Kingdom of Saudi Arabia
Ministry of Education
King Khalid University
College of Computer Science

المملكة العربية السعودية
وزارة التعليم
جامعة الملك خالد
كلية علوم الحاسب

MRI Analysis System for Brain Tumor Detection

By

Arwa Saeed	443807778
Reem Salem Ali	443814228
Sadeem Alsaleh	443804815
Mashael Ali	443806305
Faizah Alahmari	442813304

Supervised by:

Ms. Maha Asiri

Academic Year 2025-26/1446-1447 (Semester II)
Department of Computer Science
College of Computer Science
King Khalid University

This Project Documentation is submitted in partial fulfillment for the requirement for the Degree of B.Sc. in Computer Science

MAY / 2025

DHU AL-QI'DAH / 1446

COMMITTEE REPORT

We certify that we have read this graduation project report as examining committee, examined the student in its content and that in our opinion it is adequate as a project document for B.Sc. in Computer Science.

Supervisor:	Examiner1:	Examiner2:
Name: . . . . …….	Name: . . . . …….	Name: . . . . …….
Signature . . . . . . …...	Signature . . . . . . …...	Signature . . . . . . …...
Date : / /	Date : / /	Date : / /

Abstract

The early and accurate detection of brain tumors is critical for improving patient outcomes and survival rates. Traditional methods of analyzing MRI scans for brain tumor diagnosis are time-consuming and heavily reliant on the expertise of radiologists, which can lead to variability in diagnosis. To address these challenges, we propose an advanced intelligent MRI Analysis System that uses Artificial Intelligence (AI) and Deep Learning (DL) techniques for automated and precise analysis of MRI images to detect brain tumors. This system aims to assist medical professionals by providing fast, reliable, and high-accuracy diagnostic support, enabling early intervention and improved treatment planning. The proposed system analyzes MRI scans and identifies tumor regions. By integrating transfer learning with pre-trained models, the system achieves high accuracy even with limited datasets. The system is designed to classify tumor types, segment tumor boundaries, and generate detailed diagnostic reports. Additionally, it employs data augmentation and image preprocessing techniques to enhance the quality of input images and improve model performance. This AI-driven solution addresses the growing need for efficient and accurate diagnostic tools in healthcare. By reducing the time required for diagnosis and minimizing human error, the system has the potential to revolutionize brain tumor detection, ultimately improving patient care and treatment outcomes. The proposed system is a step forward in integrating AI into medical imaging, offering a scalable and accessible tool for healthcare providers worldwide.

Table of Content

Table of Figures

List of Tables

Chapter 1: Introduction and Background

1.0 Overview

In this chapter, the project is introduced by stating its goal and scope, followed by an explanation of the motivation behind its development. The chapter also outlines the aims and objectives of the project and discusses its overall significance. Additionally, it summarizes the methods utilized in the development of the system.

1.1 Project Goal

The primary goal of this project is to design and implement an AI-driven MRI analysis system that leverages advanced deep learning techniques for the automated detection and classification of brain tumors. This system is intended to support medical professionals by offering fast, accurate, and reliable diagnostic insights, thereby reducing the time required to identify brain tumors and minimizing the potential for human error. By enhancing the diagnostic process, the system aims to improve the accuracy of tumor detection, streamline workflow efficiency, and ultimately contribute to better patient care. With the integration of cutting-edge AI technology, the project envisions significantly improving clinical outcomes, optimizing treatment planning, and providing healthcare practitioners with valuable decision-making assistance in the early stages of diagnosis.

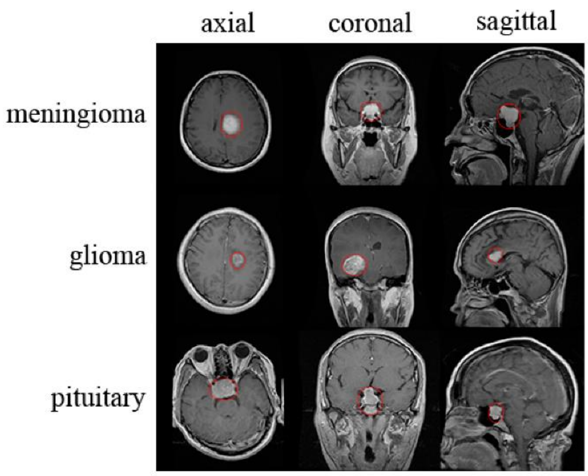

Figure 1 Detection and Classification of Brain Tumors from MRI Images

1.2 Problem Statement

The problem is that detecting brain tumors accurately and efficiently remains a challenge in the medical field. Manual analysis of MRI scans is time-consuming, prone to human error, and requires specialized expertise, which may not always be readily available. This can lead to delays in diagnosis, impacting timely treatment and patient outcomes. Additionally, there is a lack of automated systems that can assist medical professionals in identifying and classifying brain tumors with high precision. We need a solution that leverages advanced image processing and machine learning techniques to analyze MRI scans, providing faster, more accurate, and reliable detection of brain tumors to support medical decision-making.

1.3 Project Scope

The scope of the project includes the development of software that facilitates the automatic analysis of brain MRI scans to assist in the detection and classification of brain tumors. The system focuses on utilizing advanced image processing techniques and machine learning algorithms to accurately segment tumor regions and differentiate between tumor types. The targeted users for this project are healthcare professionals, radiologists, and medical institutions involved in neurological diagnosis and treatment.

Within the scope of the project, the software aims to provide a user-friendly interface that allows users to upload MRI images, automatically process and analyze them, and receive detailed visual and textual reports highlighting tumor presence and characteristics. The system will incorporate key functionalities such as image pre-processing, tumor segmentation, feature extraction, and tumor classification, ensuring a reliable and efficient diagnostic aid.

1.4 Objectives and Hypothesis

This section outlines the main goals of the MRI Analysis System for Brain Tumor Detection and presents the hypothesis that guides the system’s development. Clearly defined objectives and a testable hypothesis provide direction and help evaluate the system’s effectiveness in supporting early and accurate tumor diagnosis.

The main objectives are as follows:

- Develop an AI system to detect brain tumors in MRI scans with high accuracy and low false positives.
- Automate MRI analysis using deep learning to reduce reliance on manual interpretation.
- Improve diagnostic speed and efficiency to support timely medical decisions.
- Classify brain tumor types and segment boundaries for detailed treatment planning.
- Apply transfer learning to boost model accuracy, even with limited data.
- Enhance MRI image quality through preprocessing and data augmentation.
- Minimize human error by ensuring consistent, AI-driven diagnostic results.
- Ensure the system is scalable and accessible for global healthcare use.
- Generate detailed diagnostic reports to support clinical decision-making.
- Promote AI integration in medical imaging to advance healthcare diagnostics.

The main hypothesis are as follows:

- Can an AI-powered system detect brain tumors in MRI scans with high accuracy and minimal false positives?
- Can deep learning techniques automate MRI analysis as effectively as manual interpretation by radiologists?
- Can the system accurately classify different types of brain tumors and effectively segment tumor boundaries?
- Can the use of transfer learning enhance model performance and accuracy, even with a limited MRI dataset?

1.5 Project Motivation

The traditional methods for diagnosing brain tumors, such as manual analysis of MRI scans by radiologists, often come with challenges like long processing times, human error, and variability in results due to the reliance on the expertise of individual professionals. These issues can delay diagnosis, lead to misinterpretations, and ultimately affect patient outcomes. Early detection of brain tumors is crucial for effective treatment, as it significantly improves the chances of successful intervention and recovery. However, the complexity and volume of MRI data often make it difficult for healthcare providers to diagnose tumors promptly and accurately. This creates a critical need for an efficient and reliable solution that can assist medical professionals in detecting brain tumors at an early stage, with a high degree of accuracy. The proposed project seeks to address this problem by developing an AI-powered MRI analysis system that uses deep learning techniques to automatically detect and classify brain tumors, ultimately improving diagnostic accuracy, reducing time to diagnosis, and enhancing treatment planning for better patient care.

1.6 Project Significance

- Improved Diagnostic Accuracy: The AI-based system enhances the accuracy of brain tumor detection, reducing human error and diagnostic variability.
- Faster Tumor Detection: By automating the analysis of MRI scans, the system significantly speeds up the detection process, enabling early intervention.
- Support for Healthcare Professionals: It serves as a reliable decision-support tool, complementing the expertise of radiologists and improving diagnosis efficiency.
- High Performance with Limited Data: The integration of transfer learning with pre trained models ensures high accuracy, even with limited medical imaging datasets.
- Enhanced Image Quality: Advanced image preprocessing and data augmentation techniques improve the quality of MRI scans, resulting in better tumor classification and segmentation.
- Early Medical Intervention: By minimizing diagnostic delays, the system facilitates quicker diagnosis, leading to more effective treatment plans and better patient outcomes.

1.7 Method

1.7.1 Machine Learning Algorithms and Techniques

- **Convolutional Neural Networks (CNNs)**: A type of deep learning model particularly well-suited for image analysis. Used for analyzing MRI images and detecting brain tumors with high accuracy.
- **Deep Learning:** A subset of machine learning that automatically learns representations from data. In this project, deep learning techniques form the core of the AI system used for enhancing recognition accuracy through feature extraction and pattern recognition from MRI scans.
- **Image Segmentation:** Uses models like U-Net to identify tumor boundaries within MRI scans.

- **Classification Algorithms:** Categorizes tumors based on type and severity .

1.7.2 Frameworks and Libraries

- **OpenCV:** A computer vision library used for implementing image preprocessing techniques, including noise reduction and contrast enhancement.
- **TensorFlow:** A powerful machine learning and deep learning framework used for developing and training AI models for tumor detection and segmentation.

1.7.3 Programming and Web Technologies

- **PHP:** A server-side scripting language used for developing backend services, handling requests, and managing database interactions.
- **HTML & CSS:** Technologies used to create and design the web interface for displaying diagnostic results, reports, and user interactions.
- **Python:** The main programming language used to implement AI models, preprocessing routines, and backend logic for medical image analysis.

1.8 Summary

This chapter provides an overview of the MRI Analysis System for Brain Tumor Detection. It presents the project idea, clearly defining the goal and motivation behind the development of this system. Furthermore, the chapter emphasizes the significance of the project in improving the accuracy and efficiency of brain tumor diagnosis. The objectives and scope of the project are outlined, detailing its intended impact within the medical field. Lastly, the chapter introduces a brief explanation of the software tools applied.

