BACHELOR OF SCIENCE IN SOFTWARE DEVELOPMENT

FINAL YEAR PROJECT 1 : BSD 3106

REALTIME MEDICAL IMAGE ANALYZER

MAKINDU TRAUMA CENTER

BY

FESTUS MAITHYA KILONZO

SYSTEM REQUIREMENT SPECIFICATIONS DOCUMENT SUBMITTED IN PARTIAL

FULFILLMENT FOR THE REQUIREMENTS FOR THE AWARD OF A DEGREE

IN SOFTWARE DEVELOPMENT

PRESENTED TO : DR. LUCY MBURU

Declaration

I declare that this project is my original work and has not been presented in any other college or University for the award of a Diploma or Degree.

Student Details:

Name: …FESTUS MAITHYA KILONZO…… Date : …………22/10/2024………

Signature ………FMK………

Supervisor Details:

Name: ………………………………...…… Date : ………………………………

Signature ………………...………

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 Purpose

The purpose of this SRS is to give an overview of the Real-Time Medical Image Analysis, RMIA, system. It stipulates how the system will work, the problems that the current systems present, and also states the requirements and technological choices for the system. This document is actually the designer's view of what the client requires and it gives an elaborative view before the actual designing of the system is made.

The RMIA system is expected to support clinical decision-making in real time by analyzing medical images, such as X-rays, on the spot. Patients or any user will upload the images, and the system will respond immediately with diagnostic results and answers to related queries via a chat interface.

1.2 INTENDED AUDIENCE

The target audience of this document includes, but is not limited to:

- ✓ **Healthcare Professionals -** Medical practitioners, including doctors and radiologists, who will be the clients of the system in terms of real-time image analysis.
- ✓ **Developers -** Developers who want to collaborate in making RMIA a success.
- ✓ **Academic Evaluators** - Project evaluation committee members from KCA University.
- ✓ **Future Healthcare personnel** - Students undertaking studies in medical imaging fields.
- ✓ **IT Students** - For reference to future students who will take on software engineering at KCA University.

1.3 AIMS OF SRS

The aims of RMIA SRS are to:

- ❖ **Provide common understanding grounds between users and developers** - This document will set clear communications and understanding of the functionalities of the system, ensuring that all parties have a uniform view of the objectives of the system and its expected behavior.
- ❖ **Explain the functions in detail** – This document gives a detailed explanation of the functionalities of the system so that the users can establish if the software satisfies their needs and requirements appropriately.
- ❖ **Reduce the development effort** – The SRS document reduces ambiguity and hence rework by considering all requirements in detail before the design phase. Therefore, it assures a smooth flow of the development process.
- ❖ **Provide a basis for making estimates of project costs and timescales** – this document is a valid reference to be used for making an estimate of the cost of a project and the

timescale for the same can be availed, thus enabling effective planning and resource allocation, which is vital for project management.

- **Baseline for systems validation, verification, and enhancements** - It provides a standard reference against which the verification of the system for its requirements and the validation of the overall functionality of the system are performed. The baseline also assists in evaluating and implementing improvements in the future.
- **Allow the transfer of the software to new users or machines easily** - The structured and documented approach provided by this SRS allows for easy deployment, transfer, and configuration of the software to adapt to new users or environments with minimal effort.

1.4 PROJECT SCOPE

This will be a RMIA system on cloud infrastructure, using GCP tools for deployment. The work will involve Apache Kafka and Spark for real-time data processing and analysis. The patients shall be allowed to upload X-ray images through the system, which shall use multi-label classification models for fracture, pneumonia, and lung cancer detection. Results of the processing will be provided through a web application using Django and Streamlit.

1.5 REFERENCES

Python Documentation - Python Software Foundation. (2022). Python Documentation. Retrieved from https://docs.python.org/3/

Django Framework Documentation - Django Software Foundation. (2022). Django Documentation. Retrieved from https://docs.djangoproject.com/en/4.1/

Streamlit Documentation - Streamlit Inc. (2022). Streamlit Documentation. Retrieved from https://docs.streamlit.io/

GCP Documentation - Google Cloud Platform. (2022). Google Cloud Documentation. Retrieved from https://cloud.google.com/docs/

Apache Spark and Kafka - Apache Software Foundation. (2022). Apache Spark Documentation. Retrieved from https://spark.apache.org/docs/latest/

1.6 OVERALL DESCRIPTION

1.6.1 Product Perspective

RMIA system will be a new and independent application, hosted on a Client-Server architecture. The system will be back-ended using Django, and for the creation of the front-end, Streamlit will be used. It will be hosted and stored on GCP for scalability and reliability of the processes.

