Project Progress Report:
AI Agent for Acute Lymphoblastic Leukemia Diagnosis

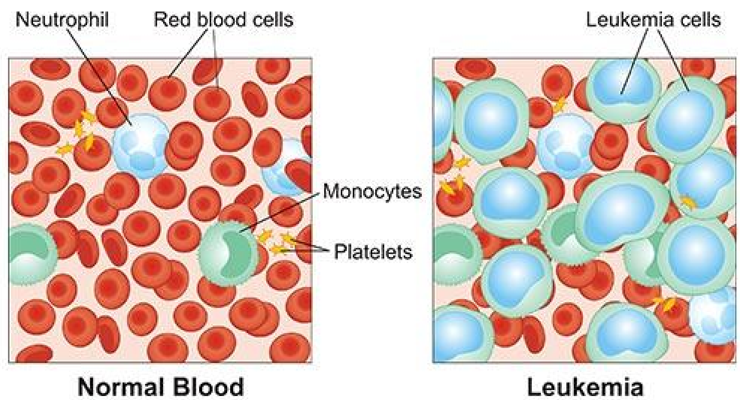
Nexus

Supervisor : Ms. Kushani Bandara
Group: 21
Cluster: 21.1

Members :
MJM.Amajith 721434768
MN.Fathima Rafaha 121428697
MMZ. Zahra 221445832
MHA. Ahmed 321422851

Introduction + Background

Acute Lymphoblastic Leukemia (ALL) is a type of cancer that affects the blood and bone marrow. It primarily occurs in children but can also affect adults. ALL is characterized by the uncontrolled growth of lymphoid cells, which are responsible for producing white blood cells.



The diagnosis of ALL is a complex and challenging process that requires careful examination of blood and bone marrow samples. Traditionally, this diagnosis has relied on the expertise of skilled hematopathologists who visually inspect and classify cells under a microscope. However, this manual approach is time-consuming, subjective, and can be prone to human error.

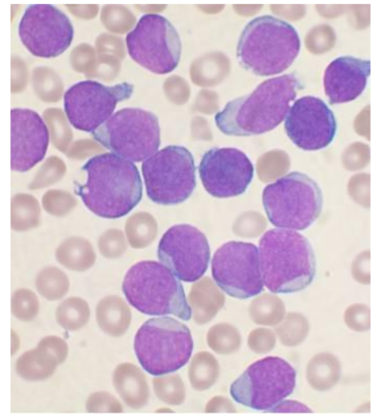
With recent advancements in artificial intelligence (AI) and machine learning, there is a tremendous opportunity to enhance the accuracy and efficiency of ALL diagnosis. By leveraging AI algorithms, we can develop a sophisticated AI agent that can assist medical professionals in accurately identifying and classifying leukemia cells.

The primary objective of this project is to develop an AI agent capable of accurately diagnosing Acute Lymphoblastic Leukemia. By leveraging deep learning techniques and analyzing a large dataset of leukemia samples, the AI agent will learn to identify and classify abnormal cells with a high degree of accuracy.

Through this project, we aim to bridge the gap between medical expertise and cutting-edge AI technology, ultimately enhancing the accuracy, speed, and objectivity of Acute Lymphoblastic Leukemia diagnosis. By combining the power of AI with the knowledge and experience of medical professionals, we can make significant strides in improving leukemia diagnosis and contribute to advancements in cancer care.

In the following sections of this proposal, we will delve into the existing systems for leukemia diagnosis, our proposed solution using an AI agent, the technologies we plan to utilize, and the project timeline. By the end of this endeavor, we anticipate the development of a robust AI agent that will have a transformative impact on Acute Lymphoblastic Leukemia diagnosis.

Problem Statement



Acute Lymphoblastic Leukemia (ALL) diagnosis is a complex and time-consuming process that relies heavily on the expertise of hematopathologists. The traditional manual examination of blood and bone marrow samples for ALL diagnosis is subjective and prone to human error. This approach often leads to delayed diagnosis and potential misclassification of leukemia cells, impacting patient outcomes.

Project Objectives

The primary objective of this project is to develop an AI agent capable of accurately diagnosing Acute Lymphoblastic Leukemia. The AI agent will leverage deep learning techniques to analyze a large dataset of leukemia samples and identify abnormal cells with a high degree of accuracy. By automating the diagnosis process, the project aims to address the following objectives:

- **Improve Accuracy**: The AI agent will be trained to recognize subtle patterns and features in leukemia cells that may not be easily detectable by human observers. By leveraging the power of AI algorithms, the project aims to enhance the accuracy of ALL diagnosis, leading to more reliable and consistent results.
- **Increase Efficiency**: The current manual process of leukemia diagnosis is time-consuming and labor-intensive. By developing an AI agent that can quickly analyze and classify leukemia cells, the project aims to significantly reduce the time required for diagnosis, allowing healthcare professionals to make timely treatment decisions.
- **Assist Medical Professionals**: The AI agent will serve as a supportive tool for hematopathologists, augmenting their expertise and providing them with additional insights during the diagnostic process. By assisting in the identification and classification of leukemia cells, the project aims to enhance the capabilities of medical professionals in making accurate and informed diagnoses.
- **Enhance Patient Outcomes**: Early and accurate diagnosis of Acute Lymphoblastic Leukemia is crucial for developing appropriate treatment plans and improving patient outcomes. By developing an AI agent that can aid in early detection and diagnosis, the project aims to contribute to better patient care and potentially save lives.

By achieving these project objectives, we aim to revolutionize the field of leukemia diagnosis by combining the power of AI with the expertise of medical professionals. The proposed AI agent has the potential to significantly impact the accuracy, efficiency, and effectiveness of Acute Lymphoblastic Leukemia diagnosis, ultimately leading to improved patient outcomes.

Introduction to Similar Type of Applications

In the real world, there are several implemented computer applications that utilize artificial intelligence (AI) techniques for the diagnosis of various diseases, including cancer. These applications have made significant advancements in improving diagnostic accuracy and assisting healthcare professionals in their decision-making process. Some examples of implemented AI applications in the medical field are:

- Aidoc: Aidoc is an AI-powered radiology platform that focuses on detecting abnormalities in medical imaging scans. It employs deep learning algorithms to analyze computed tomography (CT) scans and flag potential abnormalities, aiding radiologists in the diagnosis of conditions such as intracranial hemorrhage, pulmonary embolism, and cervical spine fractures.

- Tempus: Tempus is an AI-based healthcare platform that integrates molecular data, clinical data, and imaging data to provide personalized treatment options for cancer patients. It utilizes machine learning algorithms to analyze large datasets and identify patterns that can inform treatment decisions and improve patient outcomes.

- Proscia: Proscia is a digital pathology company that applies AI algorithms to digitized histopathology slides. Their platform, Concentriq, enables pathologists to analyze and interpret slides more efficiently and accurately. By automating routine tasks and assisting in the identification of cancerous cells, Proscia's AI-powered solution enhances the speed and accuracy of pathology diagnosis.

These implemented AI applications in the medical field demonstrate the potential and benefits of using AI for disease diagnosis. While not specific to Acute Lymphoblastic Leukemia (ALL), they showcase the successful integration of AI into medical practice and provide valuable insights for developing a similar AI agent for ALL diagnosis.

By leveraging the advancements made in these implemented applications, we aim to create an effective and reliable AI agent tailored specifically for the diagnosis of Acute Lymphoblastic Leukemia. This agent will utilize AI algorithms, deep learning techniques, and domain-specific data to enhance the accuracy and efficiency of ALL diagnosis, contributing to improved patient care and outcomes.

