



# **Anemia Detection from Eyes, Palm and Fingernails**

---

**Submitted by:**

#	ID	Name
1	42110153	Aya Mohamed Attia
2	42110396	Bakinam Mohamed AbuElmatty
3	42110024	Mohamed Salama Abdelfattah
4	42110491	Mohamed Ahmed Osama
5	42110369	Abdullah Mostafa Mohamed
6	42110320	Hassan Ali Hassan

**dissertation submitted in partial fulfillment of the requirements for the degree of Bachelor of computer science and information technology.**

**Supervised by:**

**Prof. Dina Darwish Vice Dean for Faculty of Computer Science and information technology**

**Egypt 2025**

## Committee Report

---

We certify we have read this graduation project report as examining committee, examine the student in its content and that in our opinion it is adequate as a project document for “ ”.

***Chairman:***

Name:

Signature:

Date: / /2025

***Supervisor:***

Name:

Signature:

Date: / /2025

***Examiner:***

Name:

Signature:

Date: / /2025

## **Intellectual Property Right Declaration**

---

This is to declare that the work under the supervision of **Dr. Dina Darwish** having title "**Anemia Detection**" carried out in partial fulfillment of the requirements of Bachelor of Science in Computer Science is the sole property of Ahram Canadian University and the respective supervisor. It is protected under the intellectual property right laws and conventions. It can only be considered/ used for purposes like extension for further enhancement, product development, adoption for commercial/organizational usage, etc. with the permission of the University and respective supervisor. This above statement applies to all students and faculty members.

Names:

#	ID	Name
<b>1</b>	42110153	Aya Mohamed Attia
<b>2</b>	42110396	Bakinam Mohamed AbuElmatty
<b>3</b>	42110024	Mohamed Salama Abdelfattah
<b>4</b>	42110491	Mohamed Ahmed Osama
<b>5</b>	42110369	Abdullah Mostafa Mohamed
<b>6</b>	42110320	Hassan Ali Hassan

Supervised by:

**Prof. Dina Darwish**

## **Anti-Plagiarism Declaration**

---

This is to declare that the above publication produced under the supervision of Dr. Dina Darwish having the title "Anemia Detection" is the sole contribution of the author(s) and no part hereof has been reproduced illegally (cut and paste) which can be considered as Plagiarism. All referenced parts have been used to argue the idea and have been cited properly. We will be responsible and liable for any consequence if violation of this declaration is proven.

Names:

#	ID	Name
1	42110153	Aya Mohamed Attia
2	42110396	Bakinam Mohamed AbuElmatty
3	42110024	Mohamed Salama Abdelfattah
4	42110491	Mohamed Ahmed Osama
5	42110369	Abdullah Mostafa Mohamed
6	42110320	Hassan Ali Hassan

## Acknowledgment

---

We wish to extend our deepest gratitude to all individuals whose contributions were vital to the successful culmination of this project. Foremost,

we express our heartfelt appreciation to our esteemed supervisor, Prof. Dina Darwish, Teaching Assistant Eng/ Ahmed Nousir whose profound insights and unwavering support were pivotal throughout the research journey.

Her mentorship has not only enriched our work but has also inspired us to strive for excellence.

We are profoundly grateful to our peers and fellow researchers for their invaluable perspectives and constructive feedback, which have significantly enhanced the depth and quality of our research.

Lastly, we convey our sincerest thanks to our families for their steadfast support and encouragement. Their belief in us fueled our determination and resilience during the challenges we encountered.

# Table of Contents

---

Acknowledgement.....	5
Table of contents.....	6
List of figures.....	10
List of tables.....	13
List of abbreviations and acronyms.....	14
Abstract.....	15

## Chapter 1: introduction

1.1 Overview .....	17
1.2 Motivation .....	17
1.3 Problem Statement .....	18
1.4 Objective .....	19
1.5 Aim .....	19
1.6 Scope .....	20
1.7 General constraints .....	20
1.8 Organization of the dissertation .....	21
1.9 Contributions .....	22
1.10 Role of each Team member .....	23

## Chapter 2: Background and Previous work

2.1 Background .....	25
2.2 Previous work .....	30

## Chapter 3: Technology used and Purposed Model

3.1 Technology Used .....	51
3.1.1 Artificial Intelligence.....	51
3.1.2 Machine Learning.....	53
3.1.3 Deep Learning.....	55
3.1.4 Large Language Model.....	59

3.1.5 Natural Language processing.....	61
3.1.6 Flutter.....	64
3.2 Proposed Model .....	65

## **Chapter 4: Planning and Analysis**

4.1 Planning .....	70
4.1.1 Feasibility study and estimated cost .....	70
4.1.2 Gantt chart .....	72
4.2 Analysis and limitations of existing system .....	73
4.3 Need for new system .....	73
4.4 Analysis of new system .....	74
4.4.1 User Requirements .....	75
4.4.2 System Requirements .....	75
4.4.3 Domain Requirements .....	75
4.4.4 Functional Requirements .....	76
4.4.5 Non-Functional Requirements .....	76
4.5 Advantages of new system .....	77
4.6 User characteristics .....	79

## **Chapter 5: Design**

5.1 Design and Implementation Constraints .....	81
5.2 Assumptions and dependencies .....	81
5.3 Risks and risk management .....	84
5.4 Design of database ERD .....	85
5.4.1 Entity Relationship Diagram .....	85
5.4.2 Mapping of Entity Relationship Diagram .....	85
5.5 Class diagram .....	86
5.6 Use case diagram .....	87
5.6.1 Primary Use-case Diagram .....	87
5.6.2 Use-case Scenarios .....	88
5.7 Activity diagram .....	89

5.8 Sequence diagram .....	90
5.9 State diagram .....	91

## **Chapter 6: Implementation and Testing**

6.1 Data Set.....	93
6.1.1 Classification DataSets.....	93
6.1.2 Segmentation DataSets.....	97
6.1.3 LLM DataSets.....	100
6.2 Image Classification.....	103
6.2.1 Preprocessing.....	103
6.2.2 Models.....	116
6.2.3 Evaluation .....	122
6.2.4 Results .....	123
6.3 LLM &NLP Based Medical Chat Bot.....	127
6.3.1 Model Architecture .....	127
6.3.2 Model selection and compression strategy.....	130
6.3.3 Dataset loading and preprocessing.....	131
6.3.4 exploratory data analysis and virtualization.....	133
6.3.5 Embedding generation for semantic search.....	134
6.3.6 Indexing Using FAISS.....	134
6.3.7 Hybrid Querying Function.....	135
6.3.8 Context -Aware prompt construction .....	135
6.3.9 Interference and Generation.....	136
6.3.10 Evaluation and sample questions .....	136
6.4 Software.....	139
6.4.1 Hardware Requirements.....	139
6.4.2 Software Requirements.....	139
6.4.3 Functionalities of Application.....	140
6.4.4 UI and UX of Application.....	142

## **Chapter 7: Conclusion and Future work**

7.1 Conclusion.....	147
7.2 Future work.....	148
<b>References .....</b>	<b>111</b>

## List of figures

---

Figure 2.1: Global anemia impact.....	26
Figure 2.2: LLM Workflow.....	28
Figure 2.3: Conjunctiva Pallor Detection.....	29
Figure 2.4: Body parts most used in non-invasive anemia detect.....	35
Figure 2.5: ROI of body parts.....	35
Figure 2.6: ROI of the Conjunctiva of the Eye.....	36
Figure 2.7: ROI of the palm and fingernails.....	36
Figure 2.8: Detect anemia based on 3 criteria.....	42
Figure 3.1: Alan Turing Judge.....	51
Figure 3.2: Alan Turing 4 Approach.....	52
Figure 3.3: What is AI.....	53
Figure 3.4: Traditional programming vs Machine learning.....	54
Figure 3.5: Type of machine learning.....	55
Figure 3.6: Describe Deep Learning.....	56
Figure 3.7: Forward propagation and Back propagation.....	56
Figure 3.8: Forward and Backward in RNN.....	57
Figure 3.9: RNN.....	57
Figure 3.10: CNN from input to SoftMax.....	58
Figure 3.11: RGB Matrix.....	58
Figure 3.12: Calculate convolution matrix.....	58
Figure 3.13: Large Language Model.....	59
Figure 3.14: Flutter .....	64

Figure 3.15: updated detailed description of the system.....	65
Figure 6.1: unbalanced bar chat.....	94
Figure 6.2: balanced bar chat .....	96
Figure 6.3: unbalanced data vs balanced data .....	96
Figure 6.4: Nail Dataset segmentation.....	98
Figure 6.5: Region of nail bed.....	98
Figure 6.6: palm Dataset segmentation.....	99
Figure 6.7: Region of central palm.....	99
Figure 6.8: conjunctiva Dataset segmentation.....	100
Figure 6.9: Region of inner Eyelid.....	100
Figure 6.10: Distribution of patient question lengths .....	101
Figure 6.11: Distribution of doctor question lengths .....	102
Figure 6.12: anemia-related patient queries .....	102
Figure 6.13: u-net training setup .....	106
Figure 6.14: training curves.....	107
Figure 6.15: conjunctiva implementations .....	107
Figure 6.16: fingernails implementation .....	111
Figure 6.17: palm segmentation.....	115
Figure 6.18: diagram of decision tree .....	116
Figure 6.19: random forest process.....	117
Figure 6.20: XGboost prediction .....	118
Figure 6.21: VIT architecture.....	119
Figure 6.22:CNN model .....	120
Figure 6.23: Dense NET121 Architecture.....	121

Figure 6.24: efficientnet-B0 Architecture.....	122
Figure 6.25: Random forest nails training.....	124
Figure 6.26: Random forest palm training .....	125
Figure 6.27: XGboost conjunctiva training .....	127
Figure 6.28:4-bit quantization setup .....	130
Figure 6.29: model loading.....	130
Figure 6.30: dataset loading&cleaning .....	130
Figure 6.31: text cleaning function.....	131
Figure 6.32: length distribution(before cleaning) .....	131
Figure 6.33: length distribution(after cleaning) .....	132
Figure 6.34: word cloud(before cleaning) .....	132
Figure 6.35: word cloud(after cleaning) .....	133
Figure 6.36: sentence embedding generation.....	133
Figure 6.37: FAISS index creation.....	134
Figure 6.38: hybrid retrieval logic .....	135
Figure 6.39: prompt construction (doctor vs patient).....	136
Figure 6.40: LLM generation settings .....	136
Figure 6.41: LLM generation settings .....	141

## List of tables

---

Table 2.1: Palm Metrics.....	38
Table 2.2: Fingernails Metrics.....	39
Table 2.3: Conjunctiva Metrics.....	42
Table 2.4: General Metrics.....	44

## List of Abbreviations and Acronyms

---

Abbreviation/Acronym	Meaning
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
CNN	Convolutional Neural Network
SVM	Support Vector Machine
k-NN	k-Nearest Neighbors
PCV	Packed Cell Volume
RBC	Red Blood Cell Count
TLC	Total Leukocyte Count
MCH	Mean Cell Hemoglobin
MCHC	Mean Cell Hemoglobin Concentration
MCV	Mean Cell Volume
RDW	Red Cell Distribution Width
Hb	Hemoglobin
DNN	Deep Neural Network
AUC	Area Under the Curve
ROC	Receiver Operating Characteristic
WHO	World Health Organization
TP	True Positive
FP	False Positive
TN	True Negative
FN	False Negative

## Abstract

---

This project presents an in-depth exploration of non-invasive techniques for anemia detection through the application of advanced machine learning (ML) models, specifically focusing on the analysis of images of hands, palms, and fingernails. Anemia remains a widespread global health concern, disproportionately affecting vulnerable populations, including children and pregnant women. While traditional diagnostic methods are effective, they are often invasive and may not be readily accessible in resource-constrained environments, highlighting the urgent need for alternative diagnostic solutions. By synthesizing existing research, this review evaluates a range of ML methodologies, such as Convolutional Neural Networks (CNNs) and ensemble learning approaches, examining their accuracy and reliability in diagnosing anemia through image analysis. A notable feature of this study is the incorporation of smartphone technology for image capture, which enhances accessibility and affordability in the diagnostic process.

The findings emphasize the significant potential of non-invasive, ML-based methods for anemia detection, particularly in underserved communities, while also identifying critical gaps in the current research landscape. These gaps include the need for larger, more diverse datasets and the development of algorithms capable of improving diagnostic precision and adapting to real-world applications.

Although existing models, ranging from traditional machine learning techniques to sophisticated neural networks, have demonstrated notable advancements, further innovation is essential for the effective implementation of real-time testing and practical application. By harnessing the latest developments in image processing and ML, this review underscores the promise of these technologies to facilitate timely medical interventions, ultimately enhancing health outcomes for millions affected by anemia worldwide.

# CHAPTER

# ONE

# Chapter 1: Introduction

---

## 1.1 Overview

"Through the lens of innovation, a single scan could illuminate the path to better health."

Anemia is a common medical condition characterized by a deficiency of hemoglobin or red blood cells, leading to symptoms such as fatigue, dizziness, and shortness of breath. Traditional methods for anemia diagnosis rely on blood tests, which can be invasive, costly, and inaccessible to people in remote or underserved areas. With advancements in artificial intelligence (AI) and machine learning (ML), non-invasive techniques for anemia detection have emerged as a promising alternative.

This project aims to develop an AI-powered mobile application that utilizes image processing and machine learning techniques to detect anemia from images of the conjunctiva, fingernails, and palm. The application will integrate Large Language Models (LLMs) to provide real-time feedback, answer health-related questions, and recommend further medical actions. This innovative approach enhances accessibility, offering a fast, user-friendly, and cost-effective anemia screening tool for individuals and healthcare providers.

## 1.2 Motivation

**Early Disease Detection:** Many cases of anemia go undiagnosed until symptoms worsen. A mobile-based AI solution allows for early and accessible detection, helping individuals take preventive action before complications arise.

**Healthcare Accessibility:** In many regions, access to medical facilities for blood tests is limited. A non-invasive, smartphone-based solution can bridge this gap, especially for pregnant women, children, and people in rural areas.

**Cost-Effective Screening:** Frequent blood testing can be expensive and inconvenient. This AI-driven method provides a low-cost, at-home alternative for preliminary anemia detection.

**Real-Time Health Insights:** By integrating LLM-powered chatbots, users can receive instant guidance, including dietary advice, risk assessments, and follow-up recommendations, improving overall health awareness.

**Public Health Impact:** Anemia affects a significant portion of the global population, contributing to fatigue, cognitive impairment, and maternal health risks. Widespread use of AI-assisted screening can lead to early intervention and better health outcomes.

**Digital Healthcare Integration:** The application can be linked with telemedicine services, allowing users to share their results with doctors, improving remote monitoring and healthcare efficiency.

### 1.3 Problem Statement

Anemia remains a critical global health issue affecting over 1.62 billion individuals, disproportionately impacting vulnerable populations such as children and pregnant women. Traditional diagnostic methods for anemia primarily rely on invasive blood tests that require laboratory infrastructure and trained personnel. These methods are often inaccessible in low-resource settings, delaying timely diagnosis and treatment. Moreover, subjective assessments like examining pallor can lead to inconsistent results due to variations in expertise among healthcare providers.

The challenges in early and accurate anemia detection, especially in underserved regions, underscore the need for an innovative, non-invasive, and cost-effective solution. Leveraging advancements in image processing, machine learning (ML), and smartphone technology provides an opportunity to address these limitations, enabling broader access to anemia diagnostics.

## 1.4 Objectives

- **Enhance Early Anemia Detection:** Provide a non-invasive and accessible alternative to traditional blood tests, enabling users to assess their anemia risk quickly.
- **Real-Time Analysis:** Utilize machine learning models to analyze images of the conjunctiva, fingernails, and palm instantly, ensuring fast and accurate results.
- **LLM-Powered Health Assistant:** Integrate an **LLM-based chatbot** to answer user queries, offer personalized health recommendations, and suggest follow-up actions.
- **Improve Accessibility & Usability:** Design a user-friendly mobile application that can be used by individuals, caregivers, and healthcare professionals with minimal training.
- **Support for Diverse Populations:** Ensure the model performs well across different skin tones, lighting conditions, and demographics by continuously improving with user feedback.
- **Integration with Healthcare Systems:** Enable potential integration with medical records or telemedicine platforms to assist doctors in monitoring anemia-related health trends.

## 1.5 Aim

The aim is to develop an AI-powered mobile application capable of accurately detecting anemia non-invasively using image processing and machine learning techniques. The application will integrate **Large Language Models (LLMs)** to provide real-time health insights and recommendations, making anemia detection more accessible, efficient, and user-friendly.

## **1.6 Scope**

The proposed mobile application aims to assist users in detecting and identifying anemia non-invasively using advanced machine learning techniques, including Large Language Models (LLMs). The primary features include real-time anemia assessment through image analysis of the conjunctiva, fingernails, and palm, captured using the mobile device's camera. Additionally, the application leverages an LLM-powered chatbot to provide users with personalized health insights, answer queries related to anemia and recommend follow-up actions based on the analysis. Designed for both individuals and healthcare professionals, the application enhances accessibility to early anemia detection, reducing dependency on invasive tests and improving health awareness in everyday medical activities.

## **1.7 General Constraints**

### **1.7.1 Image Sources Supported:**

The app will support non-invasive anemia detection using images of the conjunctiva, fingernails, and palm. These specific body areas are chosen due to their visible pallor changes, which can indicate anemia.

### **1.7.2 Real-Time Processing:**

The application will analyze images in real time using the mobile device's camera, ensuring minimal delay in providing results. The LLM-powered chatbot will also offer instant responses to user queries.

### **1.7.3 Camera Quality:**

The accuracy of anemia detection depends on the quality of the mobile device's camera. Users should have a camera with a resolution of at least 12 megapixels for optimal performance. Additionally, sufficient lighting is required for accurate image analysis, and the app may prompt users to enable the flashlight if necessary.

#### **1.7.4 Lighting Conditions:**

The app is designed to function under various lighting conditions, but extreme low light or overexposed images may affect detection accuracy. Users will receive guidance on optimal image capture conditions.

#### **1.7.5 Data Privacy & Security:**

To ensure user privacy, all captured images and medical data will be processed on-device whenever possible or securely encrypted before cloud-based analysis. The LLM chatbot will not store or share user interactions without explicit consent.

#### **1.7.6 Model Limitations:**

The accuracy of anemia detection may vary based on individual differences in skin tone, lighting, and camera settings. The system will continuously improve by incorporating user feedback and refining the machine learning models over time.

### **1.8 Organization of the Dissertation**

This section outlines the structure of the remaining chapters of this document, providing a roadmap for understanding the solution to the anemia detection problem.

#### **Chapter 2: Background and Previous Work**

This chapter provides a detailed review of the literature on anemia detection, including traditional methods, advancements in machine learning, and the use of image-based diagnostics. It also examines prior studies and datasets that are foundational to the project.

#### **Chapter 3: Technology Used and Proposed Model**

This chapter covers the technology employed in the development of the proposed system, with specific focus on Artificial Intelligence, Machine Learning, and Deep Learning. It also outlines the proposed model for anemia detection.

#### **Chapter 4: Planning and Analysis**

This chapter discusses the planning and analysis phase of the project. It includes the feasibility study, estimated costs, and project timelines. The limitations of existing systems are also analyzed, followed by the justification for the need for a new system, including user, system, and domain requirements.

#### **Chapter 5: Design**

This chapter focuses on the design phase of the project. It includes system architecture, database design, user interface elements, and key components. Assumptions, dependencies, constraints, and risk management strategies are discussed as part of the design process.

#### **Chapter 6: Implementation and Testing**

This chapter provides details on the implementation process, including the development of machine learning models, system integration, and application features. It also describes the testing methodologies used, including unit testing and integration testing to validate the system's performance.

#### **Chapter 7: Conclusion and Future Work**

This chapter summarizes the conclusions drawn from the research and implementation of the anemia detection system. It also provides recommendations for future work in this area and highlights the contributions made by the project.

### **1.9 Contributions**

This project contributes to the field of anemia detection through the following key aspects:

**Development of a Non-Invasive Diagnostic Tool:** By utilizing ML algorithms and image processing techniques, the system detects anemia from images of the conjunctiva, fingernails, and palms, eliminating the need for invasive procedures.

**Integration of AI for Real-Time Analysis:** The application incorporates advanced ML models, such as Convolutional Neural Networks (CNNs), for accurate, real-time image analysis.

**LLM-Powered Assistance:** A Large Language Model (LLM)-based chatbot is integrated into the application to provide users with instant health insights, dietary recommendations, and follow-up guidance.

**Enhanced Accessibility:** Designed as a smartphone application, the system is affordable, user-friendly, and available to individuals and healthcare professionals in resource-limited environments.

**Data Security and Privacy:** Ensures user data is encrypted and processed securely, adhering to global healthcare data regulations such as HIPAA and GDPR.

**Scalability for Public Health:** Offers the potential for widespread implementation to monitor anemia at community and population levels, particularly in underserved areas.

## **1.10 Role of Each Team Member**

The project team's responsibilities were distributed as follows:

**Data Collection and Preprocessing (2 Members):** Responsible for acquiring and preparing datasets of images (conjunctiva, fingernails, palms). This includes noise reduction, augmentation, and normalization to ensure high-quality inputs for ML models.

**ML Model Development (3 Members):** Focused on designing and fine-tuning ML models, including CNNs, k-NN, and Random Forest algorithms. Tasks included training, validation, and performance optimization.

**Flutter and GUI Design (2 Members):** Developed the user interface and overall application architecture using Flutter. Tasks included creating an intuitive design for image capture, results display, and chatbot integration.

**Transfer Learning and Deployment (Entire Team):** Collaboratively implemented advanced pre-trained models, integrated APIs for real-time processing, and deployed the system for testing.

**Testing and Validation (Entire Team):** Conducted rigorous testing to ensure functionality, reliability, and accuracy of the application under various conditions, including diverse user demographics and environmental factors.

# CHAPTER

# TWO

## Chapter 2: Background and Previous Work

---

### 2.1 Background

Anemia, a condition characterized by reduced hemoglobin levels or a diminished red blood cell count, affects over 1.62 billion people globally, according to the World Health Organization (WHO) [37][38]. The condition disproportionately impacts vulnerable groups such as women, children, and the elderly. Causes include nutritional deficiencies (e.g., iron, folate, and vitamin B12) [37], chronic diseases [38], parasitic infections, and genetic disorders [39]. Left untreated, anemia can lead to serious health outcomes, including fatigue, impaired cognitive function, weakened immunity, and even mortality [37][40]. Early detection is critical, especially for high-risk populations, to prevent complications and promote better health outcomes [38][42].

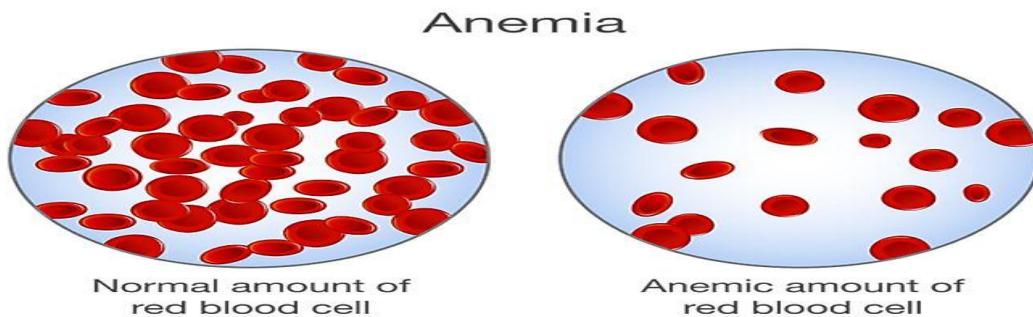


Figure 2.1: Global anemia impact

### Traditional Challenges in Anemia Detection

The standard diagnostic method for anemia involves blood tests to measure hemoglobin levels or red blood cell counts [39][41]. While effective, these methods are associated with several limitations:

- **Invasiveness:** Blood sample collection can be uncomfortable, particularly for young children and vulnerable individuals [33].
- **Accessibility Barriers:** Many low-resource regions lack the laboratory facilities, skilled personnel, and infrastructure needed for traditional blood testing [37].
- **Subjectivity:** Physical assessments, such as examining skin or conjunctival pallor, often depend on the examiner's expertise and can lead to inconsistent results [40][42].

These challenges highlight the urgent need for non-invasive, scalable, and cost-effective solutions for anemia detection, particularly in underserved areas [36][41].

## **The Role of LLMs in Healthcare**

Large Language Models (LLMs), such as GPT-based models, have shown immense potential in revolutionizing healthcare diagnostics. By integrating natural language understanding with diagnostic tools, LLMs can:

- **Enhance User Interaction:** Provide real-time feedback and answer user queries about anemia symptoms, dietary recommendations, and follow-up steps based on diagnostic outcomes.
- **Support Clinical Decision-Making:** Assist healthcare providers in interpreting results and determining appropriate treatment plans.
- **Improve Accessibility:** Enable users in remote or underserved regions to access expert-level health guidance through AI-driven chatbots.

When combined with image-based diagnostic tools, LLMs act as intelligent assistants, bridging gaps in healthcare knowledge and promoting proactive management of anemia.

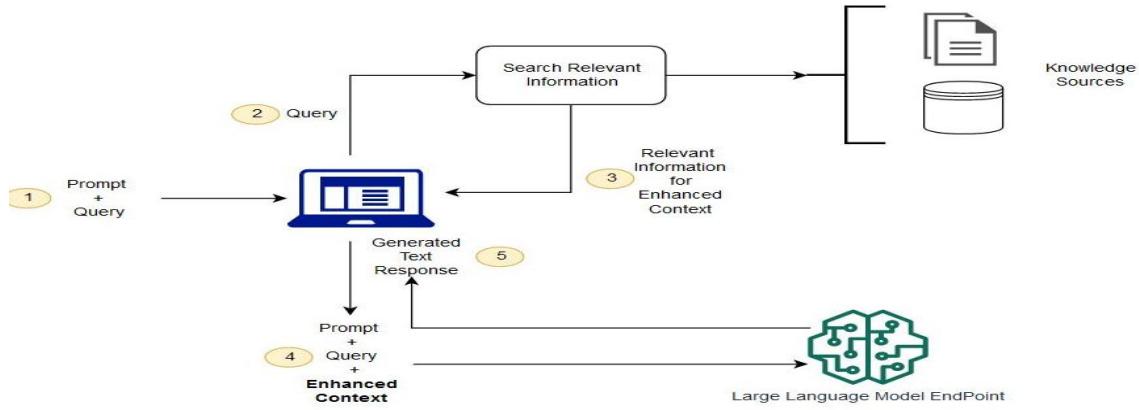


Figure 2.2: LLM Workflow

## Machine Learning and Image-Based Anemia Detection

Advancements in machine learning (ML), particularly deep learning models like Convolutional Neural Networks (CNNs), have opened new avenues for non-invasive anemia detection. CNNs excel in image classification tasks and are well-suited for identifying patterns in medical images, such as subtle changes in the conjunctiva indicative of anemia [6][7][14].

Key advantages of ML-driven anemia detection include:

- **Automated Diagnosis:** ML models can analyze images to detect anemia with high accuracy, reducing reliance on subjective human assessments [6][19].
- **Scalability:** Once trained, these models can be deployed on mobile devices for real-time diagnostics, making healthcare more accessible [14][19].
- **Integration with LLMs:** ML models provide diagnostic results, while LLMs can contextualize these findings, offering personalized recommendations and health education to users.

## Leveraging Smartphones for Anemia Detection

Smartphones, equipped with high-resolution cameras and processing capabilities, are ideal tools for non-invasive medical diagnostics. They allow users to capture images of the conjunctiva, fingernails, or palms, which ML models can analyze to detect anemia [16][33].

- **Conjunctival Analysis:** The pale appearance of the conjunctiva, caused by low hemoglobin levels, serves as a reliable indicator of anemia [6][14].



Figure 2.3: Conjunctiva Pallor Detection

- **Real-Time Feedback:** ML models deployed on smartphones can analyze images instantly, while LLMs provide users with actionable insights, such as dietary adjustments and potential next steps in care.

### Addressing Data and Model Challenges

The effectiveness of ML models in anemia detection depends heavily on the quality and diversity of the training data. High-quality datasets, such as CP-Anemic, containing images labeled for anemia diagnosis, are essential for developing robust models [6][23]. However, challenges persist:

- **Dataset Limitations:** Medical image datasets are often small, making it difficult to train highly accurate models [19][20].
- **Data Augmentation:** Techniques like image rotation, scaling, and flipping are used to expand datasets artificially, improving model performance [13][21].
- **Model Interpretability:** Explainable AI (XAI) techniques, such as Local Interpretable Model-Agnostic Explanations (LIME), highlight image regions that influence model predictions. This enhances trust in ML systems and facilitates their adoption in clinical settings [13][14][19].

## **Integration of LLMs and ML for Anemia Detection**

The combination of ML and LLMs creates a synergistic system for anemia detection. ML models provide objective, image-based diagnostic results, while LLMs enhance the user experience by:

- Simplifying complex medical information into user-friendly language.
- Offering real-time answers to health-related queries.
- Guiding users on next steps, such as consulting a healthcare provider or making lifestyle changes.

This integration has transformative potential, particularly in underserved areas, where traditional diagnostics are limited. By leveraging smartphones and AI, these systems democratize access to healthcare, enabling timely interventions and reducing the burden of anemia-related complications.

## **The Path Forward**

Despite the advancements, several challenges remain, including dataset diversity, model generalization, and ethical considerations. Robust measures for data privacy, security, and regulatory compliance are essential for deploying AI-powered diagnostic tools in real-world settings [13][16].

By addressing these challenges, this study aims to develop an AI-driven, non-invasive anemia detection system that combines ML's diagnostic power with LLMs' ability to engage and educate users. Such a solution has the potential to improve health outcomes globally, particularly in resource-limited settings.

## **2.2 Previous Work**

### **2.2.1. Traditional vs. Non-Invasive Diagnostic Approaches**

Anemia diagnosis has long relied on invasive methods, most commonly blood tests that assess hemoglobin levels [13][16]. These methods, while accurate, present notable drawbacks, especially in resource-constrained settings [18][19]. Blood tests require specialized equipment, trained personnel, and present risks such as infection and the disposal of biohazardous waste [19][21]. Moreover, blood sampling is often uncomfortable for patients, particularly children and pregnant women, who are the most vulnerable to anemia [37][40]. These limitations highlight the pressing need for non-invasive alternatives, particularly in underserved regions [16][19].

Recent advances in non-invasive anemia detection focus on reducing reliance on blood samples by analyzing visual cues from various body parts [6][14]. Commonly, pallor detection in areas such as the conjunctiva, fingernails, and palms serves as an indicator of anemia [7][25]. These areas exhibit visible signs of anemia, which can be captured and analyzed using image processing techniques, making non-invasive methods a viable alternative [13][14]. The integration of machine learning (ML) and deep learning (DL) techniques, particularly with commonly available tools such as smartphones for image acquisition, has shown great promise in improving access to diagnostic solutions in low-resource settings [6][18]. These ML models can process images to detect anemia-related symptoms, offering a scalable solution with minimal infrastructure [16][19].

### **2.2.2. Machine Learning in Healthcare Diagnostics**

Machine learning, especially deep learning, has revolutionized various fields of healthcare diagnostics, including the detection of conditions such as cancer, diabetes, and cardiovascular diseases [6][18]. ML and DL methods have now been effectively applied to anemia detection by analyzing medical images to identify patterns or anomalies that may not be visible to the human eye [19][21]. Some of the most frequently used ML models in anemia detection include

Convolutional Neural Networks (CNNs), Support Vector Machines (SVMs), Naïve Bayes, k-Nearest Neighbors (k-NN), and Decision Trees [7][25].

### **2.2.3. Convolutional Neural Networks (CNNs)**

CNNs are among the most widely used and successful models for image-based anemia detection [6][7]. These networks excel at automatically learning and extracting hierarchical features from images, which makes them particularly suited for medical image analysis [19][25]. For example, a study demonstrated that CNNs trained on datasets of conjunctiva, fingernail, and palm images achieved an impressive accuracy rate of 99.12% for anemia detection [7][9]. CNNs' ability to process high-dimensional visual data allows them to autonomously identify relevant features without manual intervention, improving diagnostic accuracy [6][19]. Furthermore, CNNs are capable of generalizing well across diverse populations and image conditions, making them highly effective in real-world scenarios [18][25].

In studies focused on conjunctiva images, the color variations detected by CNNs directly correlate with hemoglobin levels, offering a more objective and reliable diagnosis compared to traditional human visual inspection [7][19]. The high accuracy and sensitivity observed in these studies underscore the potential of CNNs for non-invasive anemia detection in clinical settings [6][19].

### **2.2.4. Support Vector Machines (SVMs)**

Although not as effective as CNNs for large-scale image classification tasks, Support Vector Machines (SVMs) have been used in anemia detection with reasonable success [6][19]. SVMs function by finding a hyperplane that best separates different classes of data—anemic versus non-anemic patients [24][25]. While SVMs require more intensive feature engineering and are not as well suited for handling complex image datasets as CNNs, studies such as those conducted by [21] have shown that SVMs can achieve accuracy rates of approximately 95.34% when applied to smaller, well-engineered datasets [19][24]. Nonetheless, CNNs typically outperform SVMs in tasks involving complex image data due to their ability to automatically learn and extract features without human intervention [19][6].

### **2.2.5. Naïve Bayes and Decision Trees**

Naïve Bayes classifiers and Decision Trees have also been explored as alternatives for anemia detection, particularly in scenarios where they work with smaller datasets [7][20]. These models are simpler than CNNs and are less prone to overfitting, making them a good choice for small-scale applications [24]. For instance, studies like those by [6][22] demonstrated that Naïve Bayes classifiers achieved 98.96% accuracy when applied to a dataset of 527 images of conjunctiva, palm, and fingernail regions [21]. Similarly, Decision Trees have been shown to perform well in this domain, providing interpretable decision-making paths [7][20]. However, both models are sensitive to noisy or inconsistent data, which can affect their performance in real-world applications [24][19].

