Brain Tumor Segmentation and Detection Using Machine Learning

Syeda Fatima Naqvi
01-136221-047

Manahil Bilal
01-136221-038

Supervisor: Dr. Adil Khan

Bachelor of Science in Artificial Intelligence

Department of Computer Science
Bahria University, Islamabad

December, 2025

Certificate

We acknowledge that the work presented in this project documentation, titled **Brain Tumor Segmentation and Detection using Machine Learning,** authored by Ms. Syeda Fatima Naqvi and Ms. Manahil Bilal, meets the required standard for the partial fulfillment of the Bachelor of Science in Artificial Intelligence degree .

Approved by:

Supervisor: Dr. Adil Khan, Assistant Professor

Internal Examiner: Dr. Muhammad Asif, Senior Assistant Professor

---

External Examiner:

Project Coordinator: Dr. Sohail Akhtar, Assistant Professor

Head of the Department: Dr. M. Khurram Ehsan, Senior Associate Professor

---

December, 2025

Abstract

The Brain Tumor Segmentation and Detection System is an advanced system that aims to help in the diagnosis of brain tumors using deep learning techniques including VGG16 and U-Net architectures. Automatic segmentation, detection, classification, and localization of brain tumors in MRI scans are provided by these models, which reduce diagnostic time and improve accuracy. This makes interpretation by radiologists less reliant on manual interpretation, thus aiding in detecting tumors more quickly and in a more reliable manner. Conventional techniques of diagnosing brain tumors rely mostly on subjective determination of MRIs by radiologists, which can cause inconsistencies, delay, and mistakes. With the increased number of MRI data in clinical practice, the burden of radiologists increases, thus creating diagnosis bottlenecks. Such difficulties require automatic, effective, and scalable solutions that can help improve diagnostic accuracy while reducing delays and ultimately improving patient care. The proposed system addresses these challenges by automating key processes in the diagnosis procedure, including tumor segmentation and MRI spot identification. It features a complete ecosystem that includes mobile apps for patients and doctors, along with a web panel for administrators. The patient app allows users to upload MRI scans and receive tumor analysis through AI assistance. The doctor app enables appointment management and consultations. The system delivers accurate and immediate results, which enhance the healthcare experience for both patients and medical personnel.

Acknowledgement

We express our heartfelt gratitude to Allah Almighty, whose mercy and blessings have guided and supported us throughout our academic journey. We are sincerely thankful to our supervisor, Dr. Adil Khan, for his continuous support, guidance, and encouragement, which greatly contributed to the successful completion of this project. We extend our deepest appreciation to our parents for their unwavering support and prayers, and to our families and friends for their constant motivation. We also acknowledge the Department of Computer Science, Bahria University, Islamabad, Pakistan, for providing a supportive academic environment. This project is a reflection of the collective support and guidance we have been privileged to receive.

Syeda Fatima Naqvi,
Manahil Bilal,
Bahria University, Islamabad, Pakistan.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Brain tumor is one of the most serious and life-threatening diseases that can be suffered by people of any age groups with a significant mortality rate, especially in developing nations such as Pakistan. The conventional techniques for diagnosis focus mainly on a human interpretation of the obtained MRI scans by the radiologists, therefore, they are slow, prone to errors, and subjective [1]. Such limitations often lead to delays in diagnosis and compromises the results of treatment.

The proposed system provides a cutting-edge AI-based solution for automating the segmentation, detection and localization of brain tumors using deep learning algorithms. CNNs, U-Net, and VGG16 are some of the models used to increase accuracy and efficiency in diagnosis. The system combines several independent models for the MRI data analysis, providing accurate tumor identification and classification.

A whole diagnostic ecosystem is created, including mobile applications for patients and doctors as well as a web-based panel. Through such platforms, the stakeholders can interact with the system depending on their roles. The backend with Flask and Firebase, provides real-time and secure data management. This solution seeks to help healthcare providers to make faster, more consistent and scalable brain tumor diagnoses without compromising reliability and patient trust.

1.1 Objectives

The objective of this study is to:

- To develop an automated system for the segmentation, detection and classification of brain tumors in MRI and CT scans using machine learning that integrates Convolutional Neural Networks (CNNs), VGG16, and U-Net architectures.[2].
- To create a complete ecosystem that extends beyond analysis, incorporating a patient mobile app for secure MRI or CT scan upload and AI-powered tumor analysis, a doctor app for appointment management and consultation, and a web panel for administrative oversight.

- To enhance diagnostic accuracy and significantly reduce the operational burden on healthcare experts and assist them in diagnoses.

1.2 Problem Description

One of the main difficulties that medical specialists are faced with when diagnosing brain tumors is the constantly increasing complexity and volume of clinical imaging data. Key problems include:

1. The conventional methods used in diagnosing brain tumors depend on manual MRI analysis by the radiologists, and this takes a lot of time and is subjective and prone to human error.
2. With the high volume of MRI scans in hospitals, there is an ever-increasing workload on radiologist, which leads to delays and reduced diagnostic accuracy.
3. Sometimes, minute differences between the tumor tissue and healthy brain region are not obvious to the naked eye leading to missed or delayed diagnoses.
4. Inter-observer variability is introduced in manual interpretation hence inconsistent evaluations and treatment decision are however made among different medical persons.
5. Lack of automated and scalable diagnostic tools prohibits real-time tumor detection and limits the ability of the healthcare system to institute early actions in an efficient manner.

1.3 Project Scope

This project introduces an intelligent brain tumor detection and segmentation system based on the use of deep learning models such as the CNNs and U-Nets to extract accurate localization and classification. The solution is the full-cycle application development for mobile and web platforms for patient-doctor communication that includes AI-provided diagnostic reports, emergency consultations, and multilingual service. The system based on Flutter, Flask API, Firebase, and Python for cross-platform app development, backend services, real-time database, and authentication, respectively, guarantees scalability and prompt responses. It provides a centralized ecosystem for improving the efficiency of diagnosis, minimizing human error, as well as facilitating timely medical interventions.

1.4 Solution to the Problem

The system is meant to expedite and improve the diagnostics procedure for tumors of the brain due to AI-equipped MRI segmentation, detection. In medical imaging, it assists radiologists in producing a faster and more steady analysis, effective treatment arrangements and decrease in diagnosis postponement. When it comes to academic life, on the research front, the platform helps in the study of patterns of growth in tumors, it helps in evaluating models to understand their performance, and on the research front, it assists in realizing innovative strategies in treatment. In addition, in medical education, the system is a useful learning tool for students as well as radiologists, whereby the system assists them in comprehending complex imaging data and behavior of tumors. Using its role-based mobile apps and web panel, the platform will allow efficient patient-doctor interaction, secure payment processing through Stripe, real time AI produced reports, and multilingual accessibility allowing the platform's usage in all forms of medical and academic environments.

1.5 Risk Involved

Here, the main challenges for the system, consisting of technical issues, issues faced while using it, legal obligations and computer security risks, are pointed out and must be taken care of properly to ensure the system works well and is reliable.

1.5.1 Technical Challenges

Generating precise segmentation models with CNNs, U-Net, and VGG16 is computationally demanding and could struggle in performance between platforms. Connecting to a real-time multi-platform environment may lead to delays or compatibility issues.

1.5.2 User Adoption

Patients and doctors can be reluctant to have faith in AI-related diagnoses. It is necessary to build a convenient interface and provide guidance to induce adoption and trust of the system.

1.5.3 Legal Compliance

Dealing with medical data demands a strict legal compliance. A violation of privacy laws or failure to obtain an informed consent may result in severe legal ramifications, and thus, the initial legal supervision appears vital.

1.5.4 Security

The data related to patients MRI is highly sensitive. Lacking strong encryption and the assurance of safe data handling, any gap in this area may hurt the trust of the users and result in reputational or legal harm.

1.6 Summary

The Brain Tumor Segmentation and Detection System using deep learning is an AI-based solution aimed at improving the diagnosis of brain tumors with the support of automatic segmentation, detection in the MRI and CT scans. It applies the deep learning algorithms such as CNN, VGG16 and U-Net to precisely locate the tumor. The system consists of Flutter-based mobile application for patients and doctors with the use of a Flask backend and Firebase authentication. It allows secure uploading of the MRI scans, AI based analysis of tumors, appointment booking, and payment through stripe. The project enhances the efficiency of diagnosis, eliminate manual interpretation, and increase the accuracy of tumor detection. It is, however, plagued by issues of model performance, data security, user adoption, and compliance to the regulations.

Chapter 2

Literature Review

2.1 Existing Applications

A number of advanced AI-powered systems have been developed for the early diagnosis and segmentation of brain tumors with the help of neural networks and medical images. DeepMedic and the U-net tools have become quite famous for the high accuracy they demonstrate in performing the automated segmentation of the tumor regions in the MRI scans, through involving convolutional neural networks for the classification of each voxel with high accuracy [3].

The BraTS challenge toolkit provides publicly available benchmark datasets and models specific for brain tumor detection available for researchers to assess and compare the performance of deep learning under uniform standards. MONAI (Medical Open Network for AI) based on PyTorch is a powerful tool for development, training, and application of healthcare imaging models with the minimum amount of effort. Another open-source platform, NiftyNet, is also targeted for deep learning in medical imaging and it can handle 3D segmentation architectures and can be used for the purpose of localizing brain tumors. Qure.ai and Aidoc utilize AI algorithms for radiology analysis in real-time, and allow tools that tag likely deviations in brain CT and MRI scans, thus helping the radiologists to make quicker decisions. Google Health has created deep learning models which are able to interpret medical images at radiologist grade performance including neurologic anomalies. InnerEye by Microsoft Research relates to research on machine learning in medical image analysis, including radiotherapy and tumor detecting purposes. AI-Rad Companion of Siemens Healthineers is an automated assistance for the radiologists for interpretation of MRI scans, improving consistency and minimizing the workload.

These systems integrate the state-of-art deep learning approaches, namely CNNs and hybrid models, to achieve the best possible performance in segmentation, detection and classification tasks. Our suggested platform extends these innovations by investigating such models as the CNNs, VGG16 and U-Net in the perspective of brain tumors' identification with the specification of their specific location in the MRI data. The system is coupled with the secure Firebase backend and the Flask-based API and is deployable for scalable inference at AWS or GCP. It also includes a full ecosystem that has a patient mobile app for uploading scans, on the spot AI analysis, emergency consultation, multilingual service, doctor finder; a doctor mobile app for tracking appointment and payment; and a web panel that can be used to manage users, specialists, and payments. This has created a viable, automated, and integrated perception for brain tumor detection and management solution.

2.1.1 Core Segmentation Architectures

2.1.1.1 DeepMedic

DeepMedic is developed as a 3D-convolutional neural network to segment brain tumors automatically using MRI scans with the use of a double-layer design that captures all information present in an image. Among the main features are:

1. The dual-pathway method enhances how the model separates each section of the tumor by processing fine as well as coarse details.
2. DeepMedic is able to identify the different kinds of tumors and various parts of the tissue when doing multi-class segmentation.
3. Because it works well in several MRI approaches, it supports both clinical findings and research studies.
4. The structure of the architecture makes it possible to provide good predictions for rare tumor types, making it reliable [4].

2.1.1.2 U-Net

Many healthcare professionals rely on U-Net which has proven itself to be effective in detecting brain tumors. Using skip connections in the encoder-decoder layout makes it possible to precisely pinpoint the regions showing tumors in MRI scans. The main features are:

1. With the encoder-decoder structure, features are extracted and details in space are reconstructed which makes accurate segmentation possible at every single voxel.
2. By using skip connections, the network keeps detailed spatial information which helps it find the edges of the tumor [5].
3. Because it usually performs well in segmentation, U-Net is considered a reference model for both studying brains and clinical scans.
4. It performs strongly even when trained on not much data which helps a lot in medical imaging where quality training data can be scarce [6].

2.1.2 Research and Development Frameworks

2.1.2.1 BraTS

BraTS Challenge Toolkit, it contains resources for brain tumor detection and gives access to pre-trained models, so AI results can be checked using the same testing conditions.

1. MRI datasets which are made public and comes with successive expert annotations for various tumor parts and MRI methods.
2. Immediately usable segmentation models (for example, U-Net variants) are ready for use.
3. The evaluation process uses a standardized set of measures (Dice score, Hausdorff distance) to make sure all algorithms are judged the same way.
4. Annual standards that help researchers see how they are doing against each other and motivate them to improve how brain tumors are analyzed [1, 7].

2.1.2.2 MONAI

MONAI (Medical Open Network for AI) is a framework designed on PyTorch, meant to help you generate AI models for medical imaging issues such as brain tumor segmentation. It makes it simpler to prepare, check and implement neural networks in clinical studies. Some important features are:

1. Special 3D medical image processing modules and pipelines that focus on tasks using MRI data.
2. Data transformations, ways of understanding error and the measures used to report results are easily tailored for healthcare purposes.
3. Capability to train large and complex networks with highly optimized and efficient programs.
4. Made with repeatability and reliability in mind, so it can be applied in both research and practical medicine [8].

2.1.2.3 NiftyNet

NiftyNet uses TensorFlow as its foundation and is developed for medical imaging, mainly to assist with segmenting 3D parts such as brain tumors in MRI scans. The modular organization means different systems can be adapted and used in different contexts. Among the most important characteristics are:

1. Specific network architectures designed for massive data and the process of separating 3D medical images.
2. An easy-to-modify setup that lets researchers suit it to their specific experiments.
3. Facilitating big datasets and GPU computing in neuroimaging studies at a large scale.
4. Enables researchers to do similar medical imaging tasks in a reliable way by sharing standardized models and configurations [9].

2.1.3 Commercial Clinical Platforms

2.1.3.1 Qure.ai

Qure.ai builds AI systems that can detect changes in real-time brain imaging such as tumor markers and notifies doctors immediately about these changes. They are made to make radiology work smarter and more efficiently. Key things about them are:

1. It is possible to spot neurological disruptions such as tumors, bleeding and shifts of brain structures, instantly during imaging.
2. Automation that can detect urgent cases and send alerts to make sure doctors respond quickly.
3. Having the software smoothly operate as part of daily radiology tasks in fast processing facilities.
4. Allows diagnostic delays to be reduced as the workload of radiologists is reduced, thanks to AI [10].

2.1.3.2 Aidoc

Aidoc is an AI-powered platform designed to assist radiologists by identifying urgent findings such as brain tumors and hemorrhages in CT and MRI scans. Its real-time deep learning algorithms integrate directly into clinical workflows to improve diagnostic speed and accuracy. Key features include:

1. Automated detection of critical abnormalities in neuroimaging, including tumors and intracranial bleeding.
2. Real-time case prioritization to ensure high-risk patients are addressed promptly.
3. Seamless integration with PACS and RIS systems for streamlined radiology operations.
4. Enhances clinical decision-making and improves patient outcomes by accelerating diagnosis [11].