1.7 PRODUCT FEATURES

1.7.1 Data Capturing

For example, patients will upload their medical images through a web application. Apache Kafka and Apache Spark allow real-time processing of data in order to present instantaneous results. The data i.e. the images uploaded will be pushed to a cloud database via a data pipeline.

1.7.2 Data Processing

The model to be used in this system is a multi-label classification model deployed on GCP infrastructure to analyze the uploaded images. Conditions detected and diagnosed by this system include, but are not limited to, fractures, pneumonia, and lung cancer, providing an accuracy in percentages.

1.7.3 Data Output

It shows the probabilities of different conditions, and through the front-end interface, the results are portrayed. Chat-based queries for further medical information are also provided in conjunction to explainable AI.

1.7.4 Data Validation and Integrity

System Attests to the uploaded file being in the correct format, such as JPEG or PNG, and of quality sufficient to enable its accurate processing. This will be of importance to reduce noise in the machine learning models.

1.7.5 Monitoring

The RMIA system will also comprise monitoring features that will track system performance, user activities, as well as system errors to alert the administrators should an action be required.

1.8 USER CLASSES AND CHARACTERISTICS

The following will use this system once its deployed:

- ✓ **Healthcare Professionals** - They shall be able to understand the medical images on their own and use this system as a diagnostic support. Training shall be provided to ensure that they know the best way to use this system.
- ✓ **The clients or users** will log in to the web interface and upload medical images. They will receive immediate results. The user interface shall be friendly, easy to use, and navigate.
- ✓ **IT Administrators** - the administrators shall update and maintain the back-end services. This will require extensive knowledge on the technologies on cloud services, Django, and Apache technologies

1.9 OPERATING ENVIRONMENT

The following are the basic requirements for utmost system functionality:

1.9.1 Hardware

- ✓ The GCP cloud servers are of high processing power, hence can be able to handle real time image processing, Compute Engine instances with at least 8GB RAM and 100GB SSD storage.
- ✓ Client Machines -Any modern device installed with a web browser will do, like Chrome, Firefox or Edge.

1.9.2 Software

- ✓ Python 3.11+ for development and deployment.
- ✓ Django will be used for the backend and API implementation.
- ✓ Streamlit will be used for frontend deployment.
- ✓ Apache Kafka and Apache Spark for data streaming and real-time processing.
- ✓ GCP services like Compute Engine, Cloud Storage, and AI Platform (Vertex AI).

1.10 DESIGN AND IMPLEMENTATION CONSTRAINTS

- ❖ **Cloud Costs** - GCP-related costs will be considered in this regard and optimized so as not to exceed the budget.
- ❖ **Latency** - The system shall be designed in such a way that there is minimal delay between the image processing and results output
- ❖ **Data Privacy** - The patients' data needs to remain private, and HIPAA and Kenya's DPA compliance should be ensured.

1.11 USER DOCUMENTATION

For the proper use of the system, the following documentation will be provided:

- ✓ **User Manual** - This is a step-by-step guide on how healthcare professionals and patients will use the system which will be provided online and also distributed as downloadable manual copies.
- ✓ **Online Documentation** - This would be made available on the web platform of the system for easy access.
- ✓ **Knowledge Base** - Common troubleshooting for both users and administrators. This will be rendered online for the early prototypes.

1.12 ASSUMPTIONS AND DEPENDENCIES

- ❖ All users will be using the internet on modern devices with web browsers since the initial deployment of the system will be a web app.
- ❖ System hosting will be done on GCP, with appropriate cloud resources allocated.
- ❖ The medical image datasets to be used for model training are dependable and representative of the conditions .

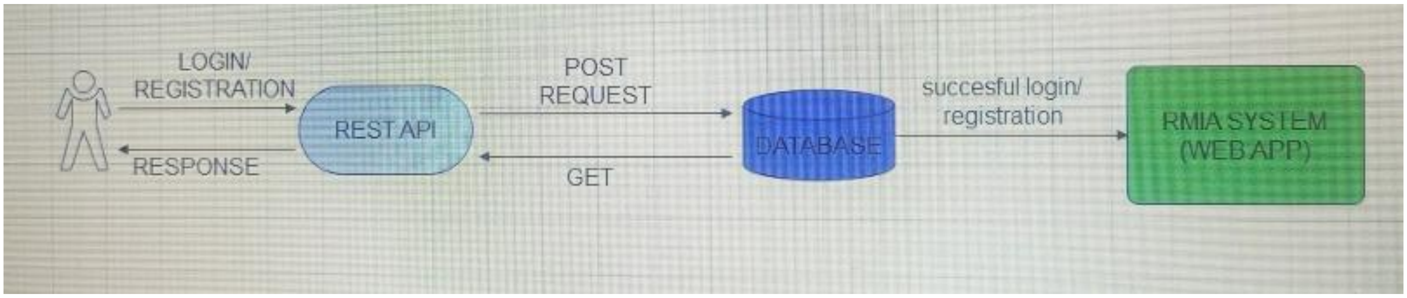
- The target users are IT literate or will be sufficiently trained. We anticipate that the digital divide between users will be very small.