### **2.2.6. Evaluation Metrics**

Across various studies, a consistent set of evaluation metrics is used to assess the performance of ML models for anemia detection [12][15]. These metrics include accuracy, sensitivity (recall), specificity, precision, and F1-score [10][18].

- Accuracy measures the overall correctness of the model [15].
- Sensitivity (or recall) indicates the model's ability to correctly identify anemic patients [12][14].
- Specificity measures the model's ability to correctly identify non-anemic patients [18].
- Precision shows how many of the predicted positives were actually correct [17].
- F1-score is the harmonic means of precision and recall, providing a balance between the two [10].

The highest reported accuracy for non-invasive anemia detection has been achieved by a CNN model, which reached 99.12% accuracy with 99.89% sensitivity [12][13]. In comparison, SVMs typically achieve accuracy rates around 95%, and Naïve Bayes and Decision Trees perform slightly lower but still above 98% [14][18]. CNN models consistently outperform other techniques in terms of specificity and precision, making them particularly valuable for reducing false positives in anemia detection [13][19].

## 2.2.7 Large Language Models (LLMs) in Non-Invasive Anemia Detection

While machine learning (ML) models, especially convolutional neural networks (CNNs), have demonstrated remarkable success in non-invasive anemia detection, the potential for integrating large language models (LLMs) into this field is emerging. LLMs, such as GPT and similar architectures, have shown promise in several areas of medical research, including image interpretation, patient data analysis, and even the automation of diagnostic workflows. The ability of LLMs to process and analyze vast amounts of unstructured text and image-related data can offer additional support to ML models in the following ways:

- 1) **Automated Report Generation:** LLMs can be used to automate the generation of diagnostic reports based on the analysis of images, such as conjunctiva, fingernail, and palm images. This can make the process faster and more efficient, allowing healthcare providers to receive detailed insights on anemia detection from images without manual intervention.
- 2) **Data Augmentation and Annotation:** LLMs can assist in generating synthetic data for training purposes. For instance, in cases where datasets are small or incomplete, LLMs could generate plausible clinical narratives, enhancing existing datasets with additional variations and ensuring that the models are exposed to a broader set of data.
- 3) **Integration with Electronic Health Records (EHR):** By combining LLMs with ML models, it is possible to create a system that not only analyzes images but also correlates these results with patient medical histories stored in electronic health records. The ability of LLMs to process and understand text-based health data could improve the accuracy of anemia diagnosis by considering a patient's full medical context.
- 4) **Multimodal Learning:** LLMs could be used in multimodal learning scenarios, where both text and image data are processed together. This would allow models to interpret a patient's image in conjunction with textual descriptions or clinical notes, providing a more holistic approach to anemia detection and potentially identifying patterns that might be missed when considering each data type in isolation.

**5) Real-Time Consultation and Assistance:** LLMs can act as virtual assistants, providing real-time support for healthcare professionals by suggesting possible diagnoses or clarifying medical terms and procedures related to anemia detection. This can enhance decision-making processes, especially in remote or under-resourced healthcare settings.

While LLMs are not yet a dominant force in non-invasive anemia detection, their potential for enhancing diagnostic accuracy and efficiency makes them an exciting area for future research and development. As advancements in both machine learning and natural language processing continue, the integration of LLMs with existing diagnostic tools could provide a more comprehensive and accessible solution for anemia detection.

### 2.2.8. Body Parts for Non-Invasive Anemia Detection

The body parts most used for non-invasive anemia detection are the conjunctiva, fingernails, and palms, each offering distinct advantages and challenges for image acquisition and analysis [20][22].



Figure 2.4 body parts most used for non-invasive anemia detection

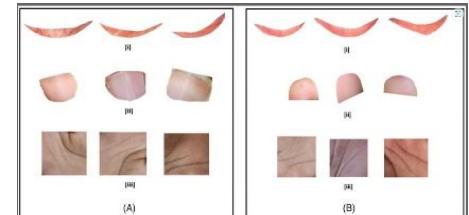


Figure 2.5: ROI of body parts

**Conjunctiva:** The conjunctiva, the thin membrane covering the white part of the eye, is a highly reliable indicator of anemia. Pale conjunctiva is a well-established clinical sign of anemia, as reduced hemoglobin levels directly affect its coloration [20]. Numerous studies have shown that CNN models can accurately classify anemia using conjunctiva images, often achieving the highest accuracy among the three body parts [21]. However, challenges remain in ensuring consistent image quality, as variations in lighting and camera quality can introduce noise into the data [22].



Figure 2.6 ROI of the Conjunctiva of the Eye

**Fingernails and Palm:** Fingernail and palm images have also been used in non-invasive anemia detection, though they provide less distinct visual cues than the conjunctiva. While fingernails often show pallor in anemic patients and palms exhibit reduced color intensity, these areas do not offer as much visual information as the conjunctiva [23]. As such, models trained on fingernail and palm images tend to achieve lower accuracy compared to CNNs applied to conjunctiva images [24]. However, models like k-NN and SVMs have been particularly effective for detecting anemia from palm and fingernail images [25].



Figure 2.7 ROI of the palm and fingernails

### 2.2.9. Types Used for Detecting Anemia

#### I. Palm

In the field of anemia detection, significant progress has been made over the years with various methods explored to improve diagnostic accuracy and accessibility. Traditional methods of anemia diagnosis, such as invasive blood tests, have always posed challenges in terms of accessibility, cost, and patient comfort. As a result, non-invasive alternatives have gained interest in recent years, aiming to provide faster, more affordable, and less intrusive solutions.

One of the prominent trends in recent work is the use of image-based approaches

for anemia detection. Medical images, such as conjunctiva, retinal, and palm images, have been utilized to identify visual signs of anemia. These methods leverage image processing and machine learning techniques to extract features

indicative of anemia-related conditions. As computational methods and imaging technologies have advanced, image-based approaches have shown promising results in automating the detection process, offering the potential for rapid screening and diagnosis without the need for complex laboratory procedures [26]. Machine learning techniques, particularly deep learning, have become integral in the development of automated anemia detection systems. Convolutional Neural Networks (CNNs) and other models have been applied to classify images and detect patterns that correlate with anemia. These methods have demonstrated improvements in accuracy, though challenges remain in ensuring the robustness and generalizability of models across diverse populations and different image types [27].

Despite progress, existing methods often face limitations, particularly with the need for large, high-quality datasets to train accurate models. Additionally, many image-based methods struggle with inconsistencies in image quality and lighting, which can impact the model's performance. Furthermore, while these approaches have made strides in improving diagnostic capabilities, there is still work to be done in making these systems accessible in low-resource settings [28].

Recent research has focused on combining multiple diagnostic techniques, including the integration of various machine learning models, to improve the overall performance of anemia detection systems. By combining different algorithms, researchers aim to address issues such as overfitting and improve the accuracy and reliability of predictions [29].

While non-invasive anemia detection has shown great potential, the field continues to evolve. Current research is increasingly exploring ways to refine and enhance existing methods, ensuring that future models can be more generalized, efficient, and scalable [30].

**Table 2.1: Palm Metrics**

Method	Accura	Sensitivity	Precision	F1 Score	Dataset
CNN [23]	99.92	99.98	99.79	99.89	2635 Images
Decision Tree (DT) [20]	95.62	97.92	98.60	98.01	Kaggle Anemia Dataset
Alex Net + Spatial Attention [1]	86.97	89.67	90.71	88.93	Mendeley Image
Proposed AMSA Model [22]	99.58	99.95	99.97	99.97	Real-time clinical trial data
Random Forest (RF) [7]	99.92	99.87	100.00	99.94	4260 from palm images

## I. Fingernails

In the field of anemia detection, significant strides have been made with various methods to enhance diagnostic accuracy and accessibility. Traditional methods, such as invasive blood tests, have always faced challenges like cost, patient discomfort, and accessibility. These limitations have spurred interest in non-invasive alternatives, which offer faster, more affordable, and less intrusive solutions for diagnosis [1][2].

One notable trend in recent research is the use of image-based approaches for anemia detection. Medical images, including those of the conjunctiva, retina, and palms, have been explored to identify visual signs of anemia [3][4]. Image processing and machine learning techniques are applied to extract features indicative of anemia-related conditions. The advancement in computational methods and imaging technologies has made these approaches promising for automating the detection

process. This has the potential to allow for rapid screening and diagnosis without relying on complex laboratory tests [5][6].

Machine learning, particularly deep learning, has become a crucial tool in the development of automated anemia detection systems. Convolutional Neural Networks (CNNs) and other deep learning models have been used to classify images and identify patterns that correlate with anemia [7][8]. These methods have shown improvements in diagnostic accuracy, although challenges persist in ensuring model robustness across diverse populations and varying image qualities [9][10].

Despite progress, existing methods face challenges, particularly regarding the need for large, high-quality datasets to train accurate models [11]. Image quality and lighting inconsistencies also present difficulties, potentially impacting model performance. While image-based techniques have improved diagnostic capabilities, making these systems viable in low-resource settings remains a significant hurdle [12][13].

Recent research has increasingly focused on combining multiple diagnostic techniques, including integrating various machine learning models, to boost the performance of anemia detection systems. By merging different algorithms, researchers aim to overcome challenges like overfitting and increase the reliability and accuracy of predictions [14][15].

Although non-invasive anemia detection methods show great promise, ongoing research is essential for refining and improving these techniques. The aim is to develop more generalized, efficient, and scalable models for broader application [16][17].

***Table 2.2: Fingernails Metrics***

Method	Accuracy	Sensitivity	Precision	F1 Score	Dataset
CNN [21]	98.33	97.44	97.64	97.54	non-anemic, anemic
KNN [21]	90.26	85	85	84	non-anemic, anemic

## I. Conjunctiva

In the field of anemia detection, significant progress has been made with the exploration of various methods to improve diagnostic accuracy and accessibility. Traditional diagnostic methods, such as blood tests, are often invasive, expensive, and difficult to access, especially in low-resource settings. As a result, non-invasive alternatives, including image-based approaches, have gained considerable attention in recent years [1][2]. These methods aim to provide faster, more affordable, and less intrusive solutions for anemia detection.

One of the major advancements in recent work is the use of medical images, such as those of the conjunctiva, retina, and palms, to detect signs of anemia [3][4]. Image processing and machine learning techniques are leveraged to extract features indicative of anemia-related conditions. With the rise of advanced computational techniques and improved imaging technologies, these image-based approaches hold the potential for automated, rapid screening, offering an alternative to traditional blood tests [5][6]. Machine learning, particularly deep learning, has played a crucial role in enhancing the accuracy and efficiency of these detection systems [7][8].

## Recent Advances and Key Papers

Several studies have contributed to this field by using deep learning models, particularly convolutional neural networks (CNNs), to classify and detect anemia from image datasets. A key feature of these approaches is their ability to automatically identify visual cues such as conjunctival pallor, a sign of anemia that can be detected through smartphone images.

## Key studies and their contributions include:

- "*Iron deficiency anemia detection using machine learning models: A comparative study of fingernails, palm, and conjunctiva of the eye images*" (Wiley, 2023): This study compares different image types (fingernail, palm, conjunctiva) for anemia detection, showing the advantages of conjunctiva images with deep learning models [9].
- "*CP-AnemiC: A conjunctival pallor dataset and benchmark for anemia detection in children*" (Science Direct, 2023): The study introduced a dataset focused on conjunctival pallor in children, aiming to standardize the evaluation of anemia detection methods using deep learning [10].
- "*Detection of anemia using conjunctiva images: A smartphone application approach*" (Science Direct, 2023): This paper proposed a smartphone-based

*application for anemia detection, emphasizing its accessibility and non-invasiveness for the general population [11].*

- "*Non-Invasive Anemia Detection Using Images Acquired from Smartphone Camera*" (Springer, 2022): This study examined the potential of using smartphone cameras for real-time anemia detection, using CNNs to process images of the conjunctiva and retina [12].
- "*NiADA (Non-invasive Anemia Detection App), a Smartphone-Based Application with Artificial Intelligence to Measure Blood Hemoglobin in Real-Time: A Clinical Validation*" (Research Gate, 2024): *The clinical validation of a smartphone application, NiADA, for real-time hemoglobin measurement, using AI-based methods to diagnose anemia through smartphone cameras [13].*

These studies collectively highlight the shift toward non-invasive, smartphone-based anemia detection using machine learning techniques. The datasets created, such as the CP-Anemic dataset, have provided valuable benchmarks for evaluating the performance of machine learning models in anemia detection [14].

## **Machine Learning Techniques and Dataset Evaluation**

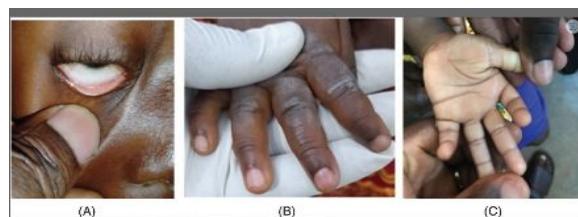
Several machine learning and deep learning models have been applied in these studies to enhance anemia detection. Convolutional Neural Networks (CNNs), ResNet50, VGG16, DenseNet121, and hybrid models combining multiple architectures have been particularly successful in detecting anemia-related features in images [15][16].

**Table 2.3: Conjunctiva Metrics**

Method	Accuracy	Sensitivity	Precision	F1 Score	AUC	Dataset
Decision tree [21]	97.32	98.49	93.67	96.02	97.70	710 images of the conjunctiva
ViT [3]	84.08	83.5	83.3	83.3	84.1	CP-AnemiC dataset includes 710 images

## II. General Detect

Non-invasive anemia detection using machine learning (ML) has gained significant attention due to its potential to overcome the limitations of traditional blood tests [1][2]. By analyzing medical images of the conjunctiva, fingernails, and palms, ML models can identify signs of anemia based on visible indicators such as pallor, which correlates with low hemoglobin levels [3][4]. These body parts are ideal candidates for this kind of analysis as their color changes directly reflect the state of the blood, offering a simple yet effective way to detect anemia without the need for invasive procedures [5][6].



**Figure 2.8: Detect anemia based on 3 criteria**

Studies have demonstrated that conjunctiva, the thin membrane covering the eyeball and inner eyelids, is particularly effective for detecting anemia. Its color tends to change in response to hemoglobin levels and using advanced ML models such as Convolutional Neural Networks (CNNs), researchers have achieved high accuracy in detecting anemia from conjunctiva images [1][2].

These models can provide precise results, with some studies reporting accuracy rates above 99% [3]. However, simpler methods like RGB thresholding have been used with less success due to their sensitivity to factors like image quality and lighting conditions [4].

Fingernails also serve as a valuable diagnostic area, with ML models such as Support Vector Machines (SVMs) and k-Nearest Neighbors (k-NN) applied to detect signs of anemia. The color of the fingernails, including pallor or discoloration, can reflect anemia [5]. However, like conjunctiva images, fingernail images are sensitive to external factors, such as lighting and camera quality, which can affect detection accuracy. To improve results, image preprocessing and augmentation techniques are often employed [6].

Palm images provide another useful diagnostic tool, as the color of the palms can change in anemic individuals. ML models like Naïve Bayes and Decision Trees have been used to analyze palm images with promising results [7]. However, the main challenge with palm images, as with fingernails, is that the datasets are often small, limiting the generalizability of the findings. Moreover, image quality can be influenced by factors like lighting, so careful standardization of image acquisition is necessary for reliable results [8].

Overall, studies have shown that while all three body parts—conjunctiva, fingernails, and palms—offer valuable clues for detecting anemia, conjunctiva images tend to provide the most accurate results [9]. The development of advanced machine learning models, particularly CNNs, has significantly improved the accuracy of these non-invasive diagnostic methods [10]. However, challenges remain, such as the variability in image quality and the need for larger, more diverse datasets to improve the generalizability and scalability of these methods [11]. Despite these limitations, the potential for ML-driven, non-invasive anemia detection is promising, especially in resource-limited settings where traditional blood tests are not accessible [12].

**Table 2.4: General Metrics**

Method	Accuracy	Sensitivity	Precision	F1 Score	Specify	Dataset
CNN	99.12	99.89	99.79	99..89	99.95	Conjunctiva, Palm, Fingernail images, augmented to 2635)
Naïve Bayes	98.96	99.97	99.97	99.97	99.98	Conjunctiva, Palm, Fingernail images, augmented to 2635)
Decision Tree	98.29	98.97	98.77	98.97	99.38	Conjunctiva, Palm, Fingernail images, augmented to 2635)
k-NN	98.92	99.89	99.79	99.89	99.98	Conjunctiva, Palm, Fingernail images, augmented to 2635)
SVM	95.34	94.59	95.99	93.23	98.97	Conjunctiva, Palm, Fingernail images, augmented to 2635)

## **2.2.10. Studies on Non-Invasive Anemia Detection Using Machine Learning**

Several studies have explored the use of machine learning (ML) for detecting iron deficiency anemia through non-invasive means, particularly focusing on medical images of the conjunctiva, fingernails, and palm. Below are key papers in this field:

### **1. Detection of Iron Deficiency Anemia by Medical Images: A Comparative Study of Machine Learning Algorithms**

This study investigates the use of ML algorithms, including CNN, k-NN, Naïve Bayes, SVM, and Decision Tree, for detecting anemia using palm images. By augmenting a dataset from 527 to 2635 samples, Naïve Bayes achieved the highest accuracy (99.96%), demonstrating the potential for cost-effective, non-invasive anemia detection [1].

### **2. Application of Ensemble Models Approach in Anemia Detection Using Images of the Palpable Palm**

This paper discusses the use of ensemble techniques (stacking, voting, boosting, and bagging) for analyzing palm images. The hybrid model achieved 99.73% accuracy, with stacking outperforming other methods. The study suggests the application of this method in mobile platforms for areas with limited resources [2].

### **3. Iron Deficiency Anemia Detection Using Machine Learning Models: A Comparative Study of Fingernails, Palm, and Conjunctiva of the Eye Images**

This comparative study explores the use of various ML algorithms (CNN, Naïve Bayes, SVM, k-NN, Decision Tree) for detecting iron-deficiency anemia. CNN achieved the highest accuracy (99.12%), particularly using palm images, which are considered reliable for anemia detection in children [3].

### **4. An Intelligent Non-Invasive System for Automated Diagnosis of Anemia Exploiting a Novel Dataset**

A machine learning-based system for detecting anemia using conjunctiva images was developed. The RUSBoost classifier achieved an accuracy of 88%

for Italian patients and 75% for Indian patients, demonstrating the robustness of the system across different ethnicities [4].

## **5. Detection of Anemia Using Conjunctiva Images: A Smartphone Application Approach**

A smartphone app for non-invasive anemia detection using conjunctiva images was created. Using CNN, Logistic Regression, and Gaussian Blur, the app achieved 92.5% accuracy, with plans for offline functionality and further data training to enhance performance [5].

## **6. A New Method and a Non-Invasive Device to Estimate Anemia Based on Digital Images of the Conjunctiva**

This study introduces a non-invasive device that estimates hemoglobin levels using conjunctiva images in the CIELAB color space. The device uses a k-nearest neighbor classifier and has potential for home or physician use, especially for large-scale screenings [6].

## **7. Iron Deficiency Anemia Detection Using Machine Learning Models: A Comparative Study of Fingernails, Palm, and Conjunctiva of the Eye Images**

This study again highlights the effectiveness of CNN in detecting anemia from images of the conjunctiva, fingernails, and palm, achieving 99.12% accuracy. The results suggest palm images as particularly reliable for anemia detection in children [7].

## **8. Detection of Anemia Using Conjunctiva Images: A Smartphone Application Approach**

This study presents a mobile app for detecting anemia through conjunctiva images, achieving 92.5% accuracy. It emphasizes the potential for cost-effective screening, particularly in regions with high anemia rates [8].

## **9. NiADA (Non-Invasive Anemia Detection App), a Smartphone-Based Application With Artificial Intelligence to Measure Blood Hemoglobin in Real-Time**

The NiADA app estimates hemoglobin levels from lower eyelid images, showing promising results comparable to laboratory methods. This tool is especially valuable in resource-limited settings [9].

## **10.Machine/Deep Learning-Assisted Hemoglobin Level Prediction Using Palpebral Conjunctival Images**

This research compares machine learning models to predict hemoglobin levels using conjunctiva images. CNNs outperformed other models in sensitivity and specificity, emphasizing their suitability for anemia detection [10].

## **11.An Overview of Using Deep Learning Algorithms for Anemia Detection**

This review highlights various AI and deep learning algorithms for analyzing conjunctiva, retina, and electrocardiogram images. The study emphasizes the potential of smartphone-based detection systems [11].

## **12.Iron Deficiency Anemia Detection Using Machine Learning Models: A Comparative Study of Fingernails, Palm, and Conjunctiva of the Eye Images**

In this comparative study, CNN was found to be the most accurate model for detecting iron-deficiency anemia, particularly from palm images, making it a strong candidate for mobile app integration [12].

## **13.Detecting Anemia Based on Palm Images Using Convolutional Neural Networks**

Using CNNs, this study achieved an accuracy of 96.43% for anemia detection based on palm images, suggesting its potential for fast, non-invasive screening [13].

## **14.Prediction of Anemia Using Machine Learning Algorithms**

This study used Random Forest and ensemble methods for predicting anemia in children under five, demonstrating an accuracy of 99%, and highlighting the importance of algorithm selection [14].

## **15.Iron Deficiency Anemia Detection Using Machine Learning Models: A Comparative Study of Fingernails, Palm, and Conjunctiva of the Eye Images**

This paper explores the reliability of palm images for detecting anemia, with CNNs achieving 99.12% accuracy, and highlights the potential for mobile applications in regions with limited medical resources [15].

## **16.Detection of Iron Deficiency Anemia by Medical Images: A Comparative Study of Machine Learning Algorithms**

This study confirmed the effectiveness of machine learning for detecting anemia from palm images, with Naïve Bayes achieving 99.96% accuracy [16].

## **17.Detection of Anemia Using Conjunctiva Images: A Smartphone Application Approach**

The study developed a smartphone app that detects anemia from conjunctiva images, achieving high accuracy and offering a quick, mobile-based solution [17].

## **18.Revolutionizing Anemia Detection: Integrative Machine Learning Models and Advanced Attention Mechanisms**

This study used an innovative ML framework with AlexNet and Multiple Spatial Attention (AMSA) modules, achieving 99.58% accuracy and improving diagnostic precision [18].

## **19.Machine Learning Algorithms for Iron Deficiency Anemia Detection in Children Using Palm Images**

A variety of ML models were used to analyze palm images for anemia detection, achieving accuracy above 99% for Decision Tree and Random Forest models [19].

## **20.Non-Invasive Anemia Detection from Conjunctival Images**

This study used a CNN-based model to detect anemia from conjunctival images, achieving 98% accuracy with plans for future improvements, including transfer learning [20].

## **21.Iron Deficiency Anemia Detection Using Machine Learning Models: A Comparative Study of Fingernails, Palm, and Conjunctiva of the Eye Images**

The study reaffirms the effectiveness of CNN in detecting iron-deficiency anemia with an accuracy of 99.12%, showing the feasibility of palm images for detection [21].

## **22.Detection of Anemia Using Conjunctiva Images: A Smartphone Application Approach**

This mobile application processed conjunctiva images for anemia detection, achieving 92.5% accuracy and offering a cost-effective solution [22].

## **23.Non-Invasive Anemia Detection Using Images Acquired from Smartphone Camera**

The study demonstrated the potential for anemia detection using smartphone-acquired images of the conjunctiva with a DenseNet-121 model, achieving 62.5% accuracy [23].

## **24.Non-Invasive Hemoglobin Estimation Using Different Color and Texture Features of Palm**

This paper presents a system for estimating hemoglobin levels using palm color changes, achieving high accuracy with RMSE of 0.597 [24].

# CHAPTER

# THREE

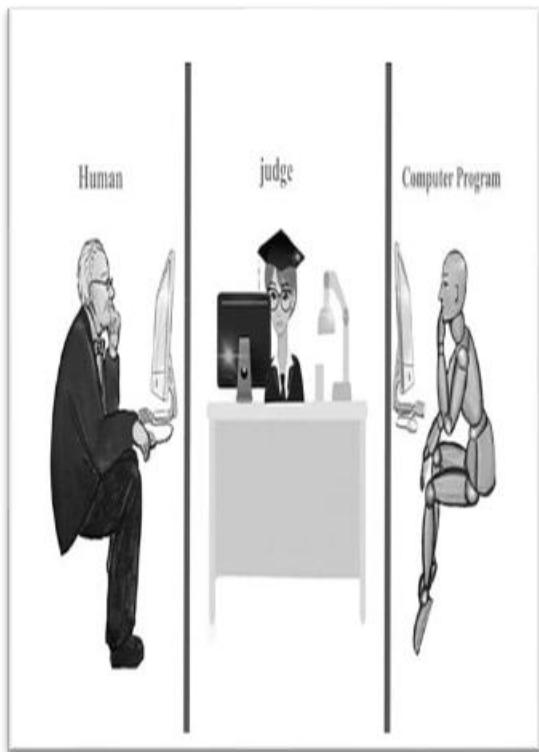
# Chapter 3: Technology used and Proposed Model

---

## 3.1 Technology Used

### 3.1.1 Artificial Intelligence (AI)

#### **3.1.1.1 AI Works and Concept:**



Artificial Intelligence (AI) is a dynamic field within computer science that seeks to develop systems capable of mimicking human intelligence. By leveraging algorithms and computational models, AI systems can perform tasks such as decision-making, language translation, problem-solving, and visual recognition. These systems analyze large datasets to recognize patterns, adapt to changes, and continuously improve their performance.

#### **Turing Test:**

Proposed by Alan Turing in 1950, the Turing Test is a benchmark for evaluating a machine's ability to exhibit human-like intelligence. In this test, a human evaluator engages in natural

Figure 3.1: Alan Turing Judge

language conversations with a machine and another human through a computer interface. If the evaluator cannot reliably distinguish the machine from the human, the machine is considered to have passed the test. This remains a foundational concept in AI philosophy and design.

### **3.1.1.2 Alan Turing Test is Based on 4 Approaches**

The Turing Test aligns with four distinct approaches to intelligence:

1. Thinking Humanly: AI attempts to replicate human thought processes, such as reasoning and problem-solving.
2. Thinking Rationally: AI emphasizes logical reasoning to achieve optimal outcomes based on defined principles.



Figure 3.2: Alan Turing 4 Approach

3. Acting Humanly: AI is designed to emulate observable human behaviors, such as responding to questions or recognizing emotions.
4. Acting Rationally: AI focuses on taking actions that achieve the most beneficial results given the available information.

### **3.1.1.3 Types of AI**

#### **1- Artificial Narrow Intelligence (ANI)**

ANI, or Weak AI, is specialized to handle specific tasks. These systems, like virtual assistants (e.g., Siri, Alexa) or fraud detection software, excel in well-defined domains but lack generalization capabilities.

#### **2- Artificial General Intelligence (AGI)**

AGI, also referred to as Strong AI, aspires to replicate human cognitive abilities, enabling systems to solve unfamiliar problems across multiple domains. Though theoretical, AGI represents a significant goal in AI research.

### 3- Artificial Super Intelligence (ASI)

ASI surpasses human intelligence in every aspect, including creativity, analytical thinking, and decision-making. While currently a hypothetical concept, ASI raises important ethical considerations regarding its potential impact on humanity.

What is AI?  
ANI vs. AGI vs. ASI

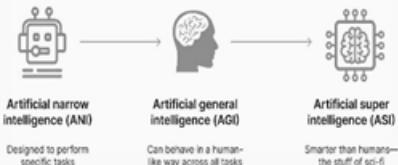


Figure 3.3: What is AI

#### 3.1.2 Machine Learning (ML)

##### Traditional Computer Programming



##### Machine Learning



Figure 3.4: Traditional programming vs Machine learning

Machine Learning may be defined as the study which provides the system ie., (computer) to learn automatically on its own experiences it had and improve accordingly without being explicitly programmed. ML is an application or subset of AI. The field of machine learning is concerned with the typical questions for how to generate computer programs that can be automatically improved with their experience.

While we are implementing an ML method requires plenty of data, which is known as training data, that is fetch into the method and based on these data, the machine learning for performing a specified task. The data such as text, images, audio, etc... It is also known as a self-learning algorithm. It must allow the machines to learn by themselves by their experience without any human intervention or help.

### 3.1.2.1 Types of Machine Learning

Machine learning uses two types of techniques: supervised learning, which trains a model on known input and output data so that it can predict future outputs, and unsupervised learning, which finds hidden patterns or intrinsic structures in input data.

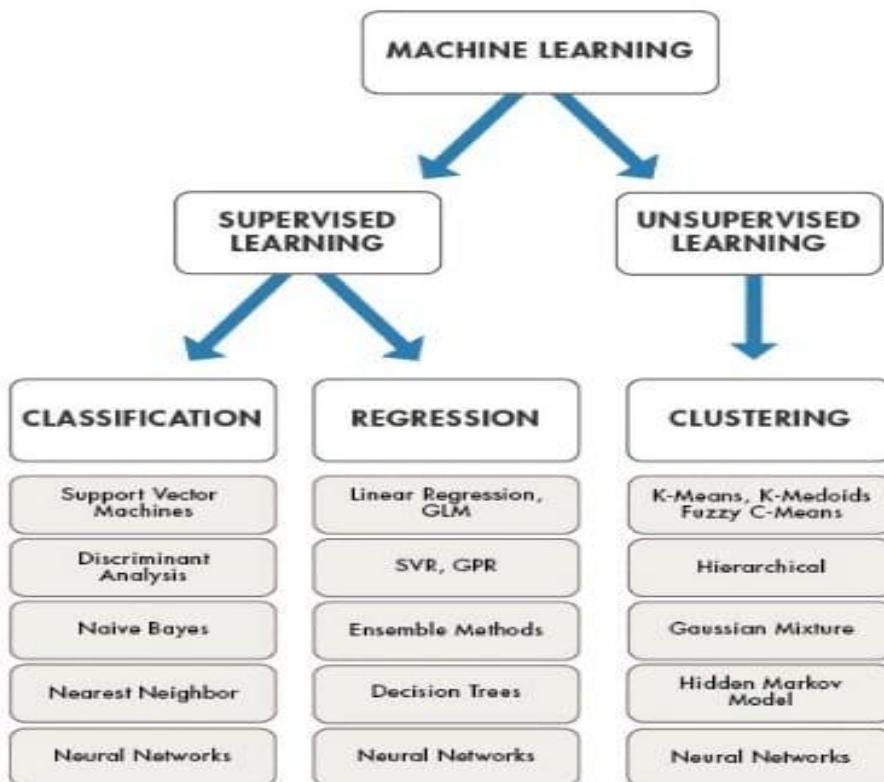


Figure 3.5: Type of machine learning

some guidelines on choosing between supervised and unsupervised machine learning: Choose supervised learning if you need to train a model to make a prediction—for example, the future value of a continuous variable, such as temperature or a stock price, or a classification—for example, identify makes of cars from webcam video footage.

Choose unsupervised learning if you need to explore your data and want to train a model to find a good internal representation, such as splitting data up into clusters

### **3.1.3 Deep Learning**

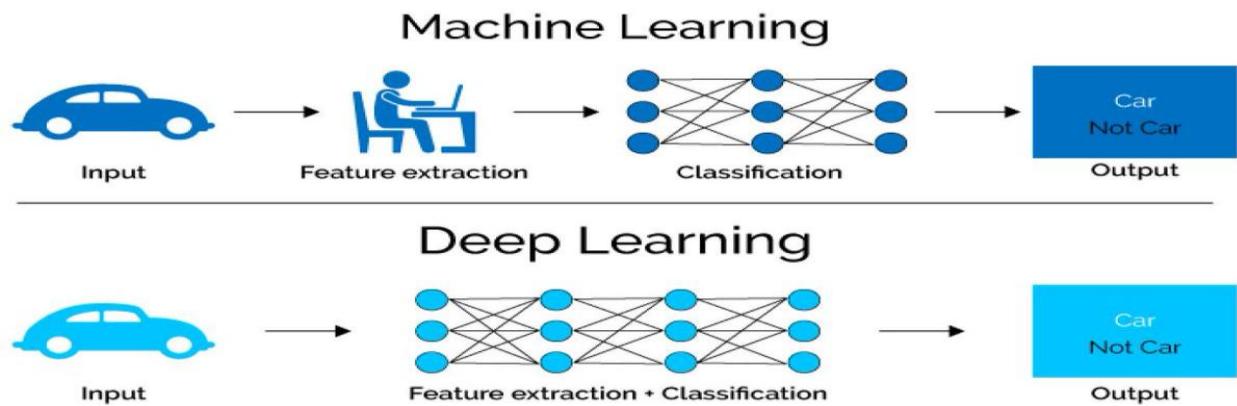


Figure 3.6: Describe Deep Learning

Deep Learning is basically a sub-part of the broader family of Machine Learning which makes use of Neural Networks (like the neurons working in our brain) to mimic human brain-like behavior.

DL algorithms focus on information processing patterns mechanism to possibly identify the patterns just like our human brain does and classify the information accordingly.

DL works on larger sets of data when compared to ML and the prediction mechanism is self-administered by machines.

#### **3.1.3.1 Types of fundamental network architectures:**

##### **1- Artificial Neural Network**

We can define neural networks as a software mathematical attempt to simulate the modus operandi of the human brain. Scientists have almost discovered how the human brain works in terms of learning ability, recollection, ability to distinguish objects and decision-making ability. The brain, as you know, consists of billions of neurons intertwined in a very complex way through the neural appendages of each cell, forming an enormous network of neurons associated with each other through these appendages.