Proposed Solution

This project aims to develop an AI agent that will significantly improve the accuracy and efficiency of diagnosing Acute Lymphoblastic Leukemia (ALL). This AI agent will leverage advanced machine learning techniques and domain-specific knowledge to analyze leukemia cells and assist hematopathologists in their diagnostic process.

Automated Cell Analysis

The AI agent will employ computer vision algorithms to automatically analyze leukemia cells from blood and bone marrow samples. It will utilize deep learning techniques, such as convolutional neural networks (CNNs), to extract meaningful features and patterns from the cell images. By analyzing these features, the AI agent will be able to differentiate between normal and abnormal cells, specifically those associated with ALL.

Enhanced Accuracy and Efficiency

The AI agent will significantly enhance the accuracy of ALL diagnosis by leveraging its ability to analyze large quantities of cells quickly and accurately. It will reduce the potential for human error and variability in manual analysis, leading to more reliable and consistent results. Moreover, the AI agent's efficiency will allow hematopathologists to process a higher volume of samples in less time, leading to faster diagnosis and treatment decisions.

Integration of Clinical Data

In addition to cell analysis, the AI agent will integrate relevant clinical data into the diagnostic process. It will consider patient information, such as age, gender, medical history, and laboratory test results, to provide a comprehensive assessment of ALL. By combining cell analysis with clinical data, the AI agent will offer a holistic approach to diagnosis, enabling hematopathologists to make more informed decisions.

Interpretability and Explainability

To build trust and facilitate collaboration between the AI agent and hematopathologists, the proposed solution will prioritize interpretability and explainability. The AI agent will provide explanations and visualizations of its analysis, highlighting the key features and patterns that contribute to its diagnostic decisions. This transparency will enable hematopathologists to understand and validate the agent's reasoning, fostering trust and confidence in its capabilities.

Clinical Integration and Validation

The proposed AI agent will undergo rigorous validation and integration into clinical workflows. It will be tested on a large and diverse dataset, consisting of samples from different populations and healthcare settings. The performance of the AI agent will be compared against the diagnoses made by expert hematopathologists to validate its accuracy and reliability.

Technology Planning to Use

In this section, we will discuss the backend and frontend technologies that are planned to be used in the development of the AI agent for Acute Lymphoblastic Leukemia (ALL) diagnosis.

Backend Technologies

For the backend development, the project will utilize Django, a high-level Python web framework. Django provides a robust and efficient framework for handling user authentication, database management, and server-side logic. It offers various features such as an Object-Relational Mapper (ORM), URL routing, and templating system, which will simplify the development process and enhance the scalability of the application. In terms of the database management system, MySQL will be employed as the backend database for storing the necessary data, including user information, patient records, and training data. MySQL is a widely used and reliable relational database management system that offers excellent performance and scalability.

Additionally, Django provides seamless integration with MySQL through its built-in support for different database backends, allowing efficient data retrieval and manipulation.

Frontend Technologies

For the frontend development, a combination of HTML, CSS, and JavaScript will be used to create an intuitive and user-friendly interface. HTML (Hypertext Markup Language) will be used for structuring the webpages, while CSS (Cascading Style Sheets) will be used for styling and layout. To enhance the interactivity and user experience, JavaScript, along with popular frameworks such as React or Vue.js, can be utilized. These frameworks provide powerful tools for building dynamic and responsive web interfaces, allowing seamless communication with the backend and providing real-time updates.

Additionally, frontend design tools such as Adobe XD, Figma, or Sketch can be used to create mockups and design prototypes, ensuring a visually appealing and user-friendly interface.

TensorFlow for Model Training

For the model training phase, TensorFlow, an open-source machine learning framework, will be used. TensorFlow provides a comprehensive ecosystem of tools, libraries, and resources for developing and training deep learning models. With TensorFlow, you can efficiently build and train neural network models for image classification, including convolutional neural networks (CNNs) tailored for analyzing leukemia cells. TensorFlow offers extensive documentation, pre-trained models, and utilities for data preprocessing and model evaluation, making it a suitable choice for developing the AI agent.

Project Progress

Planning and Requirements:

- Gathered requirements from medical experts, pathologists, and other stakeholders by asking questions and hosting interviews to understand the system's functionalities and performance expectations. **(Completed)**
- Created user classes to capture different aspects of the diagnosis system, such as data input, image analysis, reporting. **(Completed)**
- Prioritized the user classes based on their criticality and potential impact on the diagnosis process. **(Completed)**

Design:

- Developed a high-level architectural design outlining the system components, such as the user interface, image processing module, database, and reporting module. **(Completed)**

Development:

- Implementing the diagnosis system's frontend, providing an intuitive user interface for medical professionals to interact with the system. **(WIP)**
- Developing the backend components, including the image processing algorithms, data storage, and integration with the model. **(WIP)**
- Developing the model components for classification. **(WIP)**

Testing:

- Conduct unit tests for individual components, verifying the accuracy and functionality of the image processing algorithms and backend functionalities. **(TBA)**
- Perform integration tests to assess the system's interactions and data flow between various components. **(TBA)**
- Collaborate with medical experts to conduct extensive validation and testing of the system using real patient data and simulated scenarios. **(TBA)**

Deployment:

- Plan the deployment of the diagnosis system in a controlled environment, such as a pilot hospital or clinical research setting, to gather feedback and validate its performance. **(TBA)**
- Monitor the system during deployment to identify and address any issues that may arise. **(TBA)**

Feedback and Evaluation:

- Hold regular meetings with medical professionals and stakeholders to gather feedback on the system's usability, accuracy, and efficiency. **(TBA)**
- Continuously improve the diagnosis system based on feedback and address any issues identified during the deployment phase. **(TBA)**

- Conduct periodic evaluations to ensure the system's performance aligns with the desired outcomes and objectives. **(TBA)**

Project Timeline

The following timeline outlines the estimated duration for each phase of the project

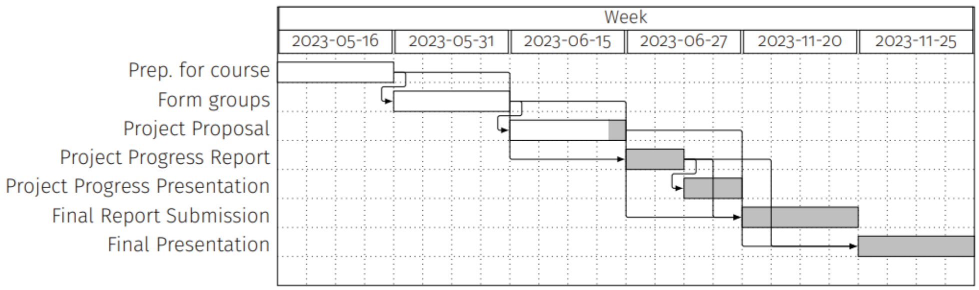


Figure 1: Gantt Chart

Please note that the timeline provided is an estimate and may be subject to change based on project requirements, resource availability, and unforeseen challenges that may arise during the development process.

Conclusion

In conclusion, this report has presented a project proposal for developing an AI agent for Acute Lymphoblastic Leukemia (ALL) diagnosis. The proposed solution aims to leverage advanced machine learning techniques, including deep learning and computer vision, to automate cell analysis, enhance accuracy and efficiency, integrate clinical data, and enable continuous learning.