1.9 Report Outline

The remaining chapters of this report are structured as follows:

Chapter 2 provides a literature review, presenting an overview of relevant research and offering deeper insight into existing methods for MRI analysis and brain tumor detection. Various existing solutions are reviewed and compared to the proposed system.

Chapter 3 focuses on project management, discussing the development approach, risk management strategies, and project planning applied throughout the creation of the system.

Chapter 4 covers the requirements and analysis phase, detailing the system requirements and showcasing the development of personas, system models, and the overall system architecture.

Chapter 5 describes the system design, including product features, user interface structure, data storage, and high-level architecture. It also presents key design diagrams such as sequence, activity, and class diagrams to illustrate system functionality and component interaction.

Chapter 6 concludes the report by summarizing the project outcomes and highlighting the system's contribution to brain tumor detection. It also outlines future work, including planned improvements, testing, and feature expansion in the next development phase.

Chapter 7 lists the literature cited throughout the report, referencing the scholarly sources, studies, and tools that informed the research and system development.

Chapter 8 provides acknowledgments, expressing gratitude to individuals and institutions who supported and contributed to the successful completion of the project.

Chapter 2: Literature Review

2.0 Overview

This chapter outlines the theoretical foundation upon which the MRI analysis system for brain tumor detection is built. It presents a review of pertinent literature related to medical image processing, machine learning techniques, and MRI-based diagnostic tools. Furthermore, it details how these underlying concepts have been applied within the context of our system. The chapter also examines several existing systems currently used for brain tumor detection, followed by a comparative analysis highlighting the differences and improvements offered by our proposed solution. Finally, a summary is provided to consolidate key insights from the chapter.

2.1 Background

Magnetic Resonance Imaging (MRI) has played a crucial role in medical diagnostics, particularly in the detection and analysis of brain tumors. Over the years, advancements in MRI-based tumor detection have been driven by the need for early and accurate diagnosis to improve patient outcomes. Early methods for tumor identification relied heavily on manual interpretation by radiologists, facing challenges such as subjectivity, variability in expertise, and time-consuming analysis. As computational power increased in the 1990s, rule-based systems were introduced, integrating predefined image processing techniques to enhance tumor detection. However, these approaches were often limited by the complexity of brain tumor characteristics and variations in imaging quality. With the rise of machine learning in the early 2000s, data-driven methodologies using techniques such as Support Vector Machines (SVM), Decision Trees, and Artificial Neural Networks improved the accuracy and efficiency of tumor classification. In recent years, deep learning techniques, including Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), have demonstrated remarkable performance in MRI-based brain tumor detection. These methods utilize spatial and temporal features from MRI scans to enable more precise, automated, and real-time diagnosis, reducing human error and enhancing clinical decision-making.

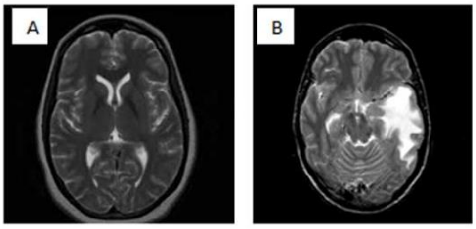

Figure 2 Two Brain images: (A) Normal brain (B) Tumor Brain

2.2 Techniques and Methodologies

To develop an efficient and accurate AI-driven MRI analysis system for brain tumor detection, various advanced techniques and methodologies are employed. This project leverages deep learning, transfer learning, and image processing methods to enhance the accuracy and reliability of tumor identification. The process involves data collection, preprocessing, tumor segmentation, classification, and model optimization to ensure high performance. Additionally, the system is designed to generate diagnostic reports and provide real-time assistance to medical professionals. The following sections outline the key techniques and methodologies used in this project:

2.2.1 Data Collection and Preprocessing

- Dataset Acquisition: MRI scan datasets will be collected from publicly available medical imaging databases.
- Image Preprocessing: Techniques such as noise reduction, contrast enhancement, and normalization will be applied to improve image quality.
- Data Augmentation: Methods like rotation, flipping, scaling, and adding synthetic noise will be used to enhance model robustness and mitigate overfitting.

2.2.2 Deep Learning and Transfer Learning

- Model Selection: Pre-trained convolutional neural networks (CNNs) such as VGG16, ResNet, or EfficientNet will be used as feature extractors.
- Transfer Learning: Fine-tuning of pre-trained models will be applied to improve accuracy while training on a limited dataset.
- Custom CNN Architectures: If necessary, a custom deep learning model will be designed and trained from scratch for tumor detection and classification.

2.2.3 Tumor Detection and Segmentation

- Object Detection Algorithms: Region-based CNNs (R-CNN, Faster R-CNN) or YOLO (You Only Look Once) will be explored for tumor localization.
- Image Segmentation Techniques: U-Net, Mask R-CNN, or other semantic segmentation models will be implemented to accurately delineate tumor boundaries.

2.2.4 Classification of Brain Tumors

- Multi-Class Classification: A deep learning model will classify different tumor types (e.g., glioma, meningioma, pituitary tumor) based on MRI scans.
- Feature Extraction and Analysis: CNNs will be used to extract relevant features from MRI images for better classification performance.

2.3 Overview of Existing Systems

Many systems have been created to solve the problem we are focusing on, using different methods and techniques. In this section, we will look at some of the systems that are currently available for MRI brain scans analysis.

Some examples of these available systems Include :

- Fuzzy C-Means (FCM) Clustering-Based System
- U-Net Based Deep Learning Systems
- Thresholding & Manual Segmentation-Based Systems

2.3.1 Fuzzy C-Means (FCM) Clustering-Based System

Overview:

The Fuzzy C-Means (FCM) algorithm is a widely used unsupervised clustering method in medical imaging for brain tumor detection. It classifies pixels in MRI scans into clusters based on their intensity values. Each pixel is assigned a degree of membership to each cluster, which helps in identifying tumor regions.

Example Tool: MATLAB

Description: MATLAB provides built-in support for Fuzzy C-Means clustering, often used in medical image segmentation tasks. Users can apply the fcm function from MATLAB's Image Processing Toolbox to segment MRI images based on pixel intensity values.

Strengths:

- Simple and computationally efficient.
- Works well on images with well-separated intensity values.
- Requires minimal prior training data.

Limitations:

- Sensitive to noise and intensity inhomogeneity in MRI scans.
- Requires manual parameter tuning, which may affect accuracy.
- Limited ability to accurately classify tumor types or segment complex tumor boundaries.
- Highly dependent on radiologists to interpret results.

2.3.2 U-Net Based Deep Learning Systems

Overview:

U-Net is a convolutional neural network architecture specifically designed for biomedical image segmentation. It has been used extensively for brain tumor detection tasks, offering precise segmentation by learning from large, annotated datasets.

Example Tool: NiftyNet (Medical Imaging Platform)

Description: NiftyNet is an open-source TensorFlow-based platform specifically designed for medical imaging. It includes pre-built U-Net models used widely for tasks like brain tumor segmentation from MRI scans.

Strengths:

- Provides high-quality tumor segmentation.
- Automates part of the tumor detection process.

- Capable of learning complex features from MRI images.

Limitations:

- Requires a large, annotated dataset for training, which may not always be available.
- Performance can degrade with limited or imbalanced data.
- Typically focuses only on segmentation without providing detailed classification reports or diagnostic summaries.
- May require significant computational resources.

2.3.3 Thresholding & Manual Segmentation-Based Systems

Overview:

Traditional systems rely on manual thresholding techniques, where radiologists adjust intensity thresholds to highlight potential tumor regions. Further segmentation is performed manually based on the radiologist's expertise.

Example Tool: 3D Slicer

Description: 3D Slicer is one of the most popular open-source software platforms for medical image analysis. It allows radiologists and researchers to manually segment brain tumors using thresholding, painting, and other manual tools.

Strengths:

- Straightforward and easy to implement.
- Useful in basic scenarios with clear intensity contrasts.

Limitations:

- Highly time-consuming and labor-intensive.
- Results vary significantly depending on the skill level of the radiologist.
- Prone to human error and inconsistent diagnoses.
- Cannot handle complex or subtle tumor features.
- Lacks automation, reporting capabilities, and advanced image processing techniques.

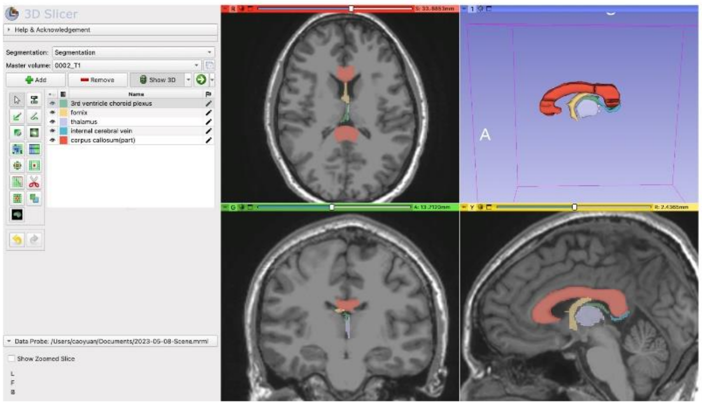

Figure 3 Manual segmentation using 3D Slicer

2.4 Advantages of Proposed Solution Compared to Existing Systems

Main Features	Our project	Fuzzy C-Means (FCM) System	U-Net Deep Learning System	Thresholding And Manual Segmentation
Platform	Web App	MATLAB	Nifty Net (TensorFlow)	3D Slice (Desktop Software)
Real-time Tumor Detection	Yes	No	Yes	No
Use of Deep Learning	Yes	No	Yes	No
Tumor Segmentation	Yes(U-Net)	Yes (Clustering)	Yes (Deep (Learning	Yes (Manual)
Classification of Tumors	Yes (AI-based)	No	No	No
Automated Report Generation	Yes	No	No	No
Noises Intensity Correction	Yes	No	Limited	No
Dependency on Radiologists	Low	High	Medium	High
All Features are Free	Yes	Requires MATLAB License	No (Open-source, but resource-heavy)	Yes (Open source)

Table 1 Comparison Table

2.5 Summary

In conclusion, this chapter provides a comprehensive and well-structured literature review that establishes a strong theoretical foundation for the project. By critically analyzing and synthesizing previous research, methodologies, and technologies, it sets the stage for the subsequent sections, facilitating a deeper understanding of MRI-based brain tumor detection systems and the advancements introduced by our proposed solution.