This interconnection of neurons gives them the ability to store information, images, sound, and other signals they reach across the five senses, and thus also to learn through repetition and error.

There are two types in (ANN):

### **1- Forward propagation neural networks**

Are one of the most famous types of neural networks, and they cause information to move across layers in a single direction from the input layer to the hidden layer to the final layer.

### **2- Back propagation neural networks**

The back propagation phase of errors is where the network recalculates the error value in each of the hidden networks. Finally, the weight update phase comes as the network recalculates all weights and compensates them with new calculated values.

The back propagation is required to be derivable. This is because in the weight update phase, the derived function of the activation function is used to calculate new values.

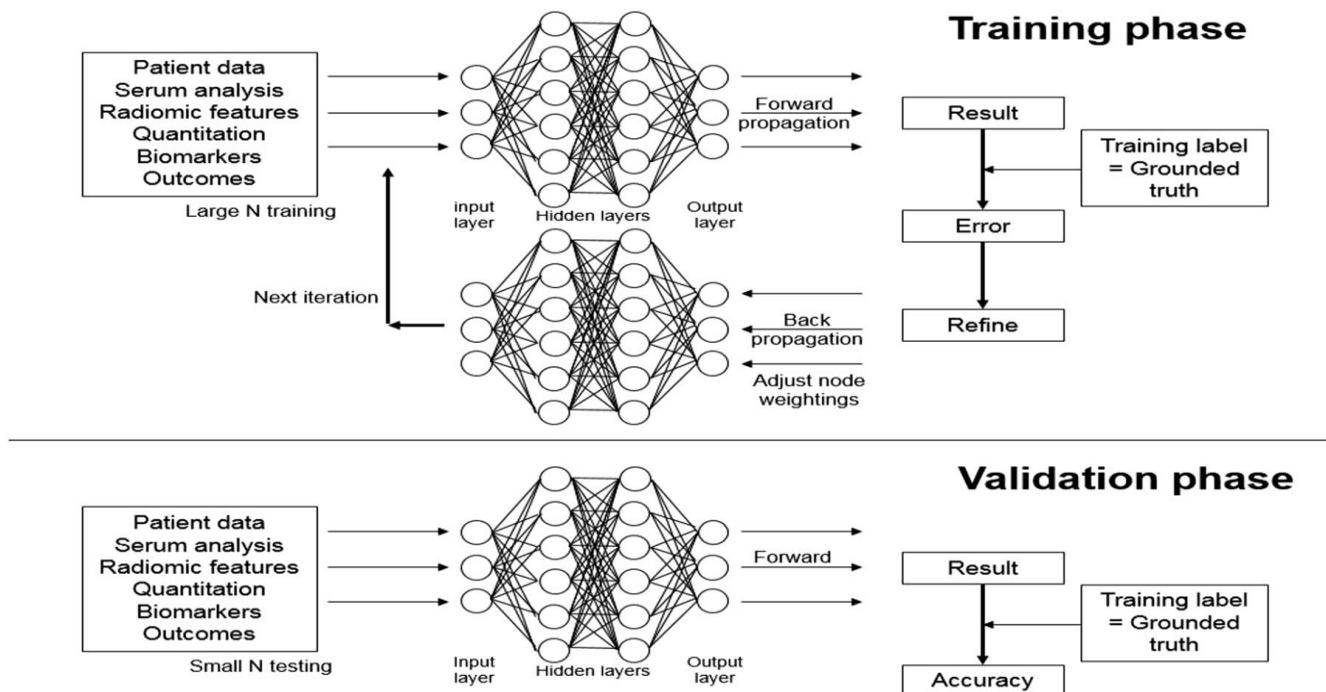


Figure 3.7: Forward propagation and Back propagation

## 2- Recurrent Neural Network (RNN)

RNN is widely used neural network architecture for NLP. It has proven to be comparatively accurate and efficient for building language models and in tasks of speech recognition.

RNNs are particularly useful if the prediction must be at word-level, for instance, Named-entity recognition (NER) or Part of Speech (POS) tagging. As it stores the information for current feature as well enables the model to predict the current output conditioned on long distance features. Below is an example of NER using RNN .

We also have forward and back propagation in RNN.

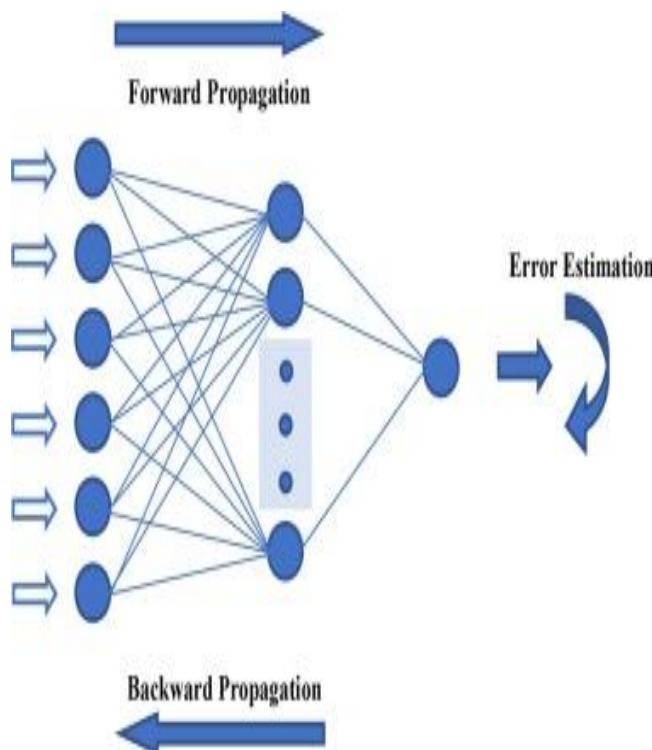


Figure 3.8: Forward and Backward in RNN

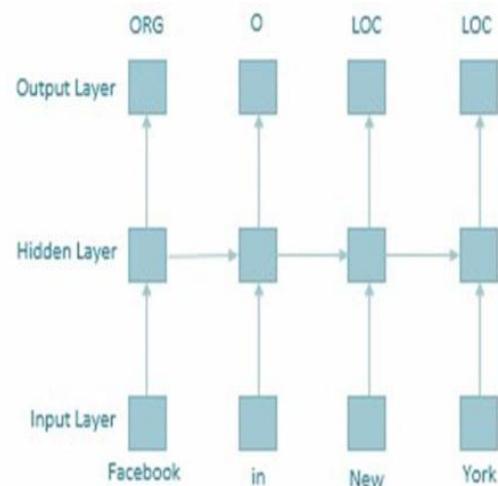


Figure 3.9: RNN

### 3- convolutional Neural Network (CNN)

In deep learning, a convolutional neural network (CNN/ConvNet) is a class of deep neural networks, most applied to analyze visual imagery. Now when we think of a neural network, we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution. Now in mathematics convolution is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other.

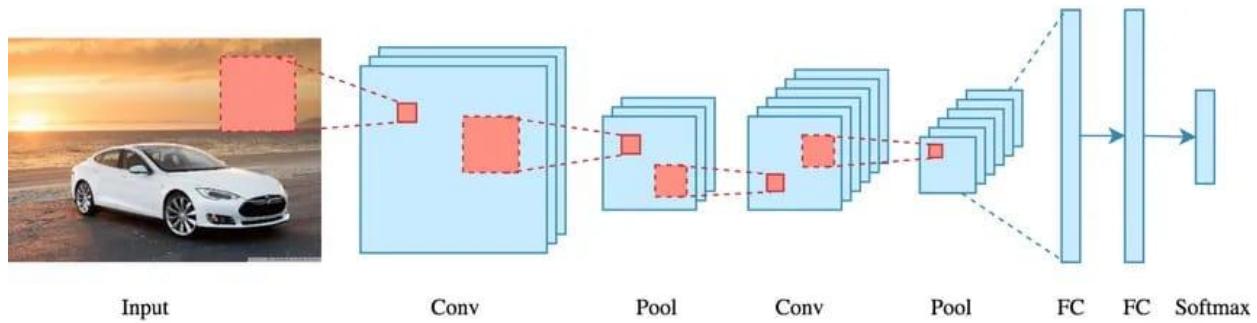


Figure 3.10: CNN from input to SoftMax

Before we go to the working of CNN's let's cover the basics such as what is an image and how is it represented. An RGB image is nothing but a matrix of pixel values having three planes whereas a grayscale image is the same, but it has a single plane. Look at this image to understand more.

3 Colour Channels			
4	6	1	3
9	7	3	2
0	35	19	25
26	13	22	16
15	4	3	7
1	8	10	53
8	0	8	1
0	3	1	3

Figure 3.11: RGB Matrix

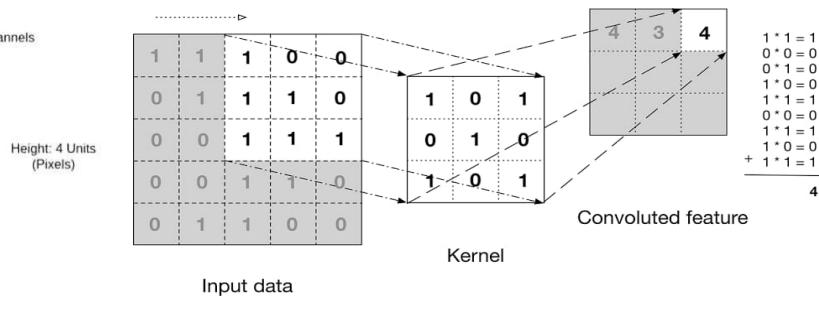


Figure 3.12: Calculate convolution matrix

The above image shows what a convolution is. We take a filter/kernel ( $3 \times 3$  matrix) and apply it to the input image to get the convolved feature. This convolved feature is passed on to the next layer

### **3.1.4 Large Language Models (LLMs)**

Large Language Models (LLMs) are a class of machine learning models designed to understand, generate, and interact with natural language text. They are built on neural network architectures, particularly transformer-based models like GPT, BERT, and their derivatives. These models are trained on vast datasets of text to learn statistical patterns and relationships in language, enabling them to perform a wide range of natural language processing (NLP) tasks. The growing computational power and availability of large datasets have significantly contributed to their rapid advancements in recent years.

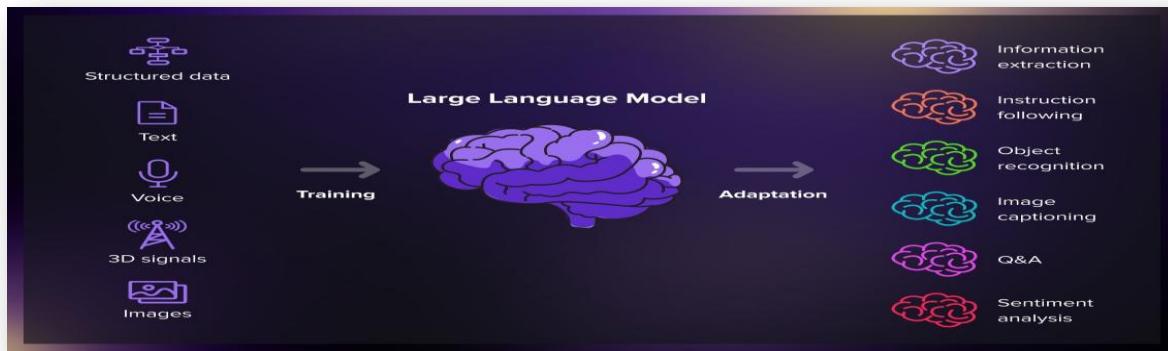


Figure 3.13: Large Language Model

#### **3.1.4.1 How Large Language Models Work**

LLMs operate on a principle like traditional NLP pipelines but in a more unified and end-to-end fashion. Instead of using discrete, specialized components for tasks like tokenization, parsing, or classification, LLMs learn to encode linguistic knowledge within their parameters. This is achieved through pretraining on large-scale corpora using objectives such as:

- Masked Language Modeling (MLM): Predicting missing words in a sentence (e.g., used in BERT).
- Causal Language Modeling (CLM): Predicting the next word in a sequence (e.g., used in GPT).

Once pretrained, these models can be fine-tuned for specific tasks or domains using smaller, task-specific datasets. This adaptability has made them a cornerstone of modern AI research and applications.

### **3.1.4.2 Applications of Large Language Models**

LLMs have transformed the landscape of NLP by enabling sophisticated solutions for a variety of use cases, such as:

**Chatbots and Conversational AI:** LLMs like ChatGPT power virtual assistants capable of maintaining contextual and meaningful conversations, offering customer support, or assisting with daily tasks.

**Content Generation:** They can create high-quality text content, including articles, stories, and summaries, often indistinguishable from human-written text.

**Sentiment Analysis and Opinion Mining:** Businesses use LLMs to extract insights from customer feedback or social media, enabling better decision-making.

**Programming Assistance:** Models like Codex help developers by generating, explaining, and debugging code in various programming languages.

**Education and Training:** LLMs provide personalized tutoring and can help students understand complex topics through dialogue-based interaction.

### **3.1.4.3 Tools and Frameworks for Working with Large Language Models**

Several frameworks and libraries are commonly used to work with LLMs:

**Hugging Face Transformers:** Hugging Face provides one of the most comprehensive libraries for deploying and fine-tuning transformer-based models. It includes prebuilt models like GPT-3, BERT, and T5, along with tools for tokenization and model optimization.

**OpenAI API:** OpenAI offers access to state-of-the-art LLMs like GPT-4 through an API, making it simple to integrate advanced language understanding and generation capabilities into applications.

**LangChain:** LangChain specializes in building applications around LLMs by allowing seamless integration of external tools like databases and APIs, thus extending their functionality.

**Rasa:** While traditionally focused on rule-based dialogue systems, Rasa increasingly supports integration with LLMs for more dynamic and responsive conversational agents.

#### **3.1.4.4 Challenges and Considerations**

While LLMs have opened new possibilities, they also present unique challenges:

- **Resource Intensity:** Training and deploying LLMs require substantial computational resources, making them less accessible to smaller organizations.
- **Bias and Fairness:** LLMs often reflect biases present in their training data, necessitating careful evaluation and mitigation strategies.
- **Interpretability:** Understanding how LLMs arrive at specific outputs remains an ongoing research area, limiting their transparency.
- **Ethical Concerns:** Misuse of LLMs, such as generating misinformation or malicious content, has raised important questions about regulation and governance.

#### **3.1.5 Natural Language Processing (NLP)**

Natural Language Processing (NLP) is a branch of Artificial Intelligence focused on enabling communication between humans and computers using natural language. This field encompasses not only text-based systems but also speech-based technologies. While researchers have been studying NLP for decades, recent advancements in computational techniques and algorithms have led to its resurgence as a highly relevant area of study.

### **3.1.5.1 How Natural Language Processing Works**

In computer science, complex problems are often made more manageable by breaking them into smaller, simpler components. This approach is particularly effective in Artificial Intelligence. For NLP tasks, various specialized components are created, each designed to solve a specific problem. These components are arranged in a sequence known as a pipeline. Input data passes through each component in the pipeline, ultimately producing the desired output.

### **3.1.5.2 Applications of Natural Language Processing**

NLP has a wide range of applications, many of which are integrated into everyday software and systems. Some examples include:

- **Virtual Assistants:** Modern personal assistants, such as those on smartphones and smart speakers, leverage advanced NLP techniques to help users' complete tasks efficiently.
- **Machine Translation:** Language translation tools have significantly improved in accuracy, making it easier for users to communicate across languages seamlessly.
- **Speech-to-Text:** As society moves at a fast pace, speech-to-text systems are invaluable for transcribing business notes, phone calls, or speeches.

### **3.1.5.3 Tools for Natural Language Processing**

Several tools and libraries are available for implementing NLP solutions. Each has unique features and benefits, making them suitable for different use cases.

#### **1- NLTK**

The Natural Language Toolkit (NLTK) is one of the most comprehensive NLP libraries. It offers functionalities such as tokenization, classification, stemming, parsing, and semantic reasoning. While its extensive documentation and versatility make it ideal for experimentation, its reliance on string-based data representation and slower performance may pose challenges for advanced applications.

## **2- Text Blob**

Text Blob simplifies the use of NLP techniques by extending NLTK with a more user-friendly interface and additional features from the Pattern library. It is particularly suitable for beginners and smaller projects, offering easy access to many NLP functionalities without requiring deep technical expertise.

## **3- PyTorch-NLP**

PyTorch-NLP is a rapidly evolving library that excels in prototyping and integrating cutting-edge research. Its frequent updates and compatibility with modern algorithms make it a popular choice among researchers and developers for building and testing innovative solutions.

## **4- SpaCy**

SpaCy is a robust competitor to NLTK, offering faster performance and a more streamlined interface. Unlike NLTK, it represents data as objects, making it easier to integrate with other tools and frameworks. While SpaCy supports fewer languages, it is highly suitable for applications requiring efficient processing and scalability.

### **3.1.6 Flutter**



Figure 3.14: Flutter

Flutter is an open-source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase.

It's flexible, which you can control every pixel to create customized, adaptive designs that look and feel great on any screen.

Multi-Platform, reach users on every screen, you can deploy to multiple devices from a single codebase: mobile, web, desktop, and embedded devices.

You can take control of your codebase with automated testing, developer tooling, and everything else you need to build production-quality apps.

Stable and Reliable, trusted by many, Flutter is supported and used by Google, trusted by well-known brands around the world, and maintained by a community of global developers.

Flutter is powered by Dart, a language optimized for fast apps on any platform.

Flutter code compiles to ARM or Intel machine code as well as JavaScript, for fast performance on any device.

Flutter is so productive, which you can build and iterate quickly with Hot Reload. Update code and see changes almost instantly, without losing state.

We think Flutter will help create beautiful, fast apps, with a productive, extensible, and open development model, whether you're targeting iOS or Android, web, Windows, macOS, Linux or embedding it as the UI toolkit for a platform of your choice

## 3.2 Proposed Model

The proposed system is an integrated platform designed to provide users with an efficient and interactive way to detect anemia and gain more information about the condition. It uses image analysis powered by machine learning models and a chatbot for guidance. Below is an updated detailed description of the system:

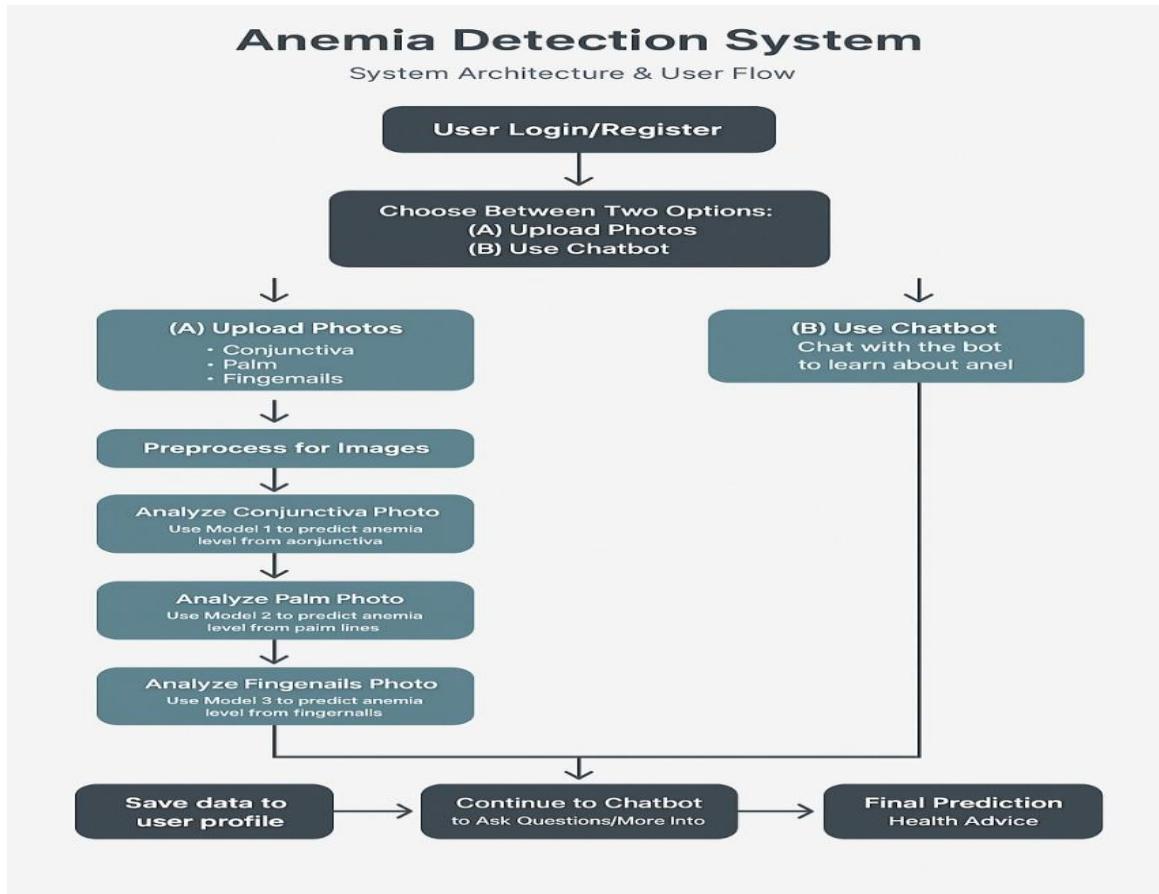


Figure 3.15: updated detailed description of the system

### 1. User Login/Register

Users must securely log in or register for personalized access. This step ensures user data is securely stored and managed.

## **2. Choose Between Two Options**

Once logged in, users can select one of two main functionalities:

*Option A:* Upload Photos for Anemia Detection.

*Option B:* Use Chatbot to learn about anemia and ask questions.

### **3. Option A: Upload Photos for Anemia Detection**

Users opting for this pathway are prompted to upload three types of images:

- Conjunctiva: Close-up photo of the inner eyelid.
- Palm: Photo of the palm of the hand.
- Fingernails: Photo of fingernails.

#### Preprocessing of Uploaded Images

Before analysis, the uploaded images are preprocessed to ensure compatibility and accuracy with the models.

### **4. Analyze Photos with Machine Learning Models**

Each image type is analyzed using a dedicated machine learning model:

Step 1: Analyze Conjunctiva Photo

Model 1: Predicts anemia based on conjunctiva and provides a confidence score.

Step 2: Analyze Palm Photo

Model 2: Predicts anemia based on the palm and provides a confidence score.

Step 3: Analyze Fingernails Photo

Model 3: Predicts anemia based on fingernails and provides a confidence score.

## **5. Combine Predictions and Decision Fusion**

The system combines the predictions from the three models to provide a final classification and recommendation:

Scenarios and Recommendations:

### **1- All Three Models Predict Anemia:**

- Recommendation: The user is highly likely to have anemia and must consult a doctor immediately for further evaluation and treatment.

### **2- Two Models Predict Anemia:**

- Recommendation: The user is likely to have anemia and should consult a doctor soon for a professional assessment and advice.

### **3- One Model Predicts Anemia:**

- Recommendation: The user has a low risk of anemia but is advised to consult a doctor for further screening if symptoms persist

### **4- No Model Predicts Anemia:**

-Recommendation: The user is in the safe zone and does not appear to have anemia. However, maintaining a healthy lifestyle and regular checkups is advised.

## **6. Final Prediction and Health Advice**

The system provides a final prediction of anemia based on the combined model outputs. Users receive health advice, including:

- Dietary recommendations.
- Medical consultation suggestions.
- Lifestyle tips to manage or prevent anemia.

## **7. Redirect to Chatbot**

Users are redirected to the chatbot after receiving their results. The chatbot allows users to:

- Clarify doubts.
- Learn more about anemia.
- Ask additional questions.

## **8. Option B: Direct Interaction with the Chatbot**

Users choosing this option directly interact with the chatbot to:

- Learn about anemia symptoms, causes, and prevention.
- Receive tips for managing anemia effectively.

## **9. Encourage Continued Chatbot Interaction**

Regardless of the chosen pathway (A or B), the system encourages ongoing interaction with the chatbot for continued learning, support, and personalized advice about anemia.

# CHAPTER

# FOUR

# Chapter 4: Planning and Analysis

---

## 4.1 Planning

Effective planning is essential for the successful execution of the anemia detection system project. This section outlines the feasibility study, ensuring the project is cost-effective and accessible for users, and provides a project timeline using a Gantt chart tailored to the team's workflow.

### 4.1.1 Feasibility Study and Estimated Cost

The feasibility study evaluates the technical, operational, and financial aspects of the proposed system to ensure its practicality and success.

#### Technical Feasibility:

- **Prototype Development:** Completed using KNN, RF, and CNN models for analyzing conjunctiva, fingernails, and palm images.
- **UI/UX Design:** Developed using Flutter for mobile applications and Tkinter for GUI prototyping.
- **Transfer Learning:** Upcoming integration of advanced pre-trained models like ResNet to enhance accuracy.
- **API and Deployment:** Planned use of APIs to integrate the models with a Flutter-based app and build LLM-powered features.

#### Operational Feasibility:

Each team member is assigned specific roles to ensure a balanced workload and effective collaboration:

- i. **Data Collection and Preprocessing:** Completed by 2 team members.
- ii. **ML Model Development:** Completed by 3 team members.
- iii. **Flutter and GUI Design:** Completed by 2 team member.
- iv. **Transfer Learning, and Deployment:** To be managed by the entire team in the upcoming phase.

The system is designed to be user-friendly, enabling non-technical users to interact easily with the app.

### **Financial Feasibility:**

**Development Costs:** The system is designed to be cost-effective and free for users. The primary requirement is access to a smartphone with a functional camera. Development costs are minimized by leveraging open-source tools and **team expertise:**

- Machine learning models and algorithms were developed using free and open-source platforms (e.g., TensorFlow, PyTorch).
- Flutter was used for creating mobile apps and GUI.
- **User Accessibility:** The system requires no additional hardware apart from a smartphone, ensuring affordability for end-users.

#### **4.1.2 Gantt Chart**

The project timeline spans from October 2024 to May 2025, covering all development, testing, and deployment phases. The work is distributed among six team members for efficient execution.

	Oct 2024	Nov 2024	Dec 2024	Jan 2025	Feb 2025	Mar 2025	Apr 2025	May 2025
Data Collection & Preprocessing								
Documentation and Refinement								
ML Model Development								
Prototype Completion								
Transfer Learning Implementation								
LLM Integration								
API Development & Setup								
Flutter App & Deployment								
System Testing & Validation								
Final Deployment & Review								

## **4.2 Analysis and Limitations of Existing System**

Current anemia detection methods rely heavily on blood tests. While accurate, these methods are invasive, costly, and require specialized medical facilities. Additionally, such tests are often inaccessible in remote or underprivileged areas where healthcare resources are limited.

Non-invasive methods, such as visual examination for pallor or the use of basic portable devices, do not provide the diagnostic precision required for reliable results. Alternatively, machine learning-based approaches to anemia detection through smartphone images present a viable solution. However, the existing systems face several critical limitations:

- Data Collection: A lack of large and diverse datasets for training machine learning models limits their generalizability and robustness.
- Algorithm Accuracy: Current models often fail to provide reliable diagnoses in real-world scenarios, especially in heterogeneous populations.
- User Accessibility: Many systems are not optimized for low-resource settings, hindering their widespread adoption.

These limitations emphasize the need for a more robust, inclusive, and scalable system to address these challenges effectively.

## **4.3 Need for New System**

Anemia remains a significant global health challenge, particularly among vulnerable populations such as children and pregnant women. While existing diagnostic methods are effective, their reliance on invasive and costly blood tests makes them impractical in regions with limited healthcare infrastructure. In many developing countries, inadequate medical facilities and the high cost of regular blood tests exacerbate the issue.

Advancements in technology, particularly the proliferation of smartphones and breakthroughs in machine learning, offer an unprecedented opportunity to address this challenge. By leveraging Large Language Models (LLMs) alongside machine learning techniques, such as Convolutional Neural Networks (CNNs), a new system can be developed to:

- Analyze images of eyes, palms, and fingernails captured via smartphones.
- Provide non-invasive, affordable, and accessible anemia detection.
- Offer actionable insights to users in a user-friendly manner, aided by LLM-powered explanations and recommendations.

### **Integration of LLMs:**

- Improving User Interaction: LLMs can serve as conversational agents, guiding users through the image capture process and providing real-time feedback.
- Data Augmentation: LLMs can aid in curating synthetic datasets by generating textual annotations or processing metadata.
- Diagnostic Support: By synthesizing multiple data sources, LLMs can generate comprehensive diagnostic reports, improving interpretability for end-users and healthcare providers.

The proposed system seeks to democratize anemia detection by delivering accessible, cost-effective, and reliable diagnostics. Such a solution has the potential to enable early detection, improve treatment outcomes, and significantly impact underserved communities.

---

## **4.4 Analysis of New System**

The new system represents a significant advancement in non-invasive anemia detection. It leverages machine learning models to analyze images captured via smartphones, focusing on visual indicators of anemia in the eyes, palms, and fingernails. By incorporating advanced AI techniques, such as CNNs and LLMs,

the system aims to simplify the diagnostic process, making it more accessible and cost-effective, particularly in underserved communities.

#### **4.4.1 User Requirements**

The system is designed for users with minimal technical expertise, including healthcare professionals and patients. Key requirements include:

- Ease of Use: An intuitive interface that allows users to upload images and receive results quickly. Integration with LLMs can guide users through the process.
- Accuracy: Reliable anemia diagnoses based on image analysis, validated against medical standards.
- Accessibility: Available on both Android and iOS devices, with multilingual support for diverse populations.

#### **4.4.2 System Requirements**

The system must meet the following technical requirements:

- Mobile Compatibility: Fully functional on Android and iOS devices, with adaptive UI for varying screen sizes.
- Machine Learning Integration: Real-time image analysis using pre-trained and fine-tuned CNN models.
- LLM Integration: Provides conversational assistance for image capture, result interpretation, and follow-up recommendations.
- Cloud Storage: Secure storage of user data, with encryption and compliance with healthcare data regulations (e.g., HIPAA).

#### **4.4.3 Domain Requirements**

The system aligns with the specific needs of the healthcare domain, emphasizing:

- Compliance: Adhering to medical standards for non-invasive diagnostics, such as FDA or equivalent certifications.
- Privacy: Ensuring patient data security and compliance with healthcare data protection laws (e.g., HIPAA, GDPR).

- Interoperability: Seamless integration with electronic health record (EHR) systems for efficient data sharing.

#### **4.4.4 Functional Requirements**

1. User Registration and Authentication: Secure user accounts with features like two-factor authentication.
2. Image Upload and Processing: Guided image capture and preprocessing for consistent quality.
3. Anemia Detection and Analysis: High-accuracy predictions using CNNs, with results explained via LLMs.
4. Results Interpretation and Suggestions: Clear, actionable feedback, including dietary advice and follow-up prompts.
5. Notifications and Alerts: AI-driven alerts for health trends or follow-ups.
6. Integration with External Healthcare Systems: Real-time data sharing with healthcare providers.

#### **4.4.5 Non-Functional Requirements**

Non-functional requirements define the system's performance attributes, scalability, security, and quality. Key NFRs include:

- Scalability: Horizontal and vertical scalability with seamless cloud integration.
- Security: End-to-end encryption, robust authentication mechanisms, and compliance with HIPAA and GDPR.
- Performance: Low latency and high throughput for real-time image processing.
- Reliability: Fault tolerance, regular backups, and 99.9% uptime.
- Maintainability: Modular architecture, automated testing, and comprehensive documentation.
- Usability: Intuitive UI/UX, accessibility features, and multilingual support.

- Interoperability: Adherence to HL7 and FHIR standards for healthcare system compatibility.

## **4.5 Advantages of New System**

The proposed anemia detection system introduces significant advancements over traditional diagnostic methods and existing alternatives, addressing key limitations in cost, accessibility, and efficiency. Below are the core benefits offered by the system:

1. Non-Invasive Testing The system eliminates the need for invasive blood draws, offering a pain-free alternative for anemia detection. This is particularly beneficial for populations sensitive to blood tests, such as children and individuals in remote areas with limited access to healthcare facilities.
2. Cost-Effectiveness By utilizing readily available smartphone cameras and machine learning algorithms, the system dramatically reduces the cost of anemia diagnosis. This affordability makes it accessible to a broader audience, particularly in low-resource settings where traditional diagnostic methods are prohibitively expensive.
3. Wide Accessibility With the increasing ubiquity of smartphones, the system provides a highly accessible diagnostic tool. Users in underserved or rural areas can easily leverage the system to monitor their health, bypassing the need for expensive medical equipment or facility visits.
4. Rapid and Timely Results Traditional blood tests often require several days for processing, causing delays in treatment. In contrast, this system delivers results within minutes of image upload, enabling users to make informed health decisions quickly and reducing the window for disease progression.
5. User-Friendly Design Designed with simplicity in mind, the system features intuitive interfaces that cater to users with minimal technical expertise. Capturing and uploading images, receiving results, and tracking health status can be accomplished with just a few taps, ensuring inclusivity across all age groups and technological proficiencies.

6. Real-Time Monitoring and Personalized Tracking The system allows users to monitor their anemia status over time, offering insights into trends and early detection of potential health issues. Continuous tracking empowers users to observe the effects of lifestyle changes, medications, or other interventions on their anemia status, fostering proactive health management.
7. Scalability for Public Health Impact Built to scale, the system can accommodate large user bases, making it ideal for public health initiatives targeting anemia in rural or resource-poor communities. It enables widespread deployment to monitor anemia cases at a community level, contributing to enhanced public health outcomes.
8. Privacy and Security Recognizing the sensitivity of health data, the system prioritizes robust data protection measures. Personal and medical information is encrypted and stored securely, with strict adherence to global privacy regulations such as HIPAA and GDPR. Users retain full control over their data, ensuring trust and transparency.
9. Seamless Integration with Healthcare Systems The system supports integration with existing healthcare infrastructure, enabling users to share diagnostic results directly with healthcare providers. This improves the continuity of care, as medical professionals can access timely and accurate data to inform clinical decisions and treatment plans.
10. Advancing Machine Learning Models Continuous user interaction enhances the machine learning model's accuracy and adaptability. As the system collects more data, it refines its diagnostic capabilities, opening avenues for detecting related conditions and expanding its scope in healthcare. This iterative improvement ensures the system remains at the forefront of technological innovation.