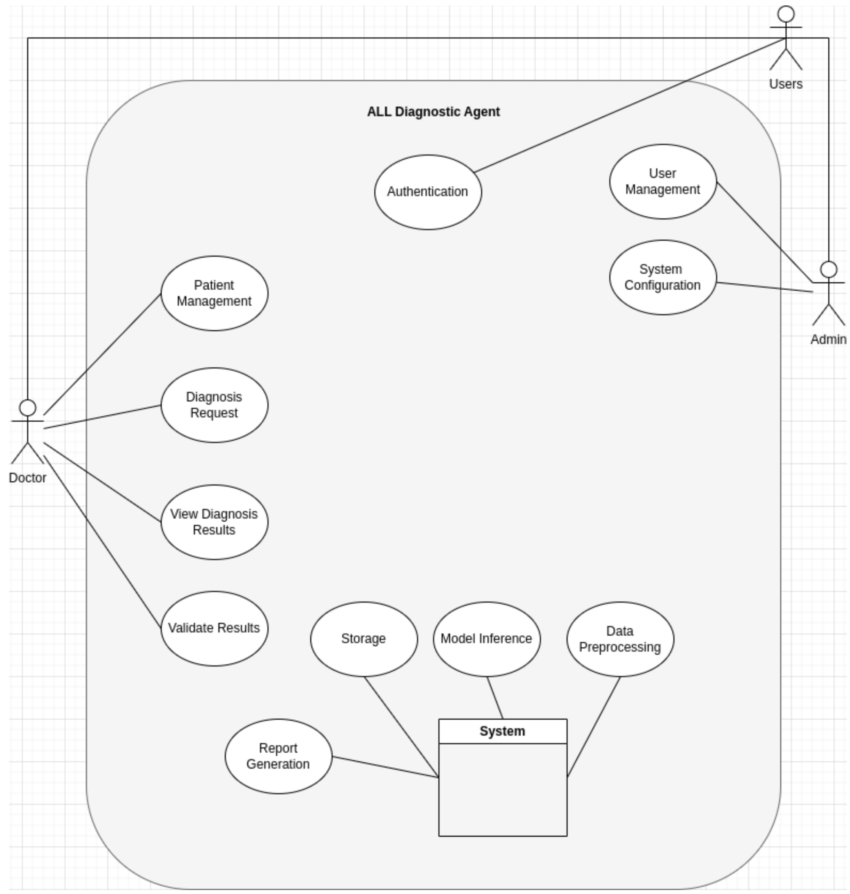
The project will utilize Django as the backend framework, MySQL as the database management system, and TensorFlow for model training. The frontend development will involve HTML, CSS, and Vanila JavaScript.

A comprehensive project timeline has been provided, outlining the expected durations for each phase, including requirement engineering, implementation, model design and training, and finalizing. However, it is important to note that timelines are subject to adjustments based on project-specific considerations and unforeseen circumstances.

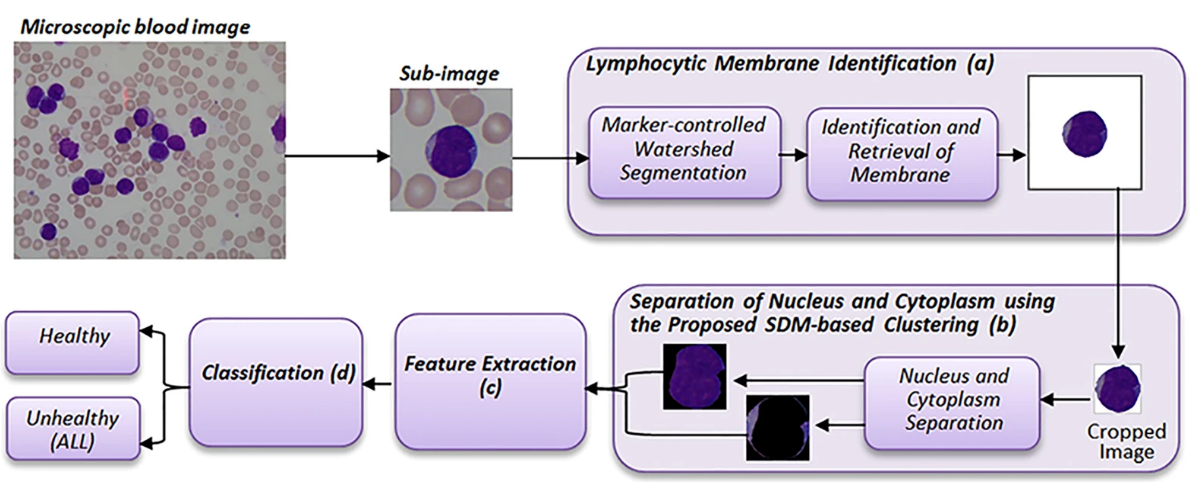
Thank you for considering this project proposal. We believe that with the planned approach, technologies, and timeline, we can successfully develop and deploy an effective AI agent for ALL diagnosis.

Appendix

High-level Use-case Diagram



Proposed Technical Solution



References

• Acute lymphoblastic leukemia (Wikipedia)
• In Childhood Acute Lymphoblastic Leukemia, Blasts at Different Stages of Immunophenotypic Maturation Have Stem Cell Properties (cell.com)
• An Intelligent Decision Support System for Leukaemia Diagnosis using Microscopic Blood Images (nature.com)
• Leukemia Classification (kaggle.com)
• Software Requirements Specification (Added)
• Project Progress Presentation (Added)

CMMI Meeting Mins

Meeting Information			
Meeting Date/Time	22/06/2023		
Participants	Fathima Rafaha Adheeb Ahmed Zeenath Zahra		
Estimated Time	I hour	**Actual Time**	40 minutes
Special Notes	Amaajith was absent		
Call/Location Information	Zoom		
Supported Documents	Null		

Agenda:

Appointment with supervisor
Supervisors approval for the project proposal
Discussion about Project management and progress tracking tool

Notes/Clarifications:

Git will be used for project management
Trello will be used to track the progress of the project

Meeting Minutes:

Scheduled a meeting with the supervisor for next week.

Supervisor approved the project proposal.

Agreed to use Git for project management.

Set up a Git repository and shared access with the team.

Decided to track progress using Trello.

Assigned tasks and deadlines to team members in Trello.

Will update the Trello board regularly to monitor progress

Meeting-Heading

Meeting Information			
Meeting Date/Time	2023-06-27		
Participants	All		
Estimated Time	20 Mins	**Actual Time**	25 Mins
Special Notes			
Call/Location Information	Teams		
Supported Documents	None		

Agenda:

Planning Requirement Gathering

Choosing SDLC methodology

Choosing right database system

Notes/Clarifications:

Meeting Minutes:

Amaajith emphasized the benefits of Agile Development

Zahra outlined a proper plan for requirement gathering

Rafaha demonstrated the seamless integration of mysql with django

Action Items:

Action item	Decision made by
Chose Agile methodology	**Unanimous**
Chose MySQL	**Unanimous**
Identified Stakeholders	**Unanimous**

Organization name 1 <Date>

Meeting-Heading

Meeting Information			
Meeting Date/Time	2023-07-17		
Participants	All		
Estimated Time	15 Mins	**Actual Time**	25 Mins
Special Notes			
Call/Location Information	Teams		
Supported Documents	None		

Agenda:

Documenting gathered requirments

Preparing SRS

Preparing PPR

Preparing PPP

Notes/Clarifications:

Meeting Minutes:

Amaajith documented all requirements from the source document

Rafaha prepared PPP and PPR
Zahra and amaajith outlined and prepared SRS

Action Items:

Action item	Decision made by

Nexus **2023-07-20**

Organization name 1 <Date>

Meeting-Heading

Meeting Information			
Meeting Date/Time	2023-07-20		
Participants	All		
Estimated Time	60 Mins	**Actual Time**	60 Mins
Special Notes			
Call/Location Information	Teams		
Supported Documents	None		

Agenda:

Choosing a right model

Preprocessing the dataset

Finalizing reports

Notes/Clarifications:

Meeting Minutes:

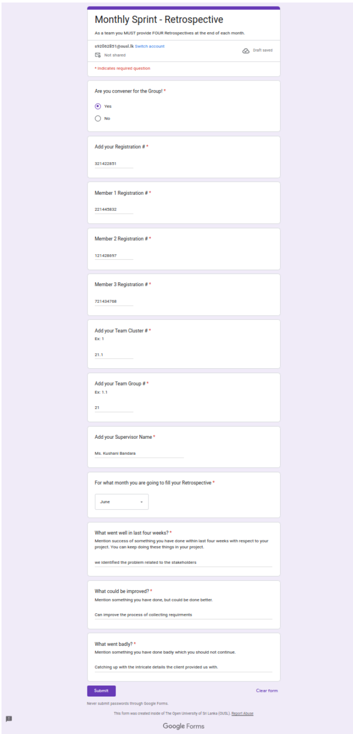
Amaajith and Zahra finalized SRS and PPR

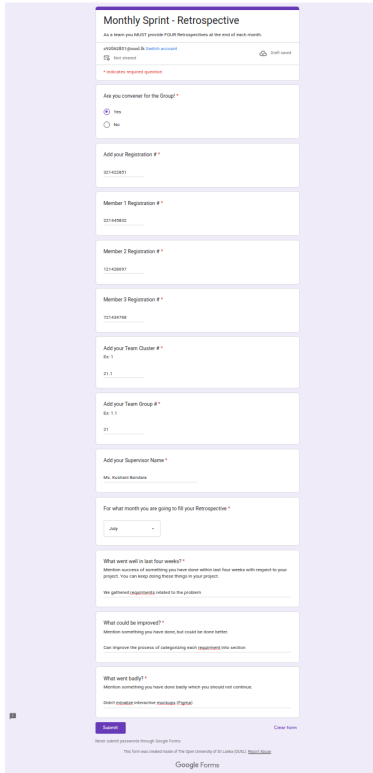
Rafaha finalized PPP
Adheeb started preprocessing the dataset and found a scalable model for training and evaluation

Action Items:

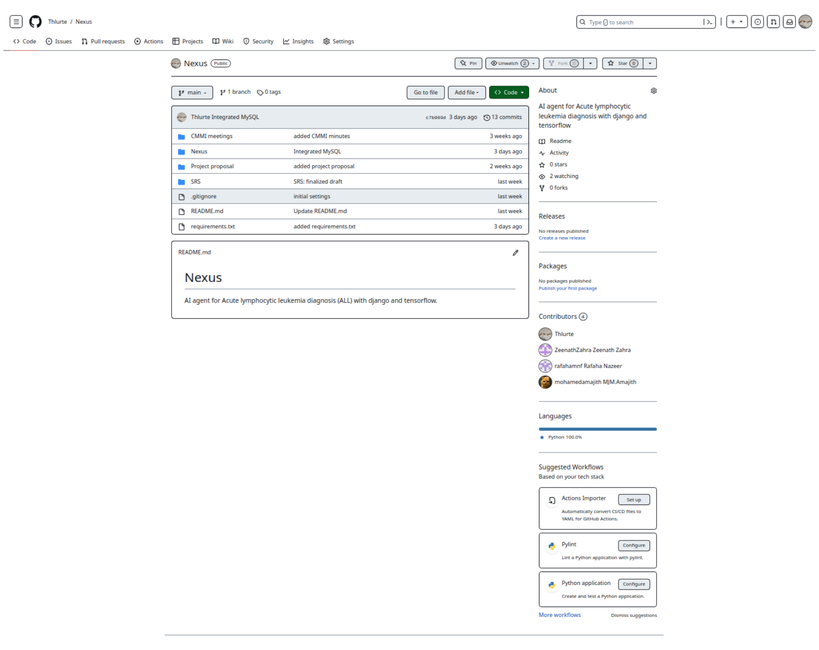
Action item	Decision made by

Monthly Retrospective

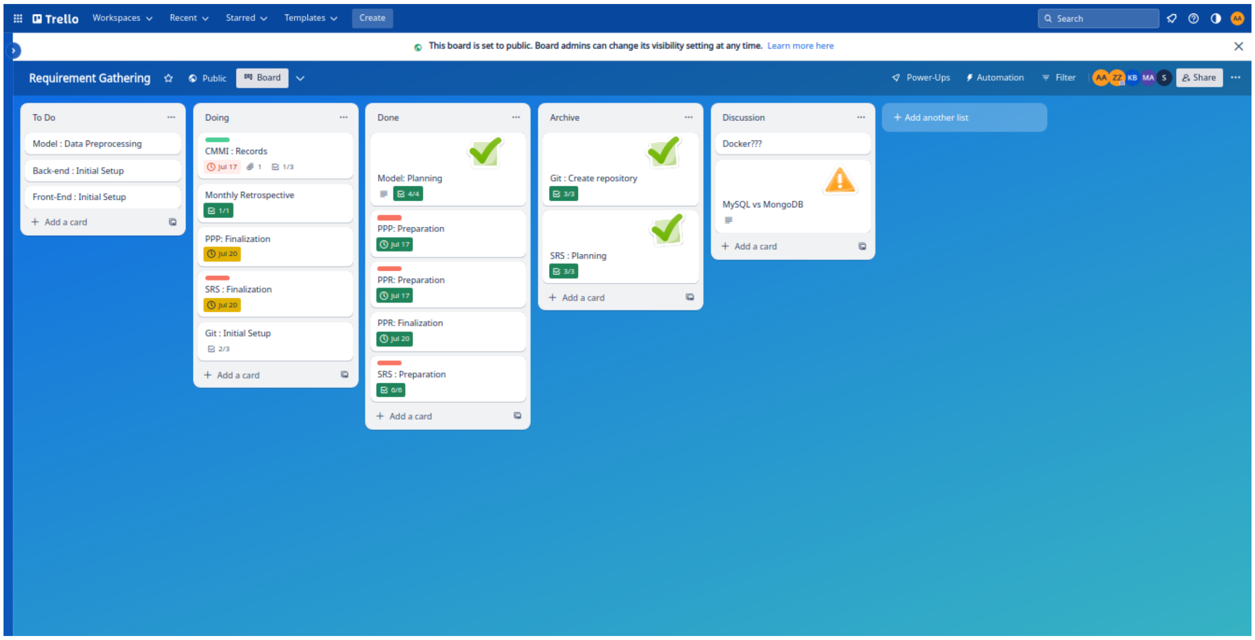




Git



Trello (Project Governance Tool)



Ms. Kushani Bandara

————————————————

Approval

SOFTWARE REQUIREMENTS SPECIFICATION

for

Acute Lymphoblastic Leukemia Cell Diagnosis System

Version 0.1

Prepared by : Nexsus
Group: 21
Cluster: 21.1

Supervisor : Ms. Kushani Bandara

July 20, 2023

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to **provide a comprehensive and detailed specification for the development of an application aimed at assisting medical professionals in the accurate diagnosis of acute lymphoblastic leukemia (ALL).** Acute lymphoblastic leukemia (ALL) is a type of cancer that affects the blood and bone marrow. It is the most common type of cancer in children, and accounts for about 30% of all childhood cancers. ALL is caused by a mutation in the DNA of a white blood cell, which causes the cell to grow and divide uncontrollably. The diagnosis of ALL is typically made by a combination of blood tests and bone marrow biopsies. However, these tests can be difficult to interpret, and there is a risk of false positives and false negatives. This software application is designed to help improve the diagnosis of ALL by using convolutional neural networks (CNNs) to **analyze pathological images of individual cells**. CNNs are a type of machine learning algorithm that are specifically designed for image recognition. They have been shown to be very effective at identifying patterns in images, and they have been used to diagnose a variety of diseases, including cancer.

