Cherry Blossom: A Polycystic Ovary Syndrome Detection Model Using Convolutional Neural Networks.

Mercy Kamau

123072

ICS 4 A

Supervisor Name

Kevin Omondi

Submitted in Partial Fulfilment of the Requirements of the Bachelor of Science in Informatics and Computer Science at Strathmore University

School of Computing and Engineering Science

Strathmore University

Nairobi, Kenya

December 2022

Declaration and Approval

I declare that this is work has not been previously submitted and approved for award of a degree by this or any other University. This work has been produced from my own research and understanding on this project. It is not subject to theft of any publication or writing of other people. Where information from other sources is used, it is well referenced in the research document.

Student Name: Mercy Njeri Kamau

Admission Number: 123072

Student Signature: _____ Date:_____

The Proposal of **Mercy Njeri Kamau** has been reviewed and approved by **Mr. Kevin Omondi**

Supervisor Signature: _____ Date: _____

Acknowledgement

I express my gratitude to everyone who assisted me in one way or the other with the completion of my documentation. First and foremost, I express my appreciation to Strathmore University for offering resources such as library materials to assist me with this research. In addition, I appreciate Mr. Kevin Omondi, my project supervisor, for his help and guidance throughout the project documentation. I appreciate my friends who assisted me in the project and to my parents for providing support and resources needed for my project. Lastly, I am grateful to God for giving me good health and strength to complete this project.

Abstract

Polycystic Ovary Syndrome is mainly metabolic, reproductive and an extremely complicated disorder among women. It is a disorder characterized by changes in the female hormone levels and the abnormal production of male hormones. This condition leads to ovarian dysfunction and causes hormonal imbalance, menstruation problems, hair loss, facial acne, facial pimple, dark skin, diabetics and obesity. Early detection is important for getting the right treatment for PCOS.

Due to the complexities of the diseases and complexities in diagnosing this disorder, it was important to find a solution to assist physicians with this process. This study investigated and used Machine Learning techniques including Convolutional Neural Networks to provide a solution that will detect PCOS using ultrasound images. The solution aimed at complementing the already existing technology, and help in the early diagnosis of the disease reducing risk of death.

Table of Contents

List of Figures

List of Abbreviations

CART – Classification and Regression Trees

CNN – Convolutional Neural Networks

DFD – Data Flow Diagram

ERD – Entity Relationship Diagram

KNN – K-Nearest Neighbor

PCOS – Polycystic Ovary Syndrome

SSAD – Structured System Analysis and Design

SVM – Support Vector Machine

Chapter 1: Introduction

1.1 Background Information

The globe has become a global village where people exchange ideas, knowledge and updated information that is always readily available when needed, thanks to technology. Amongst the most frequently looked up topics on the internet includes health related information. According to Pew Internet and American Life Project about 60% of grown up look for enough health information and 35% concentrate on diagnosis of ailments online only (Rao, 2020). However information retrieved from online sources maybe unreliable as it may not be credible and there is a need to provide accurate health systems which ensure relevancy and provides up to date information.

Currently, women increasingly use the internet and social media too access health related information in search of PCOS symptoms. Polycystic Ovary syndrome is a complex chronic disease that affects 5 to 10% of female population of reproductive age (Artini, P.G. & Simi, G. & Ruggiero, M., 2010) and is characterized by a range of symptoms including no menstrual periods or irregular menstrual periods, hirsutism, acne and ovarian cysts. PCOS can cause infertility, anxiety and cardiovascular diseases if not diagnosed. A research in the United Kingdom show about 70% of women with PCOS symptoms are undiagnosed (Sanchez, N. & Jones, H., 2016) and others diagnose themselves by searching for information online. This could cause inaccuracies as PCOS has three diagnostic classifications which include NIH criteria, transvaginal ultrasonography and biochemical hyperandrogenism (Artini, P.G. & Simi, G. & Ruggiero, M., 2010).

Diagnosis of PCOS can be challenging given that its symptoms are ambiguous. Despite the mentioned classification of diagnosis, there are various uncertainties the come with diagnosis of PCOS. PCOS has multiple heterogeneous presentations such as age, race, and obesity and may need evaluation from different physicians due to these diverse symptoms.

According to reports, women had to wait an average of two years for the right diagnosis, which left them upset, confused, and dissatisfied with the process (Abuadla, Y., Raydan, G.D., Charaf, M.Z.J., Saad, R.A., Nasreddine, J. & Diab, O.M, 2021). Based on the length of time it took to diagnose and the number of doctors the patient saw before receiving a final diagnosis, a study on patient satisfaction with PCOS diagnosis found that only 35.2% of patients were satisfied with

their experience (Abuadla, Y., Raydan, G.D., Charaf, M.Z.J., Saad, R.A., Nasreddine, J. & Diab, O.M, 2021).

With the inconsistences and complexities of the approaches taken to diagnoses women with PCOS there was a need to create a machine learning system to help with the detection of Polycystic Ovary Syndrome which would help ease women's lives suffering from PCOS as it would lead to timely treatment.

1.2 Problem Statement

Information on PCOS is provided on the internet and networking spaces regarding PCOS symptoms, long term health effects and management strategies for self-diagnosis. However information on the authors, editorial review process and publication and strength of evidence is omitted hence the information retrieved online may lack credibility. It is important that women who experience PCOS symptoms seek medical attention to accredit that they have PCOS and avoid misdiagnosis.

Polycystic ovary syndrome can be detected using biochemical examinations such as blood tests and hormonal examinations, which is an expensive investigation. PCOS can also be detected using clinical parameters such as menstrual cycle length and body mass index, however, it involves clinical knowledge and subjectivity of the clinician, as we as intrusion of privacy for the women seeking treatment. In addition the process is not automated and the emotional wellbeing is not considered.

To overcome the issue of manual diagnosis, the proposed solution was to use machine learning, deep convolution neural networks, to classify images of ultrasounds as infected or not infected by checking for ovaries containing follicular cysts. Convolutional neural networks was chosen for its high accuracy points.

1.3 Objectives

1.3.1 General Aim

The general objective was to classify images on ultrasound images to detect Polycystic Ovary Syndrome.

1.3.2 Specific Objective

i. To investigate parameters considered in detecting of Polycystic Ovary Syndrome.

ii. To review the ways of detecting Polycystic Ovary Syndrome.
iii. To investigate the challenges associated with current methods of detecting Polycystic Ovary Syndrome.
iv. To review the solutions that detect Polycystic Ovary Syndrome.
v. To develop a model to detect Polycystic Ovary Syndrome.
vi. To validate the model developed.

1.4 Research Questions

i. What are the current ways in which Polycystic Ovary Syndrome is detected?
ii. What are the challenges associated with detection of Polycystic Ovary Syndrome?
iii. What are the implemented solutions to detect Polycystic Ovary Syndrome?
iv. How was the solution designed and developed?
v. How was the solution tested and validated?

1.5 Justification

There are many challenges facing detection of Polycystic Ovary Syndrome. Some of the challenges include being over diagnosed, there is no clear division of normal variability from the abnormality of PCOS especially in young women (Copp, 2017). In developing nations there is an extreme shortage for medical personnel that are trained hence patients are forced to seek medical attention from less skilled and non-specialists physicians. Cases as high as 75% of cases are not diagnosed due to lack of care givers with knowledge (Pebolo, 2021). Due to these problems there is need for a machine learning detection model to help in detecting for PCOS.

The solution will use Convolutional Neural Networks to help with image classification. The Convolutional Neural Networks will process the image and quantify the ultrasound image as infected or not infected.

1.6 Scope and Delimitations

The project will only focus on classifying ultrasound images of PCOS and its Convolutional Neural Network.

Due to the models complexity, the project will only classify images as infected and not infected and will incorporate one dataset.

1.7 Limitations

This project is only limited to classifying images of already existing PCOS ultrasounds that have been diagnosed by a medical practitioner. This is because the system is designed only to detect for PCOS anomalies. Due to unavailability of a large dataset the model was susceptible to over fitting the images. The machine learning algorithm require a machine that had great computational power to be able to run the machine learning algorithms and constant internet for the program to work as required.

Chapter 2: Literature Review

2.1 Introduction

This chapter discusses the current methods used to detect PCOS in patients, as well as the challenges associated with the already existing systems. It also looks into related works in detection systems and gaps in related works. This chapter also illustrates the conceptual framework

2.2 PCOS Detection Avenues

2.2.1 Pelvic Examination

The doctor examines the patients pelvic visually and manually and inspects the reproductive organs for masses and other abnormalities. However this is not a comprehensive method because the patient may require additional tests and periodic checks for blood pressure, glucose tolerance, cholesterol and screening for depression and anxiety, which is detrimental for the patient in general (Sun, 2019).

2.2.2 Ultrasound

The medical practitioner checks the appearance of the ovaries and the thickness of the lining of the uterus. A transducer is placed in the vagina and it emits sound waves that are then translated into an image. However sometimes the ultrasound can show normal looking ovaries but still a patient has PCOS. This could lead to undiagnosed PCOS that can lead to infertility and worse death.