Chapter 3: Project Management

3.0 Overview

This chapter outlines the methodologies employed in the development of the MRI Analysis System for Tumor Detection, with a focus on the waterfall methodology. The waterfall model is chosen for its clear, sequential approach, which ensures a structured and systematic implementation process. This methodology is well-suited for projects requiring well-defined stages, such as system design, data preprocessing, and model training. However, its rigidity may pose challenges if technical adjustments or modifications are needed. Additionally, this chapter defines the risk management strategies and project planning framework to mitigate potential obstacles during development.

3.1 Approach

The MRI Analysis System will be developed using the waterfall methodology, as it ensures detailed project planning before implementation, helping to avoid complications and future issues. The waterfall approach guarantees high-quality outcomes by allowing testing at each phase of the project before proceeding to the next.

- **Requirements Gathering and Analysis:** This is the first and most crucial step in creating any software application. The team must first collect and analyze the system's requirements, focusing on understanding the needs of medical professionals, system administrators, and end users. After gathering the requirements, they will be analyzed to ensure they are thorough, consistent, and feasible.
- **Design:** After gathering and reviewing the requirements, the team can begin designing the system. The design process will involve using various tools and techniques, such as UML diagrams, to create a detailed architecture that aligns with the project's goals of detecting brain tumors through MRI analysis.
- **Implementation:** Next, the team will develop the system according to the approved design, writing code and integrating components to form a functioning MRI analysis system capable of processing and detecting brain tumors accurately.

- **Testing:** The system will undergo rigorous testing to ensure that it meets all specified requirements and is free of defects. This phase will include unit testing, integration testing, and system testing to verify the functionality and performance of the MRI analysis system.
- **Maintenance:** Once the MRI analysis system is evaluated, tested, and approved, it will be deployed for use by healthcare professionals. The maintenance phase will follow, during which the team will address any bugs and implement user-requested improvements to enhance the system's functionality and accuracy over time.

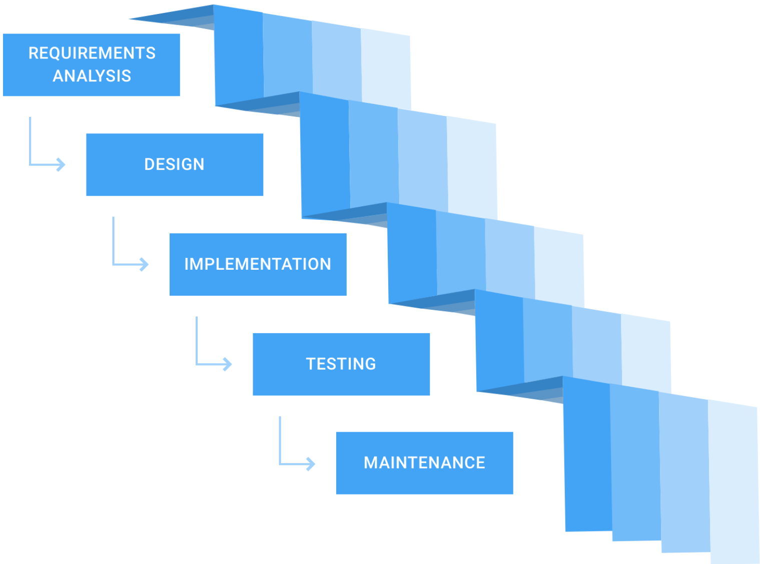

Figure 4 Waterfall Model

3.2 Risk Management

Key Risks:

- **Detection Accuracy**: The system may not always achieve 100% accuracy in detecting brain tumors, potentially leading to misdiagnoses or missed tumors. Variability in MRI scans, such as differing image quality or patient-specific factors, could affect the algorithm's performance.
- **Complexity of Interpretation**: Users, such as radiologists or healthcare professionals, may face challenges in interpreting the system's results, especially if the system provides false positives or negatives. This could result in delays in treatment or incorrect diagnoses.

- **User Adoption and Training**: Healthcare professionals may be resistant to adopting the MRI analysis system, particularly if they are unfamiliar with the technology. Additionally, training staff to use the system effectively may take time and resources, slowing down the integration into clinical practice.
- **Data Privacy and Security**: The system will process sensitive medical data, and improper handling or security vulnerabilities could expose users to risks of data breaches. This may lead to loss of patient confidentiality and non-compliance with regulations such as HIPAA (Health Insurance Portability and Accountability Act).
- **System Reliability and Downtime**: Any downtime or system failure could disrupt clinical workflows and delay critical diagnoses, leading to potential harm to patients. Additionally, reliance on technology could create challenges if the system is unavailable during emergencies.

Mitigation Strategies:

- **Enhanced Detection Algorithms**: Develop and train the detection algorithms using a large, diverse set of high-quality MRI scans, ensuring that the system can handle various types of tumors and MRI variations. Continuous improvement and validation using new datasets will help enhance the system's accuracy and reliability over time.
- **Collaboration with Medical Experts**: Collaborate closely with radiologists, neurosurgeons, and other medical professionals during the development process. This will ensure that the system's outputs are practical, interpretable, and aligned with clinical needs. Regular consultations will also help in refining the user interface and making the results more actionable.
- **Comprehensive User Training and Support**: Provide thorough training for healthcare professionals on how to use the system effectively, including clear guidelines on interpreting the results. Additionally, offer ongoing support through customer service and training resources to ensure that users feel confident in utilizing the system.

- **Strict Data Security Measures**: Implement robust encryption protocols for data storage and transmission to protect patient information. Ensure the system complies with relevant privacy laws and regulations, such as HIPAA, and establish stringent access controls to limit who can view sensitive patient data. Regular security audits will be conducted to identify vulnerabilities.
- **Reliable System Design and Redundancy**: Design the system with redundancy and fail-safes to minimize downtime. Implement a monitoring system to ensure that the MRI analysis system operates reliably in real-time. Establish backup protocols and disaster recovery plans to ensure the system is quickly restored in case of a failure.
- **Feedback and Continuous Improvement**: After initial deployment, establish a feedback loop with users to continuously assess and improve the system. This will help identify any pain points, user concerns, or areas for enhancement, allowing for incremental updates to improve the system's accuracy, usability, and effectiveness in clinical settings.

3.3 Project Plan

3.3.1 FrameTime Table

ID	Task Name	Start	End	Duration
1	Define idea of project.	15/02/2025	17/02/2025	3 Days
2	Submission of proposed idea.	17/02/2025	17/02/2025	1 Day
3	Prepare project plan.	20/02/2025	21/02/2025	2 Days
4	Identify the team roles and Responsibilities.	23/02/2025	23/02/2025	1 Day
5	Write chapter 1.	24/02/2025	02/03/2025	1 Week
6	Chapter 1 Submission.	03/03/2025	03/03/2025	1 Day

7	Conduct research on related literature.	04/03/2025	05/03/2025	2 Days
8	Write chapter 2.	06/03/2025	12/03/2025	1 Week
9	Chapter 2 Submission.	13/03/2025	13/03/2025	1 Day
10	Define the approach of the project.	14/03/2025	14/03/2025	1 Day
11	Define risk management plan.	15/03/2025	15/03/2025	1 Day
12	Write chapter 3.	16/03/2025	22/03/2025	1 Week
13	Chapter 3 Submission.	23/03/2025	23/03/2025	1 Day
14	Gather the functional and non-functional requirements.	24/03/2025	25/03/2025	2 Days
15	Design UML diagrams.	26/03/2025	27/03/2025	2 Days
16	Write chapter 4.	28/03/2025	03/04/2025	1 Week
17	Chapter 4 Submission.	04/04/2025	04/04/2025	1 Day
18	Prepare Mid Presentation.	05/04/2025	06/04/2025	2 Days
19	Define product features.	13/04/2025	15/04/2025	3 Days
20	Design User Interface prototypes.	16/04/2025	18/04/2025	3 Days
21	Write chapter 5.	19/04/2025	25/04/2025	1 Week
22	Chapter 5 submission.	26/04/2025	26/04/2025	1 Day

23	Define future work.	27/04/2025	28/04/2025	2 Days
24	Write chapter 6.	29/04/2025	02/05/2025	4 Days
25	Chapter 6 submission.	03/05/2025	03/05/2025	1 Day
26	Prepare Final Presentation	04/05/2025	05/05/2025	2 Days

Table 2 FrameTime Table

3.3.2 Gantt Chart

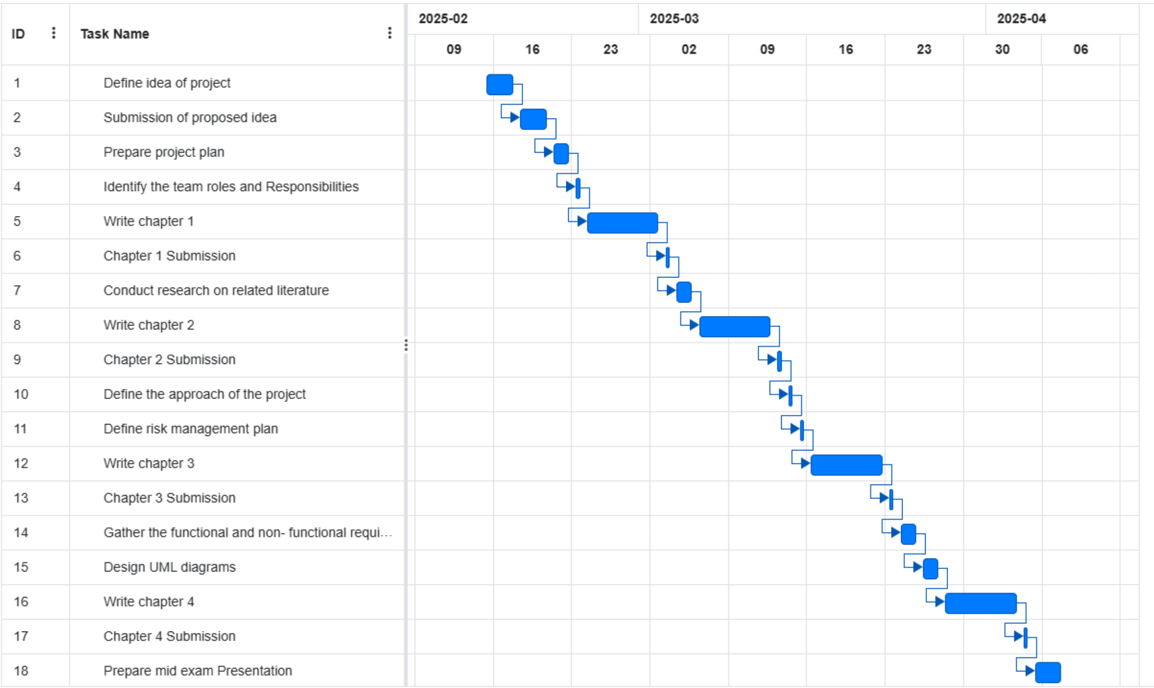

Figure 5 Gantt Chart

3.3.3 The Team: Roles and Responsibilities

- Team Roles:

Name	Role
Reem Salem Ali	Leader
Arwa Saeed Alarram	Member
Sadeem Alsaleh	Member
Mashael Ali	Member
Faizah Alahmari	Member

Table 3 Team Roles

- Team Responsibilities:

ID	Task Name	Team Member
1	Define idea of project.	Arwa Saeed
2	Submission of proposed idea.	Reem Salem
3	Prepare project plan.	All Members
4	Identify the team roles and Responsibilities.	All Members
5	Write chapter 1.	All Members
6	Chapter 1 Submission.	Arwa Saeed
7	Conduct research on related literature.	All Members

8	Write chapter 2.	All Members
9	Chapter 2 Submission.	Arwa Saeed
10	Define the approach of the project.	All Members
11	Define risk management plan.	All Members
12	Write chapter 3.	All Members
13	Chapter 3 Submission.	Arwa Saeed
14	Gather the functional and non-functional requirements.	All Members
15	Design UML diagrams.	All Members
16	Write chapter 4.	All Members
17	Chapter 4 Submission.	Arwa Saeed
18	Prepare Mid Presentation.	All Members
19	Define product features.	All Members
20	Design User Interface prototypes.	Arwa Saeed Sadeem Alsaleh
21	Write chapter 5.	All Members
22	Chapter 5 submission.	Arwa Saeed
23	Define future work.	All Members
24	Write chapter 6.	All Members

25	Chapter 6 submission.	Arwa Saeed
26	Prepare Final Presentation	All Members

Table 4 Team Responsibilities

3.4 Summary

In this chapter, the methodology used, which is the waterfall methodology, was documented. The risks that we may face while building the system and the solutions that can reduce the risks were also explained. Finally, the project plan was drawn up.

Chapter 4: Requirements and Analysis

4.0 Overview

The analysis phase is a crucial step in the development process where we define and study the requirements of the system. By doing so, we can effectively move onto the design phase with a clear understanding of what the system must accomplish and what features it must possess. This stage is an essential moment as it helps establish the foundation of the project and with careful analysis, we can create a roadmap for the project and streamline the design phase, ultimately leading to a successful implementation.

4.1 System Requirements

4.1.1 Functional Requirements

Functional requirements specify the core operations that the system must perform to achieve its intended purpose. The key functional requirements of the system include:

- MRI Image Input: The system should accept MRI scans in standard formats (e.g., DICOM, PNG, JPG).
- Image Preprocessing: The system should enhance image quality by applying noise reduction, contrast adjustment, and normalization techniques.
- Tumor Detection: The system should accurately detect the presence of brain tumors in MRI scans.
- Tumor Segmentation: The system should identify and outline tumor regions within MRI images.
- Tumor Classification: The system should classify detected tumors into different categories (e.g., glioma, meningioma, pituitary tumor).
- Report Generation: The system should generate a diagnostic report summarizing the analysis results, including tumor location, type, and confidence score.
- User Interface: The system should provide an intuitive interface for medical professionals to upload images, view results, and download reports.
- Model Training and Updates: The system should allow continuous improvement by incorporating new MRI data for model retraining and optimization.
- Integration with Hospital Systems: The system should be compatible with existing medical databases and electronic health record (EHR) systems.

4.1.2 Non-Functional Requirements

Non-functional requirements define the quality attributes and constraints that the system must adhere to, ensuring its usability, security, and efficiency. These include:

- Accuracy: The system should achieve high accuracy in tumor detection and classification.
- Performance: The system should process MRI scans and generate results within a few seconds to ensure real-time diagnosis.
- Scalability: The system should be scalable to handle an increasing number of MRI scans and users without performance degradation.
- Security: The system should implement strong security measures, including data encryption, access control, and secure authentication.
- Reliability: The system should operate with minimal downtime and provide consistent diagnostic results.
- Usability: The system should have a user-friendly interface that is easy for medical professionals to navigate.
- Maintainability: The system should be designed for easy updates, bug fixes, and improvements without major disruptions.
- Compatibility: The system should be compatible with different hardware configurations and operating systems used in medical institutions.

4.1.3 Hardware Requirements

Table 5 shows the minimum laptop specifications required to develop and test the proposed MRI Analysis System for Brain Tumor Detection web application.

Requirement	**Minimum**
CPU	Intel Core i7 (8th Gen or higher) / AMD Ryzen 7
Memory (RAM)	16 GB RAM
Hard Disk	512 GB SSD
GPU (optional)	NVIDIA GTX 1650 or better
Display	Full HD (1920x1080)

Table 5 Hardware Requirements

4.1.4 Software Requirements

The development of the proposed MRI analysis system will require the following software tools, libraries, and technologies to support front-end development, backend processing, machine learning, and data management:

- **HTML & CSS**: Required for front-end development, ensuring a user-friendly and responsive design for the web application.
- **JavaScript**: Used for front-end interactivity, including dynamic features like image display and user input handling.
- **Backend Language (TBD)**: A backend programming language (e.g., Python with Flask/Django or Node.js) will be used to handle the server-side logic, data processing, and API interactions.
- **Python**: Essential for machine learning and artificial intelligence tasks, particularly for image analysis and tumor detection using deep learning models (TensorFlow, PyTorch, etc.).
- **Database (e.g., MySQL or PostgreSQL)**: A relational database management system to store user data, MRI images, and tumor detection results.
- **Visual Studio Code**: A lightweight and powerful code editor for efficient software development.
- **OpenCV** : A computer vision library used for image pre-processing, augmentation, and possibly aiding the tumor detection process.
- **TensorFlow / PyTorch**: Deep learning frameworks for building and training models that analyze MRI images for brain tumor detection.

4.2 Personas

To better understand the users' expectations, needs, and challenges, the following personas were developed to represent key stakeholders who will interact with the MRI analysis system:

	Persona 1	Persona 2	Persona 3
Name	Dr. Sami – Radiologist at a Government Hospital	Lama – Medical Intern Student	Mr. Khaled – IT Manager at a Private Healthcare Facility
Age	45	24	38
Job	Radiologist	Medical intern	IT Manager
Location	Major medical city	University hospital	Private hospital
Goals	• Speed up MRI scan analysis for accurate brain tumor diagnosis. • Reduce human error in interpreting images.	• Learn how to identify brain tumors using MRI scans. • Use AI as an educational tool to enhance diagnostic skills.	• Seamlessly integrate the system with existing hospital IT infrastructure. • Ensure patient data privacy and security.
Needs	• Fast and user-friendly interface. • AI-powered, accurate diagnostic reports. • Ability to review and adjust AI-generated results.	• Simple interface with clear explanations of each case. • Ability to compare AI results with expert diagnoses.	• Secure system compliant with privacy regulations (e.g., HIPAA). • Compatibility with PACS and EHR systems.
Challenges	• Heavy daily workload and high number of cases. • Limited time for manual analysis of each image.	• Lack of experience in reading MRI scans. • Needs supportive and educational resources.	• Difficulty integrating new systems with legacy platforms. • Ensuring the reliability and transparency of the AI model.

Table 6 Personas

4.3 System Models

4.3.1 Use Case Diagram

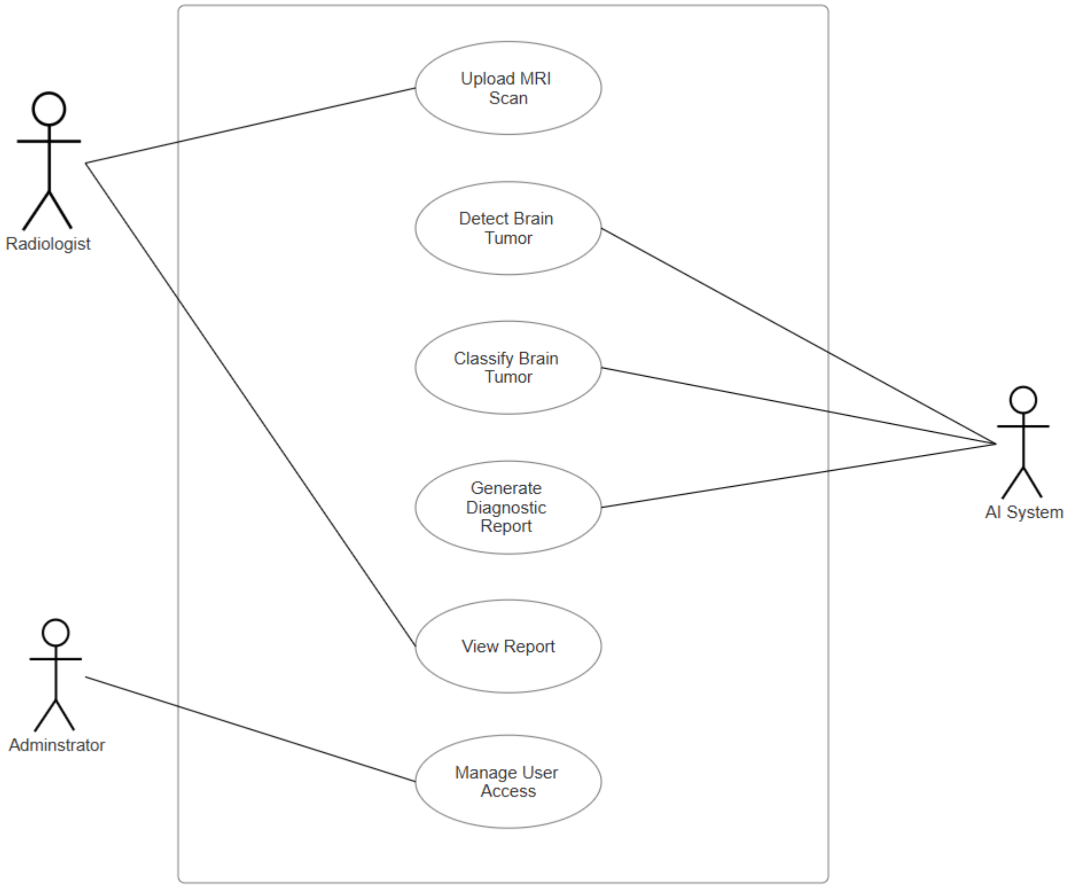

Figure 6 Use Case Diagram

- Description:

Use Case Name:	**MRI Brain Tumor Detection**
Primary Actor:	Radiologist / Medical Technician
Stakeholders and Interests:	1-Radiologist: Requires a reliable system for accurate and efficient tumor detection to support diagnosis. 2-Patient: Benefits from a quicker and more accurate diagnosis, enabling timely treatment. 3-Hospital/Medical Institution: Seeks to improve diagnostic efficiency and reduce manual workload.