1.13 SYSTEM FEATURES

The RMIA system shall acquire, process and then provide real-time diagnostic outcome for X-ray images. Further, it shall facilitate the following:

- **User Registration/Login** - Secure user registration and login functionality along with password recovery via email.
- **Image Upload** - Users can upload medical images in supported formats.
- **Real-time Analytics** - Instant analytics for diagnosis using trained models.
- **Chat Support** - Medical query live chat based on the images uploaded.

1.14 USE CASE : User Registration/Login



1.14.1 Description and Priority

The new user shall be able to sign up and then log in securely. Users who have already used the system shall be given access via a login form. Priority for this use case will be High.

1.14.2 Stimulus/Response Sequences

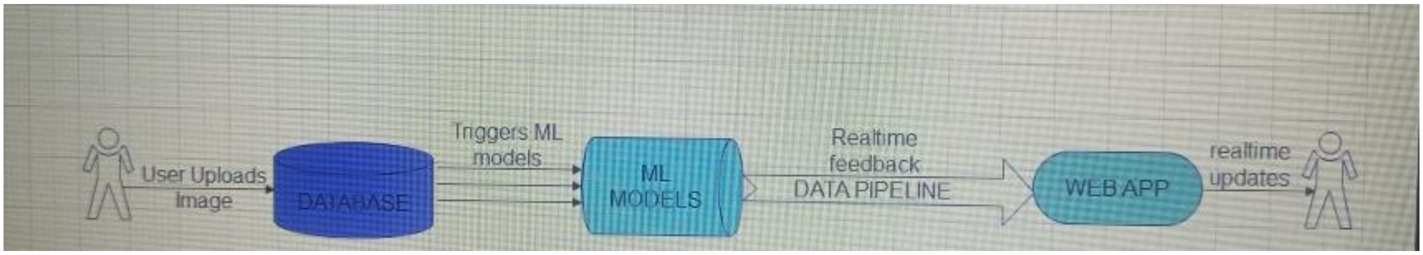
- The user opens up the registration page in his/her browser.
- The user fills in the basic minimum details in the registration form and submits it.
- The backend receives the request and stores the information related to the user.
- Success would redirect users to an image upload page.

1.14.3 Functional Requirements

REQ-1: Secure Authentication - It is expected that authentication in the RMIA system is secure and for this reason, access to the application should be granted with assurance that only authorized personnel have access. The registration process should be able to provide unique usernames and strong passwords to users. The authentication mechanism will preferably include measures like password hashing, salting, and the use of secure protocols (e.g., HTTPS) while protecting user credentials both during transmission and storage.

REQ-2: User Credential Storage and Security - Also, the system will provide a user administration module that will give password reset capabilities through email, account lockout functionality after several unsuccessful logins. The personal data of the users, like names, and other personal data, with their authentication credentials, should be stored in a database that uses encryption techniques for proper prevention against unauthorized access and data breaches. Such an approach helps maintain user privacy but also ensures that laws regarding data protection are not violated.

1.15 USE CASE: Upload and Analyze Image



1.15.1 Description and Priority

The user will upload an X-ray image to be analyzed in real time. Priority for this case will be High.