2.3 Challenges Associated with Current Methods of PCOS Detection

The current methods of PCOS detection and diagnosis are complicated. This is because a method such as ultrasound cannot be used in adolescents or menopausal women as ovarian imaging is not advised for adolescent girls. Pelvic examination is time consuming as it requires additional tests which increases risk of infertility for women with PCOS.

There has not been enough research to study correct methods of diagnosing PCOS and there is a lack of certified material and standardization measure (Streseski, 2019).

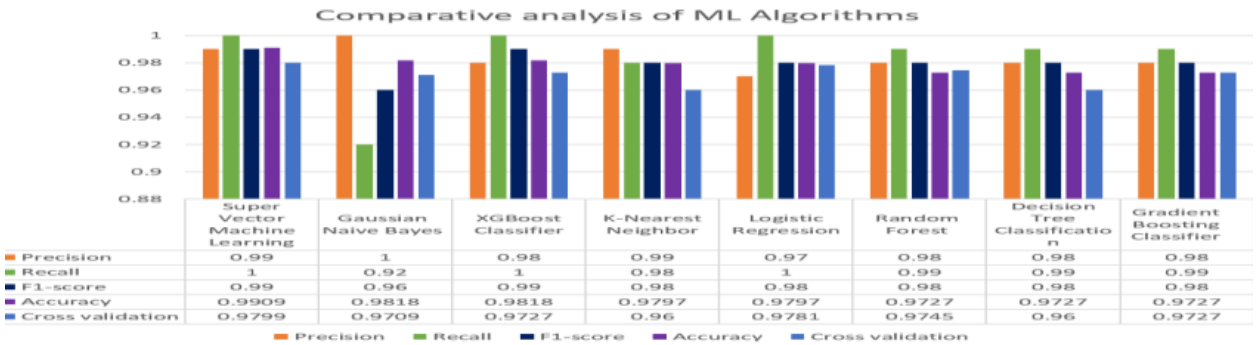
2.4 Related Works

2.4.1 Automated Detection of Polycystic Ovary Syndrome Using Machine Learning

This was a study conducted by Yasmine A. Abu Adla (Adla, 2022) with aim of investigating the possibility of building a model that could automate the diagnosis of PCOS using machine learning algorithms and techniques. The methods that were used in this study included classification and Support Vector with a linear kernel. The model performance had a performance of above 90% and recall of 80%. The model required a trade-off between precision, accuracy and recall because it did not perform well when it came to recall.

2.4.2 Machine Learning Approach: Detecting Ovary Syndrome and Its Impact on Bangladeshi Women

According to Nusrat Nabi (Nabi, 2021) early detection brings early prevention hence the research on how to use machine learning in detection of PCOS. In this research various classification models were used such as Gradient boosting, decision tree classification and super vector machine learning. With the parameters provided such as age, height, weight and depression level, the machine learning algorithms did not perform well as Artificial Neural Network gave an accuracy of 94%, K-Nearest Neighbor had an accuracy of 50% and Linear Regression had an accuracy of 100% which is impossible as there is noise during training. The model was also trained on few data points making it possible to get really high accuracy points. Figure 2.1 shows the comparative analysis of the Machine Learning Algorithms used.



	Super Vector Machine Learning	Gaussian Naive Bayes	XGBoost Classifier	K-Nearest Neighbor	Logistic Regression	Random Forest	Decision Tree Classification	Gradient Boosting Classifier
Precision	0.99	1	0.98	0.99	0.97	0.98	0.98	0.98
Recall	1	0.92	1	0.98	1	0.99	0.99	0.99
F1-score	0.99	0.96	0.99	0.98	0.98	0.98	0.98	0.98
Accuracy	0.9909	0.9818	0.9818	0.9797	0.9797	0.9727	0.9727	0.9727
Cross validation	0.9799	0.9709	0.9727	0.96	0.9781	0.9745	0.96	0.9727

Figure 2.1 Comparative Analysis of Machine Learning Algorithms

2.4.3 i-Hope: Detection and Prediction System for Polycystic Ovary Syndrome Using Machine Learning Techniques

According to Amsy Denny (Denny, 2019),i-Hope provides an optimal and minimal way of detection and prediction of PCOS. For classification of the data collected i-Hope used machine learning techniques such as Naïve Bayes classifier method, logistic regression, K-Nearest neighbor (KNN), Classification and Regression Trees (CART), Random Forest Classifier, Support Vector Machine (SVM) in Spyder Python IDE. The results showed that the best technique used was RFC with an accuracy of 89.02%. Figure 2.2 show the accuracy score of the research.

TABLE V. ACCURACY SCORE, SENSITIVITY, SPECIFICITY AND PRECISION OF VARIOUS MODELS

Algorithm Used	Accuracy score	Sensitivity	Specificity	Precision	F1 score
Logistic Regression (LR)	0.8536	0.6451	0.98039	0.952380	0.3845
K- Nearest Neighbors (KNN)	0.8658	0.8064	0.90196	0.83333	0.4098
Classification and Regression Trees (CART)	0.8292	0.8387	0.82352	0.74285	0.3939
Random Forest Classifier (RFC)	0.8902	0.7419	0.98039	0.95833	0.4182
Gaussian Naïve Bayes (NB)	0.8414	0.7419	0.90196	0.82142	0.3898
Support Vector Machines (SVM)	0.8292	0.5483	1.0	1.0	0.3541

Figure 2.2Accuracy Score, Sensitivity, Specificity and precision of various models

The algorithms used in this research had a huge tradeoff with precision and accuracies as there were huge differences in accuracies from the models.

2.5 Gaps in Related Works

Most of the related works focused on finding the optimal algorithm and focused mostly on using Support Vector Machine to find the best precision. However, with this there was a big tradeoff between the precision and accuracy. Most of the models were trained on very little data hence the models were over fitting.

In order to diagnose PCOS, physicians recommend defining at least two of the criteria, ovulatory dysfunction, hyperandrogenism or related disorders. However this this criteria is subjective as it requires a woman to carefully and fully track her menstrual cycle also normal androgen concentrations in women change with age, race, and body mass index**.** This criteria is not

sufficiently sensitive or specific to PCOS and is not standardized (Wiencek, R.J., McCartney, C.R., Chang, A.Y., Straseski, J.A., Auchus, R.J., WoodWorth, A., 2019).

2.6 Conceptual Framework

The conceptual framework below shows how the Kaggle dataset was passed through the convolution layer for feature extraction. The features extracted were then improved computationally and parameters reduced. When the data at each entry point had been transformed into value group points an activation function was used to convert the neural network to a non-linear function.

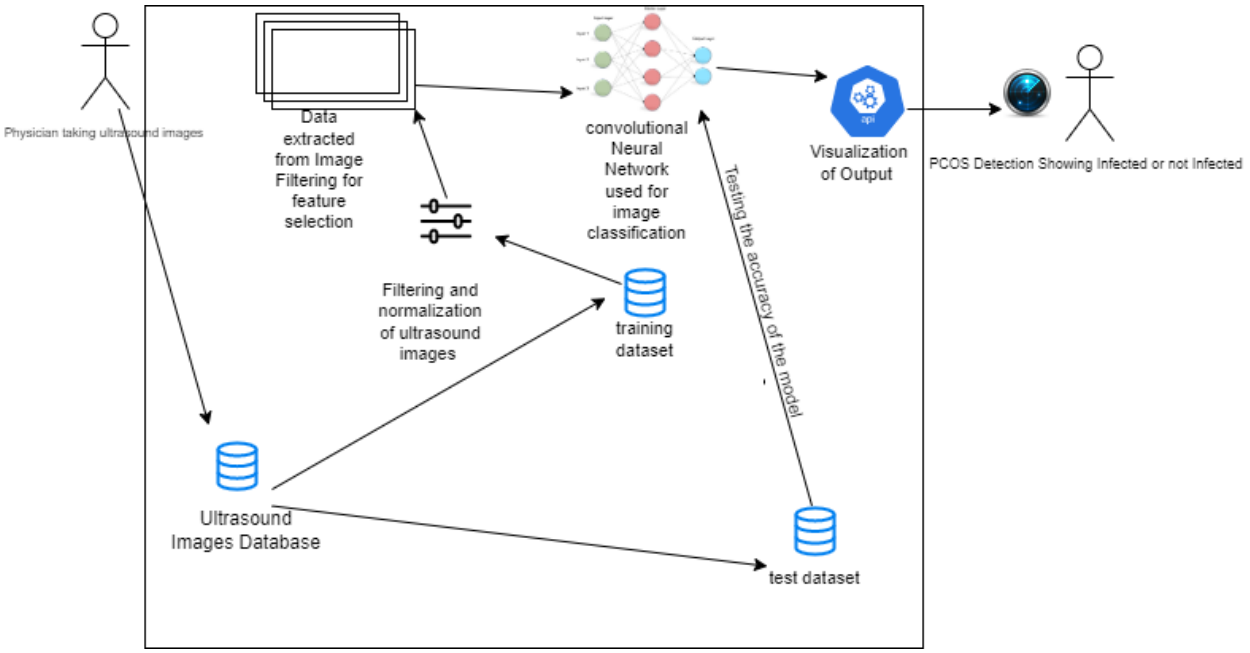


Figure 2.3 Conceptual Framework

Chapter 3: Methodology

3.1 Introduction

This chapter covered the methodology approach that was used in this model development and the steps involved. The chapter also covered analysis diagrams including the use case diagrams, sequence diagrams, system sequence diagram, ERD, context diagram and dataflow diagram.