Preconditions:	The MRI scan has been completed and is available in a compatible digital format. The system is online and integrated with the imaging database.
Postconditions:	The system provides a detailed analysis of the MRI scan, highlighting potential tumor regions. Results are stored and accessible for further medical review.
Main Success Scenario:	1-Radiologist logs into the MRI analysis system. 2-The system retrieves the MRI scan from the database. 3-The scan is processed using the tumor detection model. 4-Detected tumor regions are highlighted and classified. 5-A diagnostic report is generated, including visualizations and confidence scores. 6-The radiologist reviews the results and confirms or adjusts the findings. 7-Final report is saved and linked to the patient's medical record.
Extensions:	1-If the scan format is not supported, the system alerts the user and requests a compatible version. 2-If the model is uncertain, it flags the scan for manual review with low-confidence markers. 3-Radiologist adds notes or overrides model output based on medical judgment.

Table 7 Use Case Description

4.3.2 Class Diagram

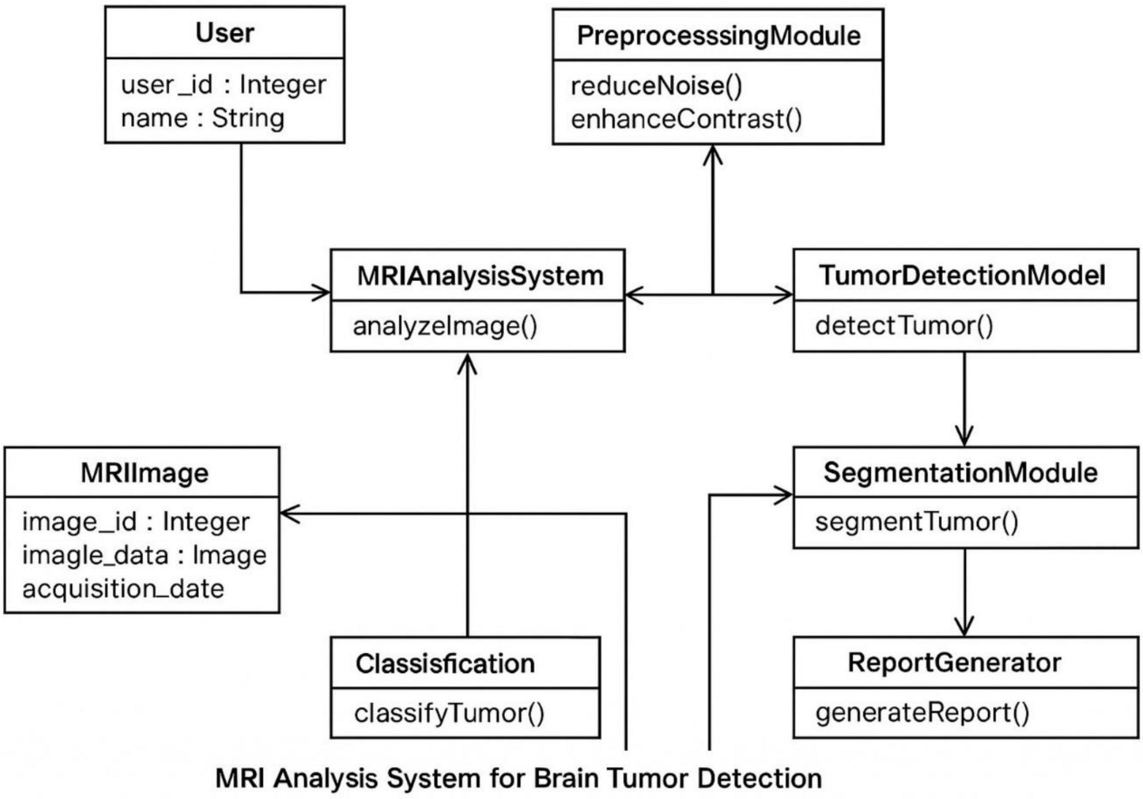

Figure 7 Class Diagram

- Description:

The class diagram illustrates the core structure of the MRI analysis system for brain tumor detection. It defines essential classes such as *Patient*, *MRIImage*, *TumorDetectionModel*, *AnalysisReport*, and *Radiologist*, along with their key attributes and methods. The diagram shows how MRI images are linked to patients and processed by the detection model to generate analysis reports. Radiologists review these reports, while system users manage access and authentication. This diagram provides a clear overview of the system's components and their relationships, forming the foundation for its functionality and data flow.

4.3.3 Activity Diagram

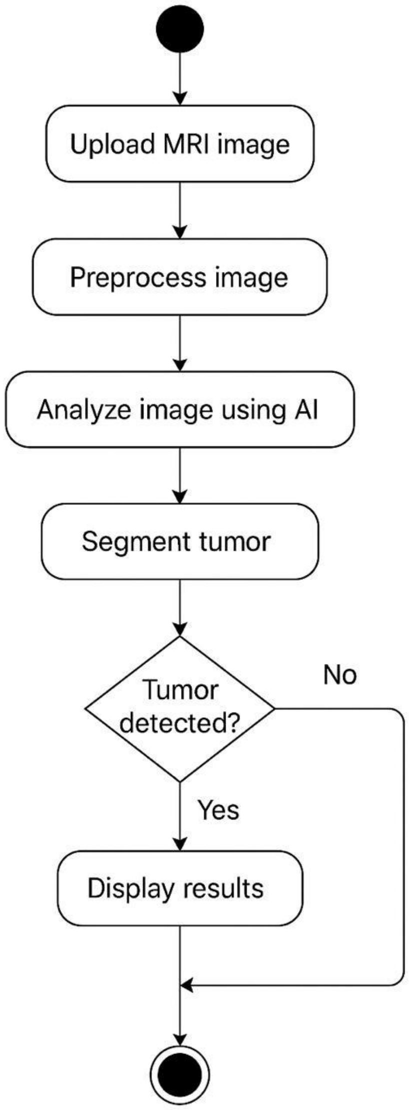

Figure 8 Activity Diagram

- Description:

This activity diagram shows the MRI-based brain tumor detection process. It starts with uploading an MRI scan, followed by image preprocessing to enhance quality. The AI model then analyzes the scan, detects the tumor, and segments it. The system displays the results, classifying the tumor as benign or malignant. Finally, the doctor reviews the report and decides on the next steps, such as monitoring or treatment.

4.3.4 Sequence Diagram

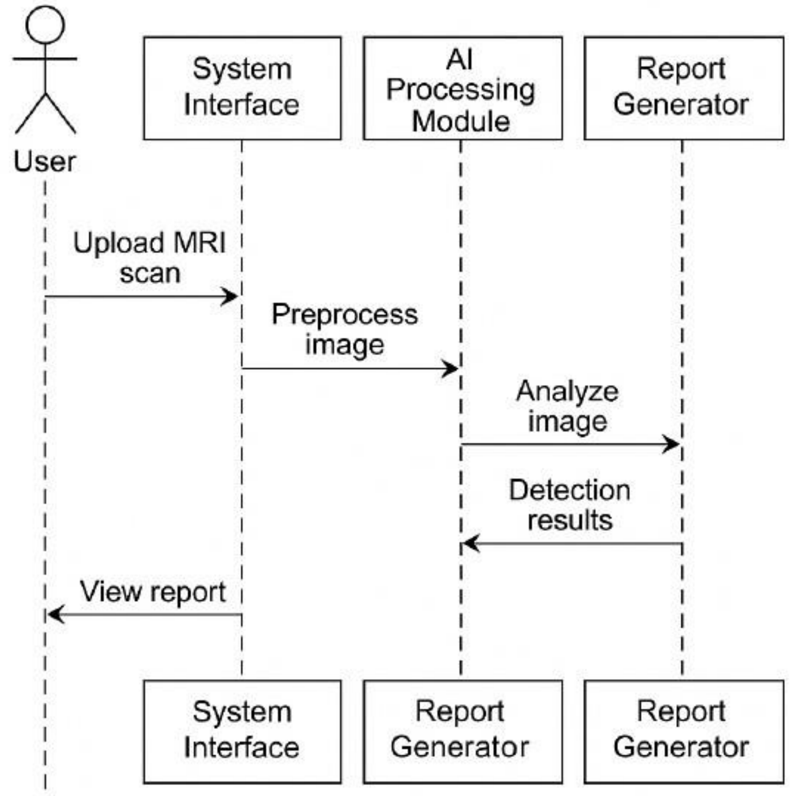

Figure 9 Sequence Diagram

- Description:

The sequence diagram illustrates the interaction between the user and the MRI analysis system throughout the scan processing workflow. The process begins when the user uploads an MRI scan via the System Interface. The AI Processing Module then handles image preprocessing, followed by tumor detection and analysis. Once the analysis is complete, the results are passed to the Report Generator, which compiles a comprehensive diagnostic report. Finally, the user can access and view the report through the same interface. This diagram clearly demonstrates the sequential flow of actions and data, highlighting the role of each system component from scan upload to report delivery.

4.4 Summary

This chapter has effectively outlined the system requirements and analysis. As a result of this chapter, team members now have a comprehensive understanding of the proposed system and a clear grasp of its intended functions.

Chapter 5: System Design

5.0 Overview

System design is a fundamental phase of software development, involving the careful planning necessary for constructing a software system. This stage encompasses the determination of data storage strategies and the creation of prototypes for system interfaces. Effective system design is crucial to ensure that the final product meets user requirements while maintaining high standards of efficiency, reliability, and scalability.