## 4.6 User Characteristics

Understanding the target users of the system is crucial for effective design:

### 1. Demographics

- Age: Designed for both children (with supervision) and adults, focusing on vulnerable groups like pregnant women.
- Gender: Equally beneficial for all genders.
- Geographic Location: Accessible globally, with special emphasis on rural and resource-poor areas.

### 2. Health Consciousness

- Targeted at individuals who prioritize health monitoring and early detection.

### 3. Technological Literacy

- Designed for users with basic smartphone skills, with LLM-powered guidance for enhanced usability.

### 4. Access to Technology

- Optimized for smartphones, including low-end devices, with offline capabilities for remote areas.

### 5. Health Condition Awareness

- Provides educational resources powered by LLMs for users with varying awareness levels.

### 6. User Motivation

- Motivated by early detection, affordability, and convenience.

### 7. Privacy Concerns

- Transparent data handling with user-controlled privacy options.

### 8. Healthcare Providers

- Supports data sharing and integration with EHR systems for improved clinical decisions.

# CHAPTER

# FIVE

## **Chapter 5: DESIGN**

---

### **5.1 Design and Implementation Constraints**

1. The application must require an internet connection to upload user images to the server for analysis and to retrieve diagnostic results.
2. The system shall undergo rigorous testing processes, including Unit Testing, Integration Testing, System Testing, and Acceptance Testing, to ensure reliability and accuracy in real-world scenarios.
3. Each user must ensure that their account credentials, including passwords, remain confidential. Passwords will be encrypted and stored securely in compliance with data protection regulations.
4. Each user must provide essential demographic details, such as age, gender, and medical history, during registration to enable personalized and accurate analysis.
5. Only authorized Administrators shall have the ability to manage system permissions, including granting or revoking access for healthcare providers and other specialized users.

### **5.2 Assumptions and Dependencies**

#### **5.2.1 Assumptions**

These are the key assumptions made during the design, development, and deployment of the system:

##### **User Assumptions:**

- i. Users possess smartphones with high-quality cameras capable of capturing clear and well-lit images of the conjunctiva, palms, and fingernails.

- ii. Users are capable of following instructions for capturing images under suitable lighting conditions to ensure diagnostic accuracy.
- iii. Users understand that the system provides diagnostic support and is not a replacement for professional medical advice.

### **Data Assumptions:**

- i. The training dataset includes a wide range of images covering various skin tones, ages, genders, and health conditions to ensure generalizability.
- ii. Image preprocessing techniques (noise reduction, cropping, normalization) can adequately handle variations in lighting and background conditions.

### **Technological Assumptions:**

- i. The smartphone app will function on devices running recent versions of Android and iOS operating systems.
- ii. Internet connectivity is available for uploading images to the server for analysis and retrieving results.

### **Healthcare System Assumptions:**

- i. Medical professionals will validate the system's results and provide follow-up care as needed.
- ii. Regulatory approval for the system is obtainable, ensuring compliance with local and international medical device standards.

## **Ethical Assumptions:**

- i. Users will provide informed consent for using their images and data for anemia detection.
- ii. Data collected by the system will be handled responsibly, with no unauthorized access or misuse.

### **4.2.2 Dependencies**

The successful implementation and operation of the anemia detection system depend on several external factors and resources:

#### **Data Dependencies:**

- i. Access to large, high-quality datasets of labeled images (conjunctiva, palms, fingernails)

#### **Technological Dependencies:**

- i. Reliable machine learning frameworks and libraries to build and deploy AI models.
- ii. Cloud-based infrastructure for image processing, model hosting, and data storage to ensure scalability and performance.
- iii. APIs and tools to integrate the app with healthcare platforms for sharing results with professionals.

#### **Hardware and Software Dependencies:**

- i. Dependence on smartphones equipped with high-resolution cameras and capable of running the app efficiently.
- ii. Compatibility with a range of Android and iOS devices to maximize user accessibility.

## **Environmental Dependencies:**

- i. Dependence on helpdesk or chatbot feature for addressing user concerns and providing guidance

## **5.3 Risks and Risk Management:**

We must ensure the anemia detection system is optimized for performance and usability while maintaining reliability across devices.

**Application Performance:** The system must be lightweight and efficient to run smoothly on low-end and mid-range mobile devices without compromising diagnostic accuracy.

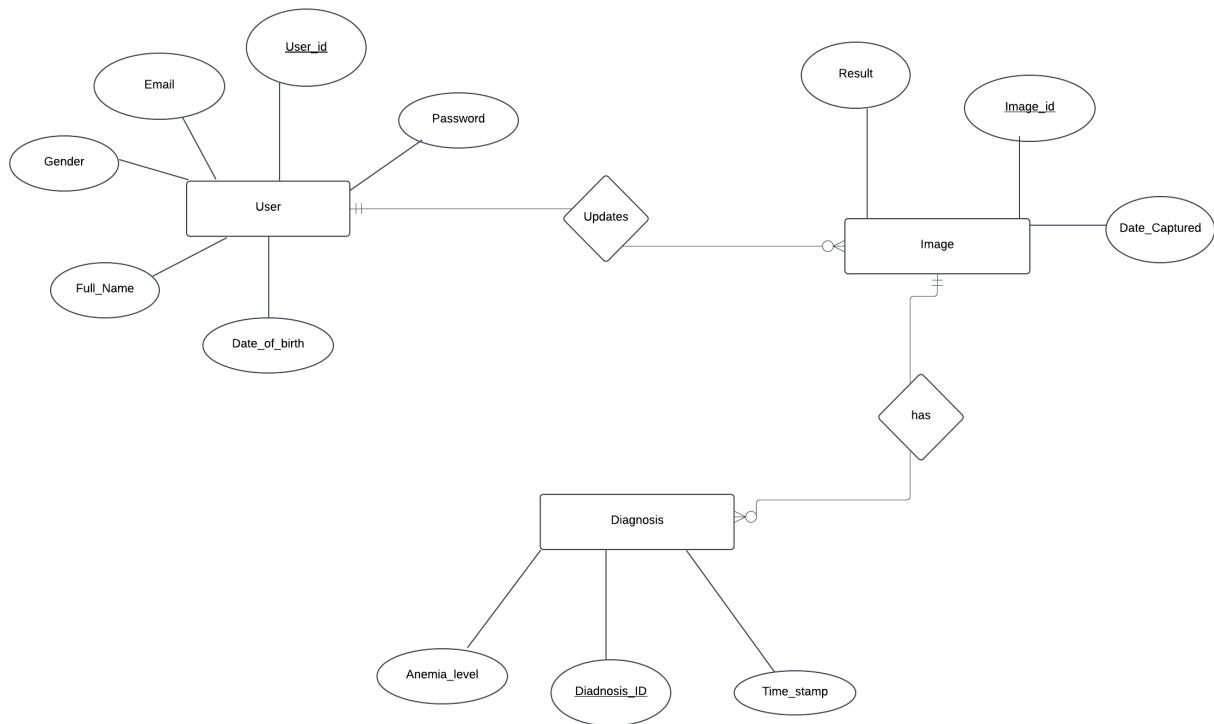
**Software Errors:** We need to ensure the application is reliable by conducting thorough testing during development to avoid functionality issues.

**Data Integrity:** The system must ensure that user data and inputs are accurate to provide correct diagnostic results. Methods such as image preprocessing and verification mechanisms should be used to enhance accuracy.

**Development Timeline:** The time required for developing, testing, and deploying the application must not be underestimated to meet deadlines without sacrificing quality.

## 5.4 Design of Database ERD:

### 5.4.1 Entity Relationship



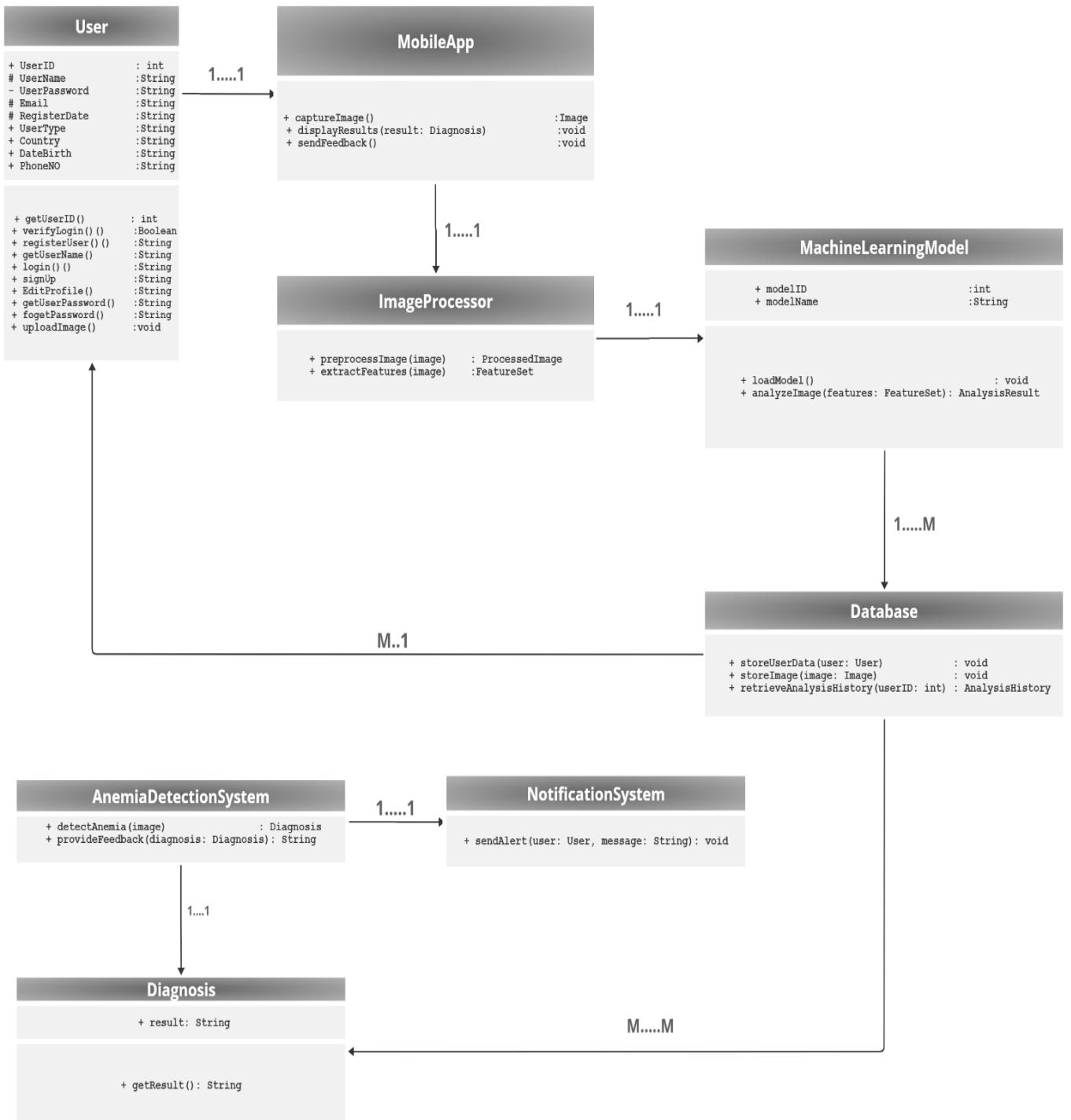
### 5.4.2 Mapping of Diagram:

User	
PK	<u>User_Id</u>
	Password
	Phone_Number
	Email
	Date_of_birth
	Full_name

Image	
PK	<u>Image_id</u>
FK	User_id
	.....
	Date_Captured
	Result

Diagnosis	
PK	<u>Diagnosis_id</u>
FK	image_id
	.....
	Anemia_level
	Timestamp

## 5.5 Class Diagram:

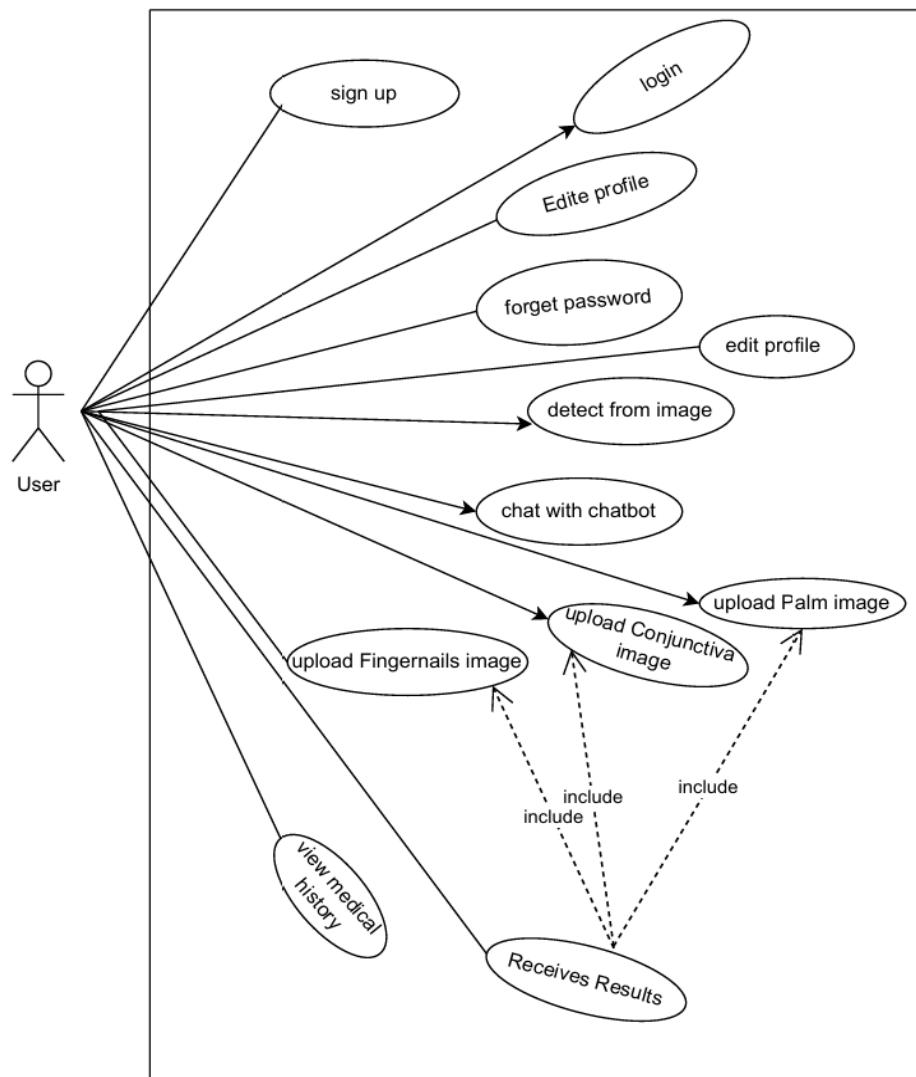


## 5.6 Use case Diagram

### 5.6.1 Primary

use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated. Use case diagrams are employed in UML, as standard notation for the modeling of real-world objects and systems.

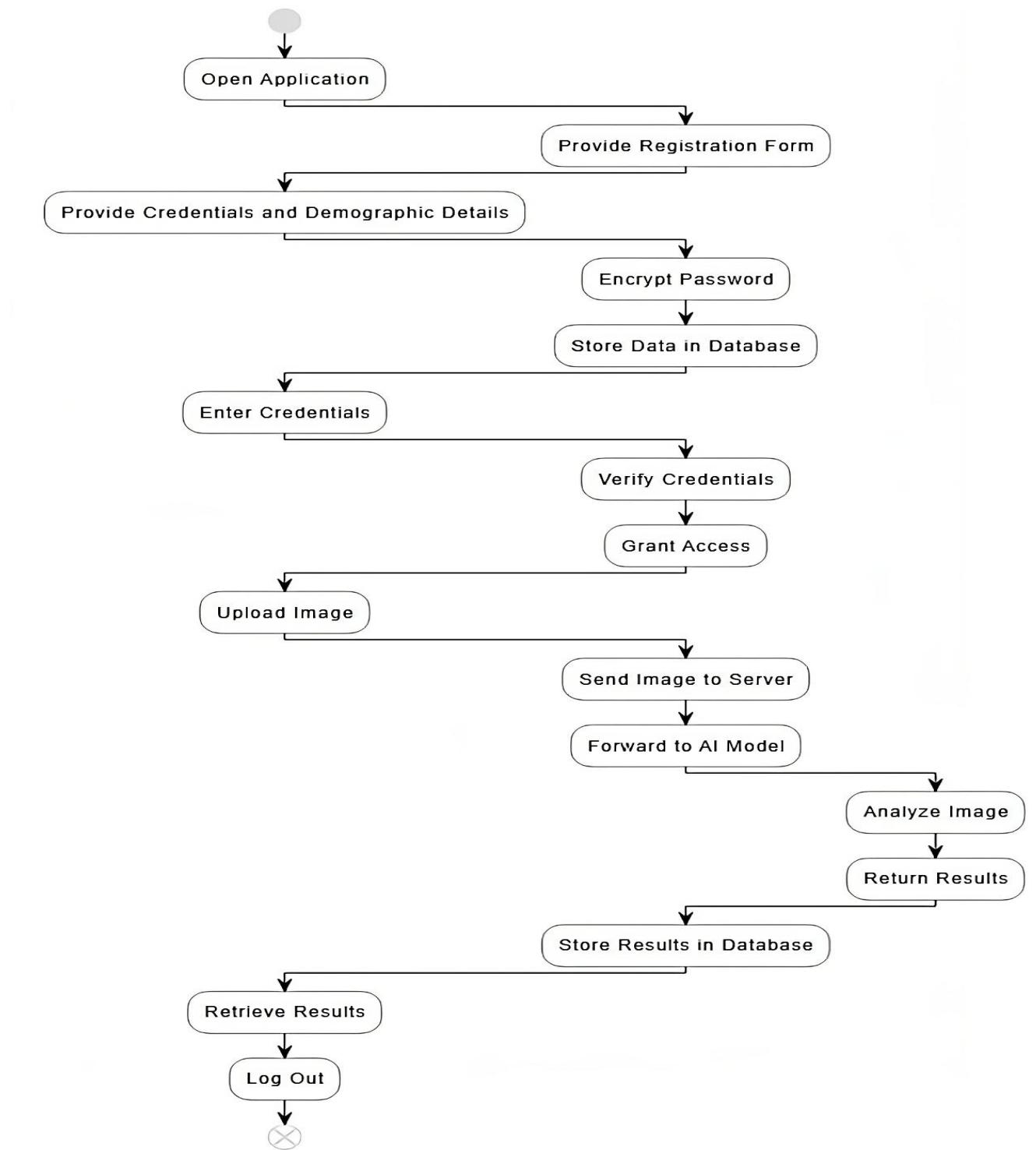
There are several benefits of having a case diagram over similar diagrams such as flowchart.



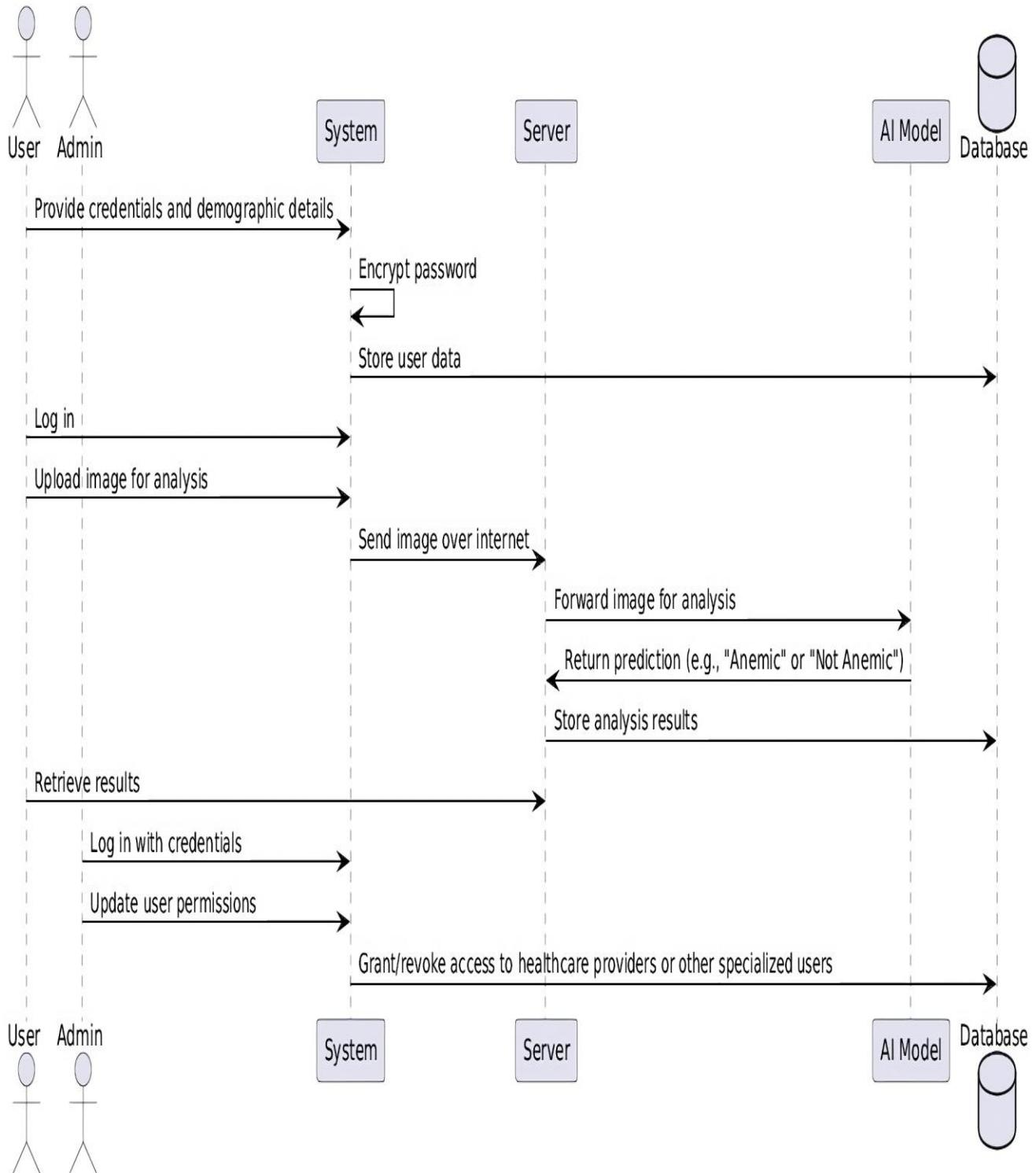
## **5.6.2 Use case Scenario**

<b>Use Case</b>	<b>Description</b>	<b>Purpose</b>
Login	Allows users to access their accounts using credentials.	Secure access to the system and personal data.
Sign Up	Enables new users to register by creating an account.	Facilitates first-time access to the system.
Edit Profile	Users can update their personal information.	Keep profiles up to date for accurate analysis.
Forget Password	Recover access if credentials are forgotten.	Improves usability by providing account recovery options.
Upload Fingernails Image	Upload images of fingernails for analysis.	Detect anemia signs based on nail appearance.
Upload Palm Image	Upload images of the palm for analysis.	Detect anemia signs from palm texture or paleness.
Upload Conjunctiva Image	Upload images of the conjunctiva for analysis.	Analyze inner eyelid color for anemia detection.
View Medical History	Access previous medical records and results.	Monitor health trends and follow-up needs.
Detect from Image	Analyze uploaded images using AI algorithms.	Core functionality to detect anemia non-invasively.
Chat with Chatbot	Ask questions or get guidance via a chatbot.	Provide instant support and improve user experience.
Receive Results	Receive diagnostic results after image analysis.	Deliver insights and recommendations for anemia detection.

## 5.7 Activity Diagram

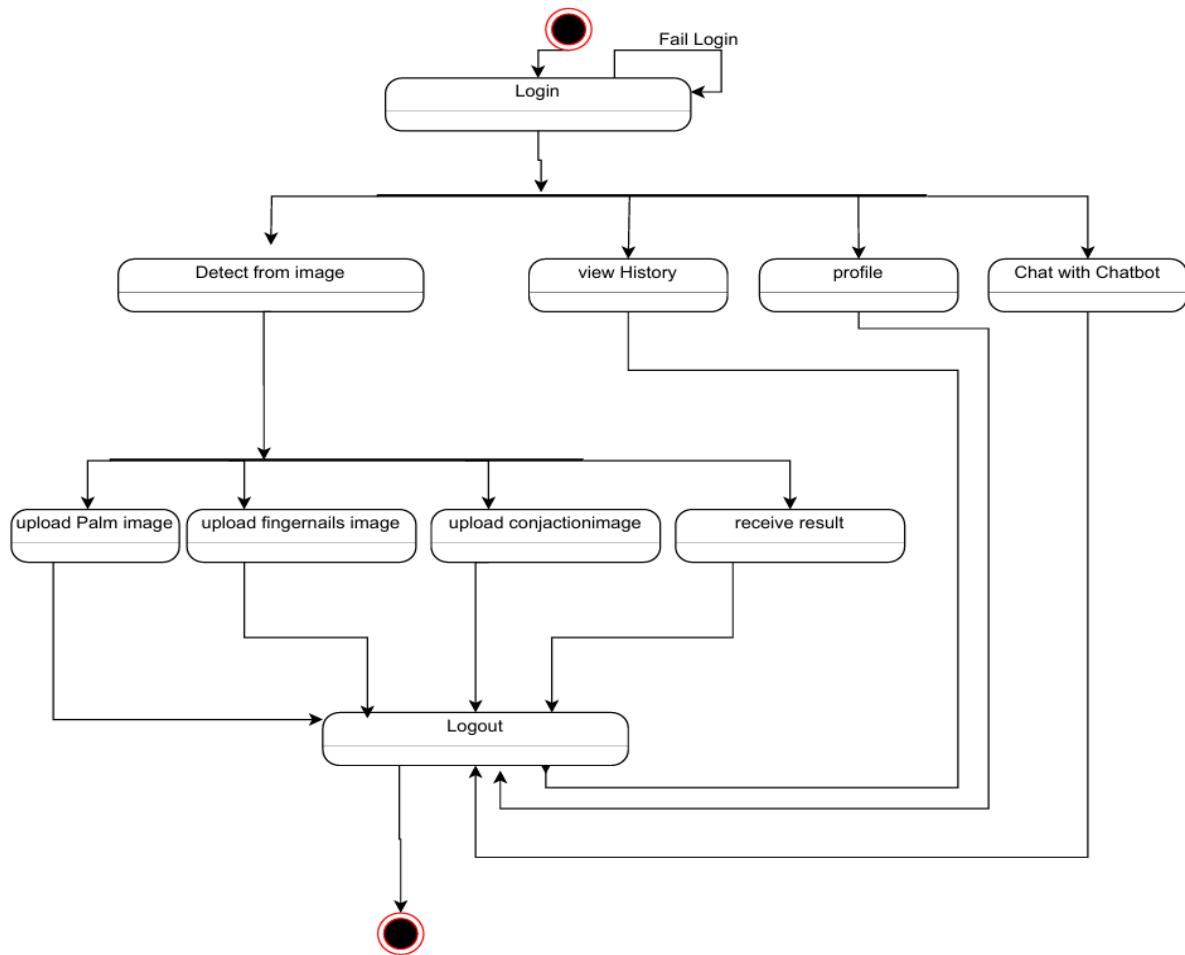


## 5.8 Sequence Diagram



## 5.9 State Diagram

A state diagram is used to represent the conditions of the system or part of the system at finite instances of time. It's a behavioral diagram and it represents the behavior using finite state transitions. State diagrams are also referred to as State Machines and State Chart Diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behavior of a class in response to time and changing external stimuli. We can say that each class has a state, but we don't model every class using State Diagrams. We prefer to model the states with three or more states.



# CHAPTER

# SIX

# Chapter 6: Implementation and Testing

---

## 6.1 Data set

### **6.1.1 Classification Datasets (Nails, Palm, Conjunctiva)**

In this section, we detail the classification datasets used to detect anemia based on visual biomarkers captured from three regions: fingernails, palms, and conjunctiva. Each dataset underwent a systematic preprocessing and balancing procedure to prepare it for training machine learning models with high generalizability and clinical relevance.

#### **A. Raw Dataset Structure and Purpose**

Each dataset consists of labeled image folders containing samples categorized into two diagnostic classes:

- Anemic
- Non-anemic

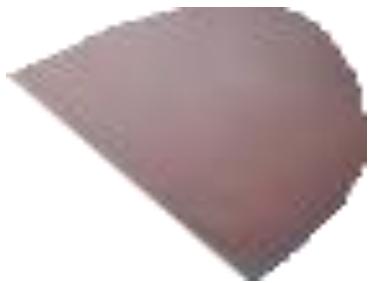
These images serve as input to classification models which predict the anemia condition based solely on visual cues. Due to variations in lighting, background, and patient skin tone, preprocessing was essential for both robustness and model fairness.



conjunctiva



Palm



Nails

## B. Data Imbalance and the Need for Balancing

Initial data exploration revealed substantial imbalance across all datasets. For instance, the unbalanced fingernails dataset had the following distribution:

Set	Class	Count
Train	Anemic	2052
Train	Non-anemic	1297
Test	Anemic	536
Test	Non-anemic	317

This imbalance risks skewing the learning process and causing biased predictions toward the majority class.

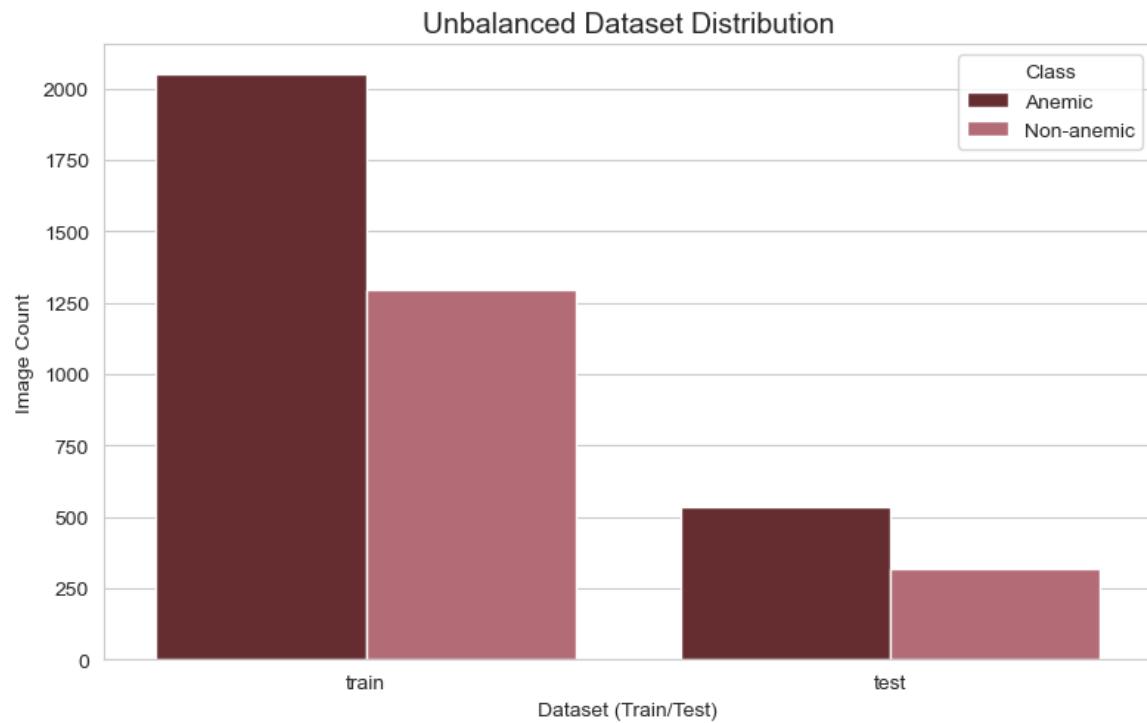


Figure 6.1: Unbalanced bar chart

## C. Balancing via Augmentation

To correct the imbalance, we employed a systematic data augmentation strategy using the Albumentations library. The augmentation pipeline applied transformations such as:

- Horizontal Flipping ( $p=0.5$ )
- Rotation (random angle up to  $\pm 10^\circ$ )
- Random 90-degree rotation

Each image was resized to  $224 \times 224$  and normalized, ensuring consistency across all data samples. The augmentation process was automated such that each class was extended to exactly 2000 images.

```
def get_train_augs():
    return A.Compose([
        A.HorizontalFlip(p=0.5),
        A.OneOf([
            A.Rotate(limit=10, p=0.5),
            A.RandomRotate90(p=0.5)
        ], p=1.0),
    ], p=1.0)
```

## D. Post-Balancing Results

After augmentation, the new distribution was:

Set	Class	Count
Train	Anemic	2565
Train	Non-anemic	2565
Test	Anemic	650
Test	Non-anemic	650

This ensured the training dataset was class-balanced, helping the model learn without bias toward the majority class.