The system design which includes dataset schema, wireframes and system architecture wer described and discussed. Deliverables such as the project proposal, model and the API were listed and discussed.

Tools and techniques that were used in the project were discussed. Some of the tools and techniques to be discussed included TensorFlow, Google collaborator, Convolution Neural Network, python, GitHub repository and Kaggle dataset.

SSAD design paradigm was used because it clearly defined each module and each step towards the development and completion of the project.

3.2 Applied Methodology

The methodology that was used in developing the model is design thinking. Design thinking methodology is an innovative problem-solving process that helps companies to come up with a desired outcome on a specific problem (Al-Deen, 2017). It is a problem solving method that places the requirements of the user first. This method first understands the problem before searching for the possible solutions. The diagram for the methodology is illustrated below in Figure 3.1

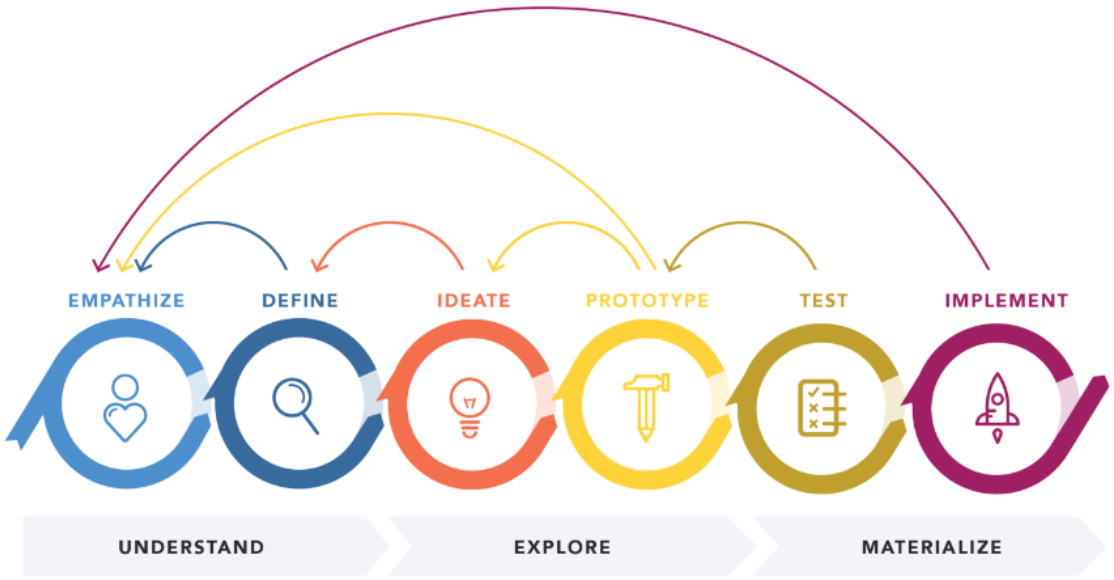


Figure 3.1Design thinking methodology (Adopted from ((Al-Deen, 2017))

3.2.1 Empathize

This was getting to understand the problem that was to be solved. The goal of this phase was to collect the necessary requirements by understanding the users' needs.

3.2.1.1 Researching User Needs

This phase was focused on gaining insights into users and their needs. Medical documents, research papers and journals were investigated and analyzed to discover new opportunities to meet the users' needs that were missing in existing systems for PCOS detection. This helped get a better understanding of the problem seeking to be solved.

3.2.2 Define

All the information gained through the empathize stage was put together to bring focus to the parameters of the problem. The goal was to conclude with a requirement statement that clearly defined the scope and parameters of the problems. The data collected from secondary documents was analyzed and organized to better understand the problems defined. This helped in gaining new ideas and to build understanding on how to use them effectively.

3.2.3 Ideate

This phase consists of generating different possible potential solutions to previously defined issues, or a portion of a solution to the proposed problem. Different solutions to the problem were generated through brainstorming and allowed for more innovative solutions to normalized problems. A system analysis of the solution was also done in this stage as seen in section 3.3.

3.2.4 Prototype

This is an experimental phase. It allowed for identification of flaws in the design thinking process while also allowing for iteration in the solution. Products with specific features were created in this phase. The aim of this phase was to identify the best possible solution for the problem stated.

3.2.5 Test

a. Unit Testing

Different modules were tested independently using glass-box method. This was important as it reviewed if the modules had met the requirements set in the initial stages of development.

b. Integration

This was be done by plotting expected results against the results produced by the model.

3.3 System Analysis

3.3.1 Approach to Design Development

The approach to be used in the solution was SSAD which is a process based on structure top-down decomposition of a system. This approach focuses on the process and procedures of the system. This approach was selected because it consists of methods that represent how data and the related processes move through an information system.

The purpose of this phase was to build a logical model of the new model. The analysis tool that was used to visualize this data is visual paradigm

3.3.2 Use-Case Diagram

A use case describes steps in a specific business process. It shows interaction of things outside the system with the system. This model helped in identifying different actors in specific use cases and their functions in the system

3.3.3 Sequence Diagram

A sequence diagram shows the interaction amongst objects during specified times. The sequence diagram was modelled to show different methods in a given class in the system. The diagram assisted in visualizing interaction between different objects of the system.

3.3.4 System Sequence Diagram

A system sequence diagram shows the interaction among processes during a specified time. A system sequence diagram graphically documents use cases by showing processes, messages and timings of then message. The diagram was constructed to show how the system interacts among each other and within itself.

3.3.5 Entity Relationship Diagram

An Entity Relationship Diagram shows relationship between system entities and their interactions. This was drawn to provide an overall view of the system, the entities, their attributes and relationships between the entities of the system and the system itself.

3.3.6 Context Diagram

The context diagram is a top level view of an information system. It shows boundaries and scopes of the system. Contains process 0, which represents the entire system but does not show how it works internally. This was drawn to show the interaction between system entities and how data flowed between them.

3.3.7 Data Flow Diagram

3.3.7.1 DFD Level 0

A level 0 DFD displays the main internal operations, data flows, and data stores. This was illustrated to show how data flowed between various processes and how data was maintained, such as how filtered images were kept on file in the system prior to feature extraction.

3.3.7.2 DFD Level 1

A level 1 DFD enlarges the system even more to reveal the earliest data flows, data stores, and processes. This was illustrated to show how data flowed during the most basic operations, such as splitting the dataset in two.

3.4 System Design

3.4.1 System Architecture

It shows the logical design of the information system into a physical structure. This includes hardware, software, methods, security and network support. The system architecture gave a brief overview of the solution and envisioned all components and interactions of the system.

3.5 Tools and Techniques Used

3.5.1 Python

This is a programming language. It was used to develop the solution. Python is a high level programming language and is best suited for programming machine learning solutions. Some of the libraries that were included are:

i. **Matpotlib**

   This library is used in creating animated and interactive visualizations in python. It will be used for data visualization

ii. **Keras**

   This library will be used to construct CNN models with TensorFlow.

iii. **Numpy**

   This library will be used for image process and manipulation of pixels.

3.5.2 Kaggle

The dataset used to train the model was retrieved from Kaggle. The dataset was spilt using cross validation for training and testing in order to increase the size of the dataset.

3.5.3 GitHub

The code prepared was stored in this online collaboration tool that acted as backup in case of any system failures in the future.

3.6 System Deliverables

3.6.1 Concept Defense

The concept note was a summary of the general idea before the project was developed. The presentation was prepared and presented to a panel for approval.

3.6.2 Proposal Document

The proposal document, is a document submitted for approval. The document was needed because it defined the objectives and steps to be followed during development of the solution.

3.6.3 Analysis and Design Diagram Document

This is the document that contains all diagrams of the system design and architecture of the approach selected. The diagrams were drawn to visualize the whole system and its processes.

3.6.4 Working Prototype

i. **Split Dataset**

Here the dataset is split into two; the training dataset and the testing dataset. This was done by cross validation of the dataset.

ii. **Filtered Images**

Filtration of noise was done to ensure more accurate detection of PCOS.

iii. **Extracted Features**

The extracted features from the images were used to train the developed CNN model.

iv. **CNN Model**

A CNN model was used for classification. This model was built. Classification was based on infected or not infected from the features extracted.

v. **Detection Module**

This module, the diagnosis was produced as product of the trained CNN model. The classification expected was either infected or not infected. This allowed for correct diagnosis.

vi. **Model Test Cases**

The system was tested to verify that all the system objectives are met and accomplished. The testing paradigm that was used was unit testing.