1.15.2 Stimulus/Response Sequences

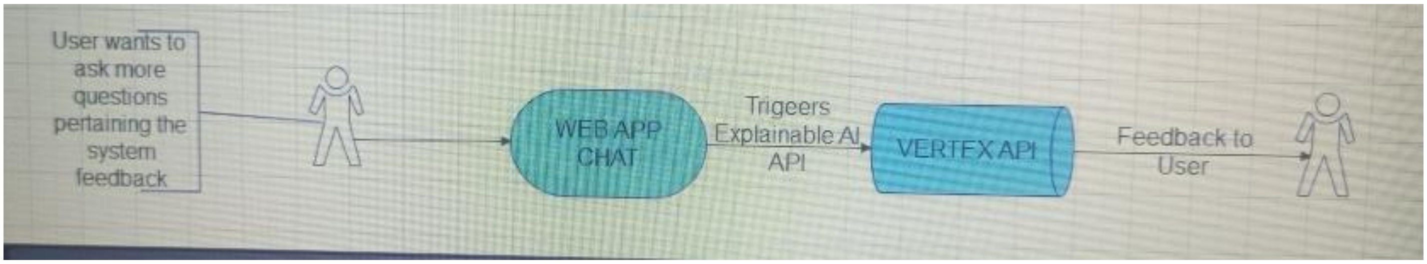
- ✓ User selects and uploads X-ray image.
- ✓ Back-end validates, processes the image.
- ✓ Results are presented to the user.

1.15.3 Functional Requirements

REQ-1: Image Format and Type Shall be Validated - The uploaded medical images should be checked by the RMIA system for acceptable types, such as JPEG, PNG, or DICOM. This check should be performed in real time on the upload of the image itself and should not allow uploads of file types not supported by the system. In a case where the file format is not right, the system should also provide feedback to users regarding the submission of valid file types.

REQ-2: Process the Image and Render the Result in Real Time - Upon uploading a valid medical image, the system should process the image using machine learning algorithms to analyze the particular medical condition specified-for instance, fractures, pneumonia, lung cancer-so that real-time rendition of the results of that analysis is direct and intuitive, along with their display for the user. This will, therefore, permit health professionals to get immediate insights from the uploaded medical images for quicker clinical decision-making.

1.16 USE CASE: Querying via Chat Option


User wants to ask more questions pertaining the system feedback
WEB APP CHAT
Trigeers Explainable AI API
VERTEX API
Feedback to User

1.16.1 Description and Priority

The "Querying via Chat Option" use case enables users to interact with the RMIA system through a chat interface. Users can ask questions related to their uploaded medical images for which they seek clarification of the results, or in general ask about the functioning of the system. The priority of this use case is high. This enhances user engagement and allows users to better understand medical analysis results more intuitively with a conversational interface.

1.16.2 Stimulus/Response Sequences

- ✓ **User Input** - The user types a query in the chat interface.
- ✓ System Processing - The system receives the input query and processes it through NLP to identify the intention of the user.
- ✓ **AI Response** - The Explainable AI component analyzes the query and returns a relevant response or explanation related to the results from the medical image analysis or about the functionalities of the system.
- ✓ **User Response** - Herein, the user gets his response from the chat interface and he can accept the information or further ask questions to get more clarity on the same.

1.16.3 Functional Requirements

REQ-1 : User Inputs Query Prompt on the Chat - The system should be able to avail the user with a chat interface through which he/she will be able to input his or her queries. The input shall allow natural language to cover many ways users may ask a question. The system shall impose a character limit per query to ensure conciseness of the communication.

REQ-2: Explainable AI Offers the Solution - The RMIA system should include an Explainable AI mechanism to obtain answers that the user will give through the queries. This module shall use the raw data from the analysis of medical images, present answers in an intelligible manner, and also support context through related information regarding the rationale behind the analysis outcomes. Responses must be tailored to answer the specific concern or question posed by the user.

1.17 EXTERNAL INTERFACES REQUIREMENTS

1.17.1 User Interfaces

The interface requirements specify how the system communicates with other software products or users for various input or output processes. The RMIA system shall interface with the user via a web-based application developed in Streamlit, to be operated using any modern web browser including but not limited to Google Chrome, Mozilla Firefox, Microsoft Edge.

1.17.2 Graphical User Interface

The GUI will be designed more user-friendly, intuitive, and consistent throughout the application. User inputs like uploading images, registering, and logging in will be provided through forms in the RMIA system; even progress bars, chat interfaces, and visualizing results shall be available. The users will interact mainly with the GUI, which will be suitable for both medical professionals and general users.

1.17.3 Command Line Interface

The RMIA system shall also have the command-line interface option for administrators and advanced users who need to perform backend configurations, database maintenance, and deployment operations.

1.17.4 Menu

It will be designed with a consistent menu appearing on all the pages of the system to enhance its usability. Menu items for navigation will include: Home, Upload Image, User Profile, and Help. Of course, each menu item will be made available to users depending on a certain level of authentication, for example, admin or a general user.