5.1 Product Features

- MRI Image Input and Preprocessing: Accepts MRI brain scans. Performs noise removal, normalization, and contrast enhancement to improve image quality.
- Tumor Detection Algorithm: Uses image segmentation techniques (e.g., thresholding, region growing). Identifies and isolates tumor regions from normal tissue.
- Feature Extraction: Extracts significant attributes such as size, shape, location, and texture of the tumor. Enables more accurate diagnosis and classification.
- Classification System: Implements machine learning models (e.g., SVM, neural networks) to classify tumor type. Helps distinguish between benign and malignant tumors.
- User Interface (UI): Provides an interactive and user-friendly interface for clinicians. Allow uploading MRI scans, viewing results, and accessing patient records.
- Performance Metrics: Evaluates system accuracy, sensitivity, specificity, and processing time. Ensures clinical reliability and usability.
- Report Generation: Generates detailed diagnostic reports based on analysis results. Includes visual and quantitative data for medical professionals

5.2 User Interface

This section presents a detailed description of the system’s user interfaces. Each interface is documented using structured tables to illustrate the expected inputs, outputs, and user interactions. These interfaces are designed to optimize usability for healthcare professionals with varying levels of technical expertise

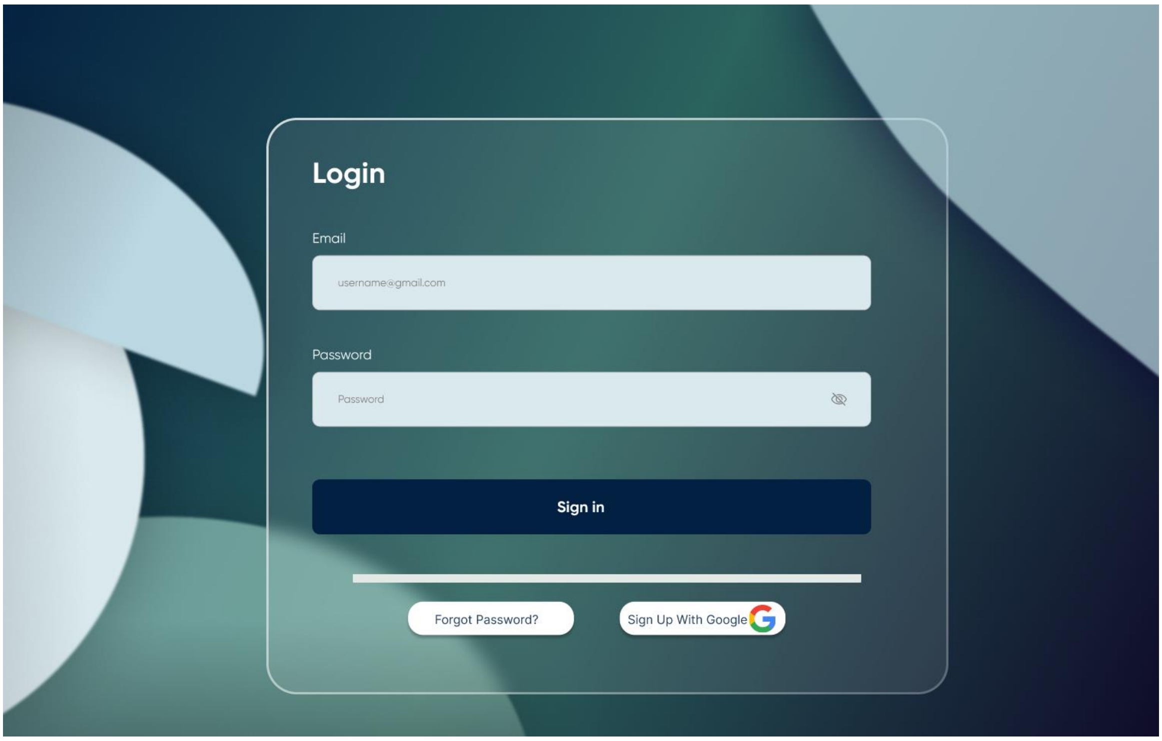

Figure 10 Login Page

Login Page		
Input	**Action**	**Output**
The User enters Email and password.	The User clicks on "Sign in" .	System verifies credentials If valid: display user account If invalid: show error message.

Table 8 Login Page

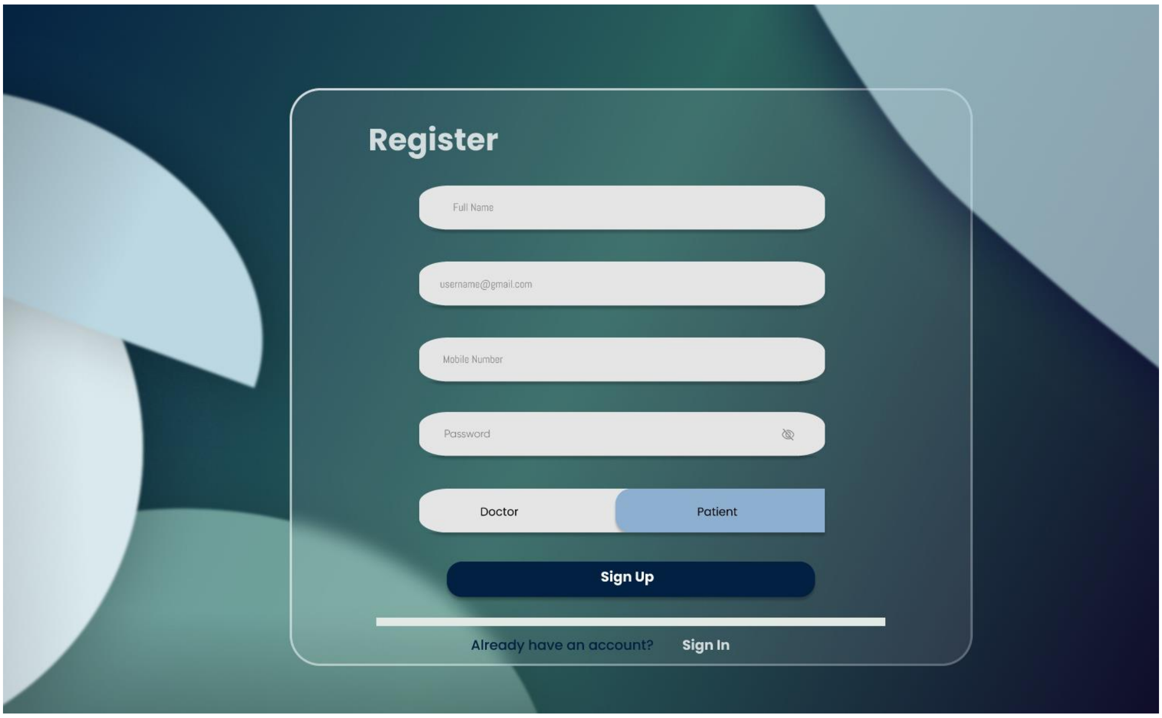

Figure 11 Registration Page

Registration Page		
Input	**Action**	**Output**
The user enters Full Name, Email, Mobile Number, Password, and selects a role (Doctor or Patient).	The user clicks on "Sign Up".	System validates input and checks for duplicates. If all inputs are valid: create an account and show success message or redirect to login/dashboard. If any input is invalid: show appropriate error message (e.g., "Email already exists", "Invalid mobile number").

Table 9 Registration Page

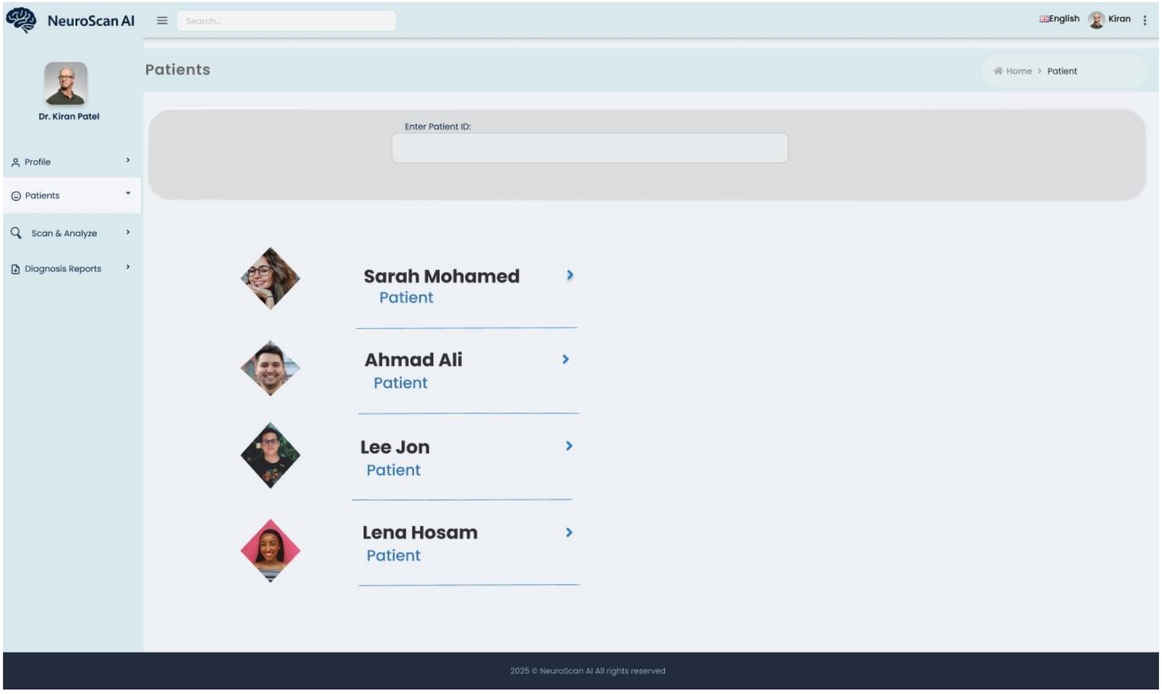

Figure 12 Patients Page

Patients Page		
Input	**Action**	**Output**
The doctor can click on a patient's name or image to select them. There is also a search box on the right side to enter the Patient ID.	When a patient is selected, the system opens their medical profile. If a Patient ID is entered, the system searches and displays the corresponding patient.	Displays a list of all patients. Shows basic patient info (name, picture). After selection, it redirects to the patient's full profile.