Chapter 4: System Analysis and Design

4.1 Introduction

In this chapter system analysis and design was done. The diagrams included in this chapter were; use case diagram, sequence diagram, system sequence diagram, entity relationship diagram, context diagram and the data flow diagram. The design paradigm used for the development of this project was structured system analysis and design, where the project was divided into modules, stages, steps and tasks.

4.2 System Requirements

System requirements were the specifications required to make the models functional and satisfy the user needs by ensuring the machine learning model runs smoothly and efficiently. Some of the system requirements reviewed in the project include:

4.2.1 Functional Requirements

i. **Authentication:** This is a plug-in, which is used for login and registration, gathers user information and compares it to users' information stored in the database. The name, email address, and password were acquired by constructing registration and login pages just for the doctor. Only the password and email address were required to log in.

ii. **Receive Input and send it to the model:** This is the module that receives information from the physician and provides it to the model for classification.It uses the patient's identification number and the ultrasound images as inputs. It was accomplished by linking the web application to the model in the backend and providing an upload button in the web application that only accepts audio files.

iii. **Extracting Features from ultrasound Images:** This is the module of the model that receives the ultrasound images and uses an image classification algorithm to extract features to be used in the neural network for classification. This was done by extracting required features to be used in classification. This module then sends the extracted features to the CNN.

iv. **Classifying findings from ultrasound images:** This is the module of the model that uses a CNN to classify the PCOS disease. It was done by developing a CNN model that takes in extracted features. It classifies the images as infected or not infected. This module then sends the output to a web application.

v. **Display output of the model:** This is the module that displays the output of the model to the doctor through the web application. It was done by using text boxes. On the display it shows the disease classification whether infected or not infected.

4.2.2 Non- Functional Requirements

i. **System Performance:** This system attribute showed how fast the system needed to operate. The system was developed in a way that it ensured reasonable response time.

ii. **Integrity:** This is how well the data is maintained by the software system in terms of authenticity and accuracy. This was achieved by using a database that only users with edit rights could manipulate the data.

iii. **Availability:** This describes how much is the system accessible to the esteemed user at a given point. The system was developed and designed to be accessible to the user if the user was connected to a working internet access.

iv. **Usability:** The system was designed to meet the users need by implementing a learnable model that was memorable and worked efficiently in reducing user error and provided user satisfaction.

4.3 System Analysis Diagrams

Some of the system analysis diagrams considered are as follows;

4.3.1 Use case diagram

The use case diagram shows how different actors of the system interacted with the model. There was only one actor in the use case diagram; the doctor; who would be able to upload ultrasound images, view ultrasound images and view the predicted results from the model.

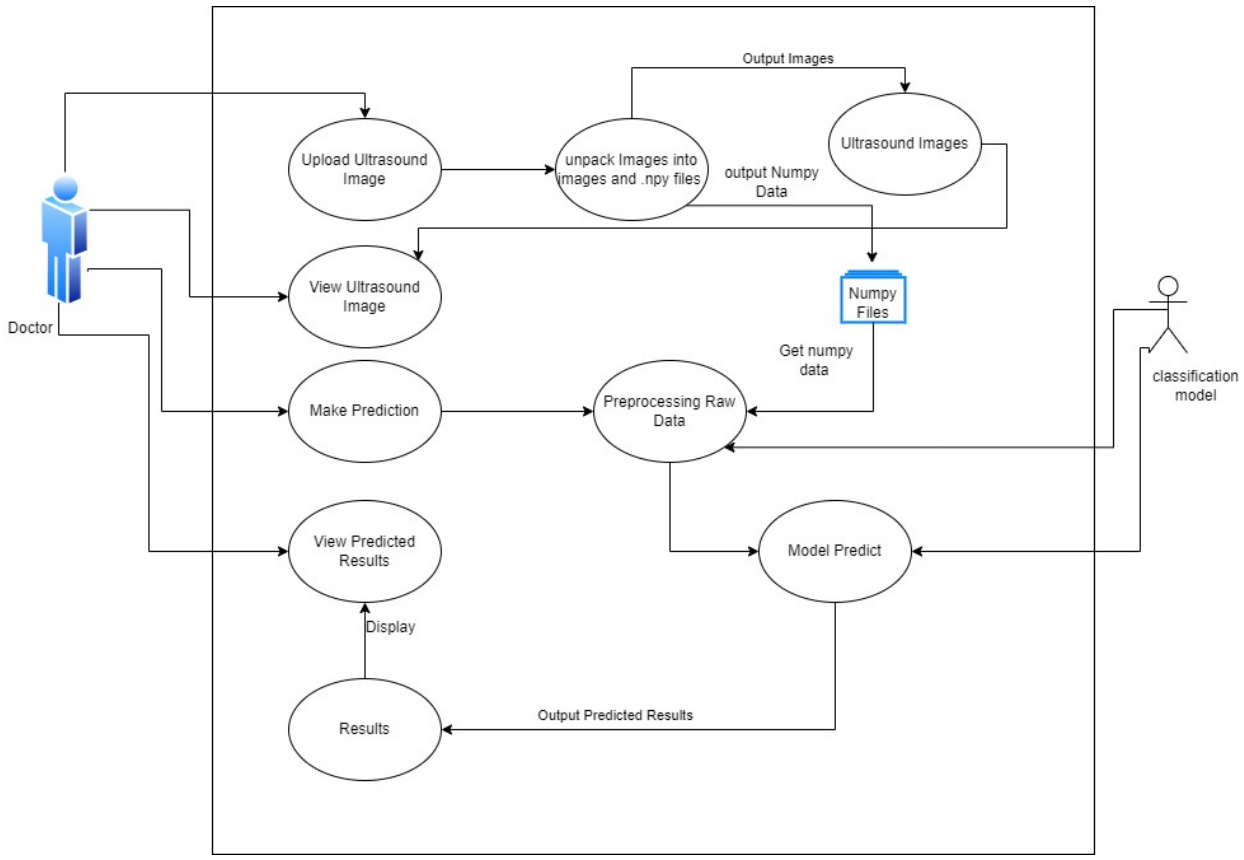


Figure 4.1 Use Case Diagram

4.3.2 System Sequence Diagram

The system sequence diagram was used to show the scenarios of the use case and the events that the external factors generate. It was used to show interaction between the doctor and the system itself.

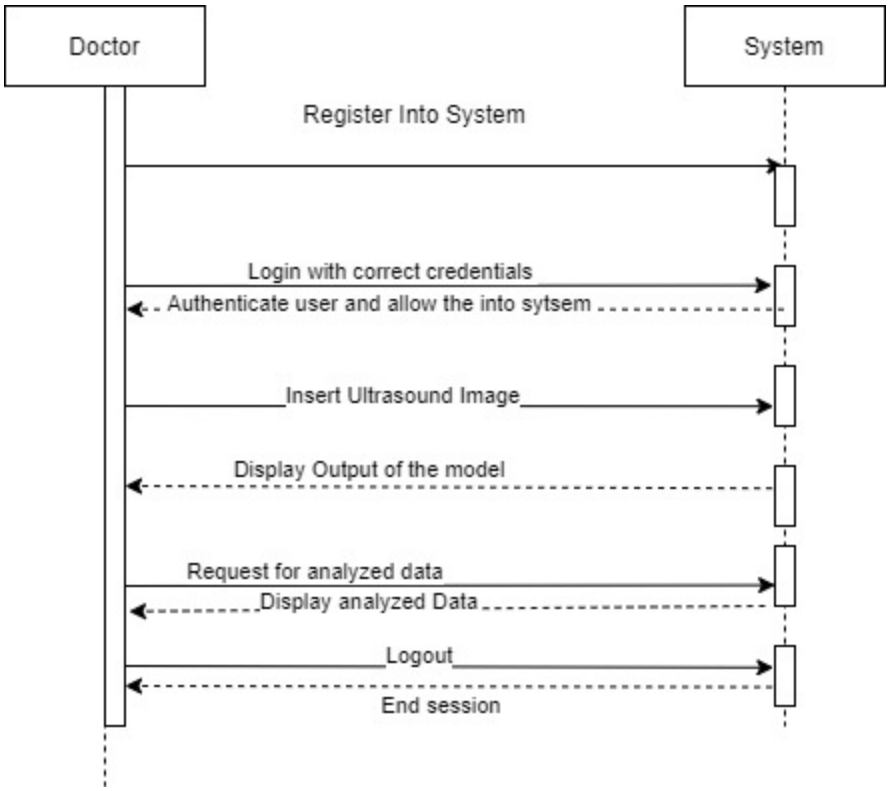


Figure 4.2 System Sequence Diagram

4.3.3 Sequence Diagram

The sequence diagram shows the timing of interaction between actors of the system and modules of the system.