1.2 Scope

The scope of the application encompasses the **processing and analysis of pathological images of individual cells** obtained from practictions through microscopic imaging techniques. It will focus on automating the **classification process to identify leukemic cells accurately**. By utilizing convolutional neural networks, the application will harness the power of deep learning algorithms to extract intricate patterns and features associated with leukemic cells, enabling medical professionals to make informed and timely diagnostic decisions. The application will not involve the diagnosis or treatment of patients but will serve as a valuable tool in assisting medical professionals with their expertise and judgment.

1.3 Intended Audience

The primary audience for this document includes:

- The software development team responsible for designing, developing, and implementing the application.

- Medical professionals specializing in hematology and oncology who will utilize the application as a diagnostic tool in their clinical practice.
- Stakeholders involved in the development and implementation of the application, such as healthcare administrators, and regulatory bodies.

2 Overall Description

2.1 Product Perspective

The application will be a **web based system designed to seamlessly integrate with existing manual procedure**. It will serve as a specialized module, receiving input images from the medical practioner and providing accurate classification results. The application will function independently, without requiring any modifications. It will provide a convenient and efficient tool for medical professionals to aid in the diagnosis of ALL.

2.2 Product Features

- Image preprocessing: To ensure optimal classification accuracy, the application will **employ advanced image preprocessing techniques.** These techniques will include noise reduction algorithms to eliminate unwanted artifacts, contrast adjustment to enhance subtle features, and image enhancement methods to improve the visibility of relevant structures. The application will strive to enhance image quality while preserving important diagnostic details.

- Convolutional neural network training: The application will utilize a deep learning approach by training a convolutional neural network (CNN). The CNN will be trained using a comprehensive dataset of labeled pathological cell images, encompassing a wide range of leukemic and regular cells. The training process will involve the extraction of distinguishing features and the fine-tuning of network parameters to achieve robust and accurate classification.

- Cell classification: Leveraging the trained CNN, the application will perform efficient and reliable cell classification. It will analyze individual cells within the input images and categorize them as either leukemic or regular cells based on learned patterns and characteristics. The classification results will be generated promptly, enabling medical professionals to make timely and informed decisions regarding diagnosis and treatment plans.

- User interface: The application will feature an intuitive and user-friendly interface, designed to streamline the workflow of medical professionals. The interface will allow users to easily upload cell images acquired from microscopy equipment, initiate the image preprocessing and classification processes, and visualize the classification results. Furthermore, the application will facilitate data management, allowing users to organize and access patient information, review past analyses, and export data for further analysis or archival purposes.

2.3 User Classes and Characteristics

- Medical professionals: The intended users of the application will primarily consist of medical professionals specializing in hematology and oncology. These users will possess a comprehensive understanding of leukemia and its diagnosis, as well as the underlying principles of microscopy imaging techniques. They will have the expertise to interpret the classification results provided by the application accurately. The application will augment their diagnostic capabilities and streamline their workflow, enabling them to make well-informed decisions efficiently.

- Administrators: Administrators will be responsible for managing and maintaining the application. They will have elevated privileges and access rights to perform tasks such as user management, system configuration, and data administration. Administrators will possess technical knowledge related to the application's infrastructure, security, and maintenance. They will ensure the smooth operation of the application, handle updates and backups, and address any technical issues that may arise.

2.4 Operating Environment

The application will be compatible with desktop and laptop computers, ensuring flexibility and accessibility. It will support multiple operating systems, including Windows, macOS, and Linux, to accommodate a wide range of user preferences and setups. To ensure optimal performance, the application will require a minimum hardware configuration, such as an Intel Core i5 processor, 8 GB of RAM, and 500 GB of storage. These hardware requirements will ensure smooth operation and allow for efficient processing and analysis of cell images, contributing to a seamless user experience.

3 Assumptions, General Development, and Deployment

3.1 Assumptions

- It is assumed that the input pathological cell images will be obtained through standard microscopy imaging techniques and will be of sufficient quality for accurate analysis and classification.
- The labeled dataset used for training the convolutional neural network (CNN) will be carefully curated, ensuring accurate labeling and representation of different cell types.
- It is assumed that the application will receive necessary support and collaboration from medical professionals and domain experts to ensure the relevance and accuracy of the diagnostic outcomes.
- The application assumes the availability of appropriate computational resources to support the training and inference processes of the CNN, such as sufficient computing power, storage capacity, and memory.

3.2 General Development

- The development of the application will follow an iterative and incremental approach, allowing for continuous feedback and refinement throughout the development lifecycle.
- Agile development methodologies, such as Scrum or Kanban, will be employed to promote flexibility, collaboration, and rapid iterations.
- The application will undergo rigorous testing at various stages of development, including unit testing, integration testing, and validation against known datasets, to ensure functionality, accuracy, and reliability.
- Continuous integration and continuous deployment (CI/CD) practices will be adopted to facilitate automated builds, testing, and deployment of the application, ensuring a streamlined and efficient development process.
- The development of the application will leverage the Django framework as the backend technology stack.

- TensorFlow and Keras, two popular deep learning frameworks, will be utilized for the development and training of the convolutional neural network (CNN).
- The application will employ MySQL as the database management system for storing and managing relevant data, such as user information, image metadata, and classification results.
- The frontend design of the application will be developed using a combination of Vanilla JavaScript, HTML, and the Tailwind CSS.

3.3 Deployment

- The application will be deployed as a web application that can be deployed on desktop or laptop computers.
- Detailed installation instructions will be provided to guide users through the setup process and ensure a smooth deployment experience.
- The application will support popular operating systems, including Windows, macOS, and Linux, to accommodate a wide range of user environments and preferences.
- Compatibility with image file formats will be ensured, allowing seamless integration with existing infrastructure and workflows.
- Adequate documentation, including user manuals and troubleshooting guides, will be provided to assist users during the deployment and usage of the application.