2.1.3.3 Siemens Healthineers' Al-Rad Companion

Siemens Healthineers' AI-Rad Companion was made to help radiologists read MRI images and notice conditions such as brain tumors. It operates with machine learning to create reports automatically and standardize how doctors diagnose. Key parts of the technology are:

1. Brain tumors and different neural diseases are located and separated by the automatic system using computer software.
2. Creating structured, same-style radiology reports to improve efficiency when handling cases [14].

2.1.4 Advanced Research Initiatives

2.1.4.1 Google Health

With advanced deep learning, Google Health systems can read brain medical images with expert accuracy. Large data collections are used to teach these models to detect issues in the brain, including tumors, helping with reliable diagnosis. This include or feature:

1. Special 3D medical image processing modules and pipelines that focus on tasks using MRI data.
2. Data transformations, ways of understanding error and the measures used to report results are easily tailored for healthcare purposes.
3. Made with repeatability and reliability in mind, so it can be applied in both research and practical medicine [12].

2.1.4.2 InnerEye

InnerEye by Microsoft Research InnerEye is a research project from Microsoft that uses artificial intelligence to help with areas such as 3D tumor mapping and radiotherapy setup in medicine. It helps doctors accurately see the tumor on MRI scans which helps with treatment choices. Some important features are:

1. Precise software tools for dividing brain tumors from other brain matter to help with treatment.
2. Merging with radiotherapy procedures to ensure patients receive powerful dose treatments, yet their healthy tissues are spared.
3. Workshops and studies allowing optimization of algorithms for effective clinical applications.
4. Makes it easier to understand patients' conditions for improved outcomes and clearer tumor recognition [13].

2.1.5 Comparison Table

Table 2.1: Summary of Existing AI Applications for Brain Tumor Detection

Name	**Approach**	**Contribution**	**Limitation**
DeepMedic, 2016 [4]	3D-CNN with a dual-pathway design for multi-class segmentation (fine and coarse details).	Achieves high-accuracy, voxel-level segmentation of different tumor types in 3D MRI scans.	Primarily a segmentation tool; does not integrate comprehensive clinical workflow or reporting features.

Continued on next page

Table 2.1 – continued from previous page

Name	Approach	Contribution	Limitation
U-Net, 2015 [5]	Encoder-decoder architecture with skip connections to achieve accurate spatial reconstruction.	Considered a reference model for high-performance segmentation, effective even when trained on limited datasets.	Focuses purely on segmentation; typically requires 3D extensions (e.g., V-Net) for full volumetric analysis.
BraTS Challenge, 2020 [7]	Standardized MRI datasets, pre-trained models and uniform evaluation metrics (Dice, Hausdorff).	Standardizes performance assessment of deep learning algorithms, enabling fair rigorous comparison across research groups.	Not a clinical application, its main role being for research benchmarking and model dissemination.
MONAI Framework, 2022 [8]	PyTorch-based framework with specific modules for 3D medical image processing and data transformations.	Simplifies the development and training of large, complex, and repeatable neural networks for medical imaging tasks.	Primarily a development framework; still requires substantial effort for final regulatory-compliant deployment.
NiftyNet, 2018 [9]	TensorFlow-based framework with modular architectures designed for large-scale 3D medical image segmentation.	Facilitates large-scale 3D segmentation studies in neuroimaging, offering high adaptability and customizability for research.	A development toolkit that requires significant customization and a separate deployment layer for clinical use.
Qure.ai, Commercial [10]	AI algorithms for real-time radiology analysis and automated tagging of likely neurological deviations.	Provides real-time alerts for urgent neurological disruptions, significantly reducing diagnostic delays and radiologist workload.	Commercial focus means that the underlying AI model architectures are typically proprietary and closed source.

Continued on next page

Table 2.1 – continued from previous page

Name	**Approach**	**Contribution**	**Limitation**
Aidoc, Commercial [11]	Realtime deep-learning algorithms integrated directly into clinical workflows (PACS/RIS).	Delivers automated case prioritization and faster diagnosis of life-threatening abnormalities such as tumors and hemorrhages.	Due to being a proprietary commercial system, transparency and third-party modification are severely limited.
Google Health/Google AI, Commercial [12]	Deep learning models trained on vast proprietary datasets for radiologist-grade image interpretation.	Achieves expert-level diagnostic performance across various neurologic anomalies, leveraging massive data resources.	Internal model structure and training data are proprietary/closed, limiting external validation and replication.
InnerEye, 2020 [13]	Open-source deep learning software for 3D tumor mapping, focusing on radiotherapy planning workflows.	Provides precise tools for tumor contouring and segmentation, directly supporting critical treatment decisions.	Primarily a research project and toolkit validated mainly for radiotherapy planning with less emphasis on broad diagnostic triage.
AI Rad, Commercial [14]	The company uses machine learning to analyze MRI images and prepare structured reports.	Standardizes radiology reporting, and consistency of diagnosis by automatically identifying and segmenting brain pathologies.	This requires deep integration with the Siemens ecosystem. Flexibility with non-Siemens hardware/software is limited.

2.2 Summary

This chapter discusses DeepMedic, U-Net and MONAI which are examples of AI methods that segment MRI scans of brain tumors with success. It points out important benchmark data such as BraTS Toolkit and it also covers diagnostic platforms on the market like Qure.ai, Aidoc and Google Health. Also, NiftyNet and InnerEye which are open-source frameworks, contribute to medical research in imaging. These technologies serve as a guide for developing our system, which integrates new models into a secure application for brain tumor diagnosis.

Chapter 3

Requirement Specifications

3.1 Existing System

At this step, we will consider various uses of artificial intelligence and neural networks for brain tumor analysis which are related to the brain tumor segmentation and detection project we are analyzing. Here, it is important to observe the current systems at work, notice where there are some issues and figure out the key problems that exist today. Problems include dependence on manual interpretation of MRI images, not having more than one stage of tumor classification, delays in diagnosing patients and no real-time system that helps both patients and doctors. Detecting these challenges allows us to better judge how our current tools are working (or not working) and where they need to be improved [7]. Having this viewpoint, we can move forward to improve our systems. From the following sections, we will discuss the new features planned, including artificial intelligence for tumor identification, applications for multiple devices and the ability to consult in real time, all meant to address present difficulties and improve both the system's accuracy and ability to serve many users.

3.2 Proposed Solution

The proposed solution helps create an integrated system that uses patient and doctor mobile applications along with a web portal, making brain tumor diagnosis, consultation, and system management simpler.

3.2.1 Patient Mobile Application

A user-friendly mobile application built just for patients is one of the main solutions we offer. In order to make brain tumor diagnosis simple, users can add their MRI scans, get feedback on them and identify nearby specialists. Patients may book emergencies if needed, receive support in various languages and preview the case summaries AI and junior doctors make before meeting the specialist.

3.2.2 Doctor Mobile Application

A dedicated mobile application built for doctors that allows healthcare specialists to manage patient cases, review AI-generated reports, and conduct online consultations efficiently. Doctors can access uploaded MRI scans analyzed by the system, verify tumor classifications, and provide medical feedback directly through the app. Doctors can schedule and manage appointments, view consultation history, and monitor payments through an integrated payment tracking module. AI-generated case summaries help doctors save time by providing pre-diagnosis insights before meeting the patient.

3.2.3 Centralized Web Panel

The web panel is where doctors are managed, payments are supervised and activity across the system is monitored. It helps coordination between users and service providers and at the same time lets administrators see what is happening in the system.

3.3 Target Audience

The main users of this brain tumor detection system are neurologists, radiologists, general physicians and hospital administrators. Patients and their families also benefit from the system since earlier diagnosis is possible and they have access to specialists. Integrating brain tumor diagnosis using AI and mobile/web apps allows the platform to intervene sooner and more accurately.

3.4 Functional Requirements

This section explains all the main requirements the system must have for its mobile apps and web panel.

3.4.1 Patient Mobile Application

The Patient Mobile Application must meet certain requirements to support secure logins, AI features for finding tumors, handling payments, searching for doctors, providing emergency help, offering multilingual support and showing the map location.

Table 3.1: Functional Requirements for Patient Mobile Application

ID	Requirement	Description
FR-001	Authentication	The application needs to offer a secure and easy way for patients to sign in with valid email and password. It is necessary for first-time users to verify themselves during registration to be allowed to access the features of the system.

ID	Requirement	Description
FR-002	Tumor Detection	The tool will make it possible for patients to send their MRI scans which will be automatically analyzed by AI. The results will consist of tumor segmented pictures and the classification result.
FR-003	Payment System	Using Stripe for a secure payment system will enable patients to pay for services and consultations through the app itself. With the app, you can review your medical payments which makes managing all your healthcare costs easier and more transparent.
FR-004	Doctor Listings	Users can find certified brain tumor specialists by location and accessibility through the app. When faced with emergencies, the application lets you communicate directly with a doctor right away and shows if anyone is available.
FR-005	Junior Doctors and AI Summary	AI is used to develop summaries of the case which help junior doctors prepare for actual consultations. The summary will aid in the pre-booking stage by giving a clear picture to both the doctor and patient.
FR-006	Emergency Consultation	Through the directory, patients will find brain tumor specialists in their area who are available at the time they are looking. When an emergency happens, the app will quickly connect users to doctors who are available at that moment.
FR-007	Multilingual Support	To support a diverse user base, the app will offer multilingual support, making it accessible to patients across various regions The interface will be designed to prioritize ease-of-use, especially for non-technical users and patients in need of urgent care.

3.4.2 Doctor Mobile Application

It lists what the Doctor Mobile Application should provide: safe login, set up appointments, view cases involving patients and handle payments to improve the work processes of specialists.

Table 3.2: Functional Requirements for Doctor Mobile Application

ID	Requirement	Description
FR-008	Authentication	The system demands that medical professionals use their valid credentials to login and access the doctor app.Verifying a doctor's identity will form part of the registration steps and only after verification will the system grant access.

ID	Requirement	Description
FR-009	Appointment Management	The system demands that medical professionals use their valid credentials to login and access the doctor app.Verifying a doctor's identity will form part of the registration steps and only after verification will the system grant access.
FR-010	Payment Tracking	With the payment tracking module, doctors can review the fees they get for their consultations and transactions.The module will also give details about upcoming payments, past transactions and financial summaries needed by the administration.

3.4.3 Web Panel

This section explains the administrative tools in the Web Panel which help with handling doctors, processing payments and observing system use from a detailed dashboard.

Table 3.3: Functional Requirements for Web Panel

ID	Requirement	Description
FR-011	Doctor Management	The web panel gives you administrative control to manage doctors registered on the site, including additions, updates and deletions. The system can be checked by administrators for accurate credentials, consultation activity and certifies only genuine specialists get included.
FR-012	Payment Management	A central management tool handles all the payments processed by patients and doctors. It is possible to follow completed payments, oversee unpaid balances and use available reports for auditing.
FR-013	User Dashboard	A detailed dashboard gives instant access to information on appointments, consultations and how people are using the app. The dashboard enables administrators to check the performance, usage of the system and guide decisions with data.

3.5 Non-Functional Requirements

This section talks about all the necessary features the system needs to offer to its mobile apps and web panel, keeping up performance, security, reliability and usability.

3.5.1 Performance Engineering Requirements

Table 3.4: Performance Engineering Requirements

ID	Requirement	Description
NFR-001	Response Time	Minimal delays are necessary in the front-end system so that MRI pictures are processed, AI segmentation results are provided and diagnosis-related insights are shown fast. It is important for good user experience, especially during emergency consultations where fast answers can guide doctors' choices.
NFR-002	Scalability	The system should handle increasing numbers of users, scans, and concurrent operations without performance degradation. .

3.5.2 Quality Standards

Table 3.5: Quality Standards

ID	Requirement	Description
NFR-003	Reliability	The Brain Tumor Detection System is designed for high availability and includes automated backups to assist patients, doctors and administrators at all times which improves trust in its accuracy.
NFR-004	Security	Ways to keep data private and secure are put in place to keep patient health information, MRI files and copies of consultations safe and legal, avoiding others seeing them.

3.5.3 Design Constraints

Table 3.6: Design Constraints

ID	Requirement	Description
NFR-005	Responsive Design	Both the Patient and Doctor Mobile Apps and the Web Panel have to function normally on iOS, Android and web browsers to ensure that every user has access to them.

3.5.4 Usability and Accessibility

Table 3.7: Usability and Accessibility

ID	Requirement	Description
NFR-006	Intuitive Design	Apps are made easy to use with a simple interface, letting users from all skill levels benefit from uploading MRI scans, arranging appointments and browsing their online reports.
NFR-007	Accessibility	Adherence to accessibility standards is crucial to ensure inclusivity and accessibility for users with diverse needs, enhancing the overall usability of the application. Regulatory Compliance

3.5.5 Regulatory Compliance

Table 3.8: Regulatory Compliance

ID	Requirement	Description
NFR-008	Data Protection	Data protection laws and healthcare standards used by the system ensure patient information and medical records are kept confidential, securely stored and legally processed.

3.5.6 Maintenance

Table 3.9: Maintenance Requirements

ID	Requirement	Description
NFR-009	Ease of Maintenance	The system should be designed for easy maintenance and updates to address evolving needs and resolve emerging issues efficiently.

3.6 Use Cases

A UML use case diagram is a visual representation showing how different components of a system interact with each other. It illustrates the collaboration between users and various functions or tasks within the system. Use case diagrams depict the system's functionalities from the user's perspective, highlighting their needs and goals. These diagrams outline the behavior of the system and provide an overview of its structure. Use case diagrams usually comprise of actors (users), use cases (tasks or functions), and their interrelations. The use case diagrams facilitate the developers in grasping the system requirements, delineating its boundaries, and providing a viewpoint external to the system regarding its functionalities. In a way, they perform the function of a tool for summarizing the system particulars along with the interactions between users and the system [15].