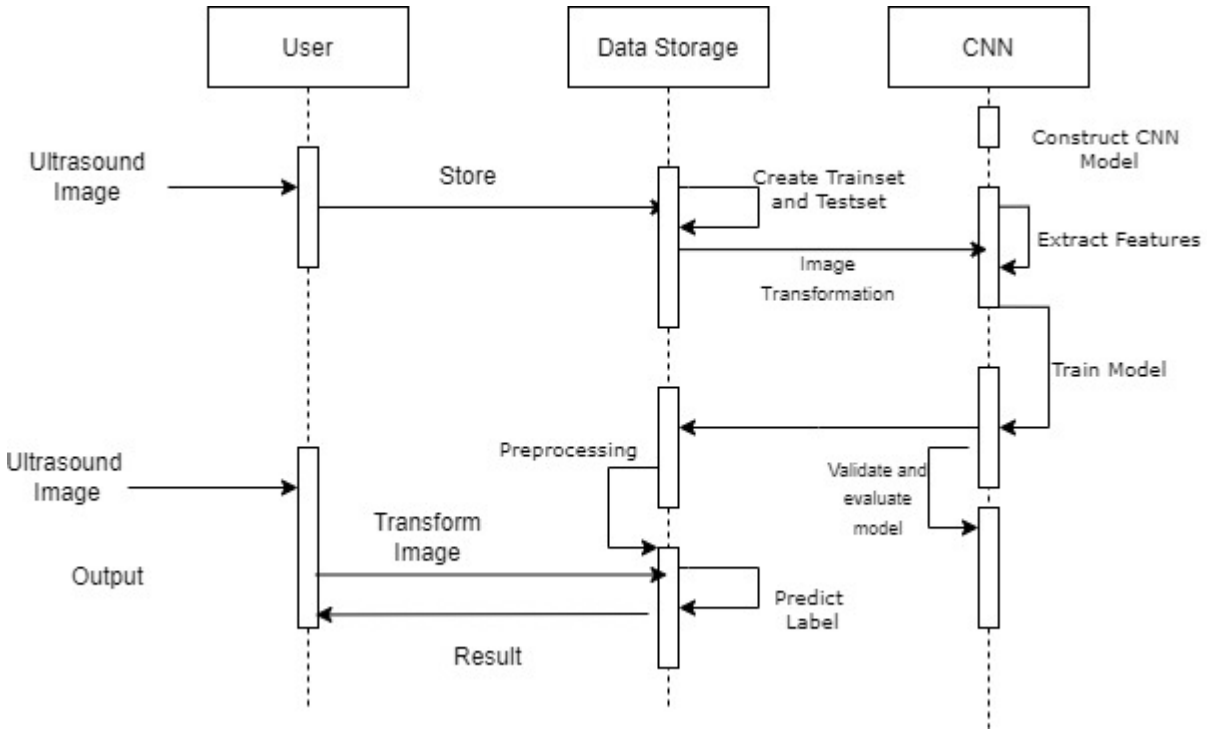


Figure 4.3 Sequence Diagram

4.3.4 Entity Relationship Diagram

The ERD diagram was used to show the entities of the system and their relationship to each other. The entities were each assigned an attribute to describe them.

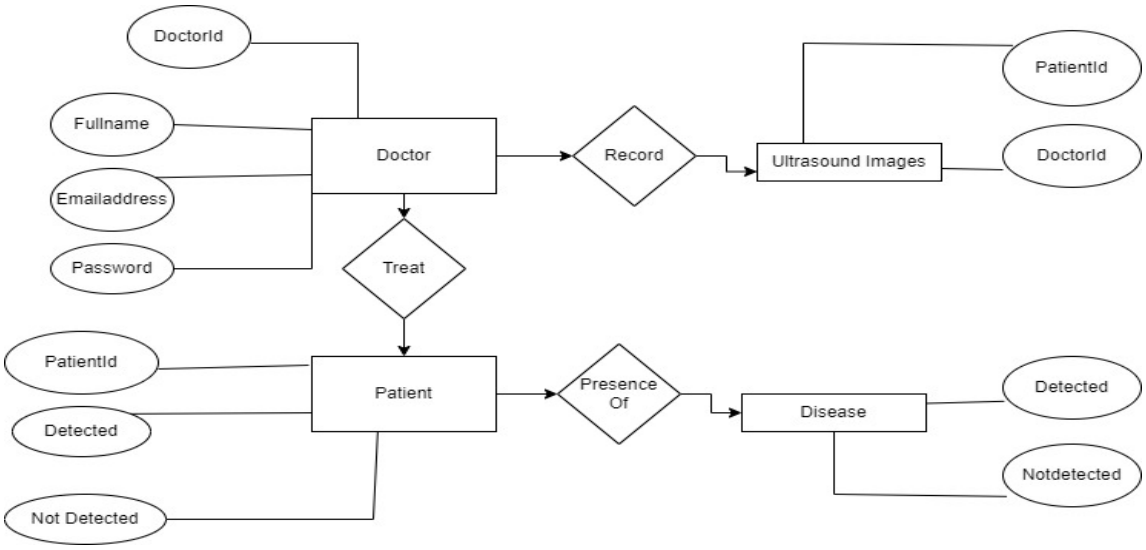


Figure 4.4 Entity Relationship Diagram

4.3.5 Data Flow Diagram

4.3.5.1 Context Diagram

The context diagram was used to display the system as a whole. It showed all the external entities and how they interact with the model.

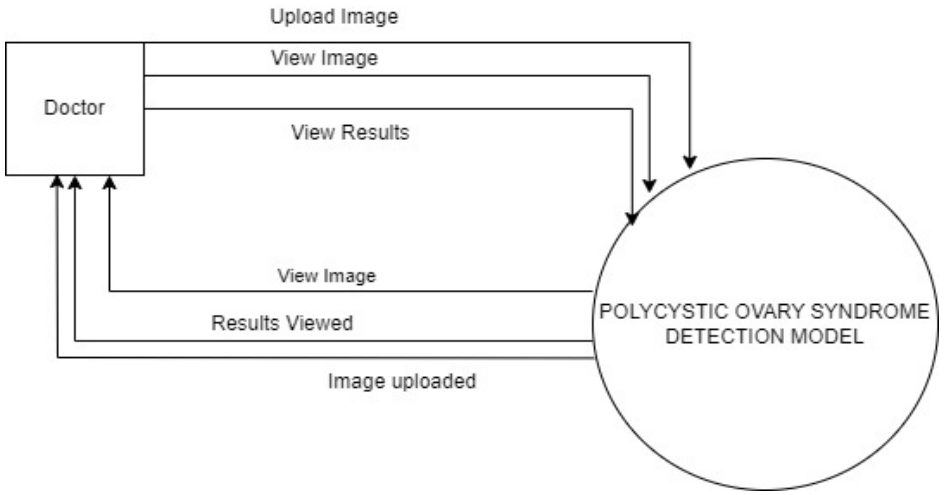


Figure 4.5 Context Diagram

4.3.5.2 DFD Level 1

The DFD Level 1 diagram was used to show how the entities of the model interact with each other and how they interact with model.