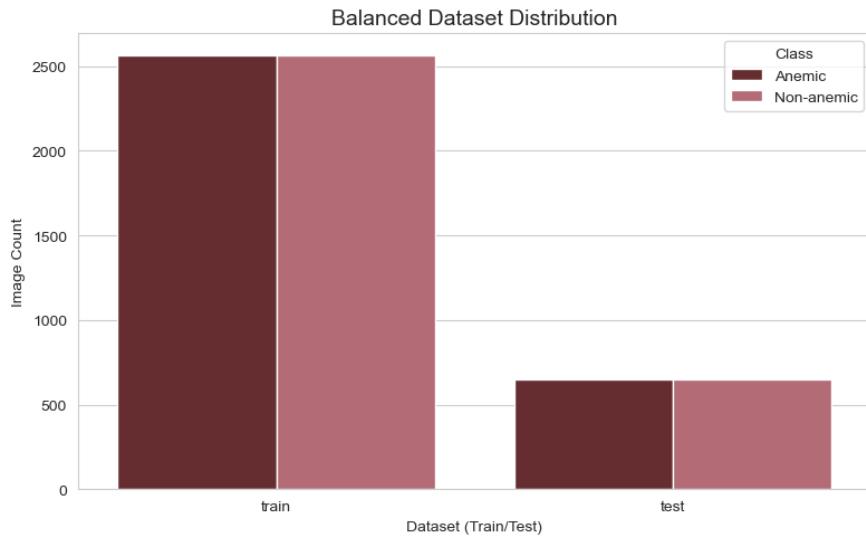


Figure 6.2: Balanced bar chart

### E. Visual Comparison (Before vs After)

To further illustrate the effectiveness of balancing, we used pie charts to visualize class distributions in the unbalanced and balanced datasets. Before balancing, the dataset had a 61.6% to 38.4% split between anemic and non-anemic. After balancing, the classes were perfectly equal.

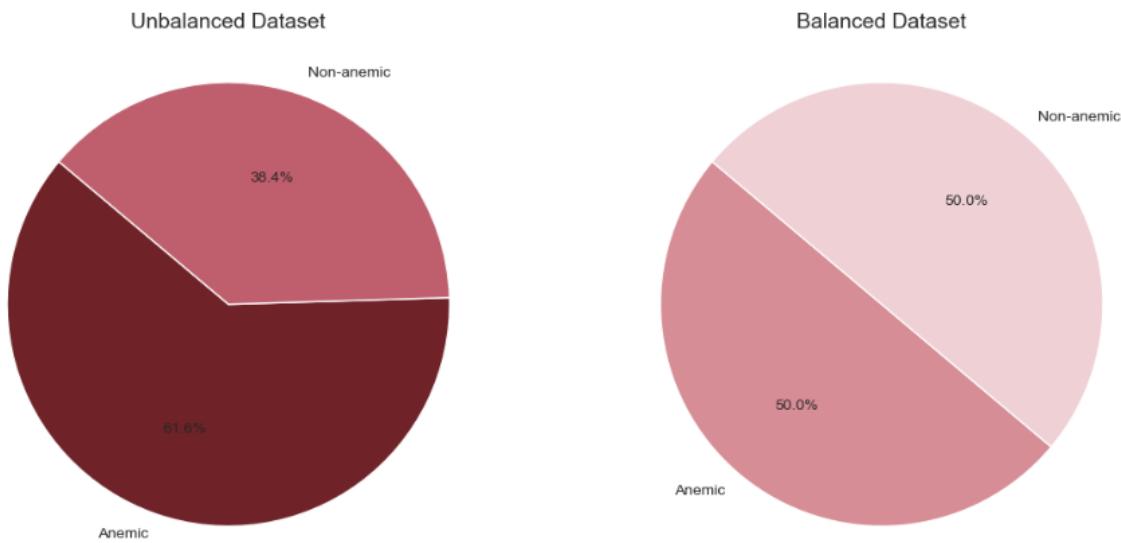


Figure 6.3: Unbalanced Data vs Balanced Data

### **6.1.2 Segmentation Datasets (Nails, Palm, Conjunctiva)**

In this section, we describe the segmentation datasets utilized to isolate relevant anatomical features from input images prior to classification. The segmentation plays a crucial role in localizing the region of interest (ROI) such as nails, palm surfaces, and conjunctiva, which enhances downstream prediction accuracy by focusing on disease-indicative zones.

#### **Purpose of Segmentation**

Segmentation was applied as a preprocessing step to remove irrelevant background noise and retain only the medically meaningful area of the image. Three datasets were used:

- Nail Segmentation: Isolates fingernail area.
- Palm Segmentation: Extracts the central palm region.
- Conjunctiva Segmentation: Masks the conjunctival area in the eye (palpebral and forniceal).

#### **A. Nail Segmentation Dataset**

The nail segmentation dataset is designed to isolate the nail bed from the surrounding finger region. This step ensures that subsequent analysis (e.g., color-based anemia detection) is performed only on the relevant area.

- Images Count: 52
- Labels Count: 52
- Format: RGB images and binary masks
- Directory Structure:
  - images/: Contains the raw fingernail images
  - labels/: Contains binary masks corresponding to the nail region

Despite its relatively small size, this dataset plays a crucial role in preprocessing nail images. Each mask defines the nail region with pixel-level precision, which is later used to extract histogram-based features for anemia classification.

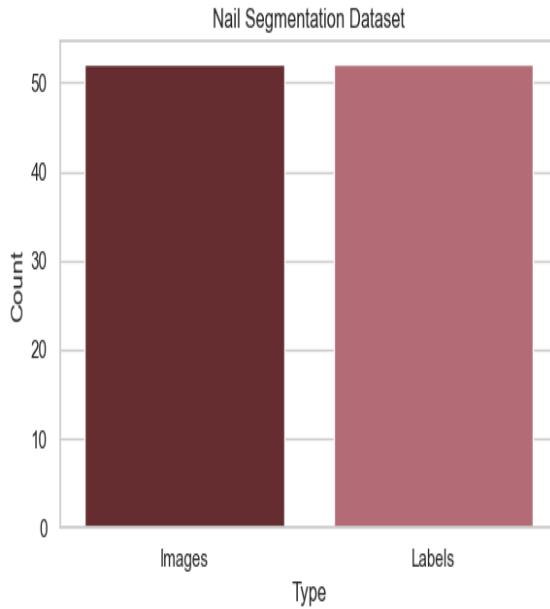


Figure 6.4: Nail dataset segmentation



Figure 6.5: Region of Nail Bed

## B. Palm Segmentation Dataset

The palm segmentation dataset is the largest among the three and is used to segment the central area of the palm, a region clinically associated with pallor in anemic conditions.

- Images Count: 420
- Labels Count: 420
- Format: RGB images + YOLO-style text label files
- Directory Structure:
  - train/images/: Raw palm images
  - train/labelTxt/: YOLO-formatted text files with bounding box coordinates
  - data.yaml: Class definition and dataset metadata

This dataset follows the YOLO object detection annotation style, where labels define the bounding box enclosing the palm region. It was exported using Roboflow, ensuring standardized annotation formatting. The palm region is later cropped using these annotations to feed into the classification model.

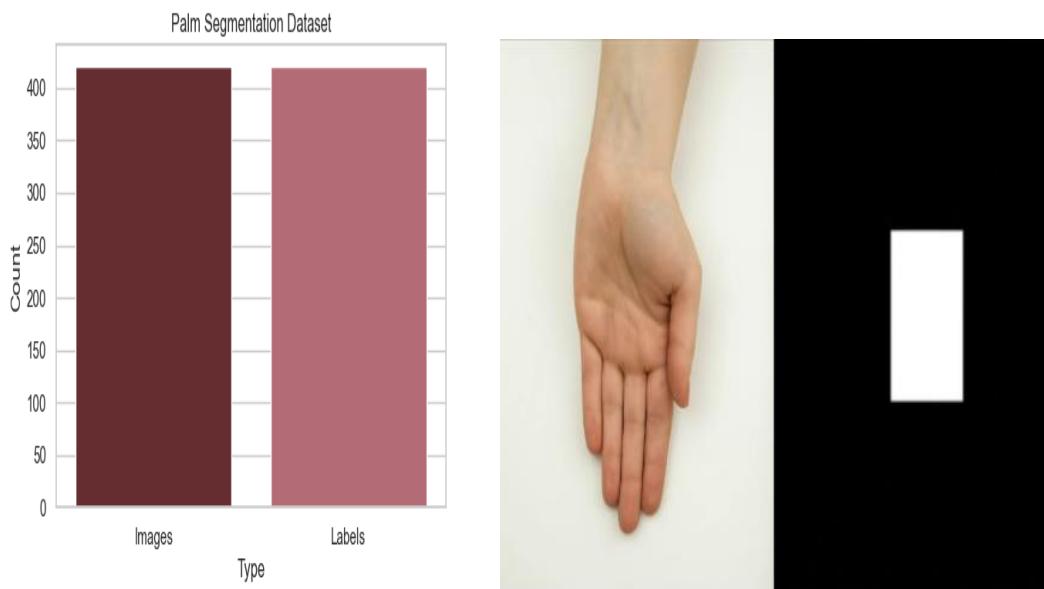


Figure 6.6: Palm dataset segmentation

Figure 6.7: Region of Central Palm

### C. Conjunctiva Segmentation Dataset

The conjunctiva segmentation dataset targets a medically significant region—the inner eyelid—where paleness is a clinical indicator of anemia. The dataset includes pixel-level annotated masks of the palpebral and forniceal regions.

- Images Count: 180
- Labels Count: 180
- Format: RGB images and grayscale or 2-channel binary masks
- Directory Structure:
  - images/: Raw eye images
  - masks/: Segmentation masks with labeled conjunctival regions

This dataset enables highly localized analysis of the conjunctival area. Each image has a corresponding mask that highlights either the palpebral region, the forniceal region, or both. The segmentation model is trained on this data to provide clean region-of-interest masks, enhancing the interpretability and precision of anemia detection based on eye color characteristics.

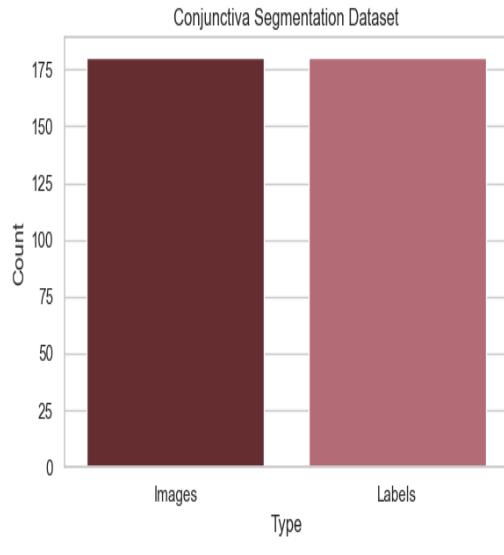


Figure 6.8: Conjunctiva dataset segmentation



Figure 6.9: Region of inner Eyelid

### **6.1.3 LLM Dataset (Medical Chatbot Conversations)**

As part of this project, we integrated a large-scale medical language dataset titled “AI Medical Chatbot” from Hugging Face, published under the identifier ruslanmv/ai-medical-chatbot. This dataset consists of over 256,000 real-world doctor-patient dialogue pairs, offering a rich resource for training and analyzing interactive medical language models.

#### **A. Dataset Structure**

Each record in the dataset contains:

- Description: Meta description of the case
- Patient: A question or concern raised by a patient
- Doctor: A medically informed response from a virtual doctor

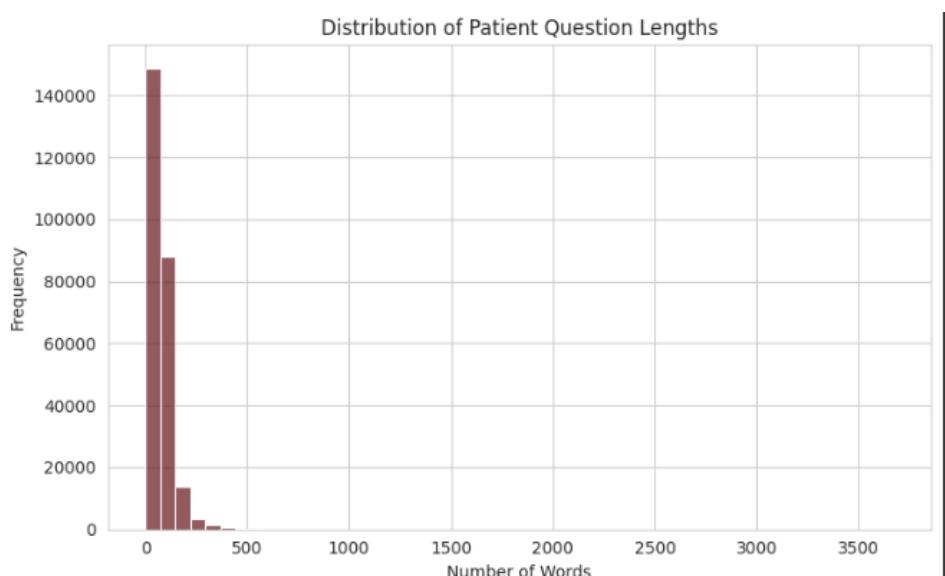
This dataset was chosen not only for its large volume and diversity but specifically because it includes multiple question-answer pairs related to anemia, which is the core condition targeted by our application.

## B. Why This Dataset Was Selected

Although the dataset covers a wide range of medical specialties, it contains a high frequency of natural questions about anemia—including symptoms (e.g. fatigue, pale skin), causes (e.g. iron deficiency), and treatment-related inquiries. These real-world interactions help us train a context-aware language model that can respond intelligently and empathetically to patients' concerns about anemia.

### . General Analysis of the Full Dataset

To understand the overall structure and linguistic complexity of the dataset, we conducted an analysis of message lengths in both the patient and doctor fields.



## 2. Distribution of Doctor Response Lengths

Doctor responses tend to be more elaborate, as shown in the distribution chart. This information was critical for defining token length and memory limits in the chatbot pipeline.

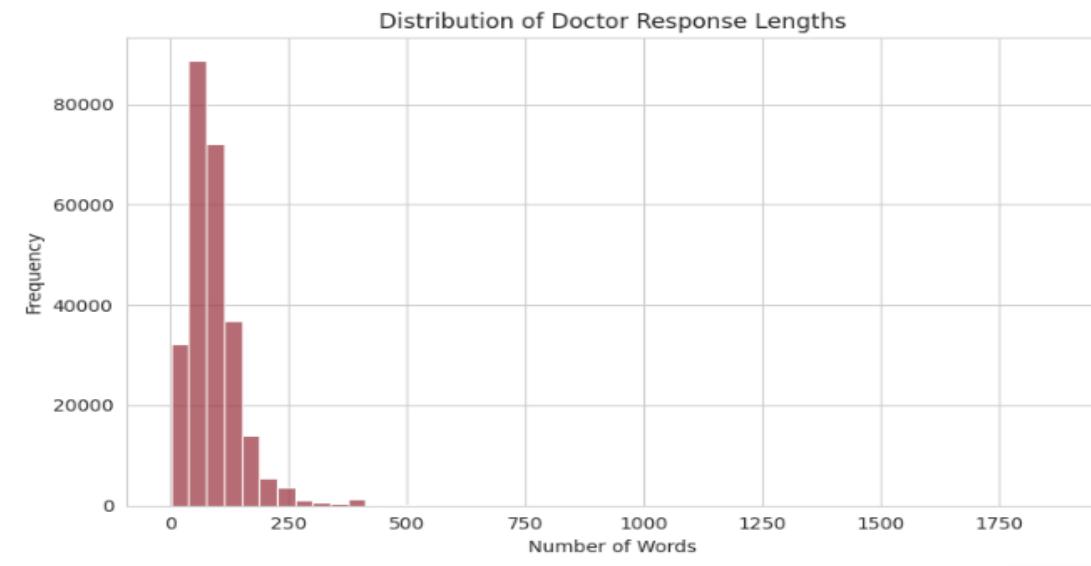


Figure 6.11: Distribution of Doctor Question Lengths

## D. Focused Extraction for Anemia

After general analysis, we applied keyword-based filtering to isolate conversations related to anemia. A total of 863 Q&A pairs mentioning terms like “anemia”, “iron”, “tired”, and “weak” were extracted.

A targeted analysis of this subset revealed the most common keywords used by patients when describing anemia-related symptoms:

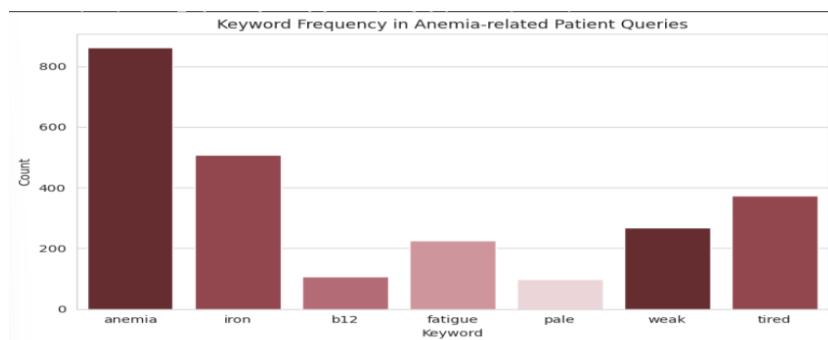


Figure 6.12: Anemia-related Patient Queries

These insights were used to:

- Design anemia-aware prompt templates for the chatbot
- Fine-tune model responses to match real patient language
- Improve user engagement and trust by ensuring medical relevance

## 6.2 Image Classification

### 6.2.1 Preprocessing

#### A. Conjunctiva Segmentation

##### Dataset Preparation and Organization

To perform accurate segmentation of the conjunctiva, we used the publicly available Eyes-Defy-Anemia dataset, which provides annotated eye images along with segmentation masks for palpebral, forniceal, and combined conjunctival regions. Each image is captured under controlled lighting with a magnification lens to ensure uniformity.

A preprocessing pipeline was implemented to:

- ✓ Traverse patient folders,
- ✓ Extract only the first and last images (to reduce redundancy),
- ✓ Automatically separate original images and mask files based on naming conventions.

##### Dataset Cleanup and Verification

We have taken multiple steps to ensure data quality:

- Verified presence of both image and corresponding mask files.
- Removed unwanted indices based on clinical or annotation inconsistencies.

##### Image and Mask Preprocessing

All images and masks were resized to 128x128 and normalized:

- ✓ Images: Scaled from 0–255 to [0,1] and converted to RGB.
- ✓ Masks: Converted to grayscale, thresholded to binary (0 or 1), and reshaped to (H, W, 1).

This uniformity is critical for efficient model convergence and compatibility with U-Net input dimensions.

## U-Net Architecture for Segmentation

We implemented a classic U-Net architecture, which is highly effective for biomedical image segmentation due to its encoder-decoder symmetry and skip connections.

```
def unet(input_size=(IMG_HEIGHT, IMG_WIDTH, 3)):
    inputs = Input(input_size)
    c1 = Conv2D(64, (3, 3), activation='relu', padding='same')(inputs)
    c1 = Conv2D(64, (3, 3), activation='relu', padding='same')(c1)
    p1 = MaxPooling2D((2, 2))(c1)

    c2 = Conv2D(128, (3, 3), activation='relu', padding='same')(p1)
    c2 = Conv2D(128, (3, 3), activation='relu', padding='same')(c2)
    p2 = MaxPooling2D((2, 2))(c2)

    c3 = Conv2D(256, (3, 3), activation='relu', padding='same')(p2)
    c3 = Conv2D(256, (3, 3), activation='relu', padding='same')(c3)
    p3 = MaxPooling2D((2, 2))(c3)

    c4 = Conv2D(512, (3, 3), activation='relu', padding='same')(p3)
    c4 = Conv2D(512, (3, 3), activation='relu', padding='same')(c4)
    p4 = MaxPooling2D((2, 2))(c4)

    # Bottleneck
    c5 = Conv2D(1024, (3, 3), activation='relu', padding='same')(p4)
    c5 = Conv2D(1024, (3, 3), activation='relu', padding='same')(c5)

    # Decoder
    u6 = UpSampling2D((2, 2))(c5)
    u6 = concatenate([u6, c4])
    c6 = Conv2D(512, (3, 3), activation='relu', padding='same')(u6)
    c6 = Conv2D(512, (3, 3), activation='relu', padding='same')(c6)

    u7 = UpSampling2D((2, 2))(c6)
    u7 = concatenate([u7, c3])
    c7 = Conv2D(256, (3, 3), activation='relu', padding='same')(u7)
    c7 = Conv2D(256, (3, 3), activation='relu', padding='same')(c7)

    u8 = UpSampling2D((2, 2))(c7)
    u8 = concatenate([u8, c2])
    c8 = Conv2D(128, (3, 3), activation='relu', padding='same')(u8)
    c8 = Conv2D(128, (3, 3), activation='relu', padding='same')(c8)

    u9 = UpSampling2D((2, 2))(c8)
    u9 = concatenate([u9, c1])
    c9 = Conv2D(64, (3, 3), activation='relu', padding='same')(u9)
    c9 = Conv2D(64, (3, 3), activation='relu', padding='same')(c9)

    outputs = Conv2D(1, (1, 1), activation='sigmoid')(c9)

    model = Model(inputs, outputs)
    return model
```

## **U-Net model (Conjunctiva)**

### **Encoder (Downsampling Path)**

- Consists of 4 levels of convolution followed by max pooling.
- Feature maps double at each level:  $64 \rightarrow 128 \rightarrow 256 \rightarrow 512$ .
- Each level captures increasingly abstract spatial features.

### **Bottleneck**

- The deepest layer with 1024 filters, representing the most abstract compressed features.
- No pooling applied here.

### **Decoder (Upsampling Path)**

- Uses UpSampling2D followed by convolution.
- Feature maps halve:  $1024 \rightarrow 512 \rightarrow 256 \rightarrow 128 \rightarrow 64$ .
- Skip connections from encoder layers are concatenated at each level to recover spatial information.

### **Output Layer**

- A  $1 \times 1$  convolution with sigmoid activation outputs a binary mask (values between 0 and 1).

The model was compiled using:

- Optimizer: Adam (adaptive learning)
- Loss function: Binary Cross-Entropy (ideal for pixel-wise binary classification)
- Metric: Accuracy

Training used callbacks:

- EarlyStopping: Halts training if validation loss stops improving
- ReduceLROnPlateau: Dynamically lowers learning rate if stuck in local minima

## Training and Evaluation

After completing the preprocessing steps and model construction, the U-Net model was trained on the cleaned conjunctiva dataset consisting of 96 image-mask pairs. The dataset was split into a training set of 76 images and a validation set of 20 images, maintaining an 80:20 ratio. Input images and masks were resized to  $128 \times 128$  and normalized prior to training.

The model was trained over 50 epochs with a batch size of 8, using the Adam optimizer and binary cross-entropy as the loss function. To improve generalization and prevent overfitting, two key callbacks were used:

- **EarlyStopping**: Monitors validation loss and halts training once performance stops improving.
  - **ReduceLROnPlateau**: Dynamically reduces the learning rate if the model plateaus.

```
model = unet(input_size=(IMG_HEIGHT, IMG_WIDTH, 3))
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

early_stop = EarlyStopping(monitor='val_loss', patience=5, verbose=1)
lr_reduction = ReduceLROnPlateau(monitor='val_loss', patience=3, factor=0.5, verbose=1)

history = model.fit(
    train_images, train_masks,
    validation_data=(val_images, val_masks),
    epochs=50,
    batch_size=8,
    callbacks=[early_stop, lr_reduction]
)
```

Figure 6.13: U-Net training setup

During training, the model showed a consistent increase in performance. Initial epochs demonstrated high variance in loss (e.g., loss = 49.80 at epoch 1), which rapidly stabilized by epoch 5. Accuracy steadily improved, reaching over 98% in later epochs on the training set and 97–98% on the validation set. Additionally, the validation loss decreased from 0.3870 in the first epoch to approximately 0.0490 before early stopping was triggered.

## Observation:

The model exhibited strong convergence with no signs of overfitting. Validation accuracy consistently tracked training accuracy, while validation loss remained low and stable.

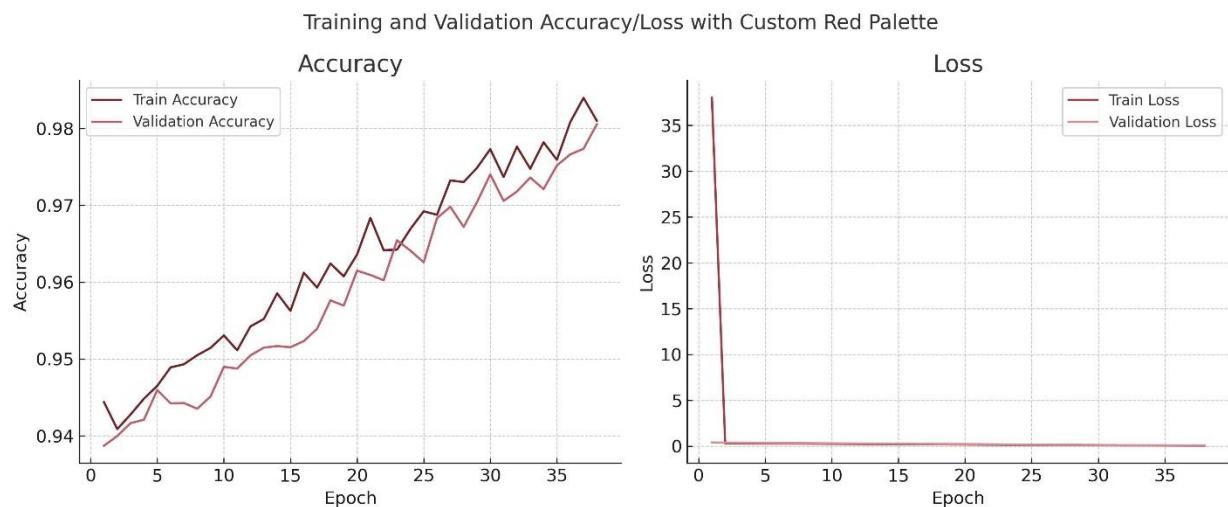


Figure 6.14: Training Curves

These plots further confirm the robustness of training. As seen, both training and validation accuracy plateau around epoch 35, while the loss curves reflect smooth and gradual minimization. Learning rate adjustments effectively fine-tuned the model's progression toward optimal weights.

## Output sample

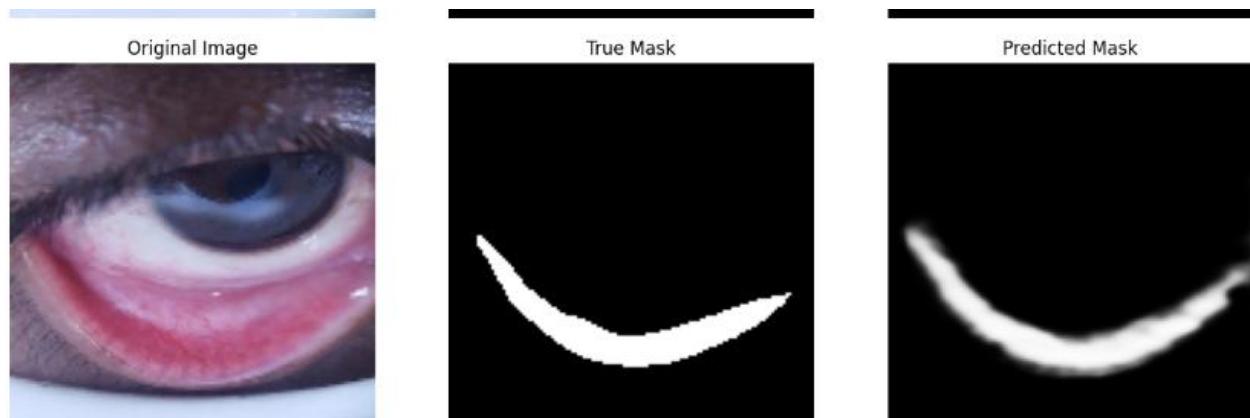


Figure 6.15: Conjunctiva Implementation

## B. Fingernail Segmentation

### Dataset Preparation and Organization

To train a robust segmentation model for fingernails, we used the publicly available Nails-Segmentation dataset from Kaggle. This dataset consists of RGB images of fingers alongside manually annotated binary masks that outline the nail regions. It serves as a suitable starting point for training deep learning models on nail-based biomedical analysis tasks such as anemia detection.

A preprocessing routine was implemented to:

- Load image filenames from the images/ and labels/ folders.
- Match each image with its corresponding mask by filename.
- Normalize and resize the inputs for model compatibility.

Both the RGB images and their corresponding masks were resized to  $224 \times 224$  pixels to match the input requirement of the DeepLabV3 architecture.

### Dataset Cleanup and Verification

To ensure that the dataset was clean and well-aligned:

- Each image was verified to have a corresponding mask of identical dimensions.
- Visualization of image–mask pairs was conducted to confirm alignment and annotation quality.
- A random split was performed to separate the dataset into training and testing subsets (45 for training, 7 for testing), ensuring balanced representation.

## **Image and Mask Preprocessing**

Preprocessing steps were applied using PyTorch's transforms module:

- Images were:
  - ✓ Resized to (224, 224)
  - ✓ Normalized using the ImageNet mean and standard deviation values:  
(mean = [0.485, 0.456, 0.406], std = [0.229, 0.224, 0.225])
  - ✓ Converted to tensors with shape (3, H, W)
- Mask Processing:
  - ✓ Converted to grayscale
  - ✓ Resized to 224×224
  - ✓ Thresholder to binary using label > 0.5 → 1.0 (nail), else 0.0 (background)
  - ✓ Reshaped to a single channel

## **Constraint: Black Background Requirement**

To meet clinical analysis requirements, all non-nail regions (background) were explicitly set to dark. This constraint ensures:

- ✓ Visual clarity during segmentation visualization
- ✓ Seamless post-processing during feature extraction
- ✓ Consistent downstream input for the anemia classification model

## **Model Architecture: DeepLabV3 with ResNet101 Backbone**

To perform accurate segmentation, we deployed the DeepLabV3 architecture integrated with a ResNet101 backbone, pretrained on ImageNet. DeepLabV3 excels in semantic segmentation by using dilated convolutions and Atrous Spatial Pyramid Pooling (ASPP) to capture multi-scale context while maintaining high resolution.

## Custom Modifications:

- ✓ The final classifier layers were modified to output a 1-channel binary mask using nn.Conv2d(256, 1, kernel\_size=1).
- ✓ This change converted the segmentation task into a binary classification per pixel (nail vs. background).

```
from torchvision.models.segmentation import deeplabv3_resnet101, DeepLabV3_ResNet101_Weights
import torch.nn as nn

model = deeplabv3_resnet101(weights=DeepLabV3_ResNet101_Weights.DEFAULT)

model.classifier[4] = nn.Conv2d(256, 1, kernel_size=1)
model.aux_classifier[4] = nn.Conv2d(256, 1, kernel_size=1)

print(model)
```

## “DeepLabV3 setup (Nail)”

## Architecture Summary:

- ✓ **Input:** RGB image ( $3 \times 224 \times 224$ )
- ✓ **Encoder:** ResNet101 (deep residual learning with batch normalization)
- ✓ **ASPP Module:** Aggregates multi-scale features using different dilation rates
- ✓ **Decoder:** Up-samples and refining features for spatial precision
- ✓ **Output:**  $1 \times 224 \times 224$  binary mask with sigmoid activation

## Training Configuration and Optimization

The training was carried out using a GPU-accelerated pipeline with mixed precision to enhance performance and reduce memory usage.

## Optimization Settings:

- ✓ **Loss Function:** Dice Loss — chosen for its robustness in handling class imbalance and small target regions.
- ✓ **Metric Evaluation:** IoU (Intersection-over-Union)
- ✓ **Optimizer:** AdamW :improves generalization through weight decay.
- ✓ **Scheduler:** CosineAnnealingLR : gradually decays learning rate to fine-tune convergence.

## Advanced Training Techniques:

- Autocast and GradScaler were used for mixed precision training, enabling faster and more memory-efficient execution.

Model convergence was stable after approximately 30 epochs, with both training and testing IoU scores exceeding 0.94, and Dice loss values dropping below 0.05. No overfitting behavior was observed.

## Inference and Mask Visualization

Post-training, the model was evaluated on unseen nail images. The predicted binary masks were:

- Thresholded at 0.5
- Displayed alongside the original input image
- Applied as a soft-masked overlay to highlight the segmented nail region

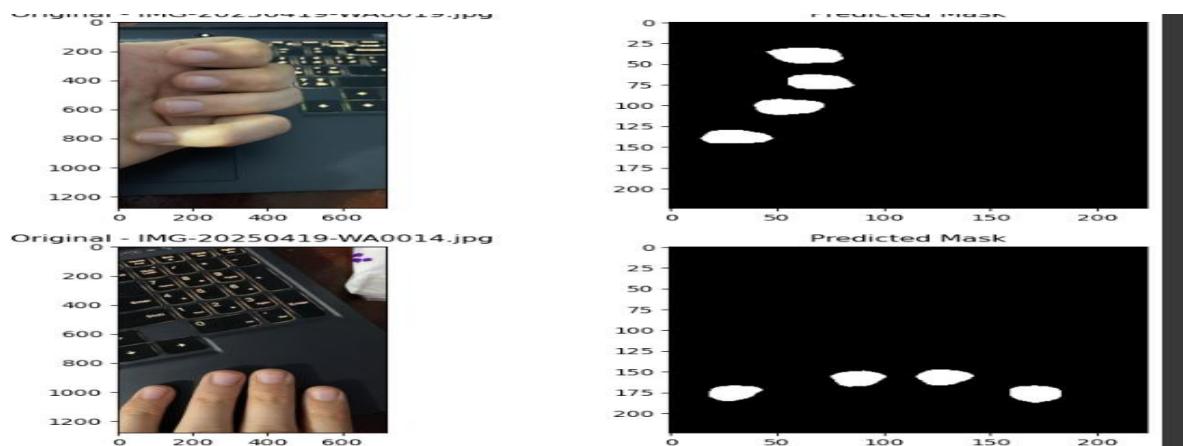


Figure 6.16: Fingernails segmentation

## C. Palm Segmentation

### Dataset Preparation and Organization

To train an accurate palm segmentation model, we utilized a YOLO-format dataset containing hand images and corresponding polygon annotations stored as .txt files. These label files define the palm region through a set of 2D coordinates forming a polygon.