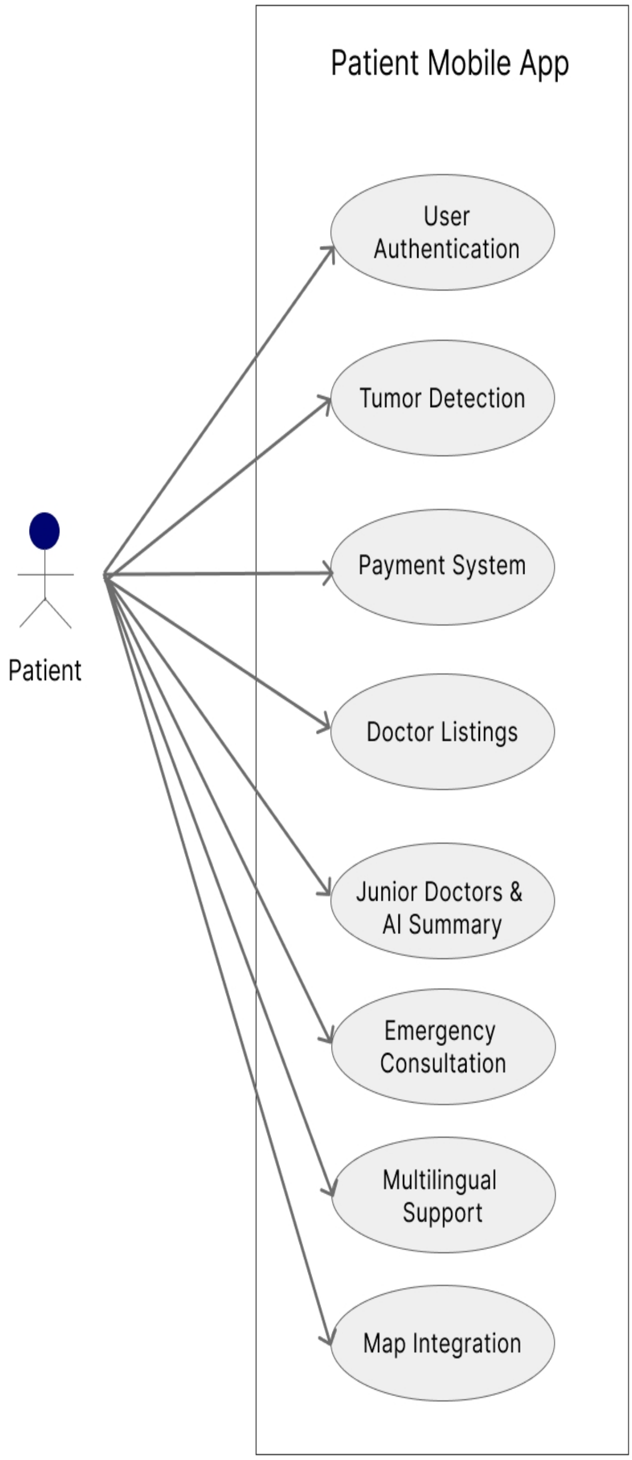

Figure 3.1: Use Case Diagram of Patient Mobile App

This use case diagram illustrates the core functionalities available to a patient user. The patient can sign-in/login, upload an MRI or CT scan for AI analysis, view results, manage payments, book doctor appointments and access all of these features in their preferred language for ease of use.

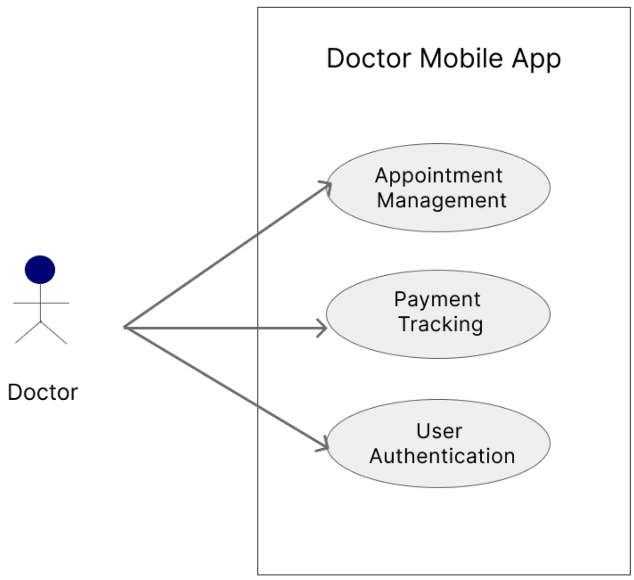

Figure 3.2: Use Case Diagram of Doctor Mobile App

This use case diagram illustrates the core functionalities available to a doctor user.Doctors can log in to their account, manage their appointments and view their payment records. The system also provides them with AI-generated case summaries to help them prepare for patient consultations efficiently.

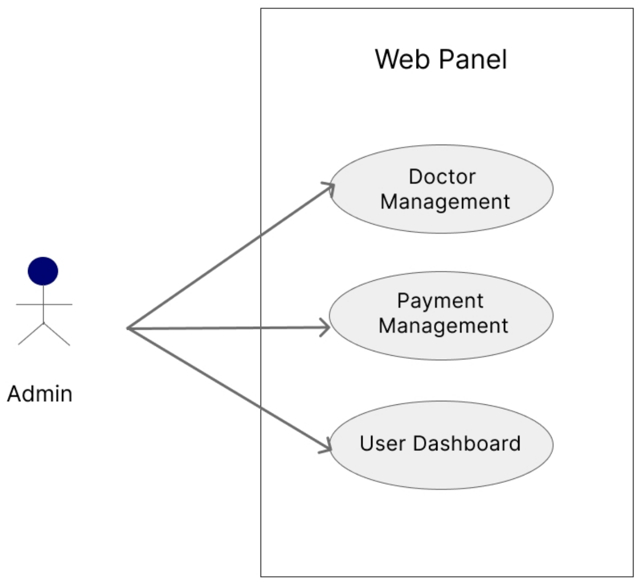

Figure 3.3: Use Case Diagram of Web Panel

This diagram shows the administrative controls for the web panel. The administrator can manage doctor accounts,admin can add or remove doctors and view financial transactions.

3.6.1 Patient Mobile Application

With this use case, patients can post their MRI results, immediately get advice from AI and benefit from healthcare services such as finding doctors and urgent help.

3.6.1.1 User Authentication

Registering and logging in for patients are important here to make sure they have proper access to the app. The following use case involved:

Table 3.10: Use Case for Authentication

Use Case ID	UC001
Actor	User
Description	Users login to their account.
Pre-Condition	User has the app installed.
Post-Condition	User successfully authenticated.
Flow	1. User opens the application. 2. User selects the "Login" option. 3. User fills in login details. 4. User verifies their account. 5. User completes the Authentication process

3.6.1.2 Tumor Detection

Patients have the option to upload their MRI results and the AI will automatically find and assess the tumor for them. Our system was used for the purpose of:

Table 3.11: Use Case for Tumor Detection

Use Case ID	UC002
Actor	User
Description	Upload the MRI scan for an artificial intelligence-powered cancer analysis.
Pre-Condition	User is authenticated
Post-Condition	The user gets a report with AI analysis results.
Flow	1. User logs in. 2. Navigates to the "Upload MRI" section. 3. Uploads an MRI image from device storage.

	4. System processes image using AI model. 5. Results are displayed.

3.6.1.3 Payment System

Secure transactions handled by Stripe for medical consultations are handled by this use case. The case study used was:

Table 3.12: Use Case for Payment System

Use Case ID	UC003
Actor	User
Description	Complete payment and consultation or services.
Pre-Condition	User is logged in.
Post-Condition	Payment is processed and receipt provided.
Flow	1. User chooses a service or doctor. 2. Clicks on the button to go to checkout. 3.Pays through the Stripe payment gateway. 4.System processes payments safely. 5.You are shown a confirmation message.

3.6.1.4 Doctor Listings

With this use case, patients have the opportunity to review a list of brain tumor specialists they may see. This use case is:

Table 3.13: Doctor Listings

Use Case ID	UC004
Actor	User
Description	Looks for and chooses doctors who are experts in treating brain tumors.
Pre-Condition	User is logged in.
Post-Condition	The appointment request is sent over.
Flow	1. User selects “Doctors Listing” from the menu. 2. Choose from results based on professionals’ specialty and location.

	3. Looks at doctor profiles and selects a doctor to visit.
	4. Asks the doctor for an appointment. 5. You are shown details about your booking such as if it is available.

3.6.1.5 Emergency Consultation

People can get in touch with a doctor right away through online consultations when they have an emergency. An example use case was:

Table 3.14: Emergency Consultation

Use Case ID	UC005
Actor	User
Description	Asks for immediate care by a medical professional..
Pre-Condition	User is logged in.
Post-Condition	Emergency consultation is provided.
Flow	1. A user picks the "Emergency Consultation" button. 2.The system is able to connect you with a doctor who is available now. 3. After consultation, the post-consultation summary is created. 4. The doctor might suggest follow-up instructions after the case.

3.6.1.6 Multilingual Support

There are many languages in this feature to help users from different cultural groups. An example of this approach was:

Table 3.15: Multilingual Support

Use Case ID	UC006
Actor	User
Description	Users can use App features in the preferred language.
Pre-Condition	User is authenticated and logged in.
Post-Condition	User can set the interface to be in your preferred language.
Flow	1.User goes to the language settings. 2. Picks the language you wish to use from the options. 3. Newly installed updates change all texts and labels in the app..

	4. The patient uses the app in the selected language.

3.6.2 Doctor Mobile Application

Doctors rely on the app to arrange appointments, get AI findings and review status updates for patients and their payments. Use cases examined:

3.6.2.1 Appointment Management

It helps doctors to keep track of, plan and manage all patient appointments quickly. Our developers tried out the following use case:

Table 3.16: Appointment Management

Use Case ID	UC008
Actor	Doctor
Description	Doctor view list of appointments.
Pre-Condition	Doctor is logged in.
Post-Condition	Apportionment is scheduled or updated.
Flow	1. Doctor clicks on the “Appointments” button. 2. System presents a list that shows the appointments. 3. Doctor decides either the time for the patient or which patient they want to see.

3.6.2.2 Payment Tracking

This use case allows doctors to monitor payments and consultation fees in real-time. The following use case was:

Table 3.17: Payment Tracking

Use Case ID	UC009
Actor	Doctor
Description	Review the money patients pay and the money earned by the organization.
Pre-Condition	Doctor is logged in.
Post-Condition	Payment records are displayed.
Flow	1. Doctor clicks on the “Payments” tab from the main menu.

	2. User gets a summary of their earnings along with the transactions made on the platform. 3. Doctor reviews each consultation payment separately.

3.6.3 Web Panel

Using this use case, administrators can handle users, specialists, scan data and process transactions from one central web interface. The following were the chosen use cases:

3.6.3.1 Doctor Management

With this use case, admin can able to add, update or remove doctor listings in the system. The case studied was about:

Table 3.18: Doctor Management

Use Case ID	UC0010
Actor	Admin
Description	Admin manages doctor profiles.
Pre-Condition	Admin is logged into the web portal.
Post-Condition	Doctor information is added, edited and deleted.
Flow	1. The admin click on "Doctor Management". 2. There is a list of all the registered doctors. 3. The system updates the doctor database following the changes. 4. The system updates the doctor database following the changes.

3.6.3.2 Payment Management

It supports dealing with and following up on the transactions made during doctor consultations. That situation was:

Table 3.19: Payment Management

Use Case ID	UC0011
Actor	Admin
Description	Admin manages doctor transactions.
Pre-Condition	Admin has access to the payment section.
Post-Condition	Payment records are managed and displayed.
Flow	1. Admin clicks on the “Payment Management”.

	2. System displays all doctor-related transactions. 3. Admin filters payments by doctor name, date, or status. 4. Can verify completed payments or flag pending ones. 5. Any updates or issues are recorded in the system.

3.7 Project Feasibility

The main users of this brain tumor detection system are neurologists, radiologists, general physicians and hospital administrators. Patients and their families also benefit from the system since earlier diagnosis is possible and they have access to specialists. Integrating brain tumor diagnosis using AI and mobile/web apps allows the platform to intervene sooner and more accurately.

3.8 Summary

Deep learning models and modern development tools are used by the system to deliver a brain tumor diagnosis service that is both efficient, scalable and accessible. Using Flutter and Flask, it can be launched on both mobile and web platforms without hassle. Because the platform speeds up MRI analysis and helps doctors decide on diagnoses, it helps patients get treated faster. At the center of this solution is AI and it handles the weaknesses in standard diagnostics, creating a helpful network for medical teams.

Chapter 4

Design

This chapter explains how the Brain Tumor Segmentation and Detection System Using Neural Networks is designed and it mentions its limitations, methodology and architecture.To speed up the diagnosis, the system replaced manual MRI analysis with an automated version, because analysis done manually can be slow, might be open to error and could be difficult for people to interpret the same way. To do a precise job of segmenting and sorting out tumors in the MRI scans, the system relies on CNN, U-Net and VGG16. Datasets and model configurations were tested to see which gave the best results [16].

Along with implementing the algorithms, the project includes working mobile apps for patients and doctors as well as an administrative dashboard used through the web. The application components are connected through a backend built in Flask and powered by Firebase and it is deployed in the cloud on services like AWS or GCP. Because scalability and usability are important, the interface provides multilingual support, doctors can be contacted in real time, care summaries are made by AI and payment transactions are secure. Using careful design and thinking about the user, the system tries to develop a platform that is accessible, smooth and ready for the future in brain tumor care.

4.1 System Architecture

Brain Tumor Segmentation and Detection System can be seen as a whole system made of smart, connected parts that are meant to simplify brain tumor management. You can see below a diagram of the system's main modules:

- **Patient Mobile App**: This application shows how patients access their account, supplying them with secure options to upload scans and receive tumor segmentation. It has multilingual options, matters regarding doctor selection, connects to Stripe for payment methods and has emergency consultation tools to keep the service open and helpful to all.
- **Doctor Mobile Application**:Developed for those in healthcare, the app lets doctors book and manage their appointments, get summaries from AI and monitor consultation payments. The interface is both secure and user-friendly which helps with workflow and interacting with patients

- **Web Panel**:The web panel gives administrators a base to handle doctors, watch over transactions and manage system data. It helps run the systems in the background and lets us watch the site as it operates while handling platform management efficiently.

- **Database**:Firebase on the cloud handles the storing of data for user profiles, medical reports, financial transactions and log records. Thanks to it, users can synchronize in real time and also use the application when there is not enough connectivity.

- **Backend and AI module**:A backend API built with Flask makes it possible for front-end apps to use the TensorFlow and PyTorch models. Such models support jobs such as tumor segmentation, figuring out cancer and localization. The AI module has been implemented using AWS or GCP for flexible and secure processing in real time.