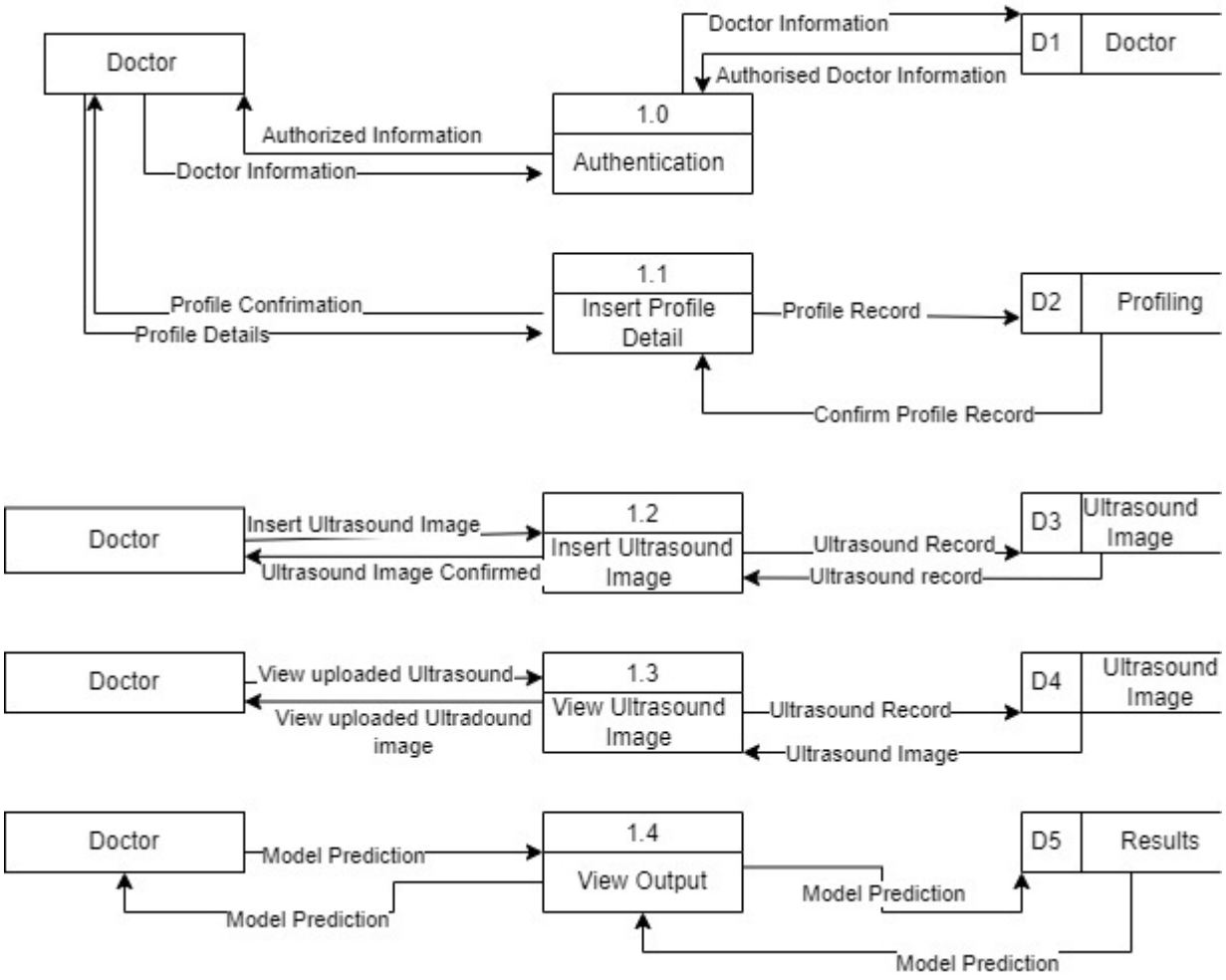


Figure 4.6 DFD Level 1

4.4 System Design Diagrams

The system design diagram consider included;

4.4.1 Database Schema

The logical database schema of the system is shown in figure 4.7

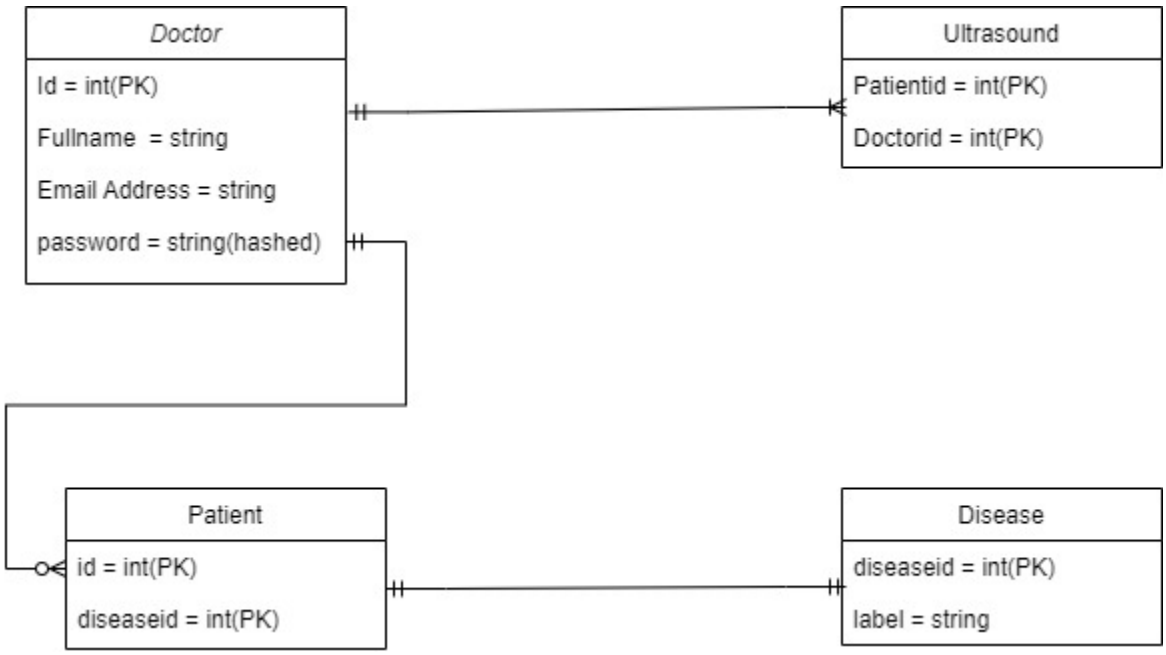


Figure 4.7 Database Schema

4.4.2 Wireframes

4.4.2.1 Registration and Login Pages

Figure 4.8 shows the systems registration and login pages.

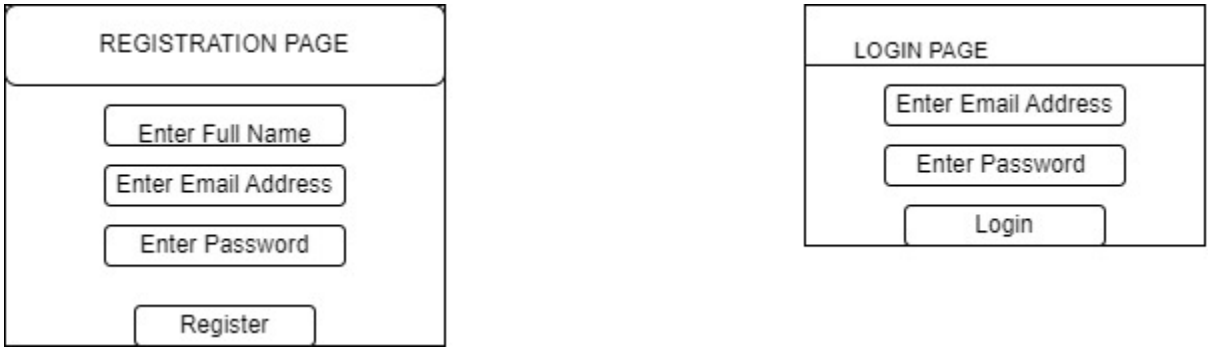


Figure 4.8 Registration and Login

4.4.2.2 Input Page

Figure 4.9 shows the page where the doctor is expected to insert an ultrasound as input for classification.

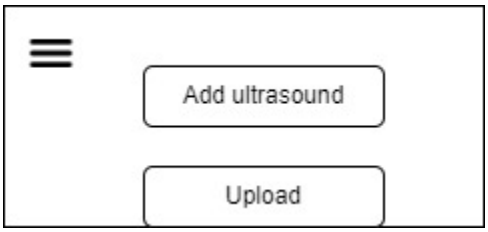


Figure 4.9 Input Page

4.4.2.3 Output Page

Figure 4.10 shows the page where the doctor receives output from the model.

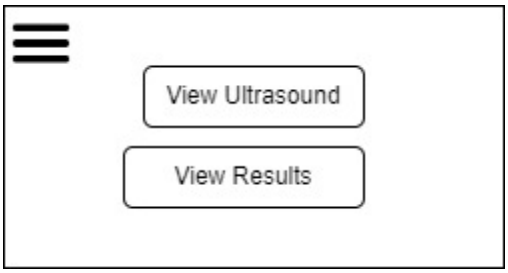


Figure 4.10 Output Page

4.4.3 System Architecture

The system architecture diagram for the developed solution used a three-tier model. The model included the presentation, application and database as shown in figure 4.10

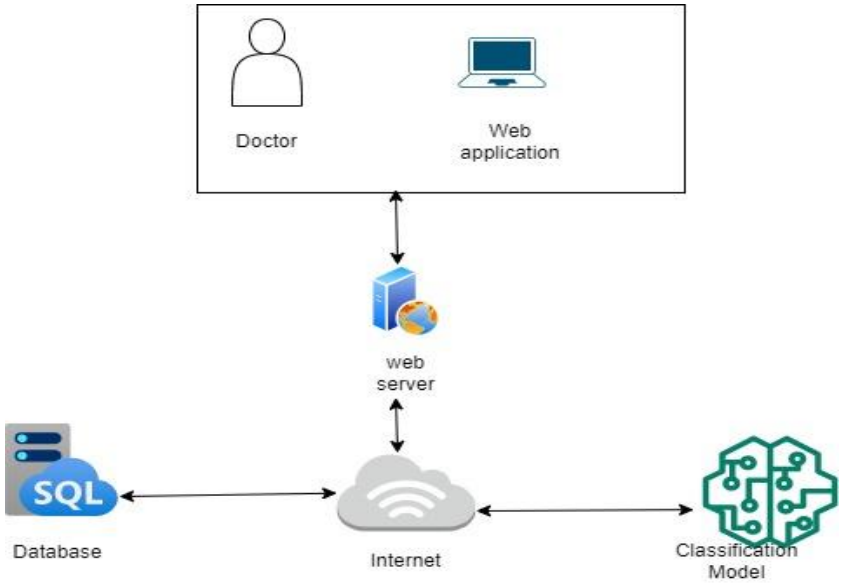


Figure 4.11 System Architecture

Chapter 5: System Implementation and Testing

5.1 Introduction

This chapter discusses the implementation of the machine learning model in detection of polycystic ovary syndrome. It highlights the dataset used, the nature of the model's input and output features.