1.17.5 Messages

RMIA system will be designed with three kinds of messages:

- ✓ Informational - Friendlier messages to let users know certain actions have occurred successfully, such as "Image uploaded successfully."
- ✓ Warning - Warns users of a potential problem, for example, "The file format is not supported. Please upload an X-ray image."
- ✓ Error: Error messages will pop up when something has gone wrong, such as "Login failed. Incorrect username or password."

1.18 HARDWARE AND SOFTWARE REQUIREMENTS

1.18.1 Hardware Requirements

Server:

- ✓ CPU: Intel Xeon or equivalent
- ✓ RAM: 16 GB minimum
- ✓ Storage: 500 GB SSD/HDD
- ✓ Internet connectivity

Client:

- ✓ A digital device like a laptop, tablet, cell phone etc.
- ✓ Memory: 1 GB RAM
- ✓ Storage: at least 2 GB ROM
- ✓ Internet connectivity to use the web application

1.18.2 Software Requirements

Server:

- ✓ Operating System: Ubuntu 20.04 LTS or later
- ✓ Web Server: Nginx or Apache
- ✓ Database: PostgreSQL or any other SQL database
- ✓ Python 3.8 or later
- ✓ Django Framework 4.0 or later
- ✓ Apache Kafka, Apache Spark for streaming and processing
- ✓ Google Cloud Platform (GCP) tools and APIs

Client:

- ✓ Operating System: Android, IOS, Windows, macOS or Linux
- ✓ Web Browser: Google Chrome, Mozilla Firefox, or Microsoft Edge etc.

1.19 SECURITY REQUIREMENTS

- ✓ **User Authentication** - The system shall enforce user authentication by implementing methods of securely identifying users to ensure access to sensitive medical data.
- ✓ **Data Encryption** - All data, user credentials, and uploaded medical images shall be encrypted in motion and at rest to avoid unauthorized data access and breach.
- ✓ **Access Control** - The system shall implement role-based access control so that users have access only to such data and features of the system that are appropriate for the performance of their tasks in their system role.
- ✓ **Audit Logs** - The system shall maintain audit logs of detailed user activities in order to track access and modifications to sensitive data for accountability and traceability.
- ✓ **Secure API Communication** - All API interactions shall utilize secure protocols of communication, such as HTTPS, to maintain data integrity and confidentiality in exchanges of data between the frontend and backend.

- ✓ **Input Validation** - The system shall validate all user inputs to prevent SQL injection, cross-site scripting, and other common security vulnerabilities.
- ✓ **Session Management** - The system should ensure that session management practices are secure, including timeout of sessions, and provide a logout for users against unauthorized access through 'inactivate' sessions.
- ✓ **Password Security** - The storage of user passwords shall be in a secure manner with the use of strong hashing algorithms; application of strong password policies needs to be enforced for security.
- ✓ **Data Backup** - The system should implement a regular data backup process in order to provide data availability and recovery in case of any incident of system failure or loss of data.
- ✓ **Regulatory Compliance** - The system shall be designed to be compliant with applicable healthcare regulations and standards, including HIPAA, GDPR and DPA, set for protection of patient information and data privacy.

1.20 SOFTWARE/SYSTEM QUALITY ATTRIBUTES

- ✓ **Performance** - The system should be able to process and return results within or less than 3 seconds after the user has uploaded the image.
- ✓ **Scalability** - The system should handle a large number of concurrent users without any degradation in performance.
- ✓ **Security** - All interactions on the part of the user and all data transmissions must be encrypted with SSL/TLS.
- ✓ **Usability** - The GUI shall be intuitive, with well-visible options to navigate and by tooltips to guide the user through it.
- ✓ **Maintainability** - The system shall be easy to update and built from modular components so as to provide further scaling and enhancement more efficiently.

1.21 CONCLUSION

The RMIA system has the objective of improving health service delivery through the direct analysis of medical images. This SRS document outlines what is required in detail and guides in the development, validation, and verification of the RMIA system.

APPENDIX A: GLOSSARY

Abbreviations

RMIA – Realtime Medical Image Analyzer

SRS – System Requirements Specifications

RAM – Random Access Memory

SSD – Solid State Drive

ROM – Read Only Memory

HDD – Hard Disk Drive

GUI – Graphical User Interface

CLI – Command Line Interface

SSL – Secure Shell Layer

TLS – Transport Layer Security

HIPAA - Health Insurance Portability and Accountability Act

GDPR - General Data Protection Regulation

DPA - Data Protection Act

HTTPS – Hypertext Transfer Protocol Secure

SQL – Structured Query Language

API – Application Programming Interface

GCP – Google Cloud Platform

AI – Artificial Intelligence