A preprocessing routine was implemented to:

- Read each image from the images/ folder and match it to its corresponding .txt label.
- Parse polygon coordinates to generate binary masks via cv2.fillPoly.
- Ensure that each pixel in the mask is either black (background = 0) or white (palm = 1).

### Constraint: Black Background Enforcement

To align with clinical and computational requirements:

- The background in all masks was explicitly set to dark.
- This ensures visual clarity, post-processing consistency, and compatibility with downstream tasks such as palm-based anemia detection.

### Dataset Cleanup and Verification

To ensure dataset reliability:

- All images were validated to have corresponding label files.
- Visualization was used to inspect and confirm mask accuracy.
- Dataset splits (train, valid, test) were preserved to ensure reproducibility and balanced evaluation.

## Image and Mask Preprocessing

Each image and its corresponding mask underwent standardized preprocessing:

### Images:

- Read as RGB using OpenCV.
- Resized to 128×128 resolution.
- Normalized to the range [0, 1].

### Masks:

- Generated from polygon annotations.
- Resized to 128×128.
- Converted to binary values: palm = 1.0, background = 0.0.
- Reshaped to a single-channel format (H, W, 1) for model compatibility.

The final processed data was wrapped into a tf.data.Dataset pipeline for efficient batching and prefetching.

## Model Architecture: U-Net for Palm Segmentation

We implemented a U-Net architecture specifically tailored for binary palm segmentation.

```
def build_unet(input_shape=(128, 128, 3)):
    inputs = layers.Input(input_shape)

    def conv_block(x, filters):
        x = layers.Conv2D(filters, 3, activation='relu', padding='same')(x)
        x = layers.Conv2D(filters, 3, activation='relu', padding='same')(x)
        return x

    def encoder_block(x, filters):
        f = conv_block(x, filters)
        p = layers.MaxPooling2D((2, 2))(f)
        return f, p

    def decoder_block(x, skip, filters):
        x = layers.Conv2DTranspose(filters, 2, strides=2, padding='same')(x)
        x = layers.concatenate([x, skip])
        x = conv_block(x, filters)
        return x

    # Encoder
    s1, p1 = encoder_block(inputs, 64)
    s2, p2 = encoder_block(p1, 128)
    s3, p3 = encoder_block(p2, 256)

    # Bridge
    b = conv_block(p3, 512)

    # Decoder
    d3 = decoder_block(b, s3, 256)
    d2 = decoder_block(d3, s2, 128)
    d1 = decoder_block(d2, s1, 64)

    outputs = layers.Conv2D(1, 1, activation='sigmoid')(d1)

    return models.Model(inputs, outputs)
```

## U-Net model (Palm)

### Architecture Summary:

- Input:  $128 \times 128 \times 3$  RGB images
- Encoder: Three downsampling blocks with Conv2D + ReLU + MaxPooling
- Bridge: 512 filters capturing high-level spatial features
- Decoder: Transposed convolutions for upsampling, concatenated with encoder features via skip connections
- Output Layer:  $1 \times 128 \times 128$  binary mask via a sigmoid-activated  $1 \times 1$  convolution

This architecture is optimal for medical image segmentation tasks, especially where fine boundaries like hand contours must be preserved.

### Training Configuration and Optimization

Training was conducted on a GPU using TensorFlow with the following configuration:

- Optimizer: Adam
- Loss Function: Binary Cross-Entropy
- Metric: Accuracy
- Batch Size: 8
- Epochs: 30

Training and validation datasets were passed using pre-built tf.data.Dataset pipelines.

### Training Performance and Results

#### Training Accuracy Progress:

- Started at 82% in Epoch 1
- Reached up to 97.6% by Epoch 30

## **Validation Accuracy:**

- Ranged from 91% to 97.8%
- Validation loss decreased from 0.16 to as low as 0.056

## **Observation:**

- The model achieved high accuracy and low loss across both training and validation sets.
- No overfitting was observed; validation accuracy closely tracked training accuracy.

## **Inference and Mask Visualization:**

After training, inference was performed on unseen palm images using the trained model. The process included:

- Loading and preprocessing test images.
- Predicting binary masks using the model.
- Thresholding predictions at 0.5 to create final binary outputs.
- Displaying:

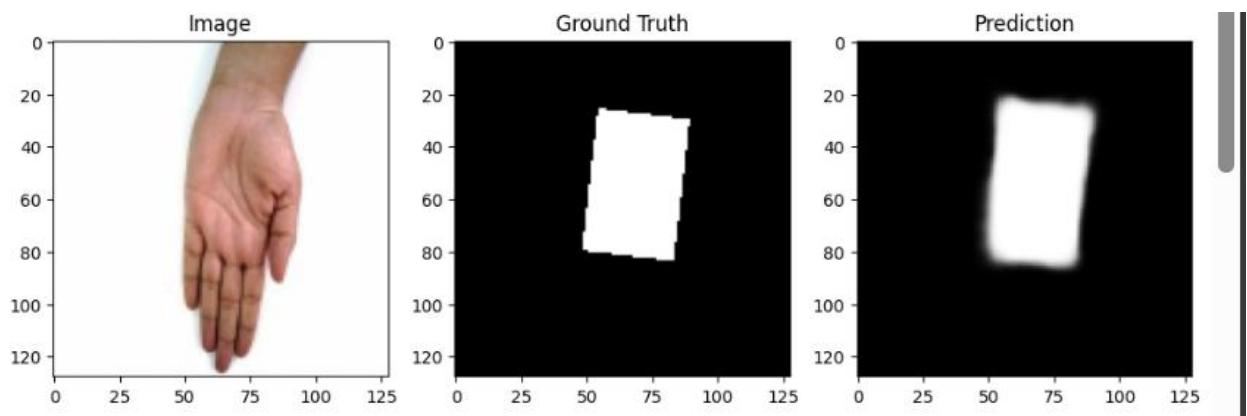


Figure 6.17: Palm segmentation

This confirmed that the model accurately delineates palm regions even under varied illumination and hand poses.

## 6.2.2 Models

This section outlines the architectural details and results of all models employed for anemia detection across three biological regions: conjunctiva, fingernail, and palm. Each model is explained in depth, including its components, learning mechanisms, and medical relevance. Performance comparisons based on standard evaluation metrics are presented at the end of this section.

### 1. Decision Tree

Decision Trees are interpretable, non-parametric models that recursively partition the feature space based on feature thresholds. Each internal node represents a decision rule, and each leaf node corresponds to a class label. In our context, they serve as a baseline classifier that enables fast inference and clear decision paths.

#### Key Features:

- Splits data based on impurity minimization (e.g., Gini or entropy)
- Shallow trees are faster but less accurate
- Easily interpretable and fast to train

Used in: Conjunctiva, Palm, Fingernail datasets

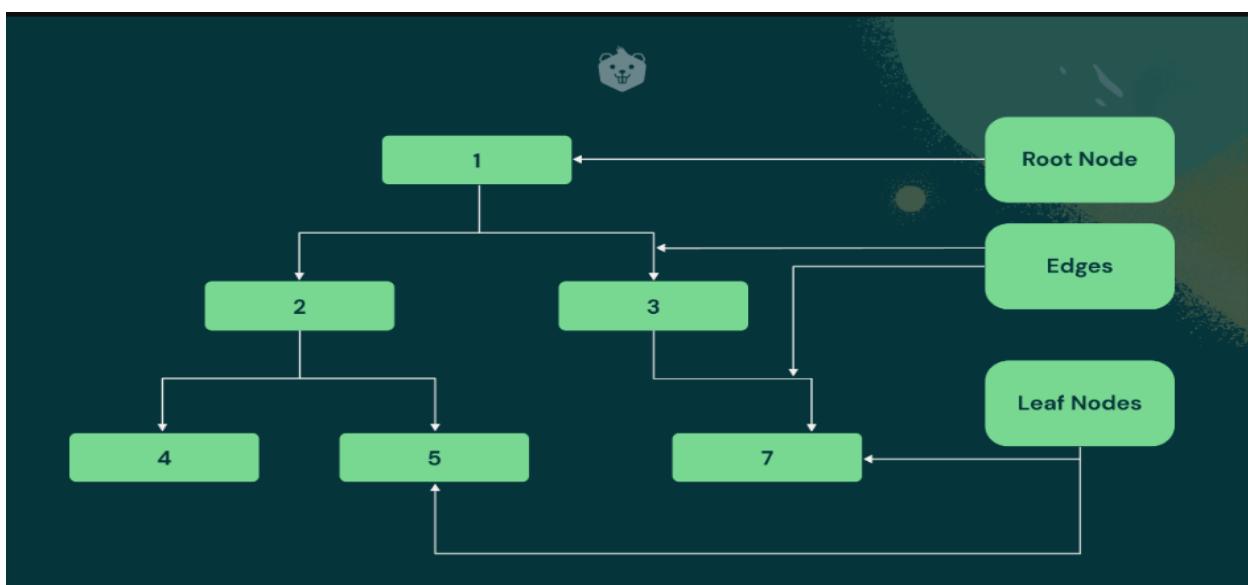


Figure 6.18: Diagram of Decision Tree

## 2. Random Forest

Random Forest is an ensemble of Decision Trees, where each tree is trained on a bootstrap sample and feature subsets. It reduces variance through averaging, leading to better generalization and resistance to overfitting.

### Key Features:

- Bagging-based ensemble of trees
- Each tree votes: majority vote determines output
- Handles noise and high-dimensional data well

Used in: All three regions (Palm, Conjunctiva, Fingernail)

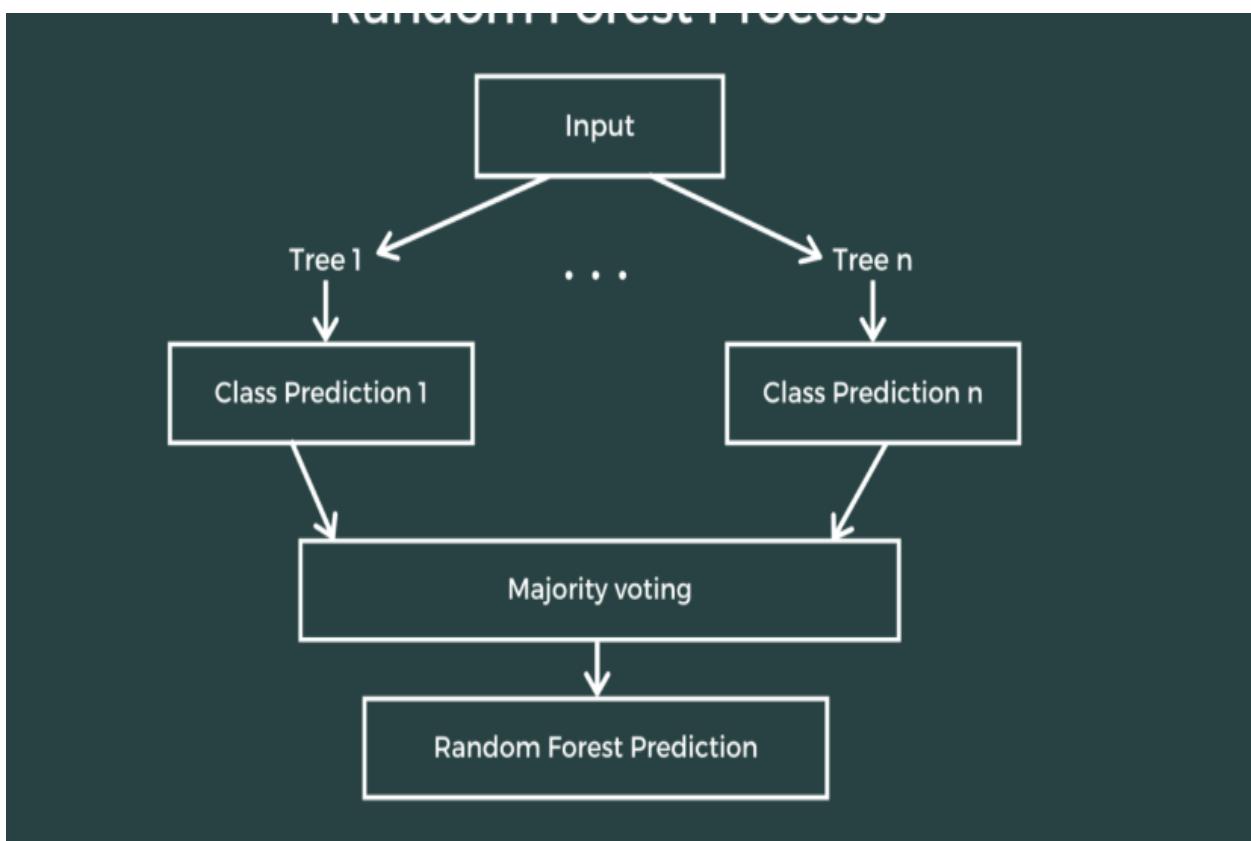


Figure 6.19: Random Forest Process

### 3. XGBoost (and Tuned Variant)

Extreme Gradient Boosting (XGBoost) is a powerful boosting algorithm that builds decision trees sequentially, where each new tree tries to correct the errors of the previous one. The tuned version involves hyperparameter optimization using cross-validation.

#### Key Features:

- Additive model with regularization
- Shrinks weights to reduce overfitting
- Efficient parallel processing and tree pruning

Used in: Conjunctiva, Palm, Fingernail datasets

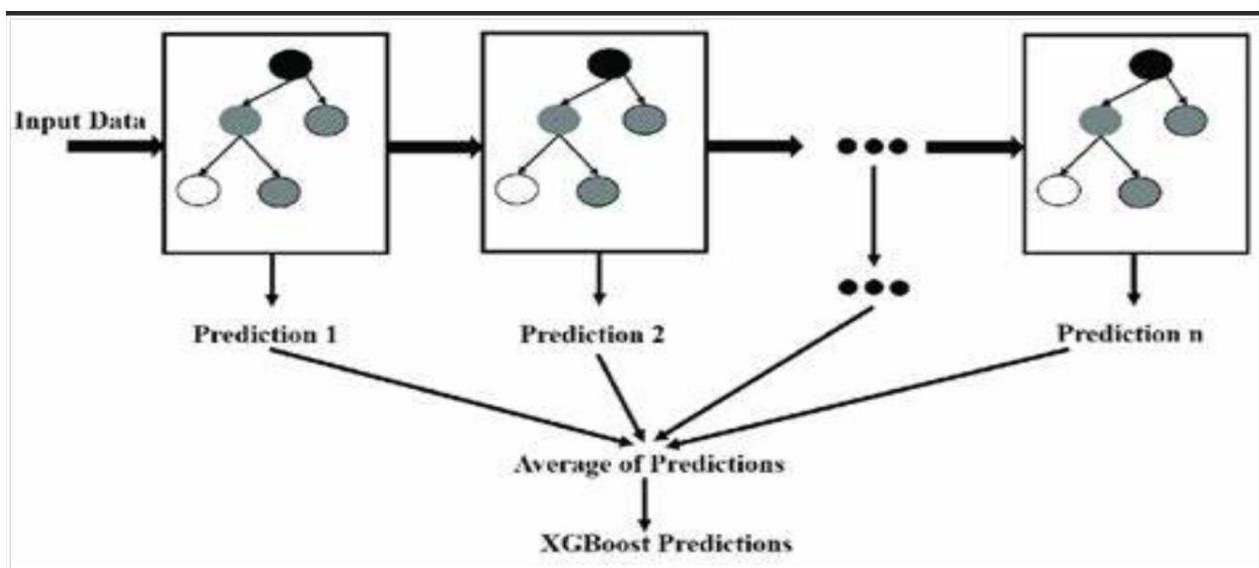


Figure 6.20: XGBoost Prediction

### 4. Vision Transformer (ViT)

Vision Transformers divide an image into patches and apply Transformer encoders to learn image representations. Unlike CNNs, ViTs rely on self-attention to capture long-range dependencies.

## Key Features:

- Patch embedding via linear projection
- Self-attention captures global spatial context
- Positional encoding preserves spatial layout

## Variants:

- ViT Balanced: Trained on class-balanced datasets
- ViT Imbalanced: No data balancing applied
- ViT + WRS: Weighted random sampling used to reduce bias

Used in: Conjunctiva and Palm datasets

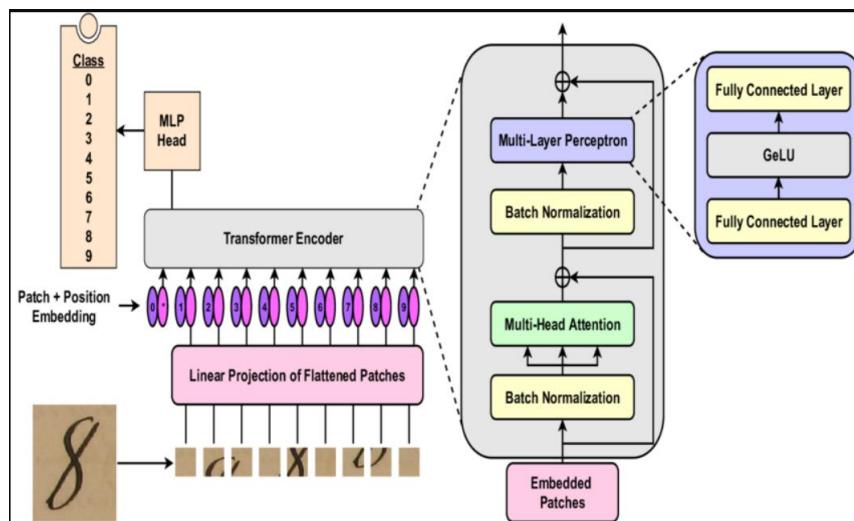


Figure 6.21: ViT architecture

## 5. Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are the foundational deep learning architecture for image classification and segmentation tasks. Unlike traditional fully connected networks, CNNs are specially designed to handle spatial data by exploiting the local spatial coherence in images through a series of convolutional layers, pooling operations, and non-linear activations.

## Core Components of CNNs:

- Convolutional Layers: Learn spatial filters applied over input images.
- Activation Functions (ReLU): Introduce non-linearity.
- Pooling Layers: Downsample feature maps.
- Fully Connected Layers: Map features to final predictions.
- BatchNorm & Dropout: Improve convergence and regularization.

## Variants Used:

- Basic 3-Layer CNN: 3 Conv2D + MaxPool + ReLU + Sigmoid output
- Attention-Based CNN: Enhances spatial focus with attention blocks

Used in: Palm (Balanced and Unbalanced), Fingernail (EfficientNet as an extension)

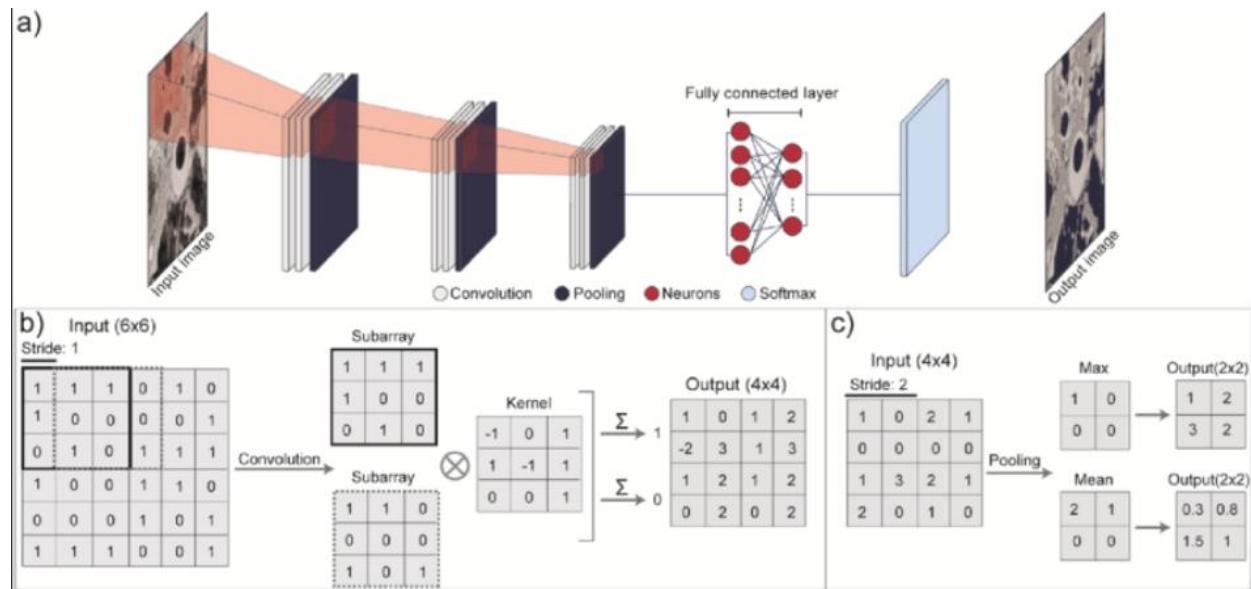


Figure6.22: CNN model

## 6. DenseNet121 (Transfer Learning)

DenseNet121 is a densely connected convolutional network where each layer receives inputs from all preceding layers. This architecture enhances feature reuse and gradient flow, making it suitable for medical image classification.

Key Features:

- Dense connections improve information flow
- Fewer parameters than traditional CNNs of similar depth
- Pretrained on ImageNet and fine-tuned on medical datasets

Used in: Conjunctiva

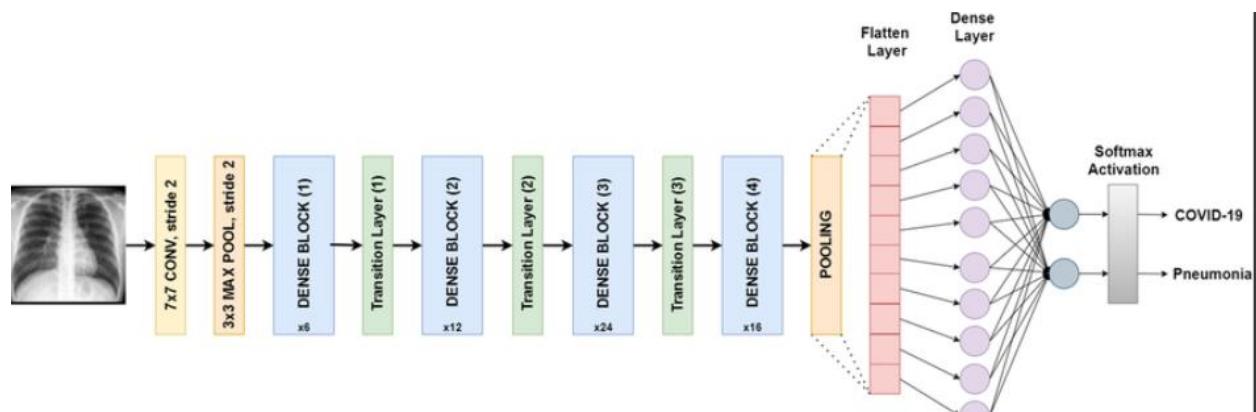


Figure6.23: DenseNet121 architecture

## 7. EfficientNet-B0

EfficientNet is a CNN architecture that uses a compound scaling method to uniformly scale depth, width, and resolution. B0 is the baseline variant optimized for efficiency.

Key Features:

- Scales up model in a balanced way
- State-of-the-art performance with fewer parameters
- Efficient for edge deployment

Used in: Fingernail-Based Anemia Detection

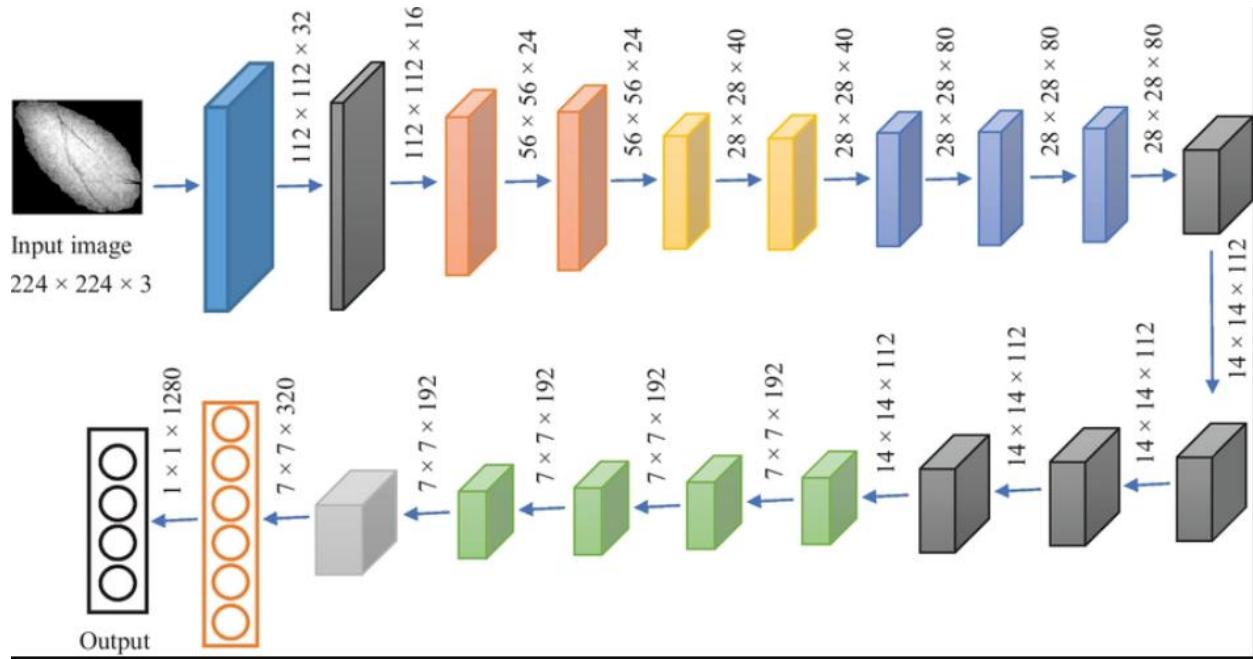


Figure 6.24: EfficientNet-B0 architecture

### 6.2.3 Evaluation

To assess the performance of the various models used for anemia detection across conjunctiva, palm, and fingernail images, we employed a standardized set of evaluation metrics commonly used in medical image analysis:

#### Evaluation Metrics:

1. **Accuracy (%)**: Measures the overall proportion of correct predictions (both positive and negative). While widely used, it can be misleading in imbalanced datasets.
2. **Sensitivity (Recall) (%)**: Represents the model's ability to correctly identify positive (anemic) cases. High sensitivity is critical in medical diagnostics to minimize false negatives.
3. **Precision (%)**: Indicates how many of the predicted positive cases are actually true positives. High precision reduces false alarms.

4. **F1 Score (%):** The harmonic mean of precision and recall. F1 Score is especially useful when classes are imbalanced.

**Note:** F1 Score was prioritized in all evaluations due to its balanced consideration of both false positives and false negatives, which is crucial in clinical screening applications where missing true cases or triggering unnecessary alerts can have significant implications.

#### 6.2.4: Results

To determine the most effective models for anemia detection across three anatomical regions (fingernail, palm, and conjunctiva), we compared multiple machine learning and deep learning architectures using four standard metrics: Accuracy, Sensitivity, Precision, and F1 Score. The F1 Score was prioritized due to its balanced handling of false positives and false negatives—essential in medical screening applications where misclassification can lead to delayed diagnosis or unnecessary intervention.

#### **Fingernail-Based Anemia Detection**

The fingernail dataset was evaluated using five different models. Random Forest, XGBoost, and Tuned XGBoost all achieved perfect performance across all metrics (100%), confirming their ability to extract relevant features from nail-based color and texture patterns. The Decision Tree also performed well (93%), but EfficientNet-B0 lagged with the lowest F1 Score (76%), suggesting it was less effective in this context despite being powerful CNN architecture.

## Fingernail -Based Anemia Detection

Model	Accuracy	Sensitivity	Precision	F1 Score
Random Forest	100%	100%	100%	100%
Tuned XGBoost	100%	100%	100%	100%
XGBoost	100%	100%	100%	100%
Decision Tree	93%	93%	93%	93%
EfficientNet-B0	77%	77%	76%	76%

### Selected Model: Random Forest

- **F1 Score: 100%**
- **Justification:** Delivered perfect performance with fast inference and low model complexity.
- **Feature Input:** HSV histograms extracted from segmented nail regions.

```
from sklearn.metrics import classification_report, accuracy_score
from sklearn.ensemble import RandomForestClassifier

rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

y_pred_rf_train = rf_model.predict(X_train)
y_pred_rf_test = rf_model.predict(X_test)
```

Figure 6.25: Random Forest Nails training

### Palm-Based Anemia Detection

The palm region exhibited high model sensitivity overall, but a few architectures struggled with precision. Random Forest outperformed all others with 100% sensitivity and F1 Score. While ViT variants and tuned XGBoost remained highly competitive ( $F1 \approx 98\text{--}99\%$ ), attention-based CNNs suffered from overfitting or underfitting on unbalanced data.

## Palm-Based Anemia Detection

Model	Accuracy	Sensitivity	Precision	F1 Score
Random Forest	99.54%	100%	99%	99%
XGBoost (Tuned)	99.45%	100%	100%	99%
Decision Tree	99%	98.99%	99%	99%
ViT (Balanced)	99%	99%	98%	100%
ViT + WRS	99%	98.48%	100%	98%
CNN (3-Layer Balanced)	91%	90.7%	89%	94%
Decision Tree (Unbalanced)	85.3%	89%	89%	89%
CNN (Attention, Unbalanced)	60%	100%	60%	75%

### Selected Model: Random Forest

- **F1 Score: 100%**
- **Justification:** Provided top performance with low inference latency. Showed greater robustness than CNN or ViT-based models on variable lighting and hand poses.
- **Feature Input:** Color and texture features derived from palm segmentation masks.

```
from sklearn.metrics import classification_report, accuracy_score
from sklearn.ensemble import RandomForestClassifier

rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

y_pred_rf_train = rf_model.predict(X_train)
y_pred_rf_test = rf_model.predict(X_test)
```

Figure 6.26: Random Forest Palm training

## Conjunctiva-Based Anemia Detection

Conjunctival images proved more challenging due to subtle features. Tuned XGBoost and color-histogram-based XGBoost showed superior performance (F1 = 99%). The ViT and Random Forest models also yielded strong results. However, simple CNN and transfer-learned DenseNet121 underperformed, suggesting these architectures may require more domain-specific fine-tuning or additional data.

### Conjunctiva-Based Anemia Detection

Model	Accuracy	Sensitivity	Precision	F1 Score
XGBoost (Color Histogram)	99%	99%	99%	99%
Random Forest (Color Hist.)	98.26%	98%	98%	98%
ViT (Balanced)	97%	97%	98%	97%
DenseNet121 (Transfer)	92.5%	92.5%	92.66%	92.42%
Decision Tree	95.78%	96%	98%	—
Simple CNN	85%	85%	85%	85%

### Selected Model: Tuned XGBoost

- **F1 Score: 99%**
- **Justification:** Balanced strong predictive power with low complexity. Particularly effective on color histogram features and did not require GPU resources for deployment.

```

from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

xgb_model = XGBClassifier(
    n_estimators=200,
    max_depth=10,
    learning_rate=0.1,
    subsample=0.8,
    colsample_bytree=0.8,
    gamma=0.1,
    reg_lambda=1.0,
    reg_alpha=0.5,
    random_state=42
)

xgb_model.fit(X_train, y_train)

y_pred_xgb_train = xgb_model.predict(X_train)
y_pred_xgb_test = xgb_model.predict(X_test)

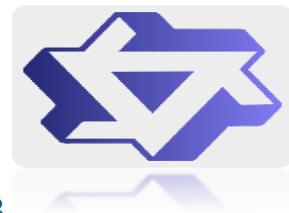
# Accuracy

```

Figure 6.27: XGBoost Conjunctiva training

### **6.3 LLM and NLP-Based Medical Chatbot System**

This section presents the detailed implementation of a hybrid Large Language Model (LLM) and Natural Language Processing (NLP) pipeline designed to simulate medical conversations between patients and doctors. The system integrates advanced language modeling, semantic search, and medical knowledge synthesis to support accurate, context-aware responses for anemia and general health inquiries.



#### 6.3.1 Model Architecture: DeepSeek-R1-Distill-Qwen-7B

The core of our medical chatbot system is the DeepSeek-R1-Distill-Qwen-7B, a powerful transformer-based decoder-only language model optimized for autoregressive text generation. This architecture is inspired by the Qwen family of models and distills the capabilities of larger models into a more efficient 7-billion-parameter variant, enabling high-quality, real-time responses in a medically sensitive domain.

## High-Level Structure

DeepSeek-R1-Distill-Qwen-7B is built using a stacked transformer decoder architecture. Unlike encoder-decoder or encoder-only structures, a decoder-only transformer is specialized for generation tasks, where each token is predicted one after the other using previous tokens as context.

Each layer in this stock includes:

- **Multi-Head Self-Attention (MHSA):** Captures relationships between all previous tokens in the input sequence. This allows the model to reason over full patient histories or prior answers during generation.
- **Feedforward Neural Networks (FFN):** After attention, each token representation is further refined using dense neural layers to capture nonlinear relationships.
- **Layer Normalization and Residual Connections:** Used throughout to stabilize training, prevent vanishing gradients, and speed up convergence.

## Autoregressive Language Modeling

As a causal model, DeepSeek-Qwen applies causal masking, which restricts each token from attending to future tokens. This ensures that generation is left-to-right, allowing the model to simulate human-like turn-based dialogue—essential in doctor-patient interactions.