5.2 Description of the Implementation Environment

5.2.1 Hardware Specifications

Table 5.1 Hardware Specifications

Item	Minimal Specifications	Recommended Specifications
Processor	Intel(R) Core(TM) i5-6200U CPU with 1.8GHz	Intel(R) Core(TM) i5-6200U CPU With 2.40GHz
RAM	4GB	8.00 GB
Hard Disk Storage	100GB of free space is recommended to run the software	

5.2.2 Software Specifications

i. 64-bit operating system, x64-based processor
ii. Python 3.8.8
iii. Tensorflow
iv. Django Framework
v. MySQL
vi. Libraries Tensorflow, Keras,
vii. Google Chrome

5.3 Dataset Used

The dataset used was a Kaggle dataset. It contains 3856 ultrasound images of ovaries. The dataset is split into two folders, training and testing, which further contain two subfolders, infected and not infected. The infected subfolder contains ultrasound images of ovaries that have PCOS while the not infected subfolder contains ultrasounds for images of ovaries that are healthy.

This dataset was collected via Google and friend. To increase the size of the dataset image augmentation was used. The dataset is used in the development of a predictive model to determine whether ovaries have PCOS or are healthy.

5.4 Model Implementation

5.4.1 Training and Analysis of the Machine Learning Model

5.4.4.1 Data Preprocessing

Data preprocessing involved showing the order of the classes, resizing the image after reading it from the disk, data transformation and shuttling and the method to use when resizing the image. The labels were encoded as float32 scalars with values of 0 to represent infected x rays and 1 to represent those not infected. An image data generator was defined to perform data augmentation and increasing the image dimensions.

5.4.4.2 Classification

The model uses supervised learning as the data provided was already pre classified and labeled. The models were trained using keras and tensorflow APIS. For image analysis, convolutional neural networks was used. The models consist of two activation layers, relu in the inside layer and sigmoid in the outer layer, pooling layers, dropout layer to avoid over fitting, flatten layer to convert the output of the convolutional layer to as single dimension that was used as input for the dense layer.

The key component is convolutional layer. The layer connects each output to only a few close inputs. The layer learns local features. Some of the important variables used during training included number of epochs, the initial running rate and number of images that should be copied to memory at a time.

```
history = model2.fit(
          image,
          validation_data = validation_iterator,
          epochs = 8
)
```

```
Epoch 1/8
43/43 [==============================] - 97s 2s/step - loss: 0.5933 - accuracy: 0.7433 - val_loss: 0.40
83 - val_accuracy: 0.7969
Epoch 2/8
43/43 [==============================] - 93s 2s/step - loss: 0.2002 - accuracy: 0.9414 - val_loss: 0.13
39 - val_accuracy: 0.9583
Epoch 3/8
43/43 [==============================] - 98s 2s/step - loss: 0.0904 - accuracy: 0.9703 - val_loss: 0.08
58 - val_accuracy: 0.9688
Epoch 4/8
43/43 [==============================] - 99s 2s/step - loss: 0.0477 - accuracy: 0.9889 - val_loss: 0.03
54 - val_accuracy: 0.9896
Epoch 5/8
43/43 [==============================] - 94s 2s/step - loss: 0.0419 - accuracy: 0.9852 - val_loss: 0.03
40 - val_accuracy: 0.9913
Epoch 6/8
43/43 [==============================] - 94s 2s/step - loss: 0.0261 - accuracy: 0.9918 - val_loss: 0.02
12 - val_accuracy: 0.9948
Epoch 7/8
43/43 [==============================] - 97s 2s/step - loss: 0.0205 - accuracy: 0.9941 - val_loss: 0.01
62 - val_accuracy: 0.9965
Epoch 8/8
43/43 [==============================] - 94s 2s/step - loss: 0.0095 - accuracy: 1.0000 - val_loss: 0.00
92 - val_accuracy: 0.9983
```

Figure 5.1 Training the model

5.4.4.3 Accuracy

The model was trained on three different models each containing different kernel sizes. The second model performed better as it had a validation accuracy 0.991 with a validation loss of 0.09.

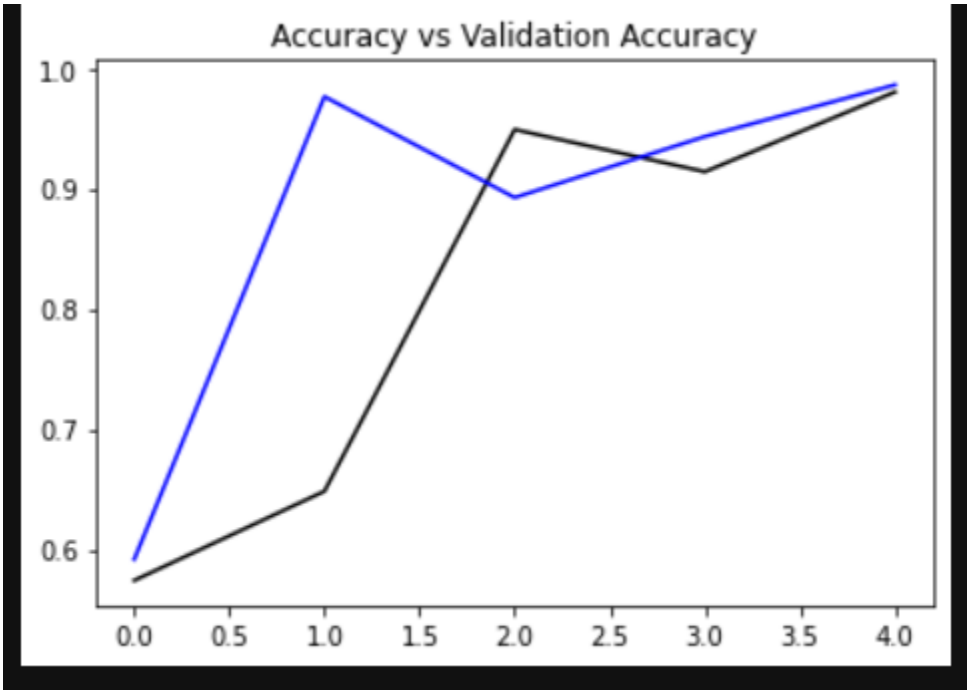


Figure 5.2 Accuracy vs Validation Accuracy

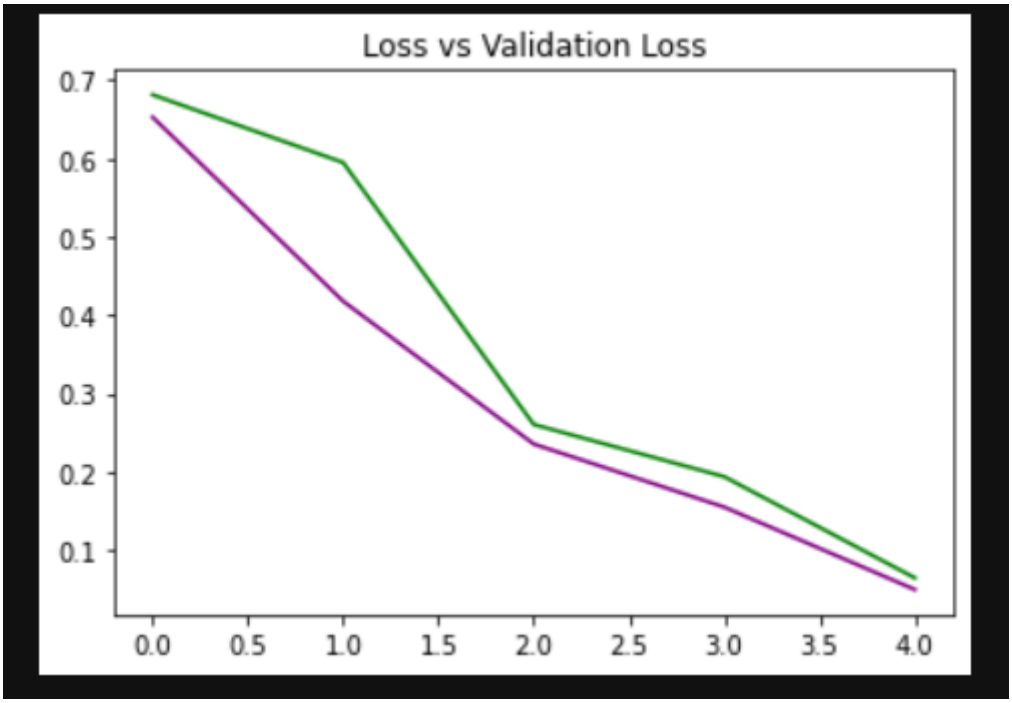


Figure 5.3 Loss vs Validation Loss

5.4.2 Testing Paradigm

5.4.2.1 Black Box Testing

Testing was done following black box testing. This technique examines the program structure without considerations of the inner workings of the system. This is testing that validates an end user experience (Hamilton, 2022).The focus is the input and output of the software in question. Black box testing was implemented in the model while feeding the test dataset to get predictions.