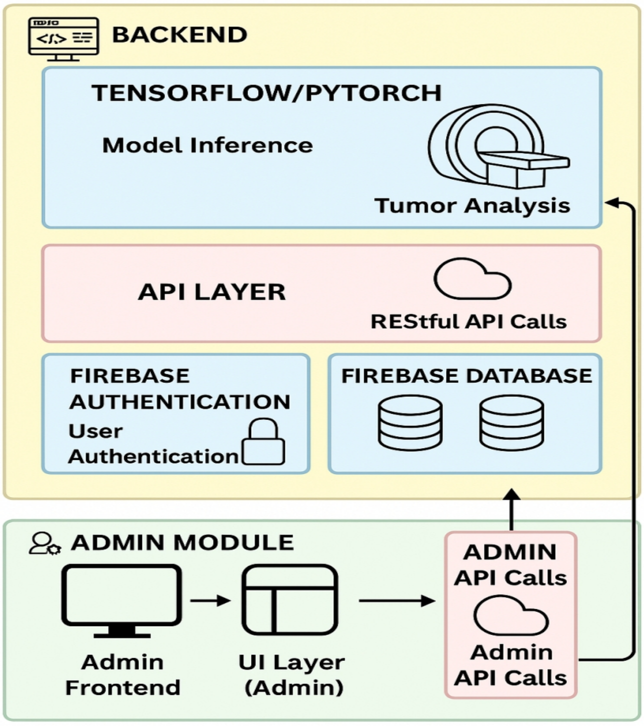

Figure 4.1: Detail System Design Architecture

4.2 Design Constraints

While planning the Brain Tumor Segmentation and Detection System, some important factors were considered:

- **Balancing Diagnostic Precision and Performance**: It was necessary to reach both the high accuracy of tumor segmentation and detection and a smooth operation on platforms like mobile apps and web panels.

- **Security and Data Privacy**: The protection of user data was the most important. Sensitive information was protected by enforcing very strong security techniques

- **Scalability and Future Growth**: The design permitted the addition of more diagnostic tools, the support of deep learning models, and the increase of the user base in a simple and flexible way. It implies that the system will be able to adapt to changes in medicine and be accepted by a larger number of clinicians in the future.

4.3 Design Methodology

The Brain Tumor Segmentation and Detection system was built with Agile methodology. Agile methodologies emphasize the importance of teams being flexible and delivering results frequently through iterations. The main concepts that guided the approach were :

- **Rapid Prototyping**: The prototypes of the models were developed and tested in small steps and we evaluated their outcomes and observed how users interacted with them on both mobile and web platforms.
- **Incremental Delivery**: The system is released in functional increments, with a new set of features and functionalities coming out in a timeline of releases. This enables the system to be continuously improved and user-friendly gradually through the feedback received.
- **Flexible Design**: The system has a variety of features to incorporate new data, alter models and offer more options for evaluating patients. It was designed to be very scalable and to facilitate change should the business environment or patient needs alter.

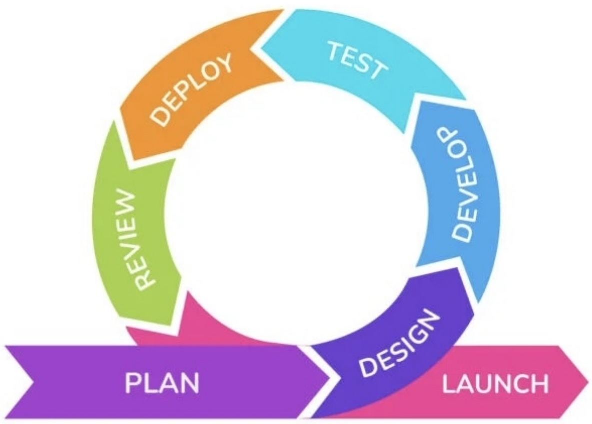

Figure 4.2: Agile Methodology

4.4 High-Level Design

Multiple methodologies were employed to analyze the working of the Brain Tumor Segmentation and Detection System:

- **Functional View**: It shows the main features that the users see and use. It gives user stories like uploading of MRI scan images, AI-assisted tumor analysis, and managing their appointments along with the option of remote consultations. These are intended to ensure that the software is very user-friendly for both patients and doctors.

- **Data Model**: A Flask API integration enables the frontend applications to easily connect to and communicate with backend services. They manage data transfer for MRI, identity verification, appointment scheduling and report delivery, making the system both robust and reactive.
- **API Design**: A Flask API integration helps the frontend applications connect to and interact with backend services without trouble. They look after data transfer for MRI, verify identities, organize appointments and handle report delivery so the system is both strong and responsive.

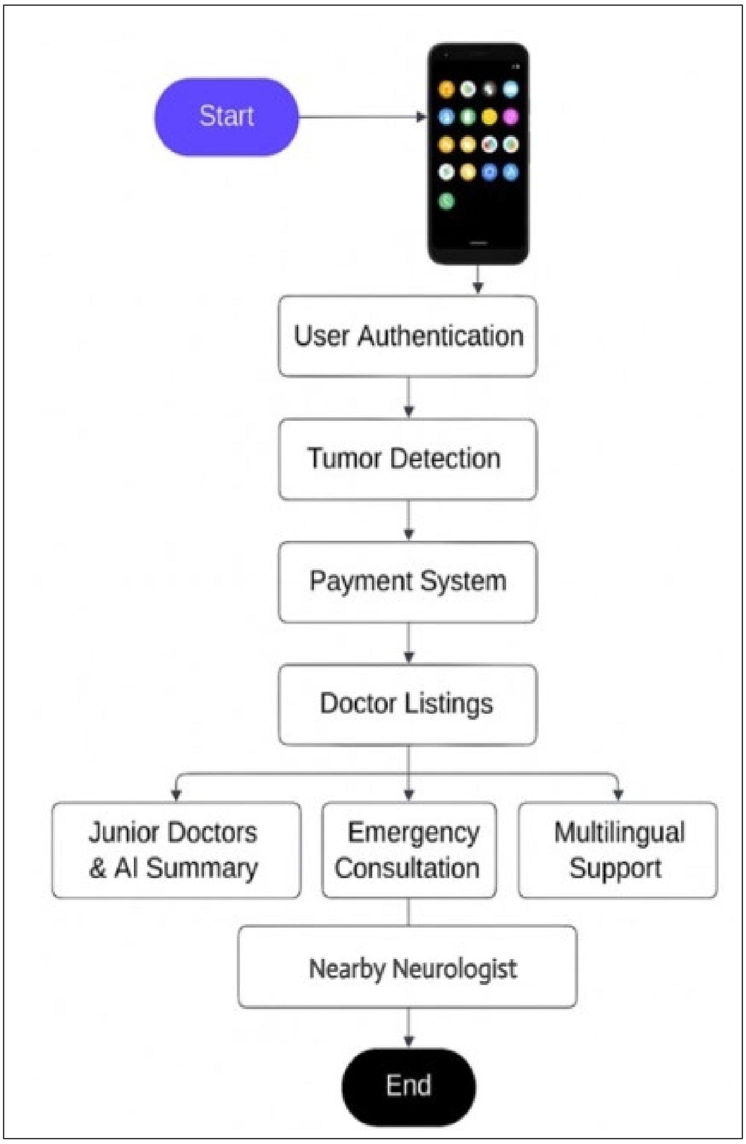

Figure 4.3: Activity Diagram – Patient Mobile App

This diagram shows a patient's step-by-step workflow using the mobile app. After starting up and logging in,user can perform several activities: tumor detection, managing his payments, finding doctors, using features for emergency consultations. The process then ends when the user exits the application after having access to all services.

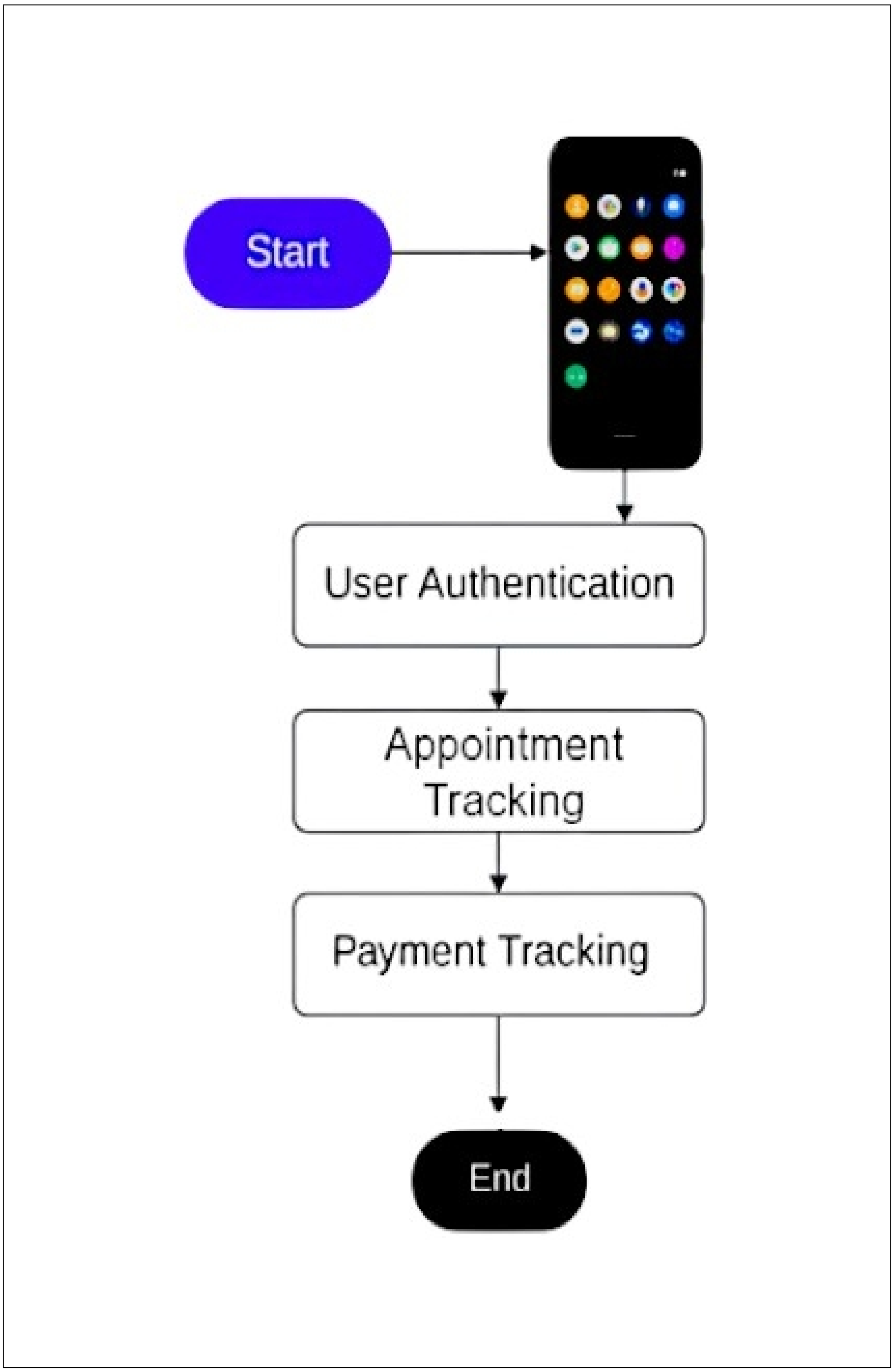

Figure 4.4: Activity Diagram – Doctor Mobile App

This diagram illustrates the workflow for doctors using the mobile application.First, a doctor needs to log into the system, then manage and keep track of their appointment and consultation payments.

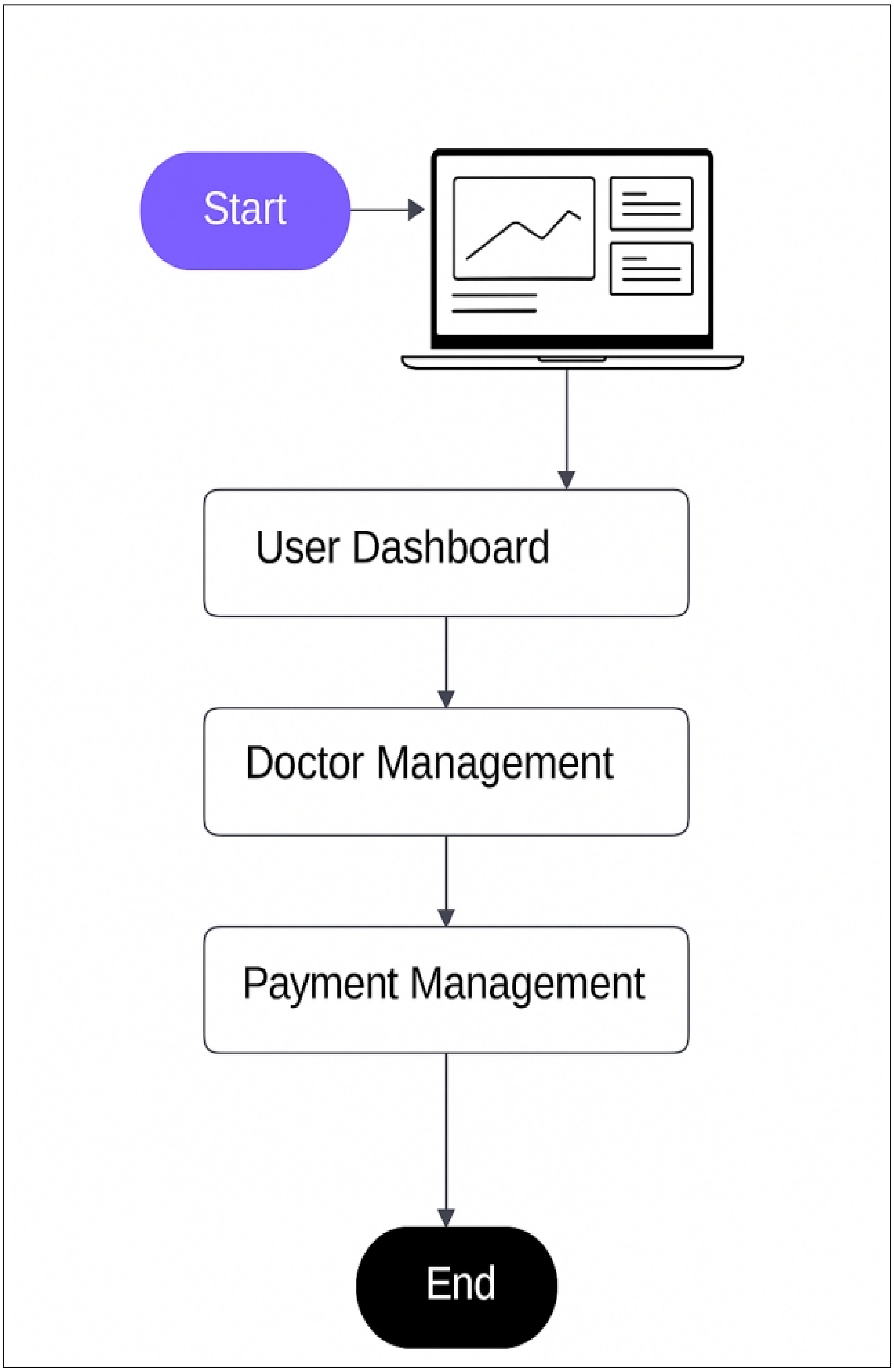

Figure 4.5: Activity Diagram – Web Panel

This diagram shows the workflow for a admin using the web panel.After logging in, the admin lands on their dashboard where they can access two main features: managing their doctors (adding and removing) and handling payments for services.