Table 10 Patients Page

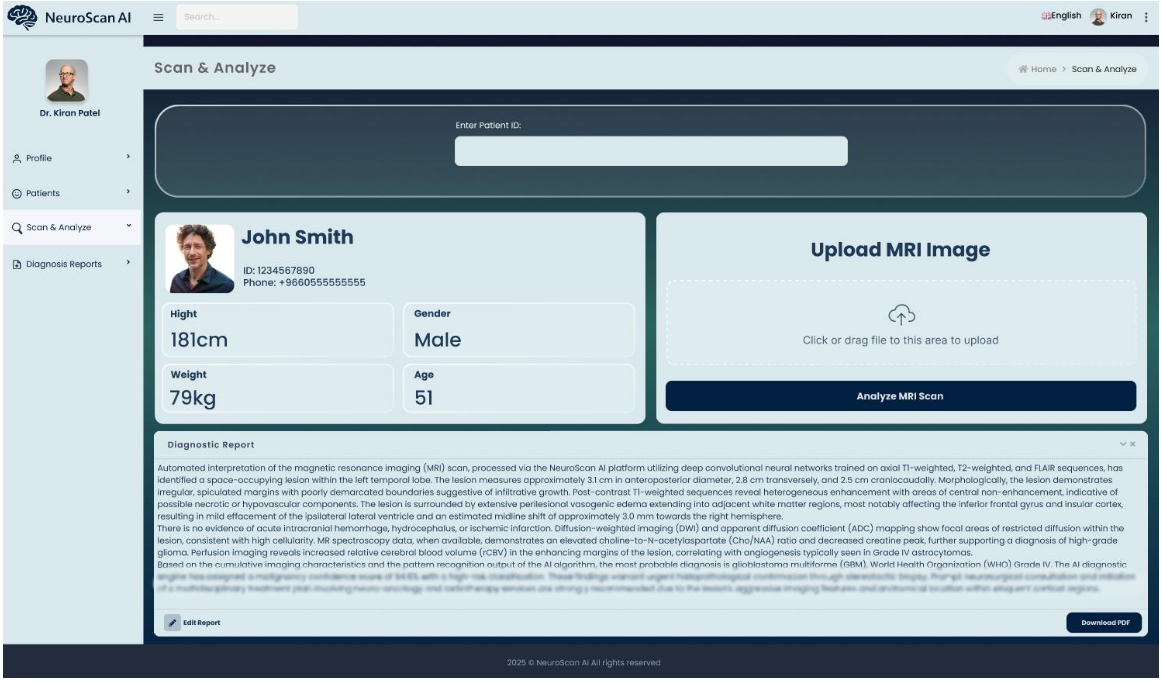

Figure 13 Scan & Analyze Page: Dashboard

Scan & Analyze Page: Dashboard		
Input	**Action**	**Output**
The doctor selects or views the patient. The doctor can upload an MRI image by dragging and dropping or clicking the upload area.	Displays patient details (Name, ID, Phone, Age, Gender, Height, Weight). Accepts an uploaded MRI image. Provides a button to analyze the MRI scan.	Shows patient information. Confirms MRI image is uploaded. Waits for the user to click "Analyze MRI Scan".

Table 11 Scan & Analyze Page: Dashboard

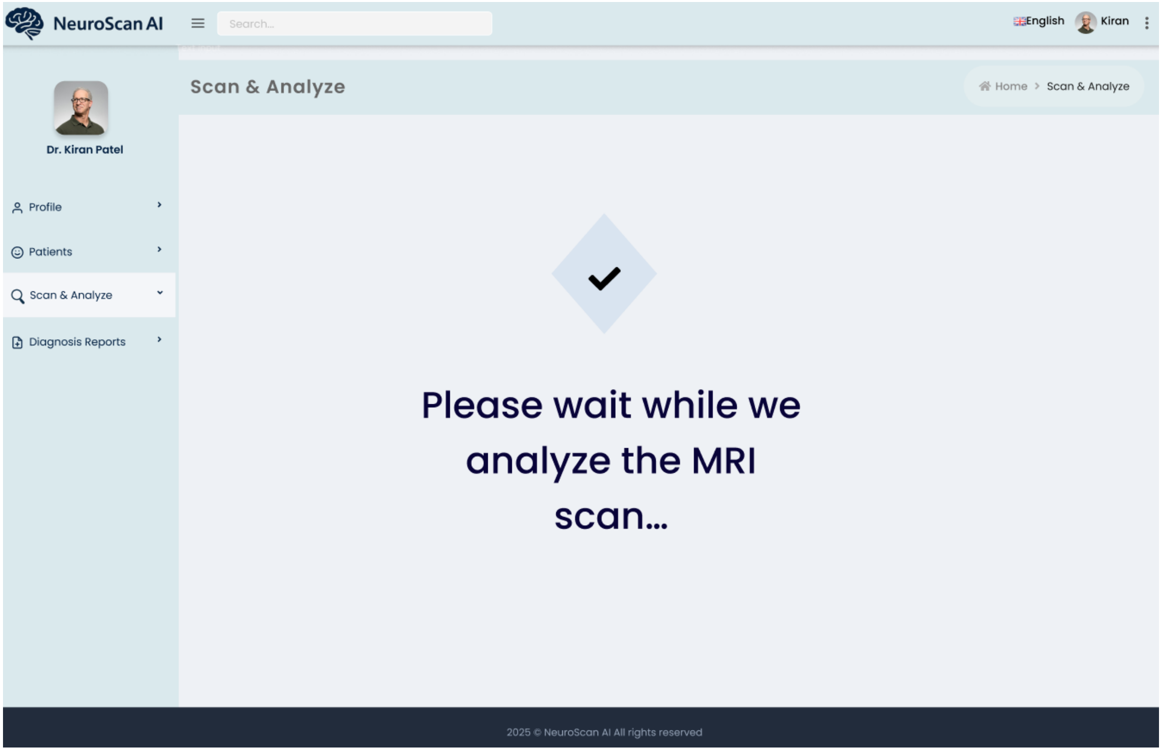

Figure 14 Scan & Analyze Page: Analyzing

Scan & Analyze Page: Analyzing		
Input	**Action**	**Output**
No input from the user at this stage.	No input from the user at this stage.	Displays a message: "Please wait while we analyze the MRI scan…" Indicates progress while the scan is being processed.

Table 12 Scan & Analyze Page: Analyzing

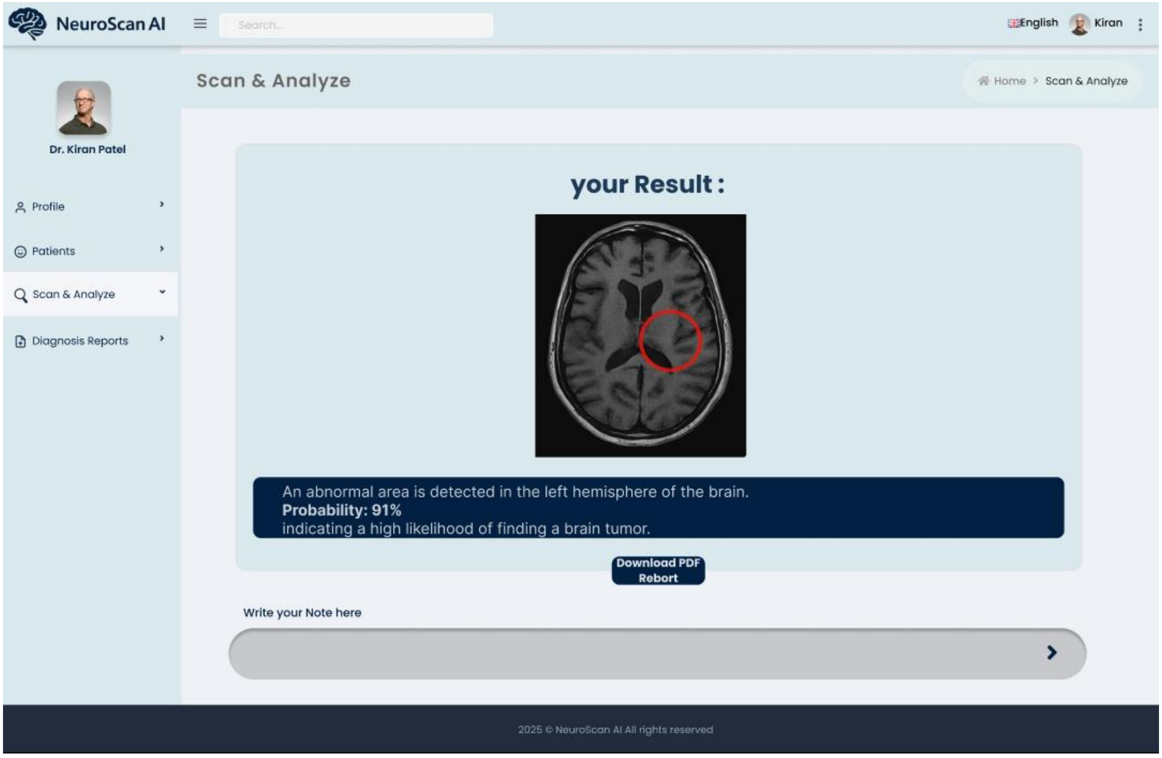

Figure 15 Scan & Analyze Page: Result

Scan & Analyze Page: Result		
Input	**Action**	**Output**
No input from the user at this stage.	Displays the AI analysis result of the MRI scan. Highlights an abnormal area in the brain and provides a probability of it being a brain tumor. Provides a button to download the report as a PDF.	Shows the analyzed brain scan image with highlighted abnormality. Displays a message indicating the presence of an abnormal area with a high probability of being a brain tumor. Offers a “Download PDF” button for the full report.

