <https://pubmed.ncbi.nlm.nih.gov/32660203/>
- Fine Needle Aspiration Biopsy of Thyroid Nodules* (2019).  
**URL:** <https://www.thyroid.org/fna-thyroid-nodules/>
- International Organization for Standardization (1994), 'Information technology — digital compression and coding of continuous-tone still images: Requirements and guidelines', ISO/IEC 10918-1.
- Leach, P. (2005), 'A Universally Unique IDentifier (UUID) URN Namespace', Internet Requests for Comments.  
**URL:** <https://tools.ietf.org/html/rfc4122>
- Manber, U. (1996), 'A simple scheme to make passwords based on one-way functions much harder to crack', *Computers and Security* **15**(2), 171–176.  
**URL:** <https://www.sciencedirect.com/science/article/pii/016740489600003X>
- National Insurance Manual* (2021).  
**URL:** <https://www.gov.uk/hmrc-internal-manuals/national-insurance-manual/nim39110>
- Newman, S. (2020), *Monolith to microservices: evolutionary patterns to transform your monolith*, O'Reilly Media, Inc.
- R. Fielding, E. (2014), 'Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content', Internet Requests for Comments.  
**URL:** <https://tools.ietf.org/html/rfc7231>
- Rescorla, E. (2000), 'HTTP Over TLS', Internet Requests for Comments.  
**URL:** <https://tools.ietf.org/html/rfc2818>
- Rivest, R. (1992), 'The MD5 Message-Digest Algorithm', Internet Requests for Comments.  
**URL:** <https://tools.ietf.org/html/rfc1321>
- T. Bray, E. (2014), 'The JavaScript Object Notation (JSON) Data Interchange Format', Internet Requests for Comments.  
**URL:** <https://tools.ietf.org/html/rfc7159>

Yergeau, F. (2003), 'UTF-8, a transformation format of ISO 10646', Internet Requests for Comments.

**URL:** <https://tools.ietf.org/html/rfc3629>