4.5 Low-Level Design

The low-level design provides detailed specifications for component development. Here, components would be broken down into smaller, manageable units (functions, modules) with well-defined interfaces. This modular approach promotes code re-usability and maintainability

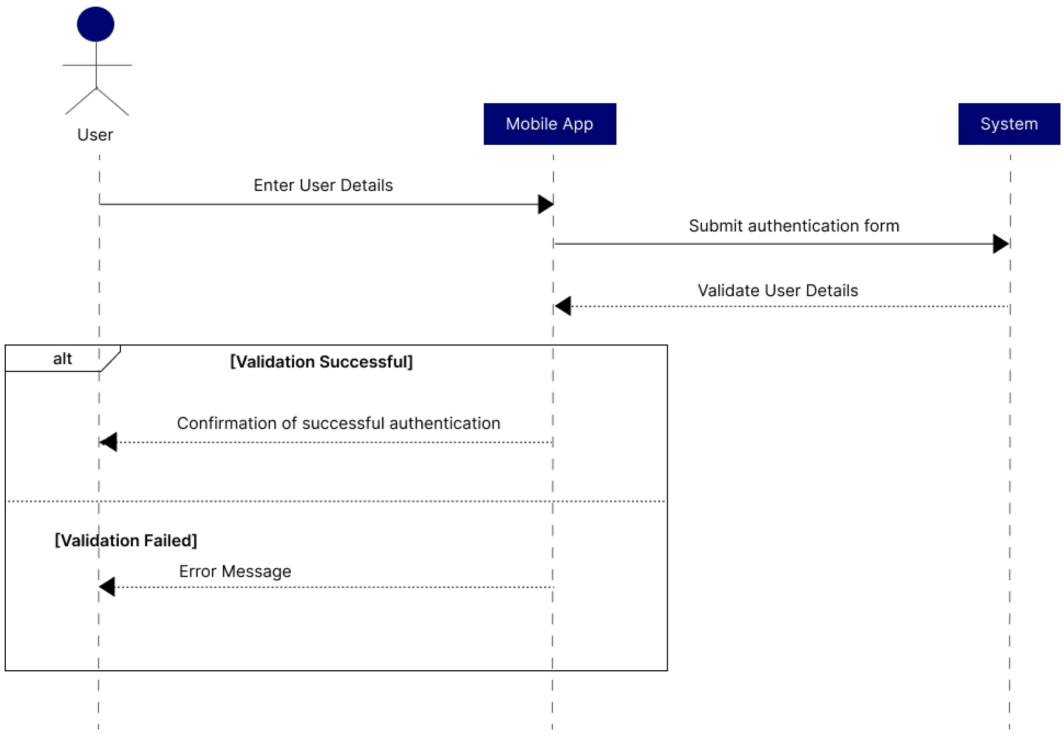

Figure 4.6: User Authentication Sequence Diagram

This diagram describes the steps involved in a user's login to the application. It shows the interaction between the user, the interface of the mobile application, the server of the system, and the database. The sequence checks the credentials for login verification, then confirms the identity with the database, and finally gives access to the application.

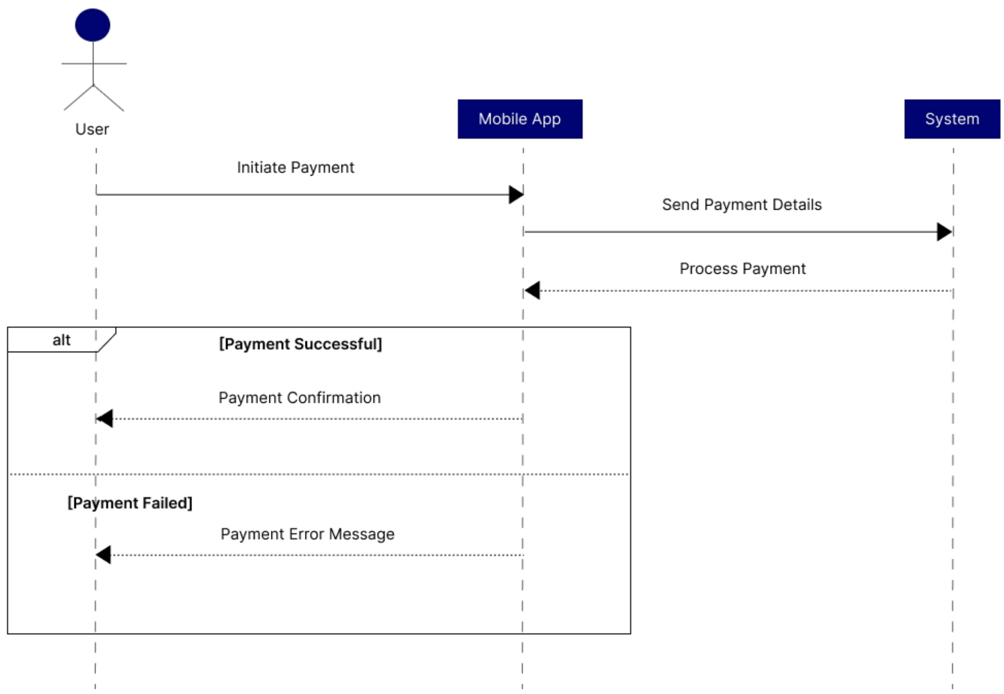

Figure 4.7: Payment System Sequence Diagram

This diagram illustrates the step-by-step process of making a payment through a mobile app. It shows the initiation of the payment from the application, the process of sending the information to the system server for processing, and then receiving the result in response to whether that process has been successful or failed.

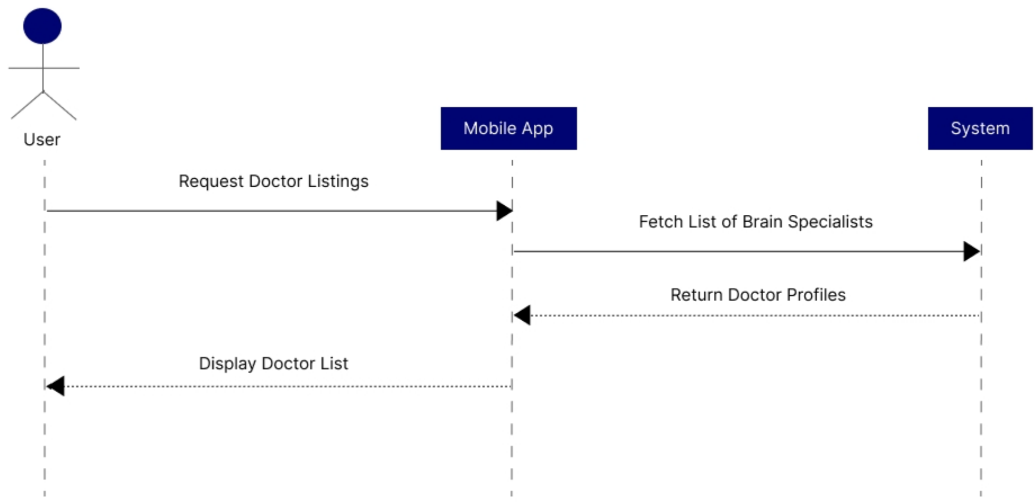

Figure 4.8: Doctor Listings Sequence Diagram

This diagram illustrates the process by which the mobile application enables a user to find and view doctors. The user requests to see available brain specialists, which triggers the app to fetch this list from the system server. Then the server responds with the profiles of qualified doctors, and finally, it lists them for the user to browse in the app.

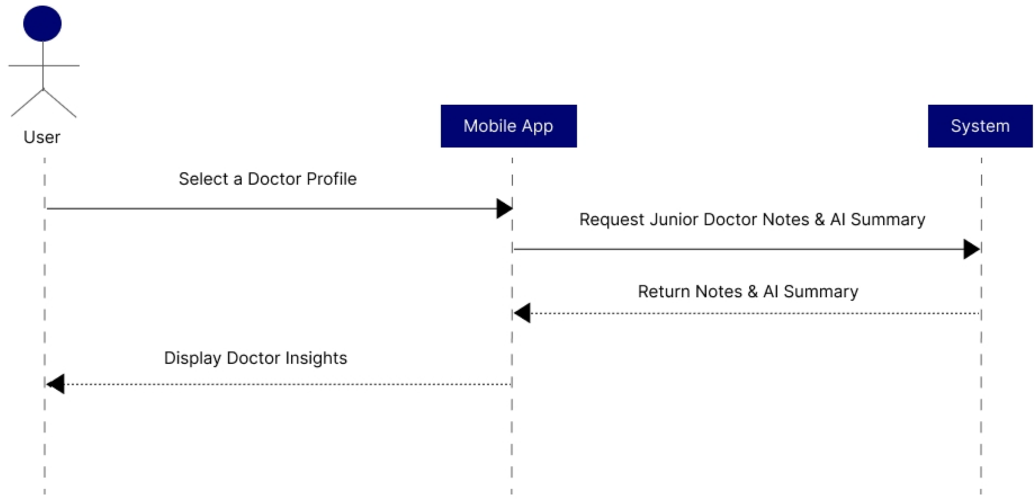

Figure 4.9: Junior Doctors and AI Summary Sequence Diagram

This diagram shows how a user gets AI-powered insights about a doctor before booking. When the user selects a doctor's profile, the app requests a summary and notes from the system. Then, the system responds with an AI-generated summary about the doctor, which is displayed by the app to help the user make an informed decision.

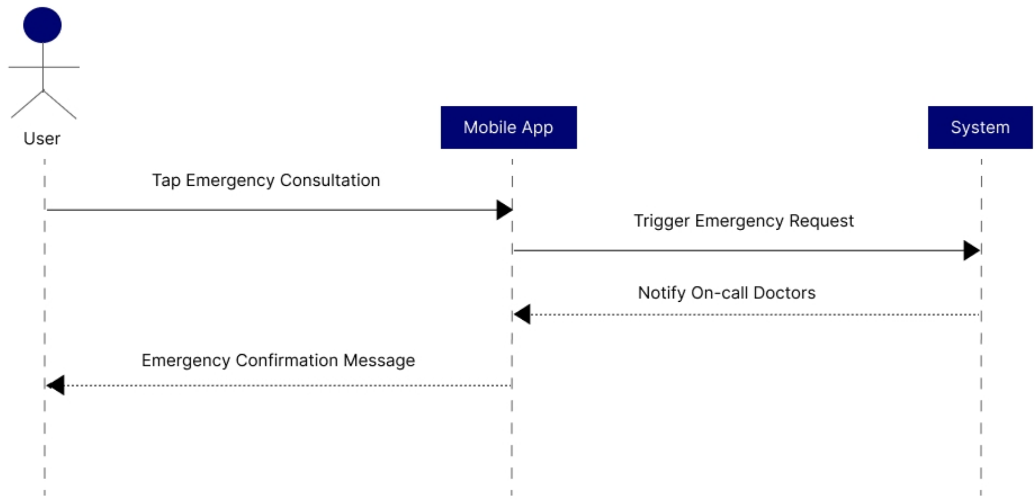

Figure 4.10: Emergency Consultation Sequence Diagram

This diagram shows the process of emergency consultation on the app. Whenever a user taps on the option for emergency, instantly, the application sends a request to the system. The system then notifies all the on-call doctors that are available, and finally, a confirmation message is sent to the user if the sending of the request is successful.

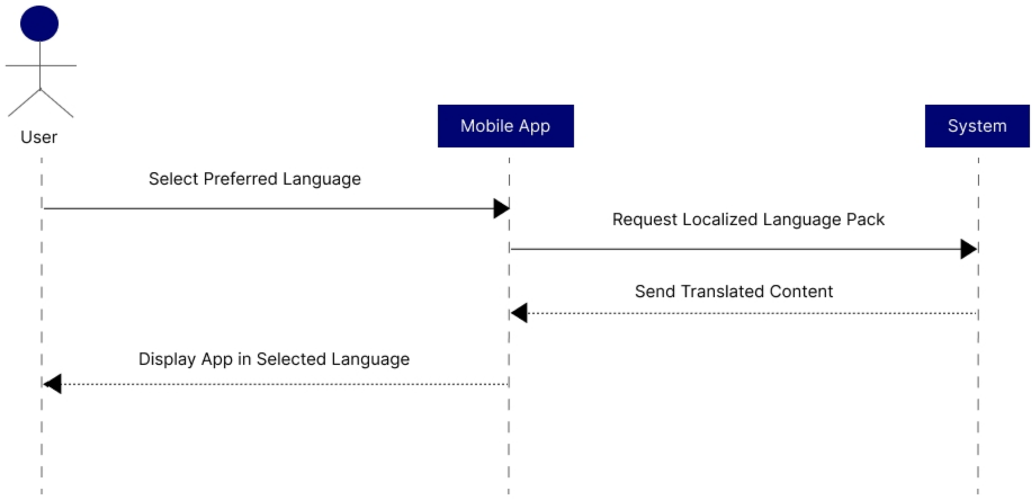

Figure 4.11: Multilingual Support Sequence Diagram

This diagram shows the flow of how the app changes languages for the user. A person selects their preferred language, and the app asks the system for the matching language pack. The system responds with all the translated text, and the app refreshes instantaneously with everything in the selected language.