4 Contextual Data Flow Diagram

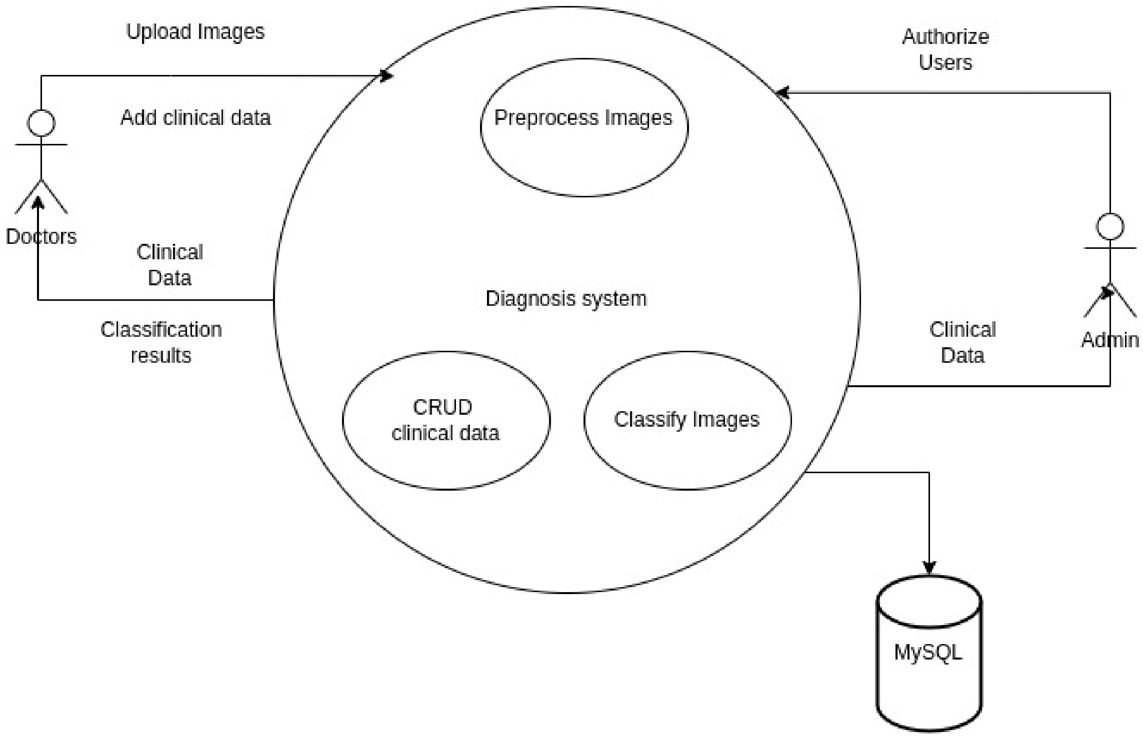


Figure 4.1: A High-level Contextual Flow Diagram

5 UI Wire-frame

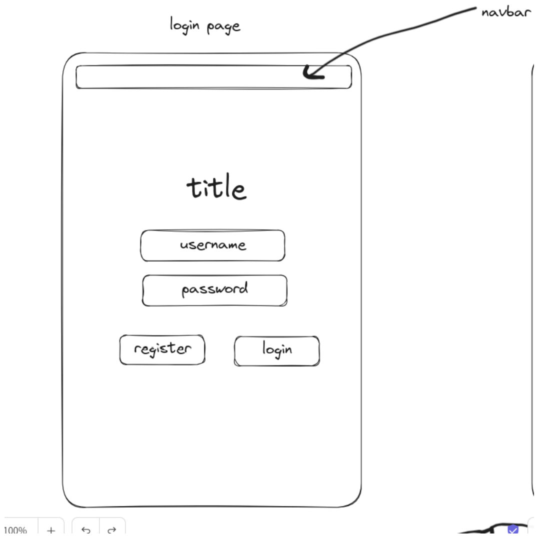


Figure 5.1: 1. Login Pag

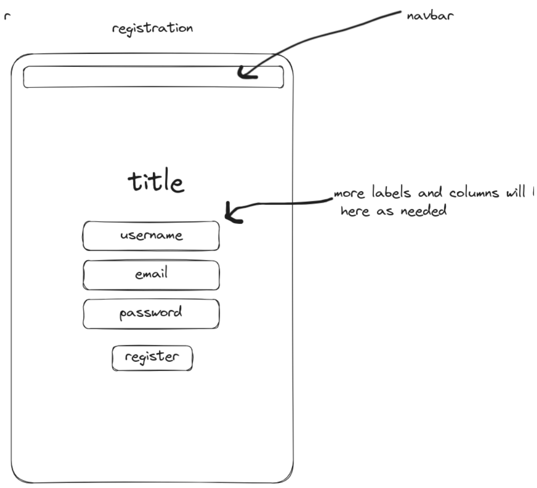


Figure 5.2: 2. Registration Page

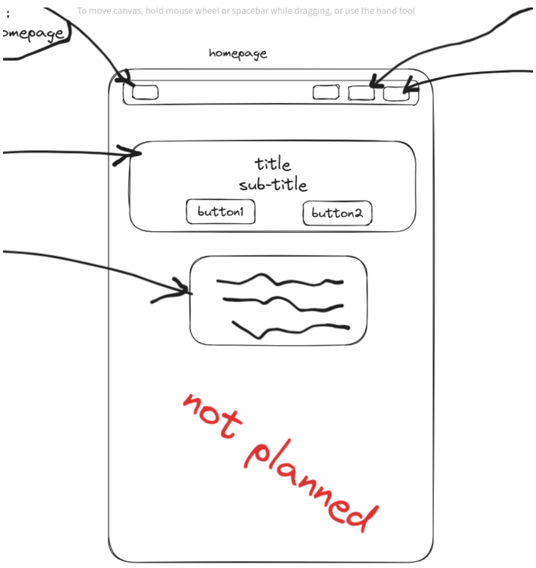


Figure 5.3: 3. Home Page

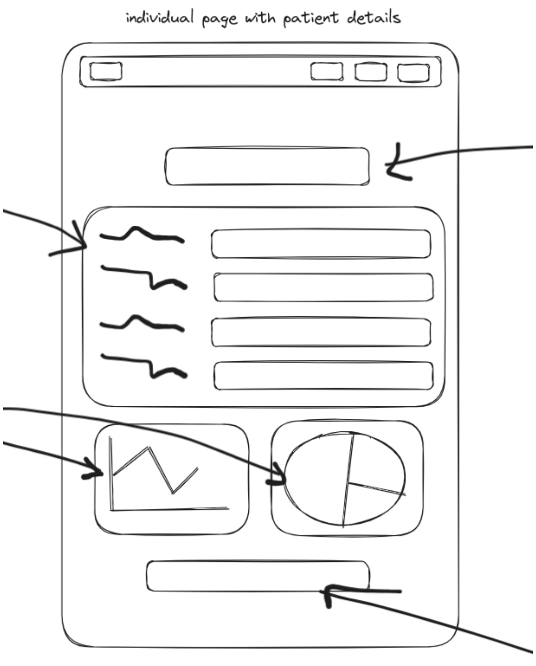


Figure 5.4: 4. Individual Patient Details

6 Use-case Diagram

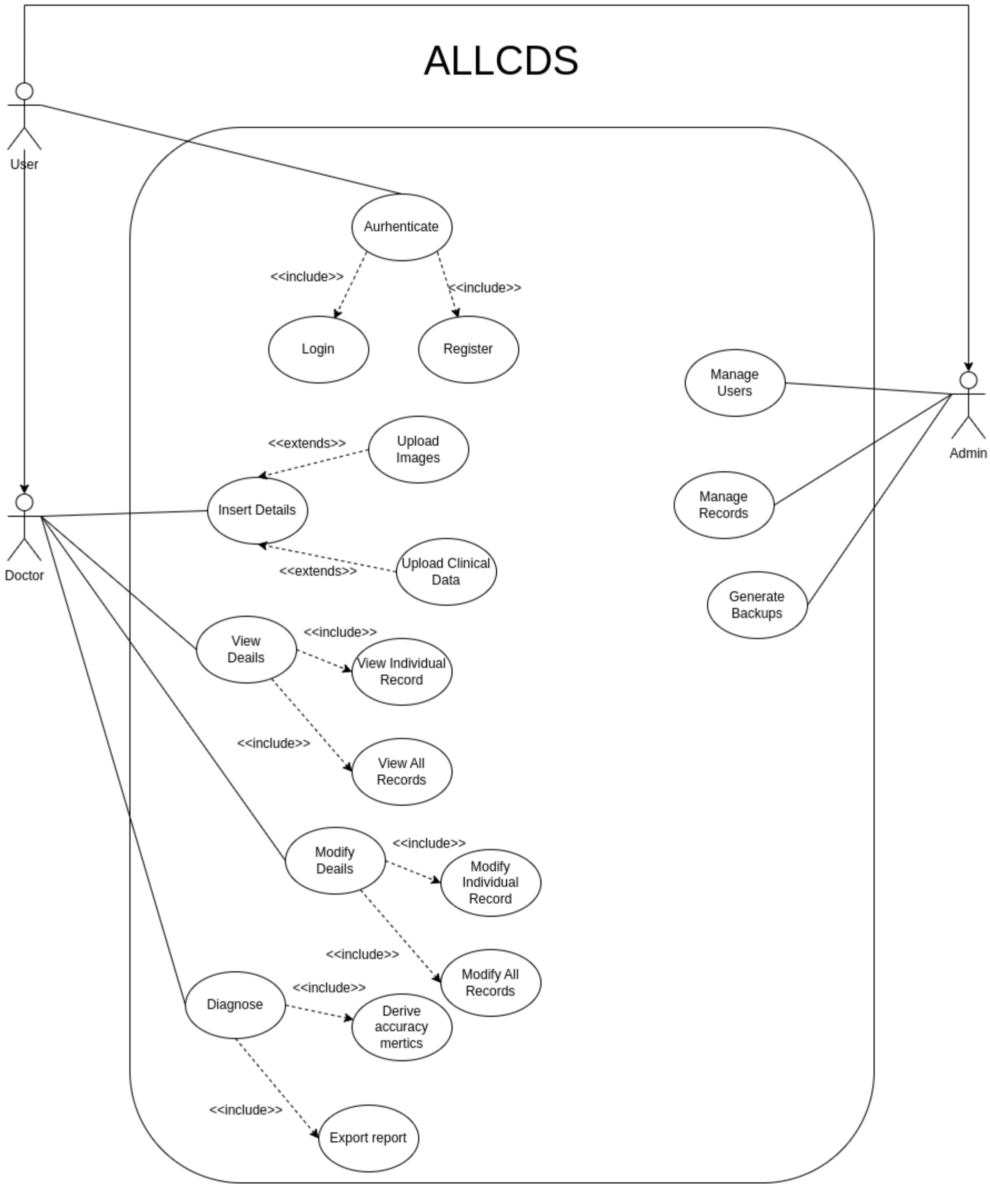


Figure 6.1: Use-case diagram

7 Functional Requirements

7.1 Image Input and Processing

- The application shall provide an intuitive user interface that allows healthcare professionals to easily upload pathological images of individual cells.
- The application shall support a wide range of common image formats, including but not limited to JPEG, PNG, and TIFF, to accommodate different imaging systems and file types.
- The application shall include image preprocessing techniques to enhance the quality and optimize the input images for CNN analysis.
  - Preprocessing techniques may include image resizing, noise reduction, contrast adjustment, and normalization.
  - The preprocessing steps shall ensure that the input images are in a suitable format and resolution for accurate analysis.

7.2 Convolutional Neural Network (CNN) Analysis

- The application shall integrate a pre-trained CNN model that has been specifically trained for the diagnosis of Acute Lymphoblastic Leukemia (ALL) using pathological cell images.
  - The choice of the pre-trained CNN model shall be based on its proven performance in similar medical image analysis tasks.
  - The CNN model shall have undergone rigorous training on diverse dataset and demonstrate high sensitivity and specificity in distinguishing between leukemia cells and regular cells.
- The CNN model shall accept the preprocessed images as input and perform cell-level classification, identifying and classifying each cell as either a leukemia cell or a regular cell.
  - The model shall utilize advanced convolutional and pooling layers, as well as activation functions, to extract relevant features from the input images.
  - The CNN model shall employ a suitable classification algorithm (e.g., softmax) to generate probabilities or confidence scores for each cell classification.

- The application shall provide a user-friendly interface that presents the classification results in a clear and understandable manner.
  - The interface may include visual indicators, such as color-coded markings or overlays, to highlight the identified leukemia cells.
  - The application shall allow users to interact with the classification results, enabling zooming, panning, and detailed examination of individual cells.

7.3 Diagnostic Reporting

- The application shall generate a comprehensive diagnostic report for each uploaded image, presenting the findings based on the classification results.
  - The diagnostic report shall clearly indicate the presence or absence of Acute Lymphoblastic Leukemia (ALL) based on the analysis.
  - The report shall include detailed information about the identified leukemia cells, including their location, quantity, and any notable characteristics.
- The diagnostic report shall provide relevant statistical information to assist healthcare professionals in evaluating the accuracy of the classification and making informed decisions.
  - Statistical information may include sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), and overall accuracy metrics.
  - The report shall also highlight any specific details related to false positives or false negatives, providing insights into potential sources of misclassification.
- The application shall allow users to export the diagnostic reports in various suitable formats, such as JSON or CSV, to facilitate further analysis, collaboration, and archiving.
  - The exported reports should maintain the integrity and readability of the information presented within the application.
  - The application shall ensure that any exported data complies with relevant data protection and privacy regulations.

7.4 User Authentication

- The application shall provide secure user authentication mechanisms to ensure that only authorized healthcare professionals can access and use the system.