## Knowledge Distillation

This variant is not trained from scratch; instead, it is distilled from a larger teacher model. Distillation compresses knowledge from a more complex model while preserving reasoning abilities. This results in:

- Significantly reduced model size
- Faster inference speeds
- Lower memory usage

This makes it ideal for our constrained hardware environment while retaining the advanced reasoning expected in medical applications.

## **Memory Optimization with Quantization**

To further reduce the computational load, the model was loaded using 4-bit quantization, a state-of-the-art technique where weights are stored using 4 bits instead of 16 or 32. This:

- Minimizes VRAM usage
- Allows deployment on mid-range GPUs
- Retains nearly full model performance due to quantization-aware optimization

We also leveraged hardware-aware parallelism so the model could automatically spread across available GPUs, thanks to support from high-performance transformer libraries.

## **Tokenizer and Input Handling**

The system uses a custom tokenizer associated with the model, which:

- Converts patient questions and doctor answers into token IDs
- Ensures long inputs (up to 2048 tokens) are handled efficiently
- Preserves clinical formatting such as bullet lists, abbreviations, and numeric values

The tokenizer is essential for transforming raw queries into model-readable format and reconstructing fluent, grammatically correct answers.

## **Why This Architecture Works for Medicine**

Medical chatbot applications require:

- Long-context reasoning (for multi-turn dialogues)
- Domain-specific accuracy (to avoid misinformation)
- Low-latency inference (to support real-time feedback)
- Efficient memory usage (to run on accessible hardware)

DeepSeek-R1-Distill-Qwen-7B addresses these with its transformer architecture, autoregressive modeling, and quantization—all while remaining modular enough to plug into a hybrid semantic search pipeline.

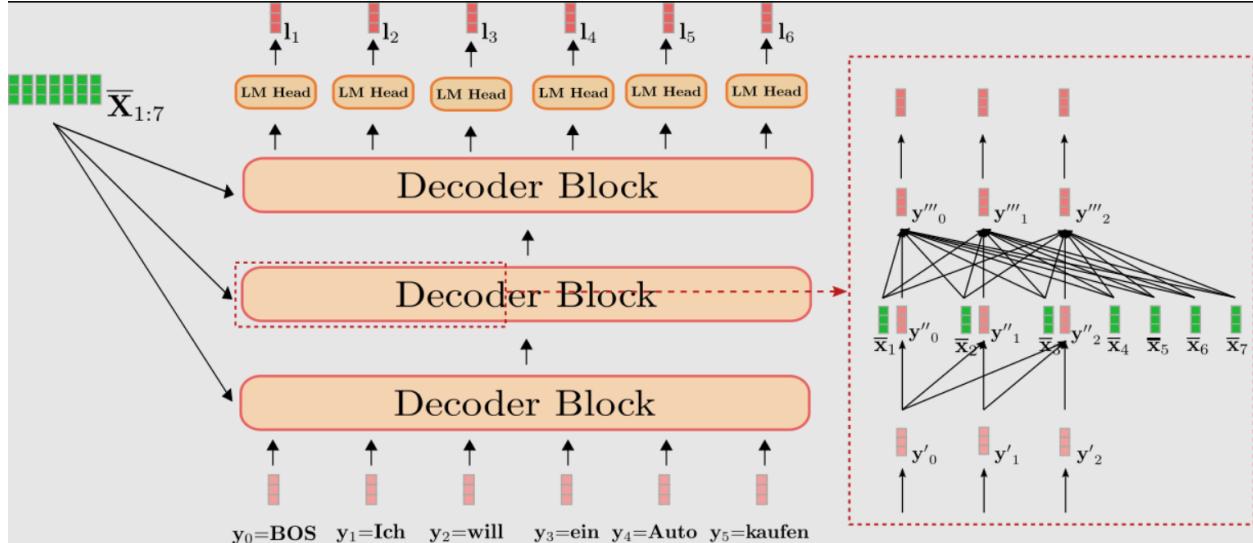


Figure 6.26: Architecture of a Decoder-Only Transformer

### 6.3.2 Model Selection and Compression Strategy

To build an efficient medical chatbot capable of reasoning over health-related queries, we employed the DeepSeek-R1-Distill-Qwen-7B, a transformer-based autoregressive language model trained on diverse biomedical data. Due to the model's large size, we applied 4-bit quantization using BitsAndBytes to reduce memory footprint without sacrificing performance:

```
#Sets up 4-bit quantization to reduce memory usage while loading the model.
bnb_config = BitsAndBytesConfig(load_in_4bit=True)
```

Figure 6.28: 4-bit Quantization Setup

The model was loaded using:

```
model_name = "deepseek-ai/DeepSeek-R1-Distill-Qwen-7B" #pre-trained model
tokenizer = AutoTokenizer.from_pretrained(model_name, trust_remote_code=True, max_seq_length=2048) #converts text to tokens
model = AutoModelForCausalLM.from_pretrained(model_name, quantization_config=bnb_config, trust_remote_code=True, device_map="auto")
```

Figure 6.29: Model Loading

This compression enabled us to deploy the model on local hardware using the accelerate package with multi-GPU support, allowing high-throughput inference.

### 6.3.3 Dataset Loading and Preprocessing

We used the open-source dataset "ruslanmv/ai-medical-chatbot" from HuggingFace, which contains real-world patient-doctor dialogues. The dataset was cleaned and filtered to remove duplicates and redundant entries.

```
▶ from datasets import load_dataset
dataset = load_dataset("ruslanmv/ai-medical-chatbot")
df = dataset['train'].to_pandas()
df = df.drop_duplicates(subset='Description')
```

Figure 6.30: Dataset Loading & Cleaning

Advanced text preprocessing included:

- Lowercasing
- Stopword removal
- Removal of greetings and conversational noise
- Tokenization

```
▶ # Step 2: Text Cleaning Function
def clean_text(text):
    # Convert to lowercase
    text = text.lower()
    # Remove greetings and conversational markers (e.g., "hi.", "hello doctor")
    text = re.sub(r'^hi|hello doctor|hi doctor|dear doctor|hello\.|hi\.\s*', '', text, flags=re.IGNORECASE)
    # Remove punctuation
    text = text.translate(str.maketrans('', '', string.punctuation))
    # Remove extra spaces
    text = re.sub(r'\s+', ' ', text).strip()
    # Optional: Remove stop words
    stop_words = set(stopwords.words('english'))
    words = word_tokenize(text)
    words = [word for word in words if word not in stop_words]
    text = ' '.join(words)
    return text

]
# Download the required NLTK resource
nltk.download('punkt_tab')

[ nlkt_data] Downloading package punkt_tab to /root/nltk_data...
[ nlkt_data]   Unzipping tokenizers/punkt_tab.zip.
True

] # Apply cleaning to relevant columns
df['Description'] = df['Description'].apply(clean_text)
df['Patient'] = df['Patient'].apply(clean_text)
df['Doctor'] = df['Doctor'].apply(clean_text)
```

Figure 6.31: Text Cleaning Function

Custom text cleaning ensured that the LLM processed only relevant clinical content. This was especially important to remove misleading linguistic patterns often present in conversational data.

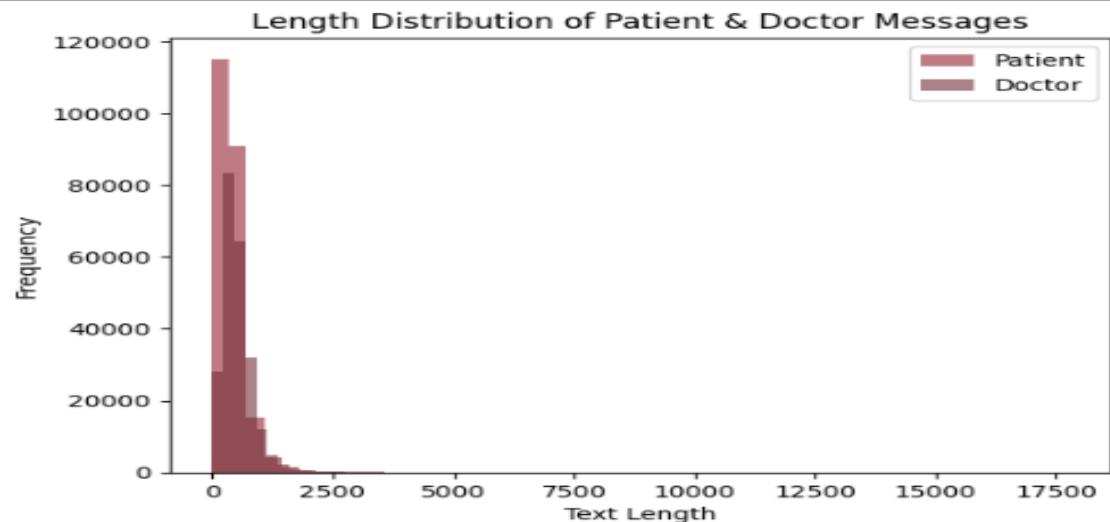


Figure 6.32: Length Distribution (Before cleaning)

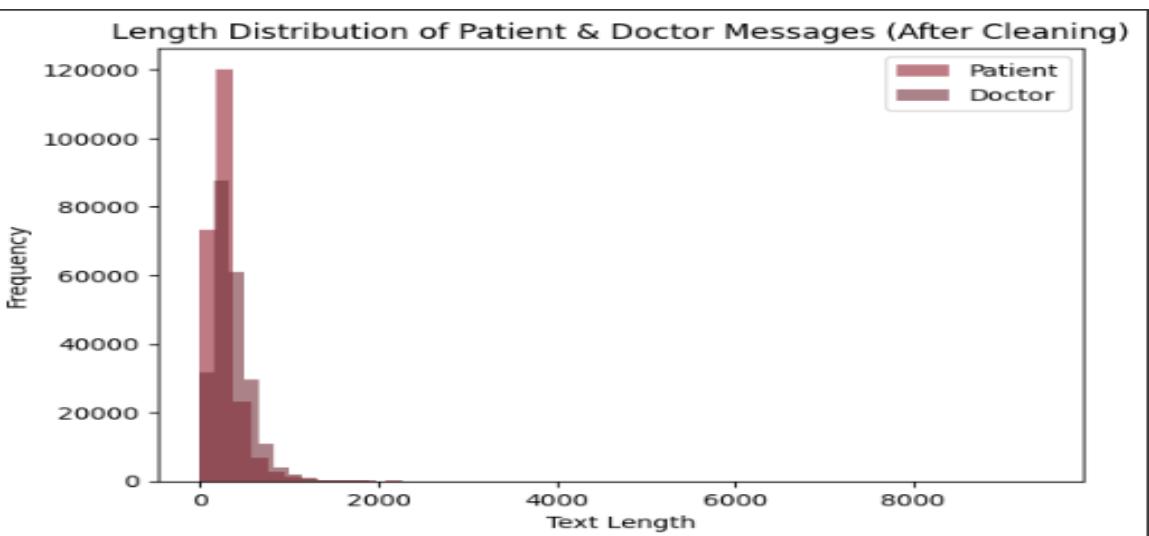


Figure 6.33: Length Distribution (After)

### 6.3.4 Exploratory Data Analysis (EDA) and Visualization

To understand linguistic diversity and user interaction complexity, we:

- Calculated the average text length for both user and doctor responses
  - Created histograms of text length distribution
  - Generated a **WordCloud** of the most common medical terms

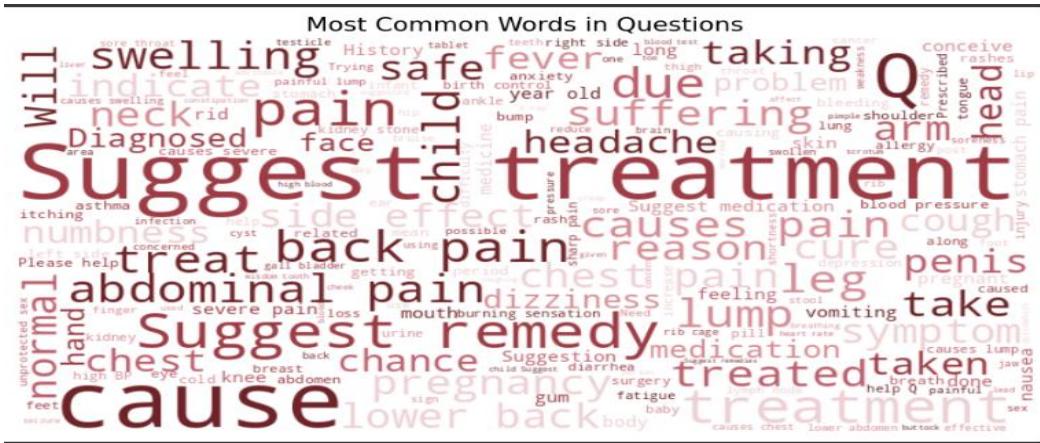


Figure 6.34: WordCloud (Before Cleaning)



Figure 6.35: WordCloud (After Cleaning)

### 6.3.5 Embedding Generation for Semantic Search

We used sentence-transformers/all-MiniLM-L6-v2 to convert concatenated Description + Patient fields into dense semantic vectors:

```
▶ from sentence_transformers import SentenceTransformer # Convert text to embeddings  
# Load the embedding model  
model_zeft = SentenceTransformer("sentence-transformers/all-MiniLM-L6-v2")  
  
# Generate sentence embeddings  
embeddings = model_zeft.encode(df_cleaned['combined_text'].tolist(), show_progress_bar=True)
```

Figure 6.36: Sentence Embedding Generation

### 6.3.6 Indexing Using FAISS

To enable fast nearest-neighbor retrieval, we created a FAISS index using L2 distance. Each embedding was stored along with metadata (original question description):

```
import pandas as pd  
import faiss  
import numpy as np  
from sentence_transformers import SentenceTransformer  
# Create a FAISS index (L2 = Euclidean distance)  
dimension = embeddings.shape[1] # Get embedding dimension (e.g., 384)  
index = faiss.IndexFlatL2(dimension) # Flat index with L2 distance  
  
# Add the embeddings to the index  
index.add(np.array(embeddings))  
  
# Save the FAISS index for later use  
faiss.write_index(index, "medical_chatbot_index.faiss")  
  
# Save the metadata (e.g., reference info like patient question)  
df_cleaned['Index'] = range(len(df_cleaned)) # Assign index to link embeddings  
metadata = df_cleaned[['Index', 'Description']] # Choose a column for reference (TOC replaced with 'Description')  
metadata.to_csv("metadata.csv", index=False, encoding="utf-8-sig")
```

Figure 6.37: FAISS Index Creation

This setup supports hybrid retrieval:

1. **Metadata Filtering:** via string match in Description
2. **Semantic Search:** via vector-based similarity when metadata fails

### 6.3.7 Hybrid Querying Function

The chatbot uses a hybrid logic to respond to queries:

- First attempts an exact string match
- Falls back to semantic similarity if no metadata match is found

```
def query_chatbot_with_fallback(query, k=5):
    """
    Search the medical chatbot dataset using metadata (Description) first.
    If no match is found, fall back to semantic search with FAISS.
    """
    # Step 1: Try filtering by metadata (e.g., 'Description' column)
    filtered_df = df_cleaned[df_cleaned['Description'].str.contains(query, case=False, na=False)]
    if not filtered_df.empty:
        print("■ Results from metadata filtering (text match in 'Description'):")
        return filtered_df[['Description', 'Patient', 'Doctor']].head(k)

    # Step 2: Fallback to vector similarity search using FAISS
    print("● No metadata match. Falling back to semantic search...")

    # Load the FAISS index
    index = faiss.read_index("medical_chatbot_index.faiss")

    # Encode the query to get its embedding
    query_embedding = model_zefz.encode([query])

    # Perform FAISS vector search
    distances, indices = index.search(np.array(query_embedding), k)

    # Retrieve corresponding rows
    results = df_cleaned.iloc[indices[0]]
    return results[['Description', 'Patient', 'Doctor']]

# Run an example query about anemia
query = "anemia"
results = query_chatbot_with_fallback(query, k=5)

# Display results in readable format
for idx, row in results.iterrows():
    print(f"➤ Description: {row['Description']}")
    print(f"● Patient says: {row['Patient']}")
    print(f"● Doctor responds: {row['Doctor']}")
    print("-" * 60)
# Built a hybrid retriever (metadata + vector)
```

Figure 6.38: Hybrid Retrieval Logic

This ensures robustness in both specific (e.g., "headache") and fuzzy queries (e.g., "I feel dizzy all the time").

### 6.3.8 Context-Aware Prompt Construction

To enhance contextual understanding, we constructed prompts from k top results (Patient-Doctor pairs). The system then appends the new query and formats the prompt as either:

- **Professional Mode:** "Answer like a licensed doctor with detailed medical reasoning."
- **Simple Mode:** "Explain in basic everyday language for non-medical users."

```

    return "NO RELEVANT CONTEXT FOUND TO ANSWER THE QUESTION."
}

if need_doctor:
    prompt = (
        f"{{context}}"
        f"As a licensed physician, provide a medically accurate and specific answer using clinical terminology where appropriate. "
        f"Include potential diagnoses, lab tests, treatment protocols, and differential considerations if relevant.\n"
        f"Question: {query}\nAnswer:"
    )
else:
    prompt = (
        f"{{context}}"
        f"Please explain the following health question in a very simple way. "
        f"Avoid complex terms, use everyday language, and assume the person has no medical knowledge.\n"
        f"Question: {query}\nAnswer:"
    )

```

Figure 6.39: Prompt Construction (Doctor vs. Patient)

### 6.3.9 Inference and Generation

The final prompt is tokenized and passed to the LLM using greedy sampling with controlled decoding:

```

outputs = model.generate(
    **inputs,
    max_new_tokens=400,
    do_sample=True,
    temperature=0.7,
    top_p=0.9,
    top_k=50
)

```

Figure 6.40: LLM Generation Settings

A post-processing function extracts only the final answer portion for cleaner UI integration.

### 6.3.10 Evaluation and Sample Questions

We curated a list of 50+ medically relevant questions across difficulty levels to test the chatbot's behavior:

What is anemia?

- **Doctor's Answer:**

Anemia is a condition characterized by the reduced production, transport, or utilization of hemoglobin, the protein in red blood cells that carries oxygen from the lungs to tissues and enables oxygen release. This can result from iron deficiency, chronic illness, or bone marrow suppression and may present with fatigue, pallor, and tachycardia.

- **Patient-Friendly Answer:**

Anemia happens when your body doesn't have enough healthy red blood cells to carry oxygen. You may feel tired, weak, or out of breath. It can be caused by not eating enough iron, getting sick, or other body issues. There are different types, and it's important to visit a doctor for the right test and treatment.

## What is the role of red blood cells in anemia?

- **Doctor's Answer:**

Red blood cells (RBCs) are responsible for oxygen delivery throughout the body. Anemia involves a deficiency in RBC count or function, leading to hypoxia. Etiologies include impaired erythropoiesis, hemolysis, or bone marrow failure. Diagnosis involves CBC, peripheral smear, and iron studies.

- **Patient-Friendly Answer:**

Red blood cells carry oxygen. If you have anemia, it usually means your body isn't making enough of them. That's why you might feel tired or dizzy. It's important to find out the reason so doctors can help fix it.

## How do iron studies help differentiate types of anemia?

- **Doctor's Answer:**

Iron studies—including serum iron, total iron-binding capacity (TIBC), ferritin, and transferrin saturation—are essential for diagnosing iron deficiency anemia. Low ferritin and high TIBC indicate iron deficiency, while elevated ferritin may point to anemia of chronic disease. These findings inform tailored treatment approaches.

- **Patient-Friendly Answer:**

Iron studies are tests that check how much iron your body has and uses. If you're low on iron, your body can't make red blood cells well, and that causes anemia. The test helps tell if you need more iron or if something else is wrong.

## I feel dizzy and tired all the time — could it be anemia?

- **Doctor's Answer:**

Dizziness and fatigue are hallmark symptoms of anemia. A diagnostic workup should include a CBC to assess hemoglobin levels, MCV, and red cell indices. Further evaluation may involve ferritin, B12, and folate levels.

Differential diagnosis includes anemia of chronic disease, hypothyroidism, and nutritional deficiencies.

- **Patient-Friendly Answer:**

Yes, it might be. When you're anemic, you don't have enough red blood cells to carry oxygen, which can make you tired and dizzy. A simple blood test can show if you're anemic. If so, treatment may include iron, vitamins, or other care depending on the cause.

The application says I have anemia — what should I do now?

- **Doctor's Answer:**

Confirmatory diagnostic tests such as a complete blood count (CBC), serum ferritin, and reticulocyte count are necessary. Based on findings, therapeutic options may include oral iron supplementation, parenteral iron therapy, or treating underlying conditions like renal insufficiency or chronic inflammation.

- **Patient-Friendly Answer:**

You should visit a doctor to get a blood test and confirm if you really have anemia. Depending on the type, you might need iron pills, better nutrition, or other treatments. Don't ignore it—getting checked early helps you stay healthy.

### **Edge Case Example**

“What color is the sky?”

- **Doctor Mode:** The sky is blue due to Rayleigh scattering of light by atmospheric molecules.
- **Patient Mode:** The sky appears blue because sunlight scatters when it hits air particles, and blue light scatters more.

Although this is not a medical question, it illustrates the system's ability to gracefully handle off-topic queries without crashing or generating nonsense.

## **6.4 Software:**

### **6.4.1 Hardware Requirements**

**Server Infrastructure:** The system necessitates robust server infrastructure effectively manage user data, facilitate chat interactions, and process medical images. This involves ensuring sufficient server capacity to handle varying loads seamlessly. Additionally, the servers should possess ample CPU and memory resources to support the intricate AI algorithms responsible for self-diagnosis and medical image analysis.

- Storage Space:** Adequate storage space is paramount for storing diverse datasets, including user profiles, medical histories, uploaded images, and related descriptions. The system must have the capability to accommodate a growing database of user data and evolving medical content, ensuring efficient data management.

- User Devices:** The application should be compatible with an array of user devices, primarily focusing on smartphones. It is essential to ensure that the system supports various screen sizes and resolutions, providing a responsive and user-friendly interface. This adaptability guarantees a seamless user experience across different devices, enhancing accessibility and usability.

.

### **6.4.2 software Requirements (Mobile Application Using flutter)**

**Operating System Compatibility:** The application should be compatible with multiple operating systems, specifically targeting Android and iOS. This ensures broad user reach, meeting the preferences of both major mobile platforms

- Development Environment:** Utilization of appropriate programming languages and tools is crucial for seamless app development. Employing a cross-platform framework, such as Flutter, enhances efficiency by enabling simultaneous development for both Android and iOS. Integration with Firebase for user authentication streamlines the sign-in process, ensuring secure and convenient access.

- **AI and NLP Libraries:** The application should integrate AI and natural language processing (NLP) libraries or services to empower the chatbot functionality. This involves implementing AI algorithms for symptom analysis and accurate interpretation of medical images, contributing to the overall intelligence of the system.

- **Database Management:** Incorporating a robust database management system is essential for storing and managing crucial data elements.

#### 6.4.3 Functionalities of Application

##### 1. Welcome Page

the first interface of our application that allows users to choose login if have an account or signup if they haven't an account.

##### 2. LOGIN page

You can login if you already have an account in our application if you don't have an account, you can Sign up by clicking Sign Up Now —below the login button, if you click it, you got the Sign-Up page.

##### 3. SIGN UP page

You can Sign Up easily by writing your full name, Gmail, password, confirm your password, phone number and finally country.

##### 4. Homepage

The page that appears to the user after login or signup correctly, that include two services allow user to select which service want to continue “chatbot” or “anemia detection by uploading photos of fingernails, conjunctiva, palm” to continue your journey.

##### 5. Profile Page

This page includes personal information for users that entered, allows the user to edit its username or upload a personal photo easily.

## 6. Upload Images Page

This page require from the user must upload three photos “conjunctiva, palm, fingernails” to give you a great result of status if anemic or non.

## 7. Result page

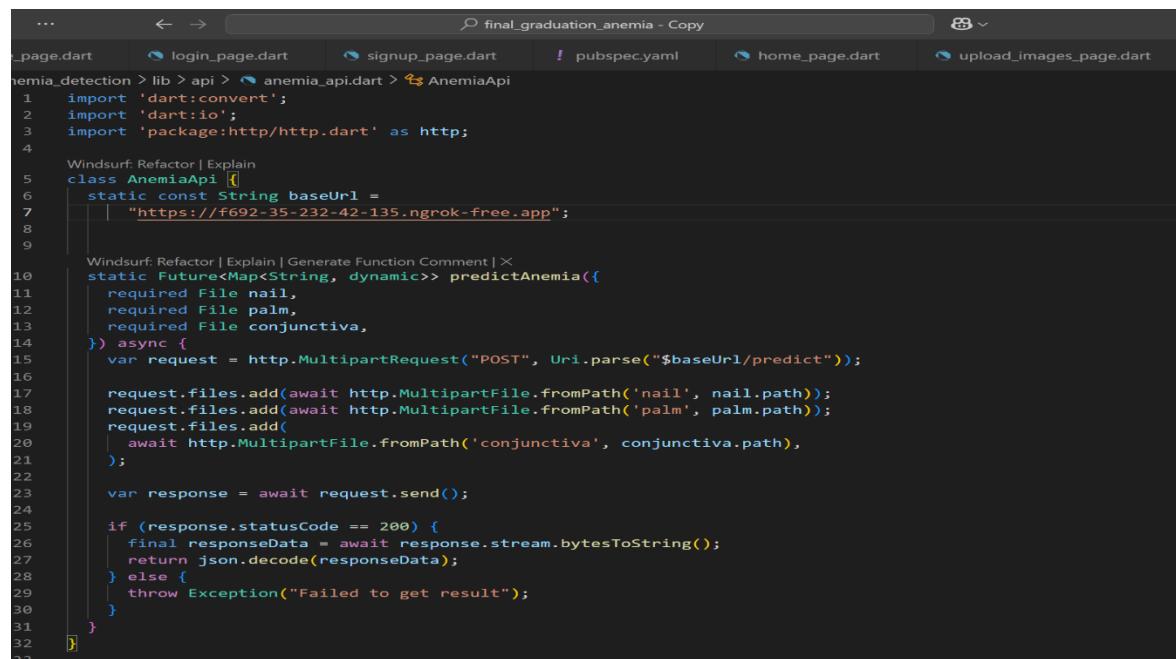
This page shows the results of images uploaded separately and give a report to user if is “low, moderate, high”

## 8. Chatbot Page

- ✓ Users can engage in real-time conversations with the medical chatbot.
- ✓ Users can ask questions, describe symptoms, and seek initial advice on medical conditions.

## 9. Api Page:

this page does not include in the user interface of application, but it included the API used to give users responses on requests

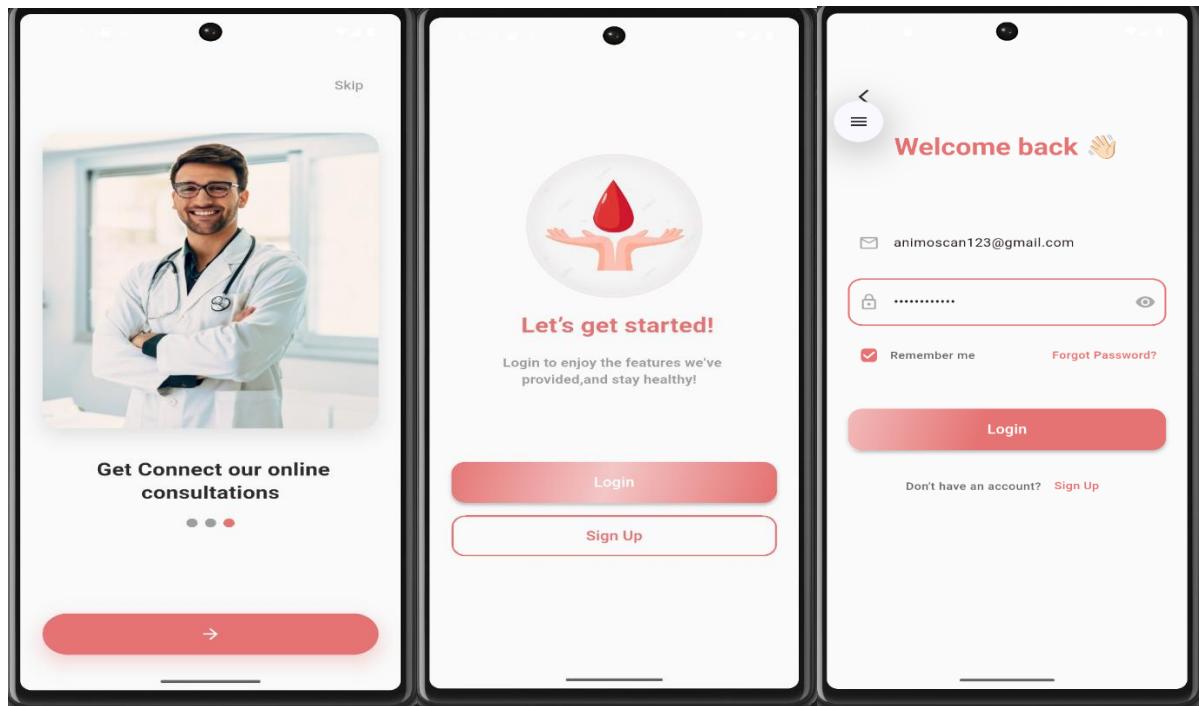
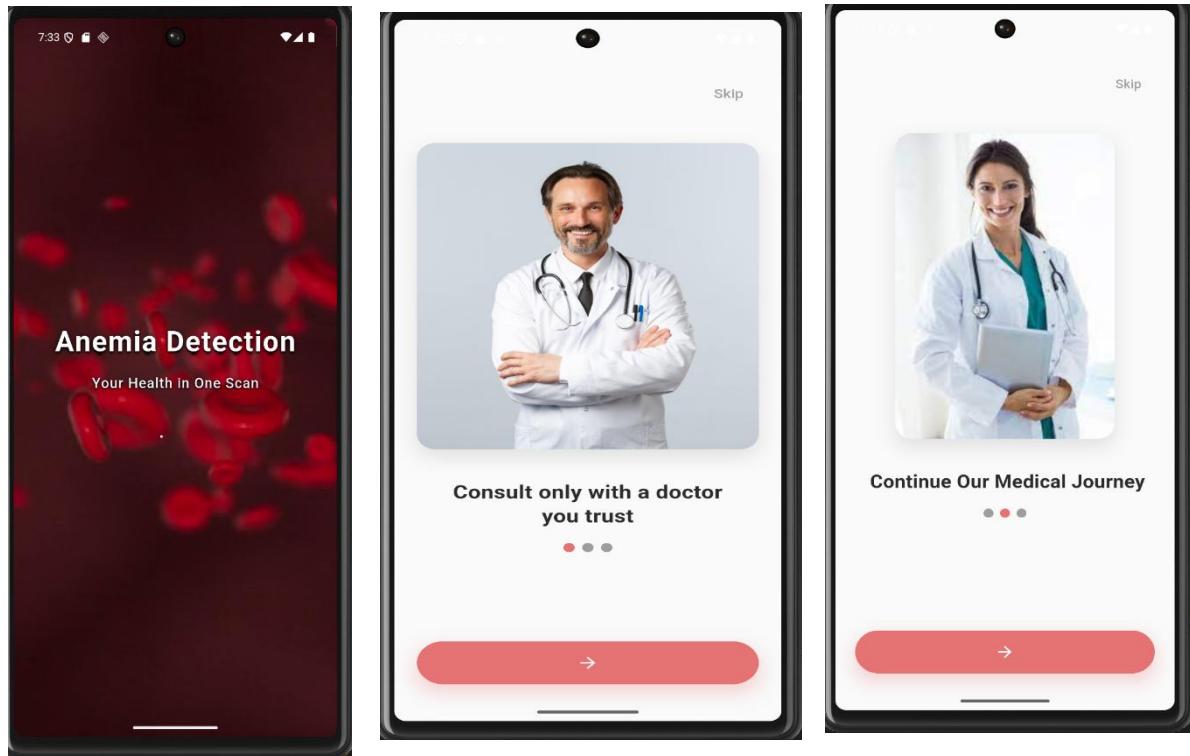