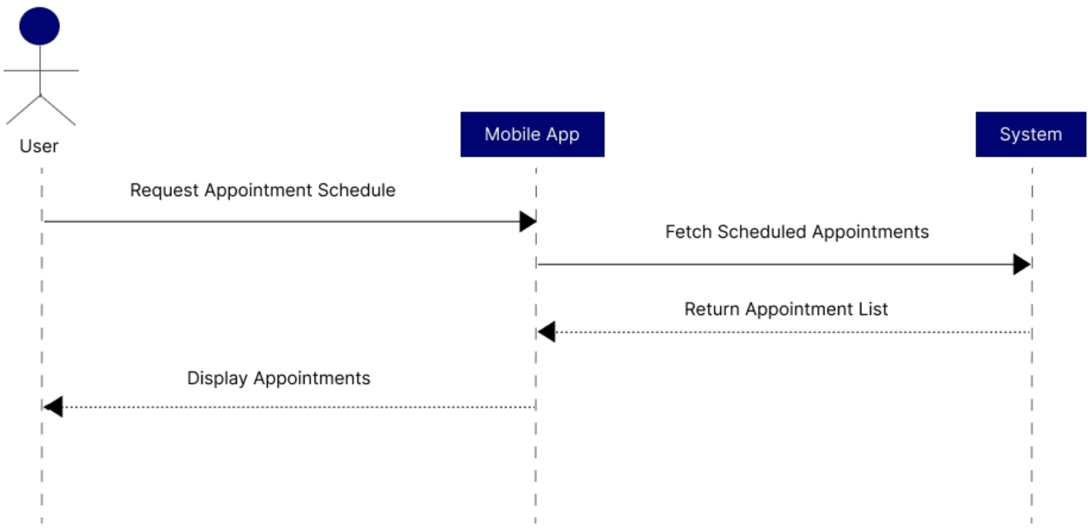

Figure 4.12: Appointment Management Sequence Diagram

This diagram illustrates the step-by-step process that is followed in managing appointments through the app. It shows the user selecting an appointment slot, the mobile app sending this request to the system, and the system confirming the booking. The successful confirmation is then displayed back to the user in the app interface.

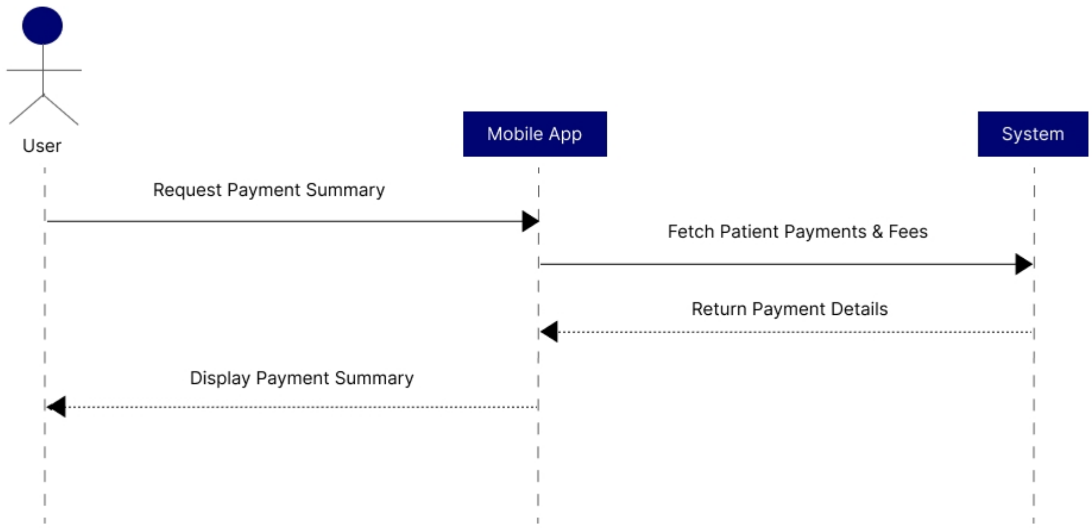

Figure 4.13: Payment Tracking Sequence Diagram

This diagram demonstrates how a user can look at their payment history within the application. Once the user requests a summary of their payments, the app connects with the system to obtain the details of the user's payments and fees. Financial information is returned by the system to be sorted and summarized by the app for easy review by the user.

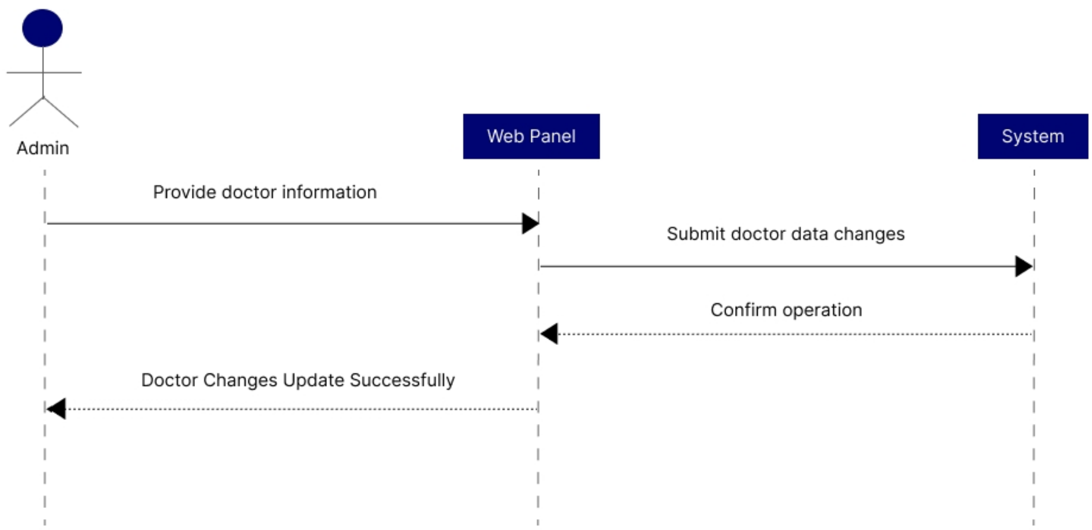

Figure 4.14: Doctor Management Sequence Diagram

This diagram illustrates how an administrator navigates through the web panel to update a doctor's information. An admin submits the changes to a doctor's profile, which then gets forwarded to the system for processing. After that, confirmation from the system states that the update was successful, and the admin is shown confirmation of that doctor's information being successfully changed in the system.

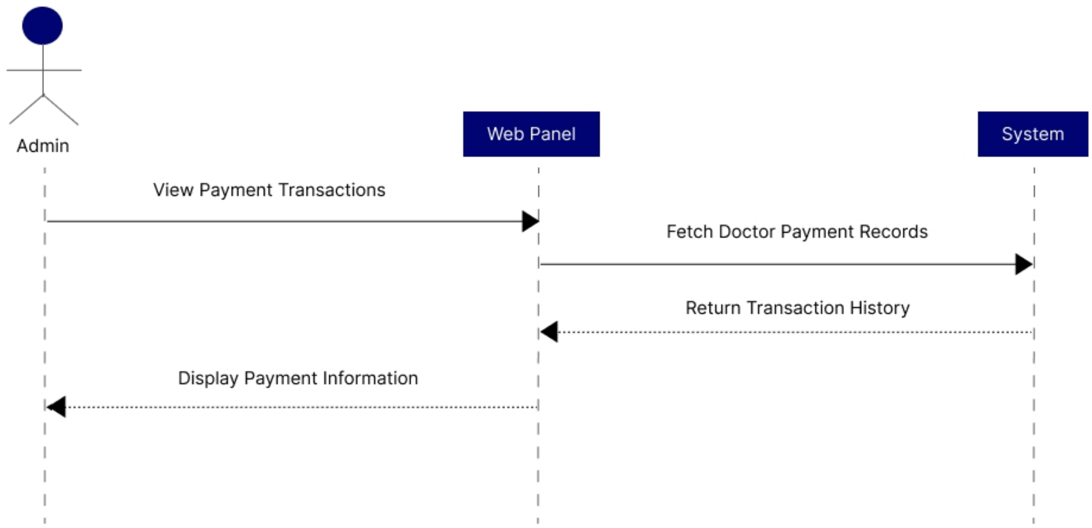

Figure 4.15: Payment Management Sequence Diagram

This diagram shows how an administrator in the system would monitor financial transactions. The admin, through the web panel, requests a view of the payment records. The system retrieves all histories of doctor payments. Thereafter, the system returns the transaction data, and the web panel displays full payment information to the admin.

4.6 GUI Design

The Brain Tumor Segmentation and Detection system utilizes the power of advanced neural networks to increase the accuracy and efficiency of brain tumor diagnosis and treatment. It has been built in such a way that it caters to the specific needs of patient, doctor, and administrator through separate mobile apps and a web panel. The main features and factors for each platform are discussed below:

- **Navigation**: The navigation of the app should be straightforward and simple to use. Think about using a tabbed layout or a bottom navigation bar that provides quick access to the main features.
- **Search Functionality**: The search must be fast and should facilitate filtering by such parameters as date, place, person details, and desired intake.
- **Virtual Assistant Integration**: The virtual assistant (powered by Flask API) should be integrated impeccably. Users would be able to inquire about brain tumors, get instant results, and even have their insurance details analyzed for further advice.

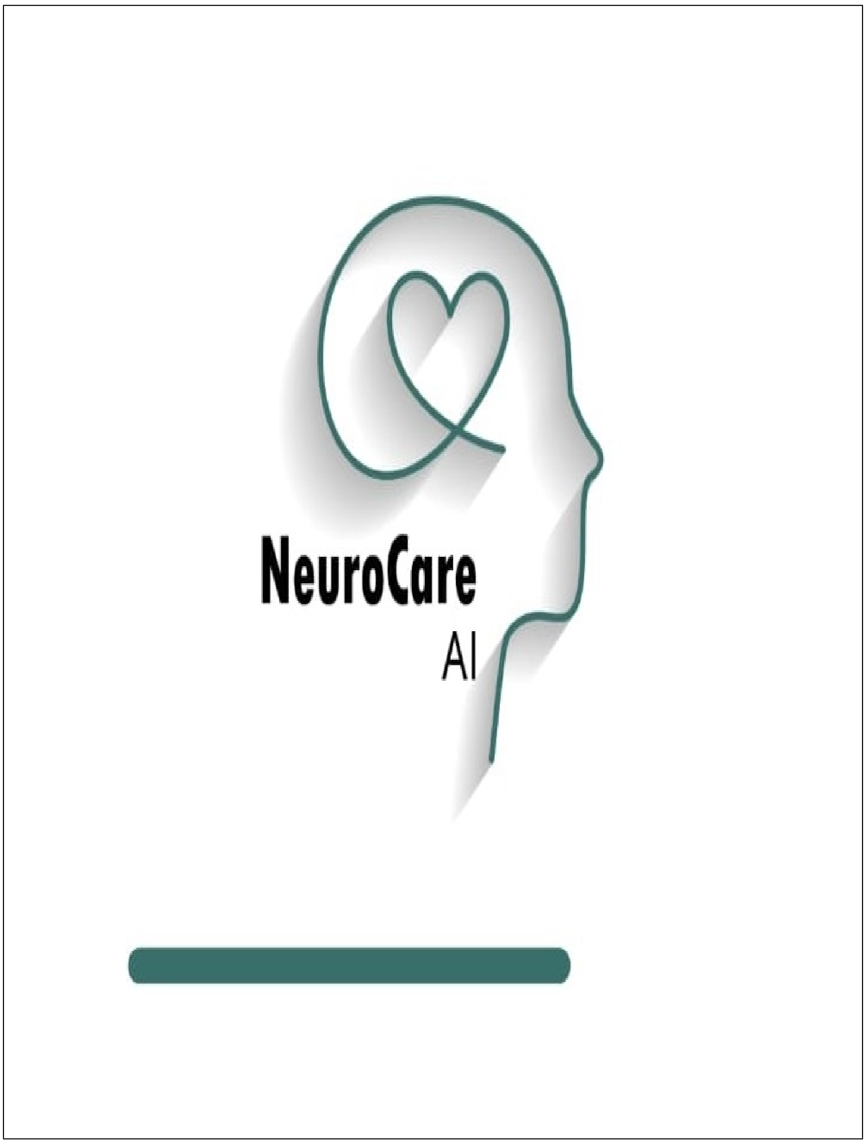

Figure 4.16: Patient App Launch Screen

The launch screen appears when patients open the app, displaying the “NeuroCare AI” logo as the system loads.

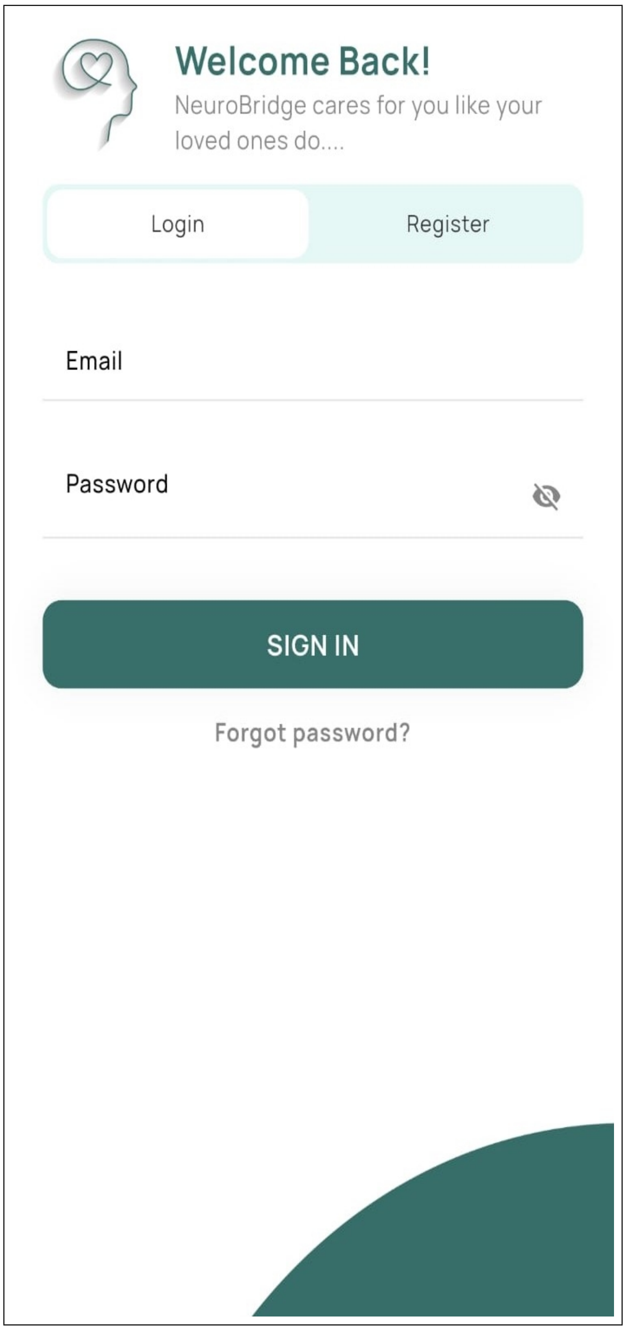

Figure 4.17: Login Page

The patient authentication screen appears when users access the app, presenting secure login fields for existing patients and registration options for new users as the system initializes.