  - The authentication process should require users to provide valid credentials, such as a username and password or other appropriate authentication factors.
  - The application should enforce strong password policies, including minimum length, complexity requirements, and regular password expiration.

- The application shall support role-based access control to restrict certain functionalities or data access based on the user's assigned role or privileges.
  - User roles may include administrators, medical practitioners, pathologists, or other relevant roles within the healthcare organization.
  - The access control mechanisms should be configurable, allowing system administrators to define and manage user roles and associated permissions.
- The application shall implement secure session management to maintain user sessions and protect against unauthorized access or session hijacking.
  - The system should automatically terminate idle sessions after a specified period of inactivity to ensure security and privacy.
  - The application shall provide mechanisms to handle session timeouts, session revocation, and secure logout functionalities.

7.5 Historical Diagnosis and Clinical Details

- The application shall maintain a comprehensive database or storage system to retain all past diagnosis details and associated clinical information for each patient.
  - The database should be designed to securely store and organize a large volume of historical diagnosis records.
  - The storage system should support efficient retrieval and querying of patient-specific information based on various search criteria.
- The application shall capture and store relevant clinical details of each patient, including but not limited to:
  - Personal information (e.g., name, date of birth, gender)
  - Medical history (e.g., previous illnesses, treatments, surgeries)
  - Genetic information (if available and relevant)
  - Laboratory test results (e.g., blood tests, genetic tests)
  - Radiology or imaging reports (e.g., X-rays, CT scans)
  - Other pertinent clinical data (e.g., symptoms, family history)
- The application shall associate each diagnosis with the corresponding patient's clinical details, allowing for easy retrieval and correlation of relevant information.
  - Each diagnosis entry should include a unique identifier or reference to the patient's record for proper linkage and identification.
- The application shall support the ability to view and review past diagnosis records, allowing healthcare professionals to access the complete details of previous diagnoses for reference and comparison.

- The system should present the information in a clear and organized manner, including timestamps, diagnosis descriptions, and any supporting diagnostic data or images.

- The application shall facilitate the updating and modification of clinical details and diagnosis records to ensure the accuracy and relevance of the stored information.
  - Authorized users should have the ability to add new clinical information, modify existing records, or annotate past diagnoses with additional findings or treatment outcomes.

7.6 User Interface

The user interface (UI) of the Acute Lymphoblastic Leukemia Cell Diagnosis System (ALLCDS) plays a critical role in ensuring a seamless user experience. The following requirements detail the necessary features and functionalities of the UI:

7.6.1 Intuitive and User-Friendly Interface

The ALLCDS shall provide an intuitive and user-friendly interface that facilitates efficient usage and navigation for both novice and experienced users. The interface should adhere to established UI design principles, including:

- Clear and concise labeling: All interface elements, buttons, and controls shall be labeled appropriately to convey their purpose and functionality clearly.
- Consistent layout: The UI shall maintain a consistent layout throughout the application, ensuring that similar features are consistently positioned across screens.
- Responsive design: The UI shall adapt and optimize its layout and usability for different screen sizes and resolutions, including desktop and mobile devices.
- Minimal user input: The interface shall minimize the need for user input by leveraging automation and preconfigured settings wherever possible.
- Contextual help and documentation: The UI shall provide contextual help, tooltips, and documentation to assist users in understanding the system's features and functionality.