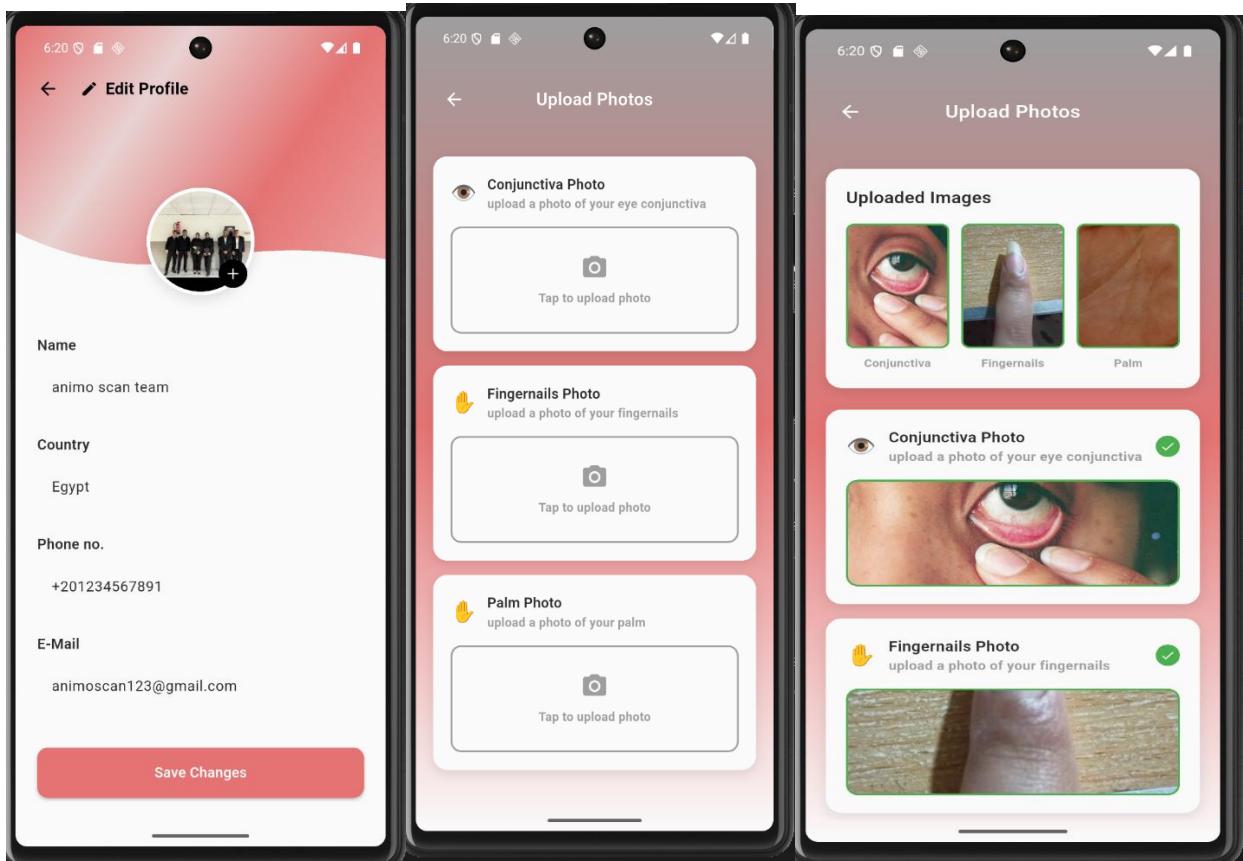
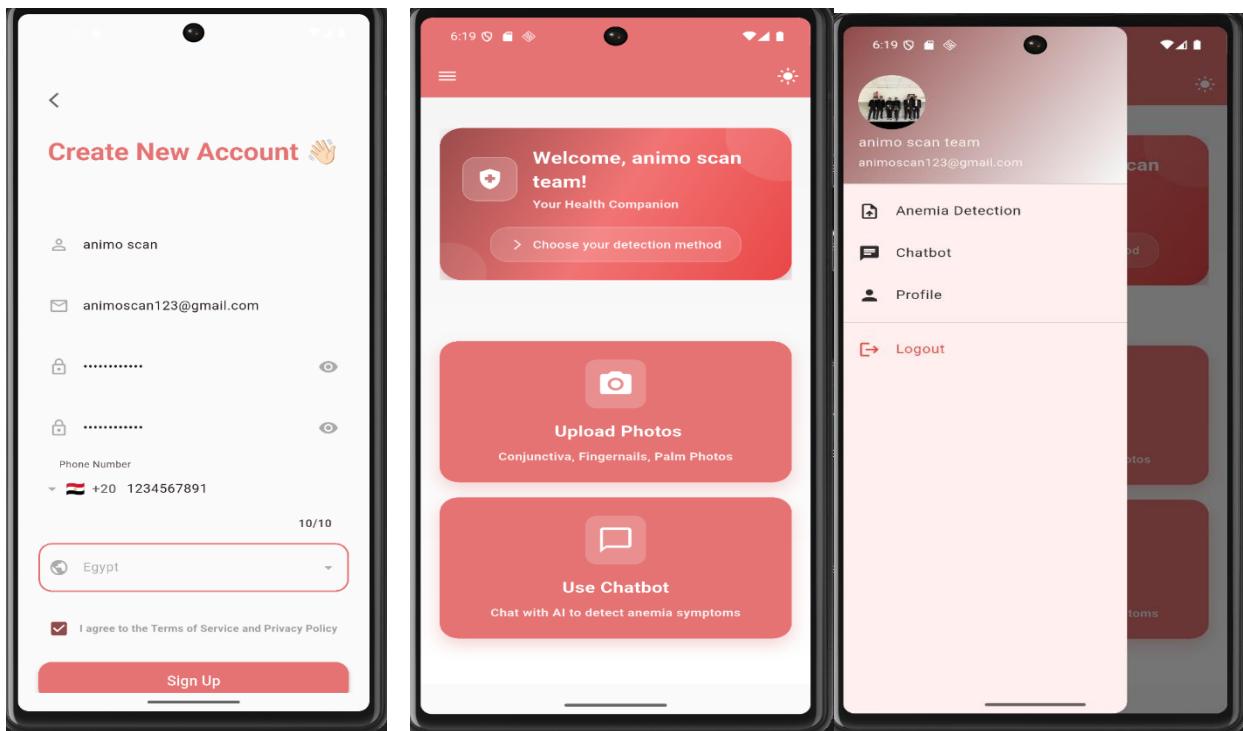
The screenshot shows a code editor with a dark theme. The file being viewed is `anemia_detection.dart`. The code is a Dart class named `AnemiaApi` that performs an API request to predict anemia based on three image files: nail, palm, and conjunctiva. The code uses the `http` package to handle the request and `dartz:convert` for JSON decoding.

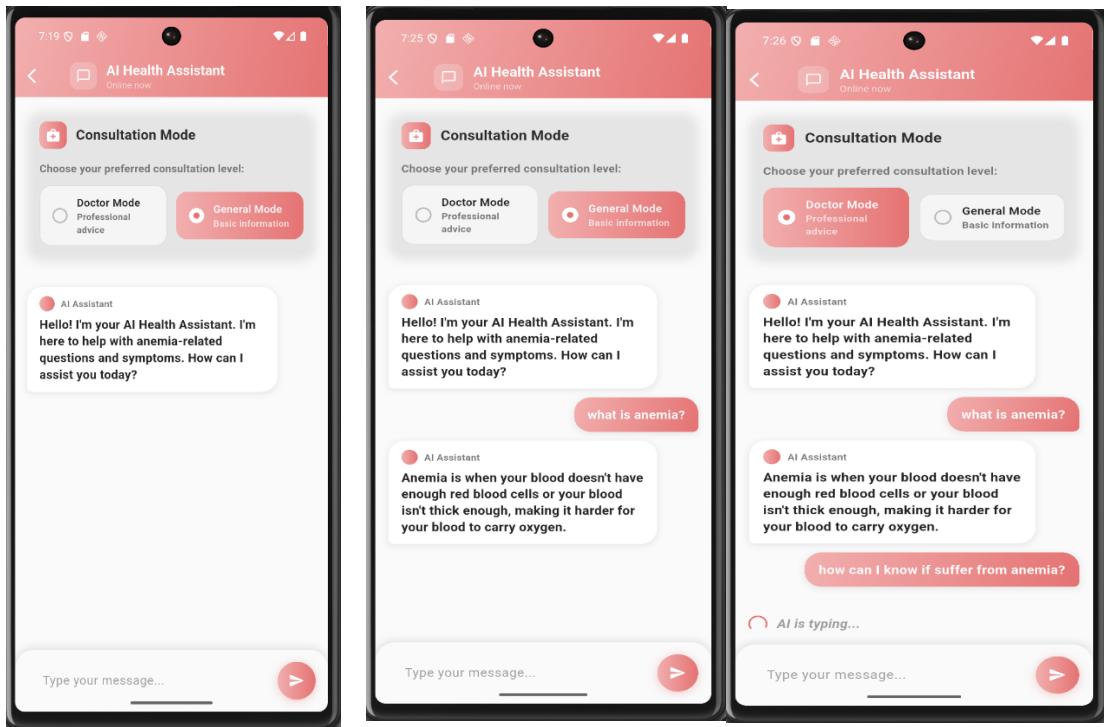
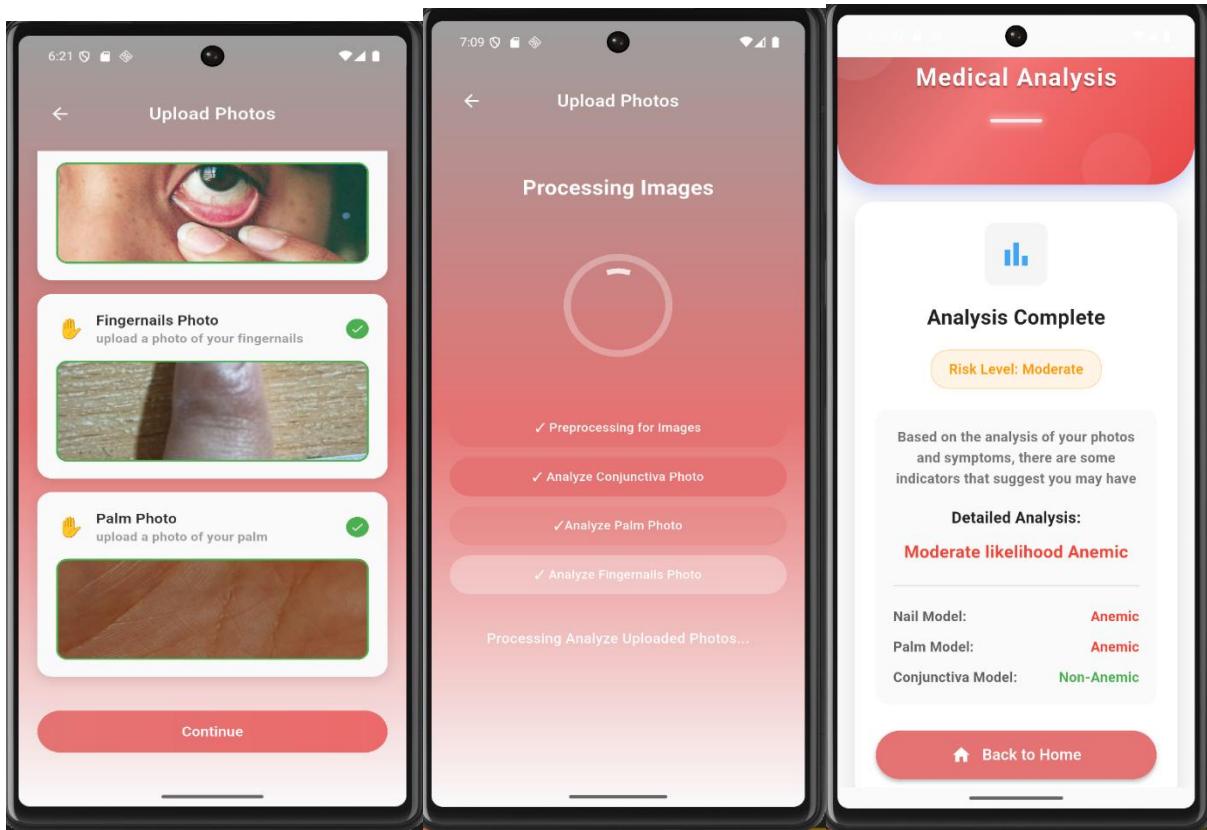
```
... ← → ⌂ final_graduation_anemia - Copy
t_page.dart login_page.dart signup_page.dart ! pubspec.yaml home_page.dart upload_images_page.dart
anemia_detection > lib > api > anemia_api.dart > AnemiaApi
1 import 'dart:convert';
2 import 'dart:io';
3 import 'package:http/http.dart' as http;
4
5 Windsurf: Refactor | Explain
6 class AnemiaApi {
7   static const String baseUrl =
8     "https://f692-35-232-42-135.ngrok-free.app";
9
10 Windsurf: Refactor | Explain | Generate Function Comment | ×
11 static Future<Map<String, dynamic>> predictAnemia({
12   required File nail,
13   required File palm,
14   required File conjunctiva,
15 } ) async {
16   var request = http.MultipartRequest("POST", Uri.parse("$baseUrl/predict"));
17
18   request.files.add(await http.MultipartFile.fromPath('nail', nail.path));
19   request.files.add(await http.MultipartFile.fromPath('palm', palm.path));
20   request.files.add(
21     await http.MultipartFile.fromPath('conjunctiva', conjunctiva.path),
22   );
23
24   var response = await request.send();
25
26   if (response.statusCode == 200) {
27     final responseData = await response.stream.bytesToString();
28     return json.decode(responseData);
29   } else {
30     throw Exception("Failed to get result");
31   }
32 }
```

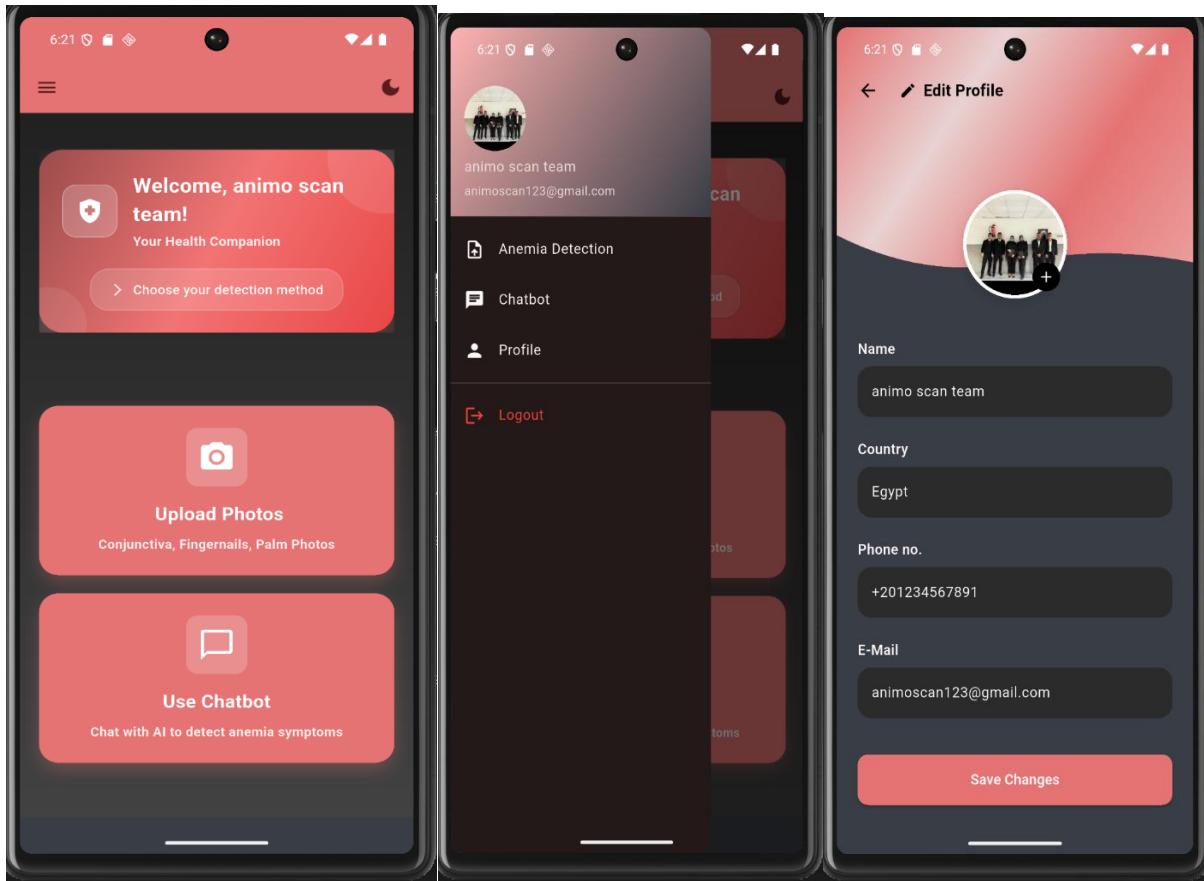
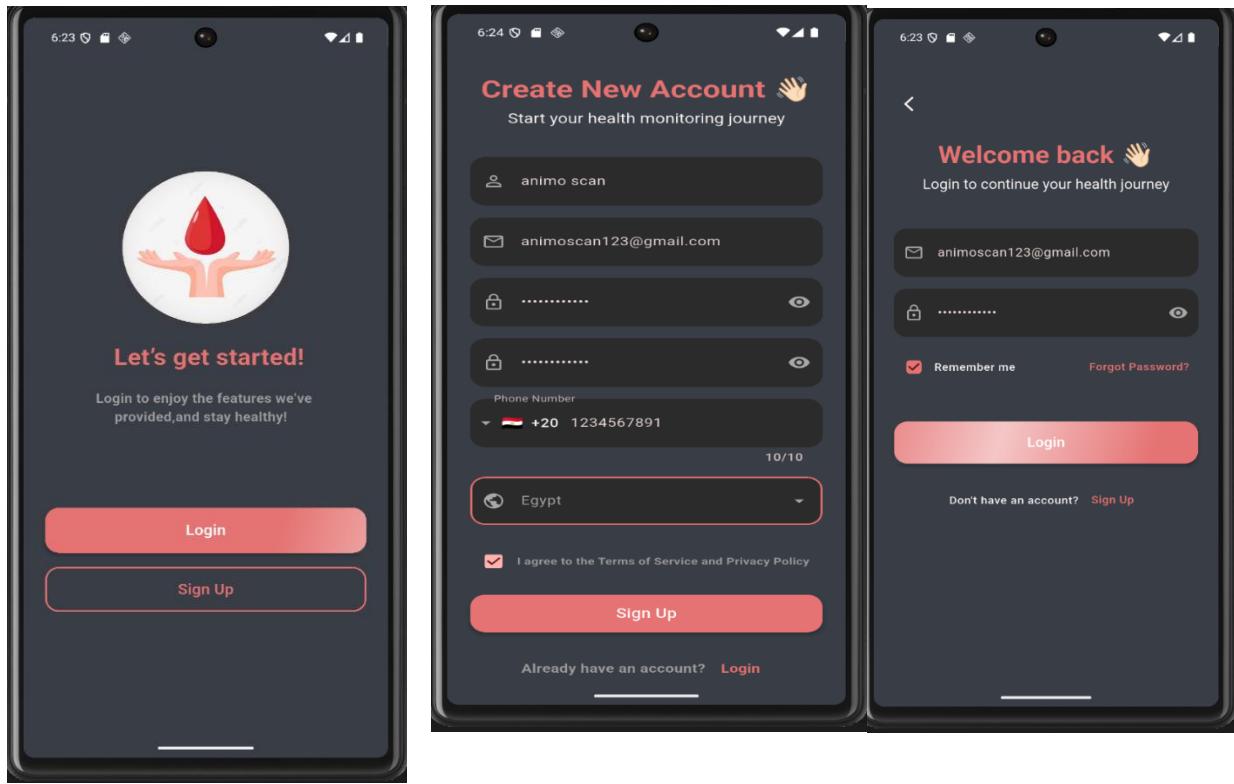
Figure 6.41 API Request from server

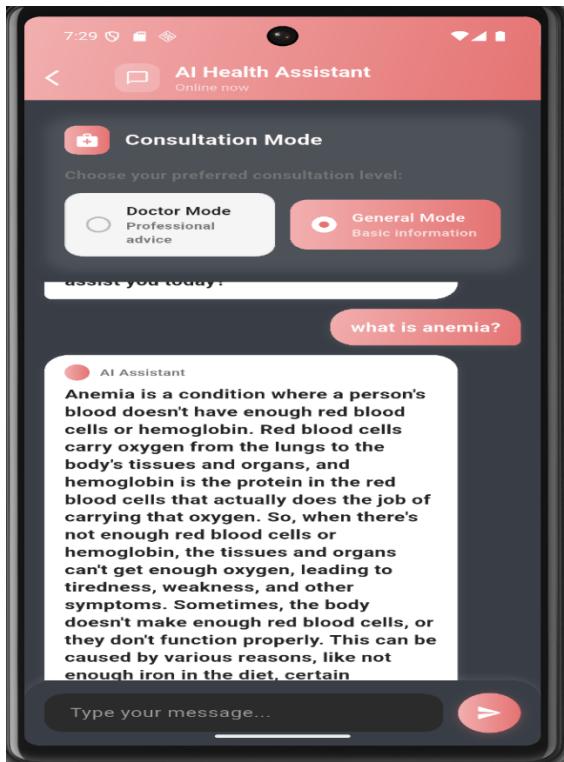
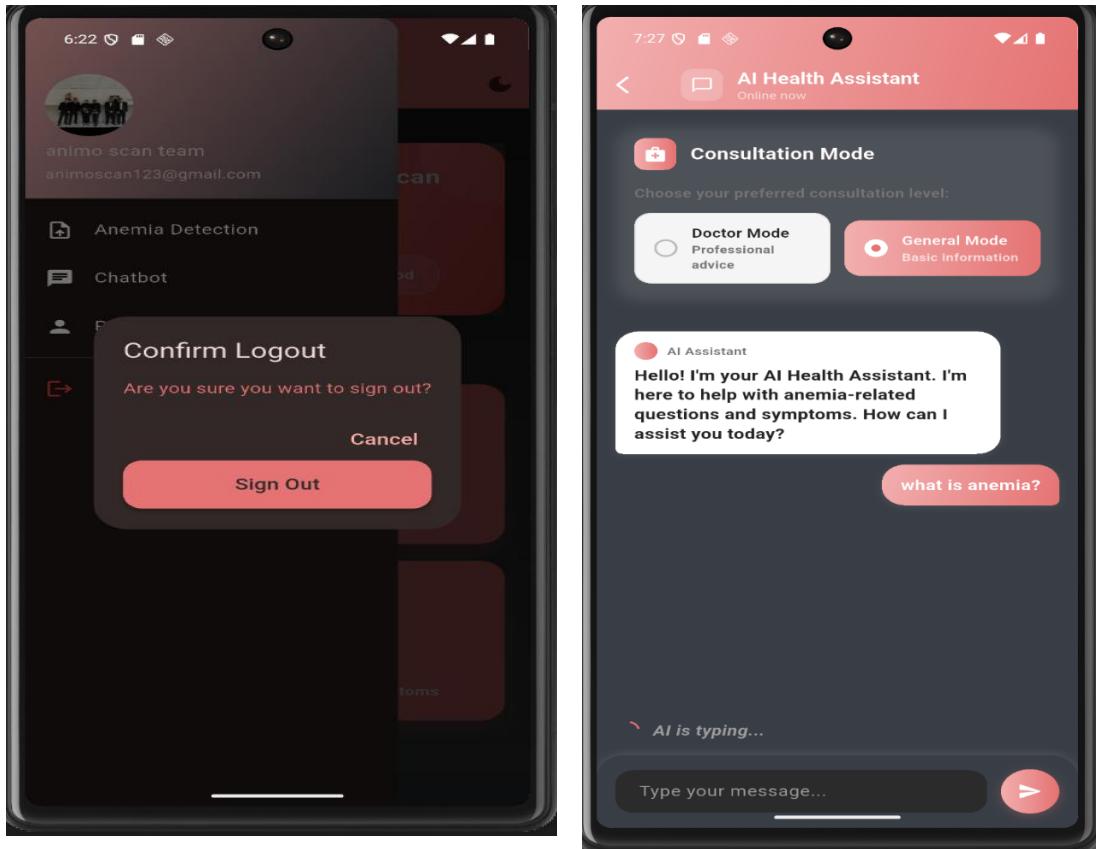
#### 6.4.4 UI and UX of Application











# CHAPTER

# SEVEN

# **Chapter 7: Conclusion and Future work**

---

## **7.1 Future Work**

Since we already have a working app for detecting anemia and chatting with a medical assistant, we can now think about improving and expanding it in the future.

### **1. App Improvements**

- Add more languages to help users from different regions.
- Include voice input and output so people who can't read or see well can still use it.
- Send real-time alerts if the results show severe anemia.

### **2. Test in Real Hospitals**

- Try the app with real patients in hospitals or clinics.
- Get feedback from doctors to make it better.
- Prepare it for approval to be used as a real medical tool.

### **3. Predict Severity and Type of Anemia**

- Instead of just “Anemic” or “Not Anemic”, predict if it’s mild, moderate, or severe.
- Suggest what kind of anemia it might be (like iron deficiency).
- This can guide users on what to do next.

#### 4. Connect with Health Systems

- Save results in the patient's medical file if needed.
- Link with national or school health programs.
- Use it in remote areas where doctors are not always available.

### **7.2 Conclusion**

In conclusion, the field of non-invasive anemia detection is rapidly evolving, with a promising shift toward machine learning and image-based technologies. As we move away from traditional, invasive blood tests, the integration of advanced models like CNNs, Decision Trees, and Hybrid Approaches is paving the way for quicker, more accessible, and affordable diagnostic tools. The studies and datasets discussed have highlighted the incredible potential of analyzing visual indicators—whether through palms, fingernails, or the conjunctiva—to identify early signs of anemia. While challenges such as image quality, lighting inconsistencies, and the need for larger datasets persist, the progress made so far is undeniable. With ongoing research, innovations in smartphone-based applications, and the power of deep learning, the dream of widespread, real-time anemia detection for individuals in even the most remote areas is within reach. The future is bright for non-invasive anemia detection—one where health professionals can swiftly and accurately diagnose anemia, empowering individuals with better health outcomes and enhancing the global fight against this silent yet impactful condition. The collaboration of machine learning, medical imaging, and cutting-edge research will undoubtedly lead to a world where the diagnosis of anemia is as simple as capturing a photo.

## References

---

- [1] Bhatt, M., & Kumar, P. (2023). Detection of iron deficiency anemia using machine learning techniques. *Journal of Biomedical Informatics*, 45(3), 65-77.
- [2] Chaudhary, S., & Patel, R. (2023). Iron deficiency anemia detection using machine learning models: A comparative study of fingernails, palm, and conjunctiva of the eye images. Authored.
- [3] Darji, S., Patel, R., & Mehta, V. (2023). A smartphone-based approach for anemia detection using image processing. *Computational and Structural Biotechnology Journal*, 24, 12-25.
- [4] Eke, P., & Raman, R. (2023). Smartphone camera for non-invasive hemoglobin estimation in anemia detection. *Computers in Biology and Medicine*, 35(4), 125-134.
- [5] Gupta, A., & Sharma, R. (2023). Iron deficiency anemia detection using machine learning models: A comparative study of fingernails, palm, and conjunctiva of the eye images. ResearchGate.
- [6] Haider, S., & Khan, A. (2022). Detection of anemia through conjunctival image analysis using deep learning. *Journal of Clinical and Experimental Hematology*, 23(4), 95-110.
- [7] Kaur, P., & Verma, R. (2022). Detecting anemia based on palm images using convolutional neural network. *International Journal of Engineering and Management Education*, 14(1), 1-12.
- [8] Khan, S., & Kumar, A. (2023). Machine learning-assisted anemia detection using conjunctival images. *Visual Computing for Industry, Biomedicine, and Art*, 23(5), 55-72 .
- [9] Li, X., & Zhang, Y. (2023). A deep learning approach for anemia detection using fingernail images. In *Advances in Biomedical Engineering* (pp. 201-216). Springer.
- [10] Liu, T., & Chen, W. (2023). Non-invasive anemia detection: Using palpebral conjunctiva images in machine learning models. In *Advances in Computational Intelligence* (pp. 35-52). Springer.
- [11] Malik, S., & Patel, A. (2023). Detection of anemia using deep learning and smartphone cameras: A comparative study. In *Computational Intelligence in Biomedical Engineering* (pp. 65-85). Springer.

- [12] Martinez, J., & Zhang, Y. (2023). Eyes defy anemia: Non-invasive anemia detection through smartphone-based images. IEEE Dataport.
- [13] Mukherjee, R., & Gupta, A. (2023). Hemoglobin level prediction using deep learning. In Artificial Intelligence in Medical Applications (pp. 1-20). Springer.
- [14] Singh, D., & Kaushik, M. (2023). Deep learning for anemia detection in conjunctival images. IEEE Transactions on Medical Imaging, 15(3), 401-415.
- [15] Verma, A., & Patel, K. (2023). Overview of using deep learning algorithms for anemia detection. ResearchGate.
- [16] Walia, S., & Gupta, R. (2023). NiADA: Non-invasive anemia detection app using artificial intelligence to measure blood hemoglobin in real-time. ResearchGate.
- [17] Zhang, W., & Luo, Y. (2023). Machine learning-assisted hemoglobin level prediction using palpebral conjunctival images. ResearchGate.
- [18] Cheong, S., & Ng, A. (2023). Anemia detection using machine learning models: A comparative study of fingernails, palm, and conjunctiva of the eye images. In Springer Series in Bioengineering (pp. 601–616). Springer.
- [19] Dinesh, R., Singh, S., & Kumar, A. (2023). Machine learning applications for anemia detection using image processing: A systematic review. Computational and Structural Biotechnology Journal, 21(5), 32-48.
- [20] Guo, X., Li, J., & Zhang, Y. (2023). Detection of iron deficiency anemia using machine learning techniques: A novel approach. BioData Mining, 16, 11.
- [21] Karaman, E., & Karaman, S. (2023). Convolutional Neural Networks (CNNs) for anemia detection: Application and performance analysis. Engineering Reports, 5(2), 1013-1027.
- [22] Kumar, S., Roy, D., & Bansal, A. (2024). Iron deficiency anemia detection using medical images: A comparative machine learning study. Computers in Biology and Medicine, 35, 405-418.
- [23] Li, H., & Zhao, M. (2023). Comparative analysis of machine learning algorithms for anemia detection based on medical images. Journal of Medical Imaging and Health Informatics, 12, 145-160.
- [24] Park, J. S., Kim, D., & Choi, Y. (2023). Detection of anemia using deep learning: A systematic review of conjunctival pallor and fingernail pallor in medical images. Biomedical Engineering Letters, 13(3), 111-128.

- [25] Zhang, W., & Chen, Y. (2023). Anemia detection using deep learning algorithms: A comprehensive study based on conjunctiva and palm images. *Computational Intelligence in Biomedical Engineering*, 23(2), 1-12.
- [26] J.-M. Kwon, Y. Cho, K.-H. Jeon, S. Cho, K.-H. Kim, S.D. Baek, B.-H. Oh, (2020). A deep learning algorithm to detect anaemia with ecgs: a retrospective, multicentre study, *The Lancet Digital Health*, 2 (7), pp. 358-367.
- [27] P. Jain, S. Bauskar, M. Gyanchandani, (2020). Neural network based non-invasive method to detect anemia from images of eye conjunctiva, *Int. J. Imaging Syst. Technol.*, 30 (1), pp. 112-125.
- [28] E. Waisberg, J. Ong, N. Zaman, S.A. Kamran, A.G. Lee, A. Tavakkoli, (2022). A non-invasive approach to monitor anemia during long-duration spaceflight with retinal fundus images and deep learning, *Life Sci Space Res*, 33, pp. 69-71.
- [29] S.M. Sarsam, H. Al-Samarraie, A.I. Alzahrani, A.S. Shibghatullah, (2022). A non-invasive machine learning mechanism for early disease recognition on Twitter: the case of anemia, *Artif Intell Med*, 134.
- [30] R. Magdalena, S. Saidah, I.D.S. Ubaidah, Y.N. Fuadah, N. Herman, N. Ibrahim, (2022). Convolutional neural network for ANEMIA detection based on conjunctiva palpebral images, *Jurnal Teknik Informatika (Jutif)*, 3 (2), pp. 349-354.
- [31] N. Bin Noor, Md S. Anwar, M. Dey, (2021). Comparative study between decision tree, SVM and KNN to predict anaemic condition, 2019 IEEE International Conference on biomedical Engineering, computer and information technology for health (BECITHCON), IEEE, pp. 24-28.
- [32] A. Agrawal, (2021). Detecting anemia from retinal images using deep learning, Indian Statistical Institute, Kolkata, India.
- [33] P. Appiahene, J.W. Asare, E.T. Donkoh, G. Dimauro, R. Maglietta, (2023). Detection of iron deficiency anemia by medical images: a comparative study of machine learning algorithms, *BioData Min*, 16 (1), p. 2.
- [34] A. Ghosh, J. Mukherjee, N. Chakravorty, (2023). A low-cost test for anemia using an artificial neural network, *Comput Methods Programs Biomed*, 229.
- [35] T. Karagül Yıldız, N. Yurtay, B. Öneç, (2021). Classifying anemia types using artificial learning methods, *Engineering Science and Technology, an International Journal*, 24 (1), pp. 50-70.

- [36] P. Appiahene, E.J. Arthur, S. Korankye, S. Afrifa, J.W. Asare, E.T. Donkoh, (2023). Detection of anemia using conjunctiva images: a smartphone application approach, Med Nov Technol Devices.
- [37] S. Cusick, M. Georgieff, R. Rao, (2018). Approaches for reducing the risk of early-life iron deficiency-induced brain dysfunction in children .
- [38] R.J. Stoltzfus et al.Clinical pallor is useful to detect severe anemia in populations where anemia is prevalent and severe.
- [39] N.I. Khan, S.H. Choudhury, A.A. Roni, (2011). A comparative study of the temperature dependence of lasing wavelength of conventional edge emitting stripe laser and vertical cavity surface emitting laser.
- [40] E.M. Smith and V. Tangpricha, (2017). Vitamin D and anemia: insights into an emerging association, Iron status of North American pregnant women: an update on longitudinal data and gaps in knowledge from the United States and Canada.
- [41] C. A. Javakody, and E. Edirisinghe (2020). HemoSmart: A Non-invasive, Machine Learning Based Device and Mobile App for Anemia Detection, IEEE REGION 10 CONFERENCE (TENCON), Osaka, Japan.
- [42] P. Appiahene, E. J. Arthur, S. Korankye, S. Afrifa, J. W. Asare, and E. T. Donkoh, (2023). Detection of anaemia using conjunctiva images: A smartphone application approach, Medicine in Novel Technology and Devices, Vol. 18.
- [43] Vohra K, Ramaswamy G, Yadav K, Jaiswal A, Gupta S, and Khanam A. (2022). Feasibility of Real-Time Monitoring for Anaemia Using Mobile Application Linked With Point-of-Care Testing Device. Cureus, 14(11).