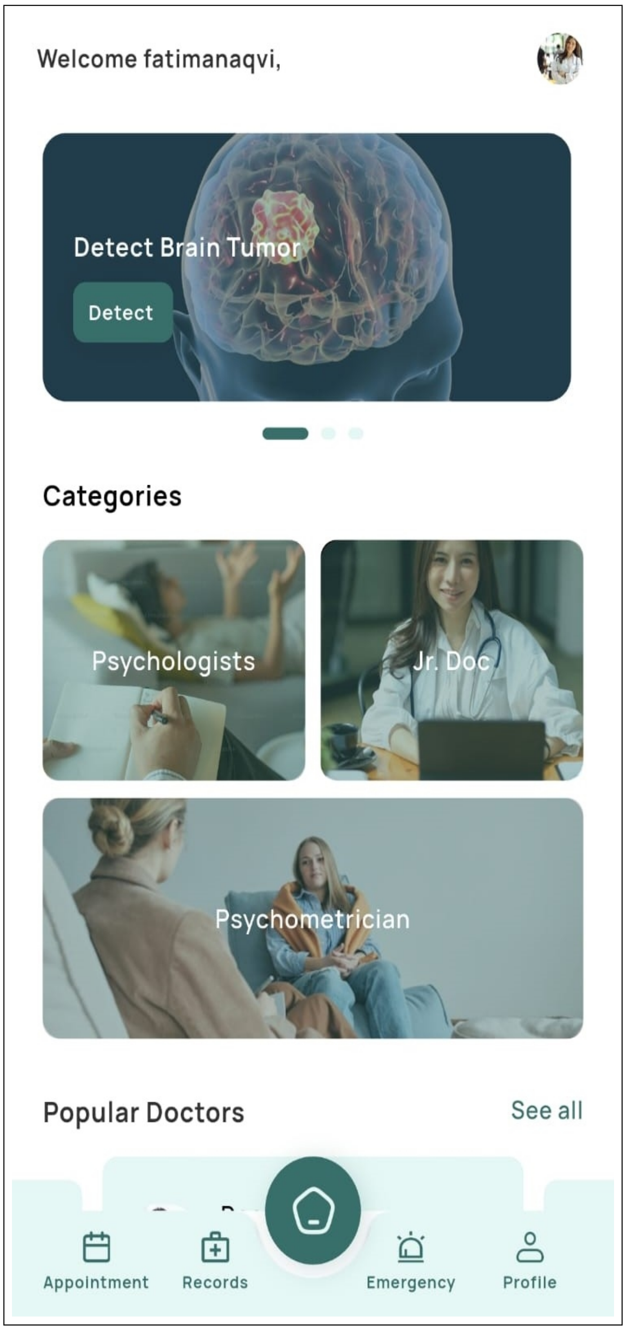

Figure 4.18: Patient App Dashboard

The patient dashboard provides accessible options for tumor detection and enables patients to consult various healthcare professionals, including senior doctors junior doctors, and psychometrists.

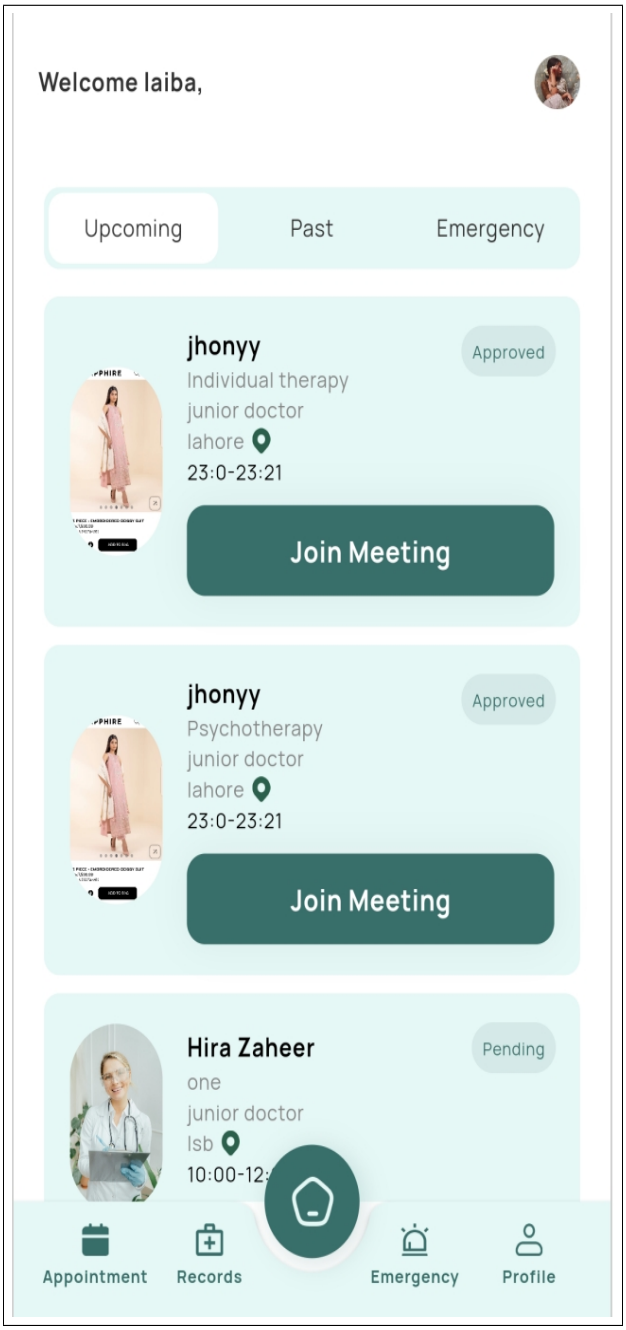

Figure 4.19: Patient App Appointment Management

The appointment management screen allows patients to view their upcoming, past, and emergency appointments, ensuring efficient tracking and streamlined scheduling.

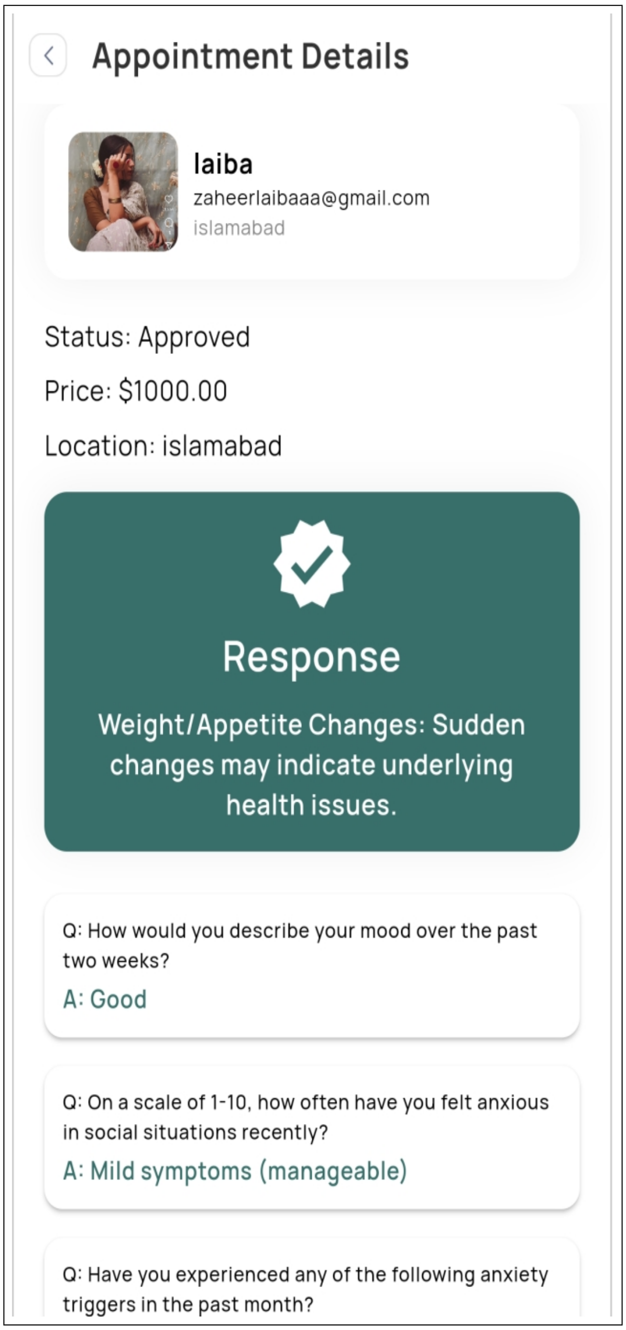

Figure 4.20: Appointment Details

The detailed appointment view presents the appointment status, location, price, patient health responses, assessment question answers, and an AI-generated health summary, enabling clear and effective tracking.

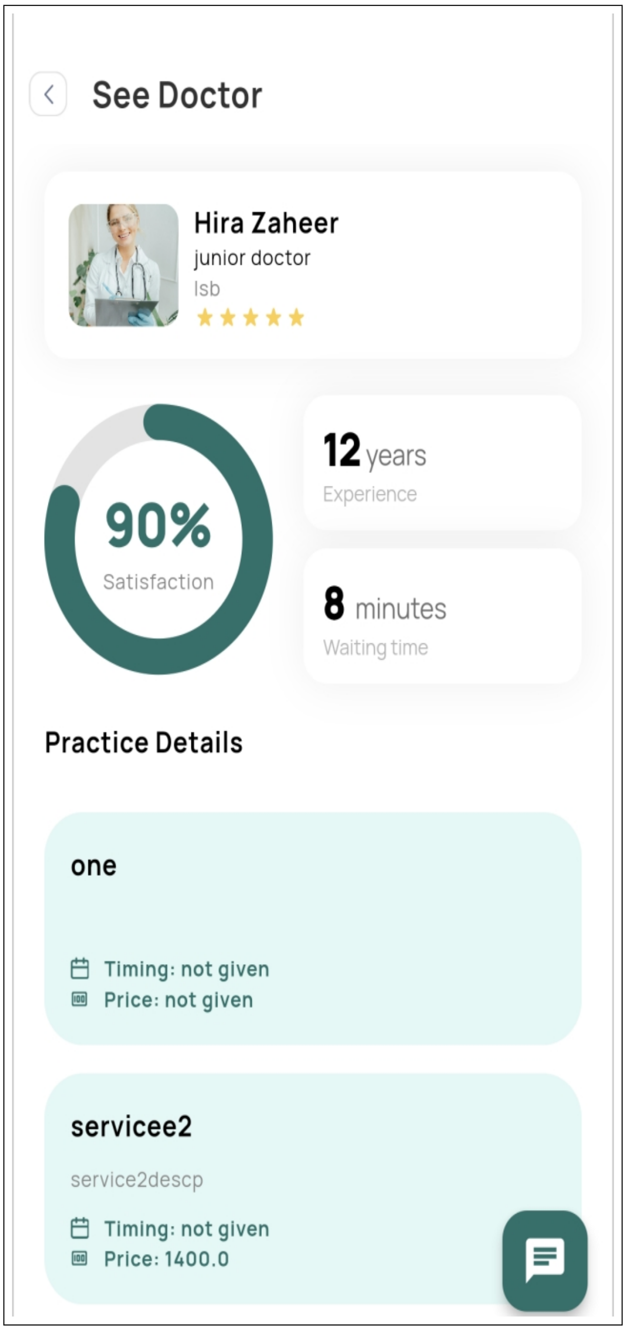

Figure 4.21: Doctor Listings

The detailed doctor listing displays each doctor's name, position, location, experience, waiting time, consultation fees, availability, and patient ratings, enabling effective selection and comparison.

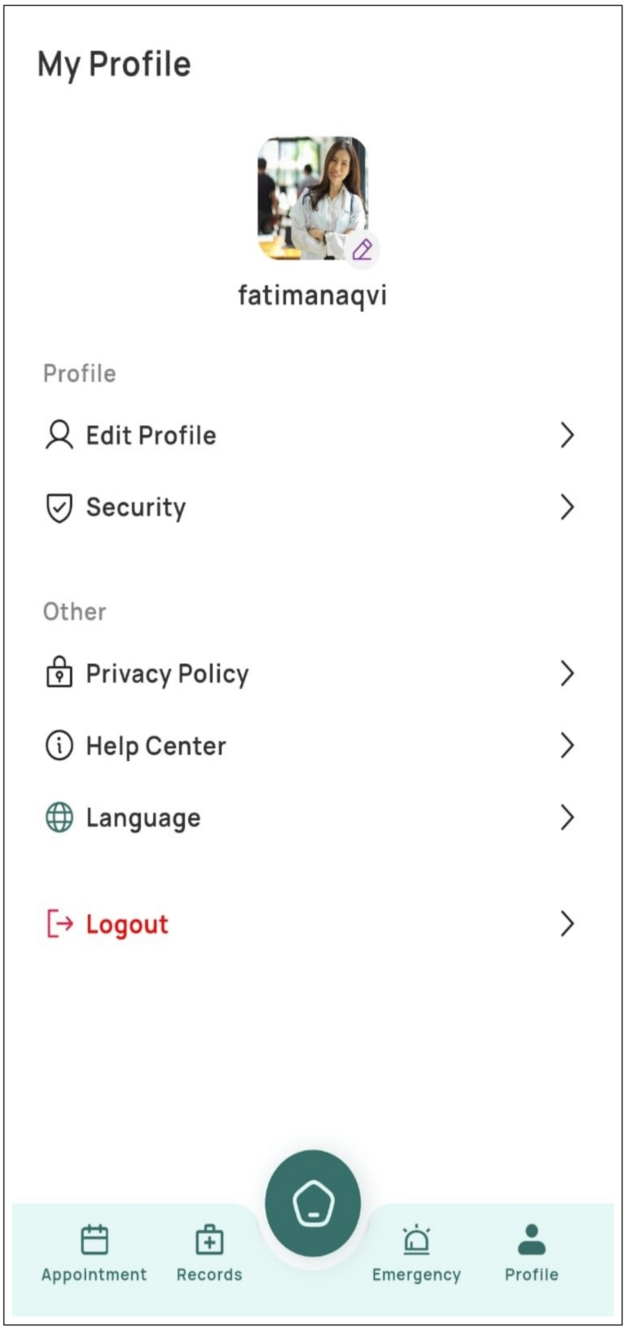

Figure 4.22: Edit Profile

The patient profile settings screen provides options to update personal information, manage security and privacy preferences, access the help center, change the app language, and log-out, ensuring effective account management.