5.4.2.2 White Box Testing

5.4.2.2 White box testing

This technique tests the software's internal structure, design and coding are tested to verify input-output flow and improve design usability and security (Hamilton, 2022).It revolves around the inner workings of the software. This technique was implemented while training the model. Upon reception of the performance metrics, the optimization of processes within the system occurs. Data preprocessing was part of white box testing.

5.4.2.3 Unit testing

Each individual component was tested first as an individual unit rather than a whole component. Testing was carried out by providing the component with appropriate input, relying on its specification, testing was done against expected output. Images were examined to determine if it gave satisfactory results after being fed to the model. This helped in building confidence that each module was performing as expected.

5.5 System Testing Results

This particular section shows various tests done on the model and their results.

5.5.1 Model Testing

Table 5.2 Model Testing

Test Case	Description	Test Data	Results	Verdict
TC001	Checking Accuracy Levels	CNN Model 1 - 5	Model 2 had highest accuracy	Pass
TC002	Library Exploration	Tensorflow, Numpy Pandas	Tensorflow had more functionalities	Pass

5.5.2 Login Module

Table 5.3 shows the testing taken on authentication module of the system.

Table 5.3Login Module

Test Case	Description	Test Data	Experimental Outcome	Result	Verdict
TC001	Registration of Physician	Username – Mercy Email – macykamau03@gamil.com Password – xyz123!	Physician will be redirected to the Login page.	Physician was redirected to the Login page.	Pass
TC002	Login with correct Username	Username – Mercy Email – macykamau03@gamil.com Password – xyz123!	User will be redirected to the home page.	User is able to login to their respective accounts	Pass
TC003	Login with incorrect details	Username – Mercy Email – macykamau03@gamil.com Password – xyz123	Login denied and user alerted about the wrong username and password	Login denied and user alerted about the wrong username and password	Pass
TC004	Login without detail	Username and password is left blank and login is attempted	User is alerted to fill in required fields	User is alerted to fill in required fields	Pass

5.5.3 Model Integrity

Table 5.4 Model Integrity

Test Case	**Description**	**Test Data**	**Experimental Outcome**	**Result**	**Verdict**
TC001	The data used for model should not have null value	Test Dataset	No null values	The data did not have null values	Pass

5.5.4 Model Prediction

Table 5.5 Model Prediction

Test Case	**Description**	**Test Case**	**Experimental Outcome**	**Result**	**Verdict**
TC001	The physician should be able to upload an X-Ray Image	Upload an image	The physician is able to upload an image successfully	Physician is able to upload an image and receive correct prediction	Pass

Chapter 6: Conclusions, Recommendations and Future Works

6.1 Conclusions

The main goal of this developed solution was to ensure the model is able to detect PCOS in women and present it to the medic. This in turn would facilitate proper diagnosis and treatment. The system was able to sufficiently and effectively provide doctors with a proper diagnosis with high confidence factors such as high accuracy. The developed solution has a high chance of improving healthcare by detecting anomalies in the ovaries early. As the system is adopted over the years, it will reduce infertility rates and possible endometrial cancer.

6.2 Recommendations

To optimize the given solution, a machine with better hardware specifications, that is, processors and the Random access memory should be used in processing of available datasets to ensure faster results. The model should be fine-tuned by training it more and using more datasets to increase its accuracy

6.3 Future Works

In case the developed solution is to be improved and used, its scalability aspects should be worked on to ensure good performance for real time use. To improve security, technologies such as blockchain can be used to produce more secure CNN models.

References

Abuadla, Y., Raydan, G.D., Charaf, M.Z.J., Saad, R.A., Nasreddine, J. & Diab, O.M. (2021). Automated Detection of Polycystic Ovary Syndrome Using Machine Learning Techniques. 2. doi:https://doi.org/10.1109/ICABME53305.2021.9604905

Adla, Y. A. (2022). Automated Detection of Polycystic Ovary . 208-212.

Al-Deen, S. G. (2017). Innovative Development Methodologies with Design Thinking.

Artini, P.G. & Simi, G. & Ruggiero, M. (2010). Best Methods For Identification and Treatment of PCOS. *Minerva Ginecologica, 62*(1), 33. Retrieved from https://pubmed.ncbi.nlm.nih.gov/20186113/

Copp, T. (2017). Overdiagnosis and Disease Labels: The Case of Polycystic Ovary Syndrome .

Denny, A. (2019). i-HOPE: Detection And Prediction System For . 673-678.

Hamilton, T. (2022, November 5). *What is BLACK Box Testing? Techniques, Example & Types.* Retrieved from www.guru99.com: https://www.guru99.com/black-box-testing.html

Nabi, N. (2021). Machine Learning Approach: Detecting Polycystic Ovary Syndrome & It's Impact.

Pebolo, F. P. (2021, January 29). Polycystic Ovarian Syndrome: Diagnostic Challenges in resource-poor Setting Ugandan Perpective. *5*.

Rao, V. (2020, February). Medicine Recommendation System Based On Patient Review. *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*, *9*(02), 3308. Retrieved from https://www.ijstr.org/final-print/feb2020/Medicine-Recommendation-System-Based-On-Patient-Reviews.pdf

Sanchez, N. & Jones, H. (2016, June 2). Less Than A Wife: A Study of Polycystic Ovary Syndrome Content in Teens and Women's Digital Magazines. (G. Eysenbach, Ed.) *Journal of Medical Internet Research, 18*(6), 3. doi:10.2196/jmir.5417

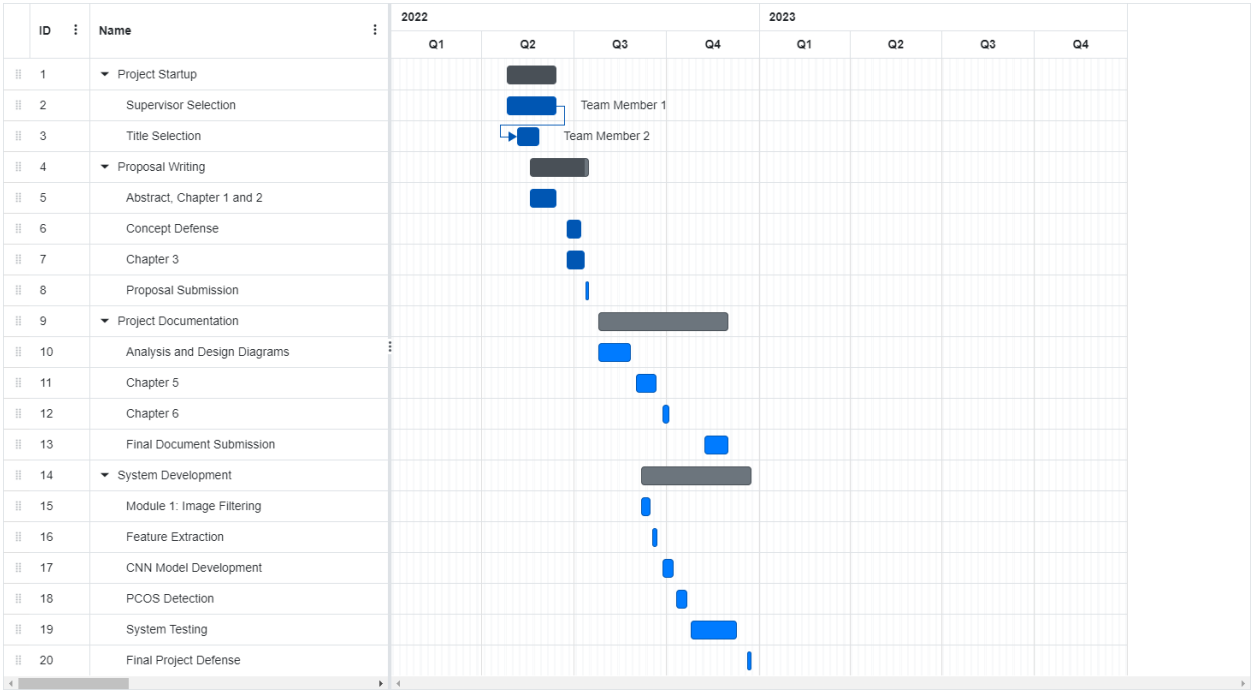
Streseski, J. (2019). Diagnosing PCOS: What Are the Challenges and How Can We Improve? *Scientific Shorts*.

Sun, T. (2019). PCOS Diagnosis: The Role of Pelvic Ultrasound. *ULTRASOUND TECHNOLOGY & INNOVATION*.

Wiencek, R.J., McCartney, C.R., Chang, A.Y., Straseski, J.A., Auchus, R.J., WoodWorth, A. (2019, March 01). Clinical Chemistry. *Challenges in the Assessment and Diagnosis of Polycystic Ovary Syndrome, 65*(03), 370-377. doi:10.1373/clinchem.2017.284331

Appendix

Appendix 1 Gantt Chart



Appendix 1 Gantt chart