Table 13 Scan & Analyze Page: Result

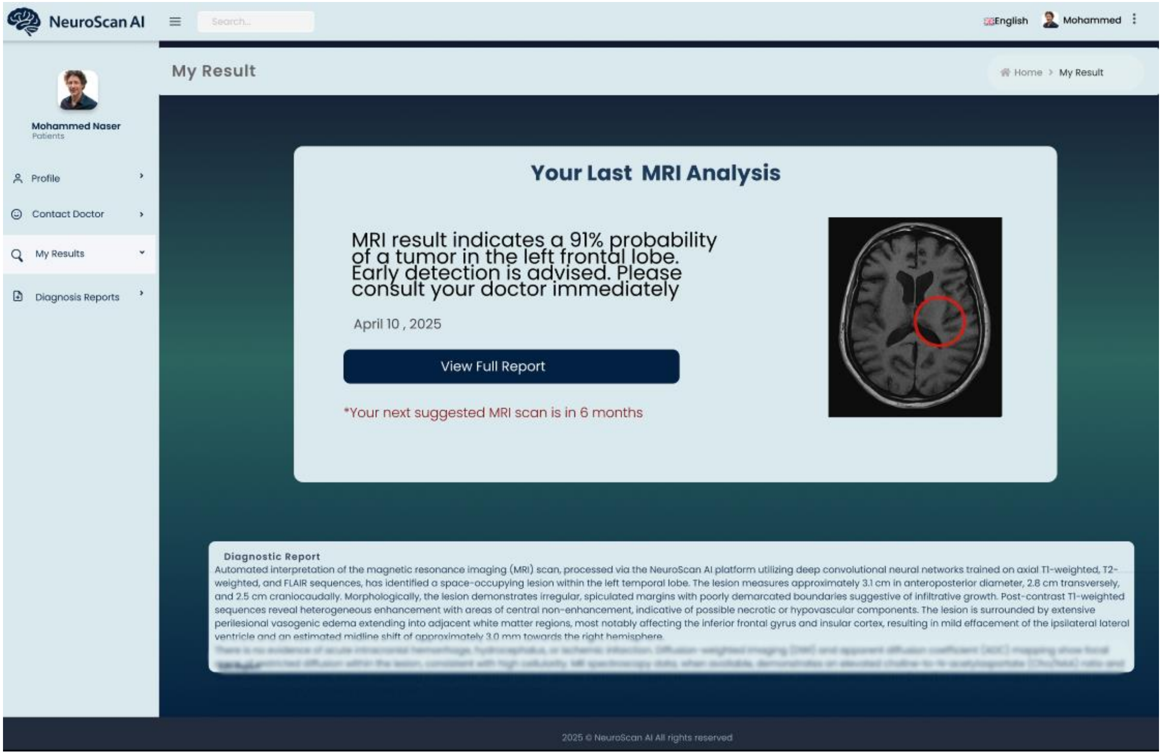

Figure 16 Patient Page: My Results

Patient Page: My Results		
Input	**Action**	**Output**
The doctor uploads the patient's MRI scan and analysis report to the system.	The system processes the MRI scan using AI, analyzes it, and sends the results to the patient's account.	A probability diagnosis (e.g., 91% chance of tumor) A medical recommendation A button to view the full report The MRI image with highlighted area

Table 14 Patient Page: My Results

5.3 Dataset

For the development and evaluation of the MRI Analysis System for Brain Tumor Detection, the dataset utilized is titled "Brain Tumor MRI Dataset" by Masoud Nickparvar, which is publicly available on Kaggle. This dataset is open for research purposes and educational use, making it suitable for academic projects involving machine learning and medical imaging.

The dataset consists of a total of 7,053 MRI images, categorized into four classes based on tumor types:

- Glioma Tumor
- Meningioma Tumor
- Pituitary Tumor
- No Tumor

These images are stored in JPEG (.jpg) format, organized into labeled directories that support supervised learning and image classification tasks. The dataset provides both training and testing sets, facilitating model evaluation and reducing the risk of data leakage during machine learning development.

Key Features:

- **Total Images:** 7,053
- **Training Set:** 5,214 images
- **Testing Set:** 1,839 images
- **Classes:** 4 (Glioma, Meningioma, Pituitary, No Tumor)
- **Image Format:** JPEG (.jpg)
- **Labeling:** Folder-based classification
- **Availability:** Publicly accessible via Kaggle under an open-use license for research and non-commercial use.

This dataset was selected due to its:

- **High quality and balanced class representation**, which is essential for training accurate deep learning models.
- **Real-world MRI data**, which aligns with the practical goals of the system in detecting actual brain tumors.
- **Clear labeling and organization**, reducing preprocessing complexity and allowing rapid development of CNN-based classifiers.
- **Public availability**, ensuring ethical use and transparency in research.

Moreover, its inclusion of multiple tumor types allows the system to not only detect the presence of a tumor but also assist in identifying the specific type, which is valuable for radiologists in diagnosis and treatment planning.

5.4 Data Storage

Data storage is a critical component of the system, responsible for managing both input data (such as uploaded MRI scans) and output data (such as diagnostic reports and classification results). The storage architecture is divided into two main layers: a file-based storage system for handling MRI images and generated reports, and a relational database (SQLite) for managing structured metadata and system interactions

5.4.1 MRI Images Storage

Uploaded MRI scans are saved in a secure directory within the system's backend (/static/uploads/). Each image is named using a unique identifier that includes patient ID and timestamp, ensuring traceability and avoiding filename duplication.

5.4.2 Database Structure

The system uses SQLite to store essential metadata. The main tables include:

- patients: Stores patient information (ID, name, age, gender).
- mri_scans: Links each MRI image to a patient and stores scan details such as upload date and image path.
- tumor_results: Contains analysis output such as tumor type (e.g., glioma, meningioma), confidence score, and severity.
- reports: Stores file paths of generated diagnostic PDF reports.

5.4.3 Report Generation and Storage

After analysis, a diagnostic report is automatically generated in PDF format and saved under /static/reports/, linked to the corresponding entry in the reports table.

5.4.4 Security Measures

To protect sensitive medical data and ensure system integrity, the following security measures will be implemented:

- Access control: The system will enforce secure user authentication and authorization protocols to restrict access based on user roles (e.g., doctors, patients, administrators).
- Data Encryption: All stored files, including MRI scans and diagnostic reports, will be protected using AES (Advanced Encryption Standard) to prevent unauthorized access.
- Regulatory Compliance: The system will adhere to medical data privacy regulations such as the Health Insurance Portability and Accountability Act (HIPAA) and the General Data Protection Regulation (GDPR) to ensure legal and ethical handling of patient information.

5.5 High Level Design

The MRI Analysis System was designed using an object-oriented approach, with detailed design diagrams presented in Section 4.3. The Sequence diagram captures the dynamic interactions between system components such as the doctor interface, analysis module, and patient dashboard, clearly outlining the order and flow of messages to reveal system behavior during image upload, processing, and result retrieval. The Activity diagram illustrates the overall workflow, including user authentication, MRI image submission, AI-based tumor analysis, and report generation, enabling a better understanding of the logical progression of system tasks. Additionally, the Class diagram defines the static structure of the system by showing the key classes—such as User, MRIImage, AnalysisResult, and Repot and their relationships. This provides a foundational view of how the system is organized and how its components interact, which is essential for both development and future maintenance.

5.6 Summary

This chapter explained the system design of the MRI Analysis System for Brain Tumor Detection. It covered the main features of the product, such as MRI image upload, AI-based tumor detection, and report generation. The user interface was designed to be simple and user-friendly, with separate views for doctors and patients. The chapter also described how data is stored securely in a database. Finally, it included high-level design diagrams like sequence, activity, and class diagrams to show how the system works and how its parts interact.

Chapter 6: Conclusions and Future Work

6.0 Overview

Accurate and timely diagnosis of brain tumors is critical in improving patient outcomes and supporting medical professionals in making informed decisions. This project presents a web-based MRI analysis system that utilizes artificial intelligence and machine learning to assist doctors and radiologists in detecting brain tumors from MRI images. The system features a user-friendly interface, allowing medical professionals to upload MRI scans, perform automated analysis, and generate detailed reports, while patients can securely access their results online. Designed with accessibility, simplicity, and security in mind, the platform aims to enhance the efficiency and accuracy of the diagnostic process. By reducing the manual burden on radiologists and offering quick AI-based insights, the system contributes to faster decision-making and improved healthcare delivery. Although the current version is limited to detecting brain tumors from pre-labeled datasets, it establishes a strong foundation for future enhancements. Its strength lies in the use of intelligent analysis, structured data handling, and an accessible interface for both doctors and patients.

6.1 Future Work

In the next development phase, the system will move toward full implementation and testing to ensure it meets functional and non-functional requirements. Future improvements may include expanding the model to detect additional types of brain anomalies, integrating real-time image processing, and enhancing the AI algorithm for greater accuracy. Security features and multilingual support will also be considered to broaden usability.

6.2 Summary

The proposed solution is a web-based AI-powered application designed to assist in the early detection of brain tumors using MRI images. It simplifies the diagnostic process for doctors and offers patients access to their medical reports. By applying machine learning to medical imaging, the system improves the speed and accuracy of tumor detection. With further enhancements, it can evolve into a more comprehensive diagnostic tool supporting advanced analysis and broader medical use.

Chapter 7: Literature Cited

References

- Abbas, Q., Solomon, C., & Mouton, A. (2013). Automatic localization of brain tumor from MR images using symmetry information. Computer Methods and Programs in Biomedicine, 111(3), 555–565. https://doi.org/10.1016/j.cmpb.2013.05.015
- Chollet, F. (2015). Keras [Computer software]. GitHub. https://github.com/fchollet/keras
- Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115–118. https://doi.org/10.1038/nature21056
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25, 1097–1105. https://papers.nips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf
- Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., … & Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. Medical Image Analysis, 42, 60–88. https://doi.org/10.1016/j.media.2017.07.005
- TensorFlow Developers. (n.d.). TensorFlow: An end-to-end open source machine learning platform. https://www.tensorflow.org/
- Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A., & Gee, J. C. (2010). N4ITK: Improved N3 bias correction. IEEE Transactions on Medical Imaging, 29(6), 1310–1320. https://doi.org/10.1109/TMI.2010.2046908
- Nickparvar, M. (2021). *Brain Tumor MRI Dataset* [Data set]. Kaggle. https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset

Chapter 8: Acknowledgements

First and foremost, we are deeply grateful to Allah for granting us the strength, health, and perseverance to complete this project successfully.

We would like to express our sincere appreciation to Ms. Maha Asiri for her invaluable guidance, support, and encouragement throughout the development of this project. Her expertise and constructive feedback were essential to the successful completion of our work.

We also extend our gratitude to King Khalid University for providing a supportive academic environment and the necessary resources that enabled us to carry out this research.

Lastly, we are truly thankful to our families and friends for their continuous support, patience, and motivation. Their unwavering encouragement has been a constant source of strength throughout this journey.

This project is the result of the collaborative efforts of the following team members: Arwa Saeed, Reem Salem, Sadeem Alsaleh, Mashael Ali, and Faizah Alahmari.