7.6.2 Display of Uploaded Images, Classification Results, and Confidence Scores

The UI shall effectively present the uploaded pathological cell images, classification results, and associated confidence scores to the user. The following requirements outline the necessary display elements:

- Image upload functionality: The UI shall include an image upload feature that enables users to select and upload pathological cell images for analysis. The system should support various image formats commonly used in medical imaging (e.g., JPEG, PNG).
- Image preview: Upon upload, the UI shall provide a preview of the uploaded images, allowing users to review the images and ensure accuracy before initiating the analysis process.
- Classification results: The UI shall display the classification results for each uploaded image, indicating whether the cell is identified as pathological or regular.
- Confidence scores: Alongside the classification results, the UI shall present confidence scores or probabilities associated with each classification. These scores provide an indication of the certainty level of the CNN model's classification decision.
- Visualization options: The UI may provide additional visualization options to enhance the interpretation of results, such as overlaying highlighted regions on the images to indicate the areas of interest or abnormalities detected.

8 Nonfunctional Requirements

The non-functional requirements of the Acute Lymphoblastic Leukemia Cell Diagnosis System (ALLCDS) encompass various aspects such as performance, accuracy, security, and scalability. These requirements ensure that the system operates efficiently, reliably, and securely while allowing for future growth and expansion.

8.1 Performance Requirements

8.1.1 Processing Time

The system shall process pathological cell images and provide classification results within an acceptable time frame. Specifically, the processing time should be less than 10 seconds per image. The system's image processing pipeline, including preprocessing and CNN-based analysis, should be optimized for efficient execution to deliver prompt results to users.

8.1.2 Scalability

The system must be capable of handling a large volume of images and classifications simultaneously. It should support concurrent analysis of at least 100 images, allowing medical professionals to upload and process multiple images in parallel. This requirement ensures that the system can handle the demands of high-throughput environments, such as busy clinics or research laboratories.

8.2 Accuracy

8.2.1 Classification Accuracy

The system's CNN model must achieve a high accuracy rate in differentiating between pathological and regular cells. The minimum accuracy requirement is set at 90%, indicating that the model should provide accurate classifications for the majority of cases. Regular model evaluation and performance monitoring should be conducted to ensure that the accuracy remains consistently high as new data becomes available.

8.2.2 Consistency of Results

The system should maintain consistency in classification results for the same input image over multiple runs. This requirement ensures that the system produces reliable and reproducible results. Inconsistencies in classification outcomes can undermine trust in the system's diagnostic capabilities. Therefore, the CNN model and associated processes should be designed to provide consistent results when presented with the same image.

8.3 Security Requirements

The system must ensure the confidentiality and integrity of patient data. This includes implementing appropriate security measures such as encryption and access controls to protect sensitive information from unauthorized access or tampering. Adherence to relevant data protection regulations and best practices is essential to maintain patient privacy and prevent data breaches.

8.4 Scalability

The system architecture should be scalable to accommodate future enhancements and changes. It should be capable of incorporating additional pathological cell types if required, allowing the system to expand its diagnostic capabilities. Moreover, the architecture should support multi-site deployments, enabling the system to be deployed across multiple locations or institutions while maintaining centralized management and data synchronization.

9 Quality Assurance Requirements

Testing plays a crucial role in ensuring the reliability, accuracy, and performance of the Acute Lymphoblastic Leukemia Cell Diagnosis System (ALLCDS). It is important to thoroughly test the system to identify and rectify any potential issues or shortcomings. The testing section outlines various types of testing that should be conducted during the development and deployment of the system.

9.1 Unit Testing

Unit testing focuses on testing individual components and modules of the system in isolation. The following unit testing requirements should be considered:

- Unit tests shall be developed for each module and component of the system, including image preprocessing, CNN model, classification algorithms, and user interface components.
- The unit tests shall cover both normal and boundary cases, validating the correctness and expected behavior of each component.
- Test cases shall be designed to assess the accuracy of the classification algorithm, the effectiveness of image preprocessing techniques, and the functionality of individual UI elements.

9.2 Intergation Testing

Integration testing verifies the interactions and compatibility between different components of the system. The following integration testing requirements should be fulfilled:

- Integration tests shall be conducted to ensure proper communication and data flow between the image preprocessing module, the CNN model, and the user interface.
- Integration tests shall cover various scenarios, including successful image upload and analysis, error handling, and edge cases.
- The integration tests should validate the system's ability to process images, generate accurate classification results, and display them correctly in the user interface.

9.3 Performance Testing

Performance testing focuses on assessing the system's performance under different workloads and conditions. The following performance testing requirements should be considered:

- Performance tests shall be conducted to measure the system's response time and resource utilization during image processing and classification.
- The system's performance should be evaluated under both normal and peak load conditions to ensure it can handle the expected volume of image uploads and concurrent analysis.
- Performance tests should be performed on various hardware configurations and network conditions to identify any performance bottlenecks and optimize the system accordingly.

9.4 Acceptance Testing

Acceptance testing is performed to validate that the system meets the specified requirements and satisfies the needs of the stakeholders. The following acceptance testing requirements should be met:

- Acceptance test cases shall be developed based on the functional and non-functional requirements specified in the Software Requirements Specification (SRS).
- The acceptance tests should cover various user scenarios, including image upload, classification, result display, and system configuration.
- The system should undergo acceptance testing with involvement from domain experts and medical professionals to ensure it meets their expectations and aligns with their diagnostic processes.

9.5 Security Testing

Security testing aims to identify vulnerabilities and weaknesses in the system's security measures. The following security testing requirements should be considered:

- Security tests shall be conducted to identify potential vulnerabilities in data transmission, storage, and access control.
- The system's security measures, such as encryption of sensitive data, authentication mechanisms, and authorization controls, should be thoroughly tested.
- Security tests should be performed by experienced security professionals or ethical hackers to ensure the system can withstand potential attacks and protect patient data.

9.6 Usability Testing

Usability testing evaluates the system's user interface and assesses its ease of use, intuitiveness, and effectiveness. The following usability testing requirements should be addressed:

- Usability tests should involve representative users, including medical professionals, to evaluate the user interface's intuitiveness and user-friendliness.
- The usability tests should focus on tasks such as image upload, result interpretation, and system configuration to assess the ease of completing these tasks.
- Feedback from users should be collected to identify areas for improvement in the user interface and overall user experience.

10 Conclusion

The Acute Lymphoblastic Leukemia Cell Diagnosis System (ALLCDS) employs convolutional neural networks (CNNs) to unravel the complexities of pathological cell images. Its user-friendly interface effortlessly facilitates image upload, classification result display, and confidence score examination. The system swiftly processes each image in under 10 seconds, adeptly managing a high volume of concurrent analyses. With a minimum accuracy requirement of 90%, ALLCDS ensures precise differentiation between regular and pathological cells. Security measures, including encryption and access controls, protect patient data. Scalability accommodates future enhancements and 1deployments. Rigorous testing, encompassing unit, integration, performance, acceptance, security, and usability, guarantees reliability, ushering a new era of leukemia diagnosis.

11 Miscellaneous

11.1 Team Info

- MJM.Amajith 721434768
- MMZ. Zahra 221445832
- MN.Fathima Rafaha 121428697
- MHA. Ahmed 321422851

11.2 Document Revision

Date	Version	Description
2023/07/20	0.1	Initial SRS

Ms. Kushani Bandara

Approval