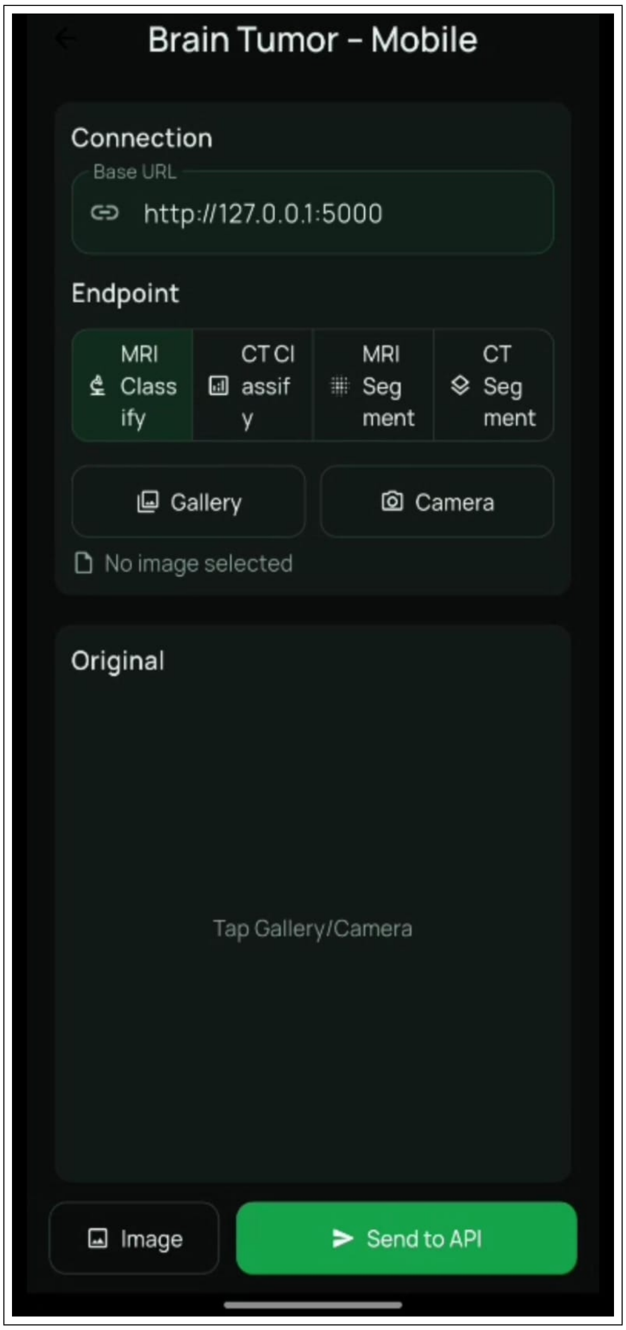

Figure 4.23: Tumor Detection

The tumor detection feature allows patients to upload MRI images from the gallery or capture them via camera, choose between classification or segmentation, and receive AI generated analysis results for brain tumor detection.

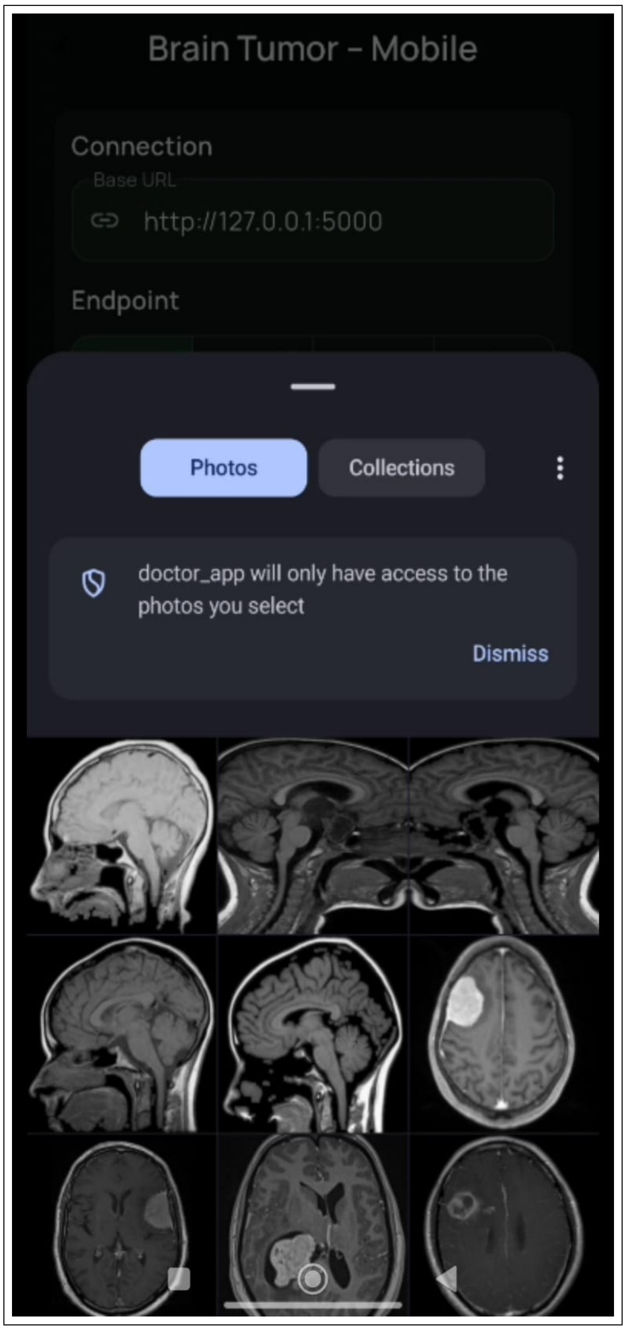

Figure 4.24: Upload MRI/ CT Scans

The MRI/CT scan upload interface enables patients to select images from the gallery or capture them via camera for brain tumor analysis.

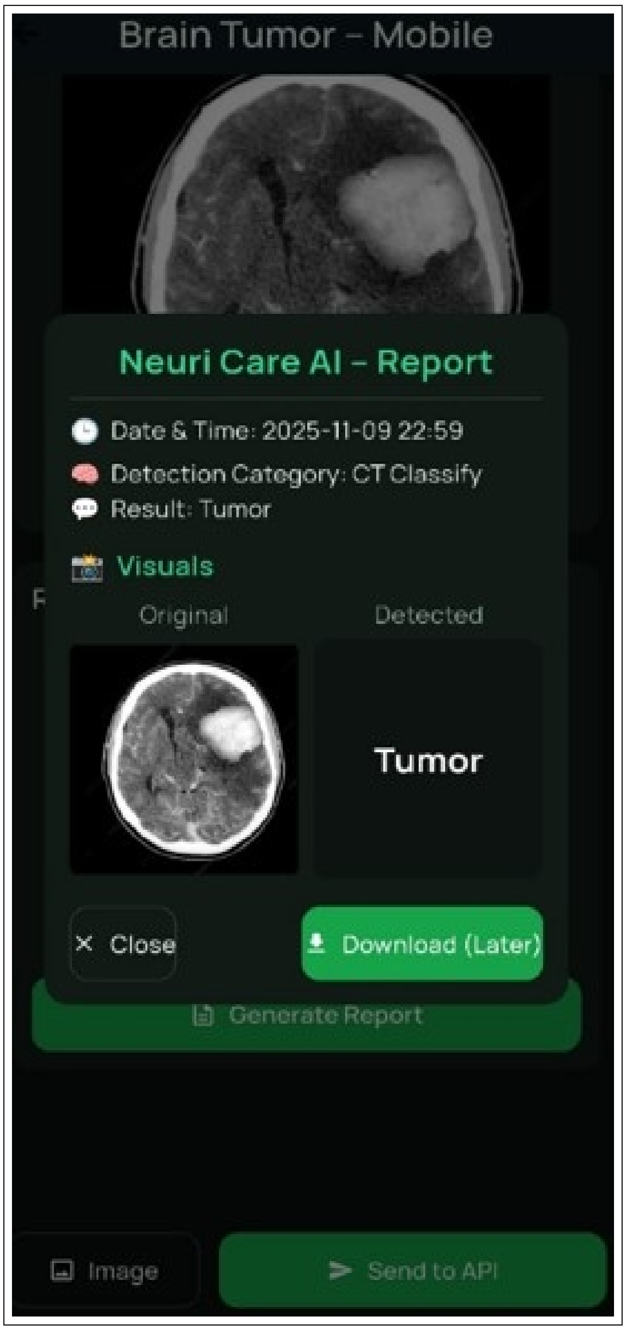

Figure 4.25: Result: Tumor Detected

The result view displays a scan in which a tumor has been detected, indicating the presence of a brain tumor based on the analysis of the uploaded MRI and CT scans.

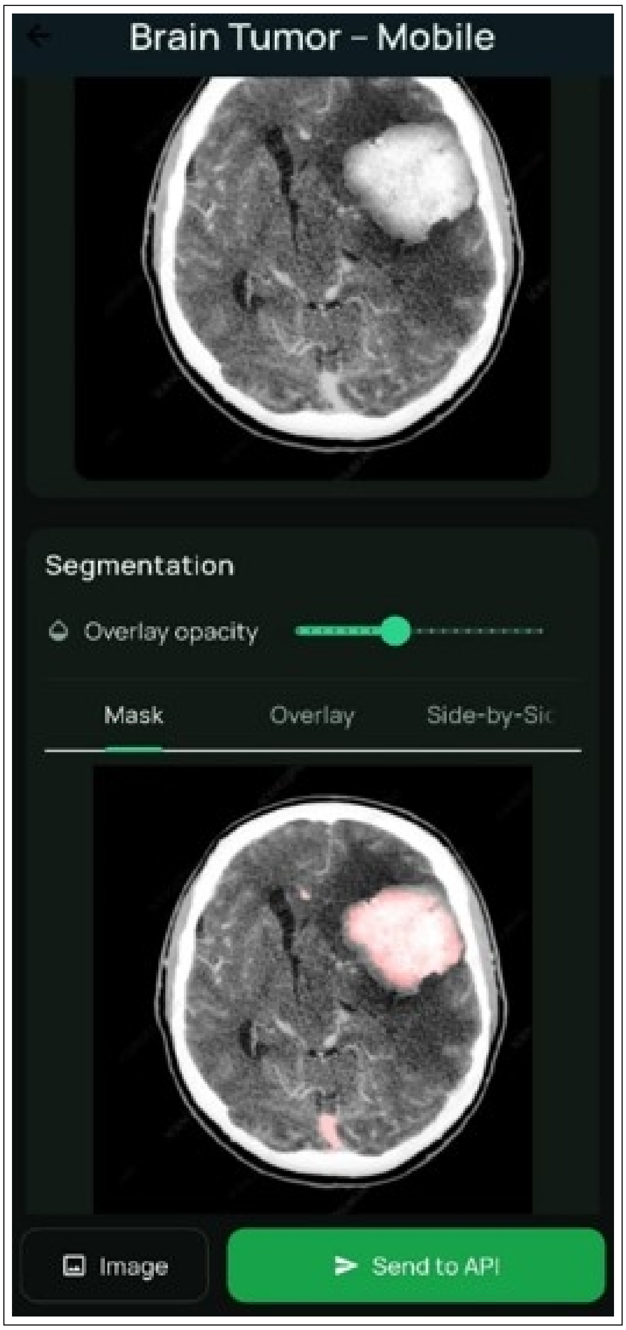

Figure 4.26: Segmentation Result

The result view displays a scan with the detected tumor highlighted, based on the analysis of the uploaded MRI and CT scans.

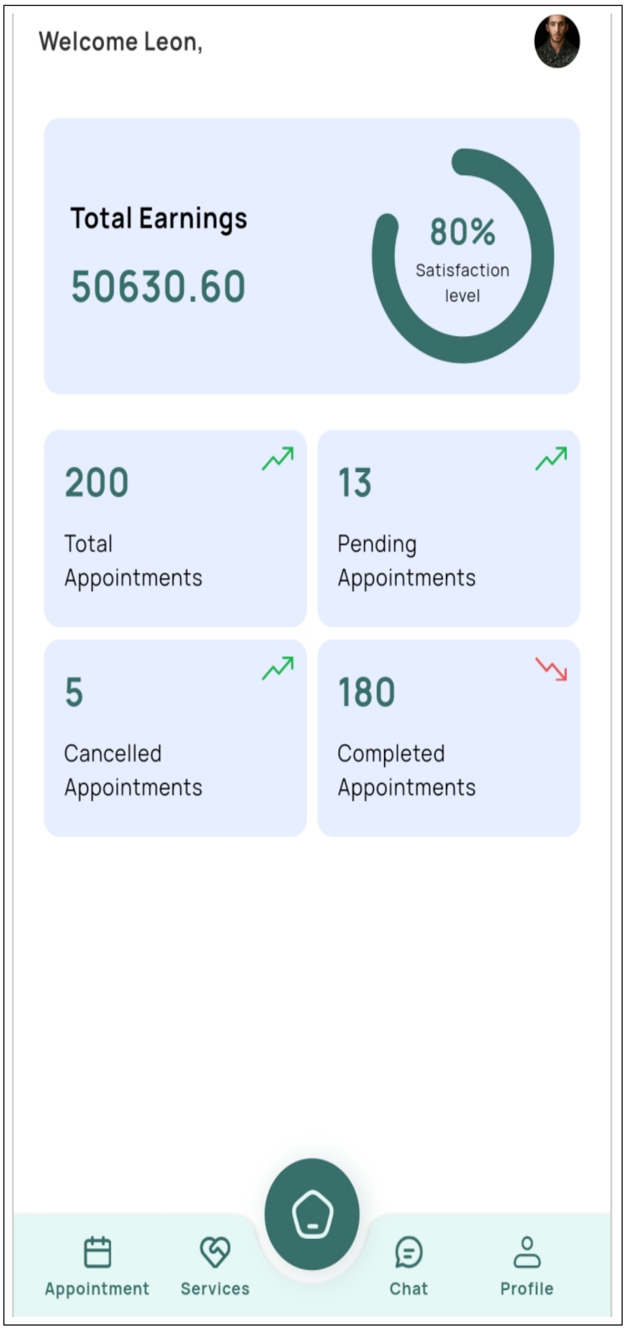

Figure 4.27: Doctor App Dashboard

The Doctor App dashboard displays total earnings and detailed appointment statistics, including total, pending, canceled, and completed appointments, enabling effective management and tracking.

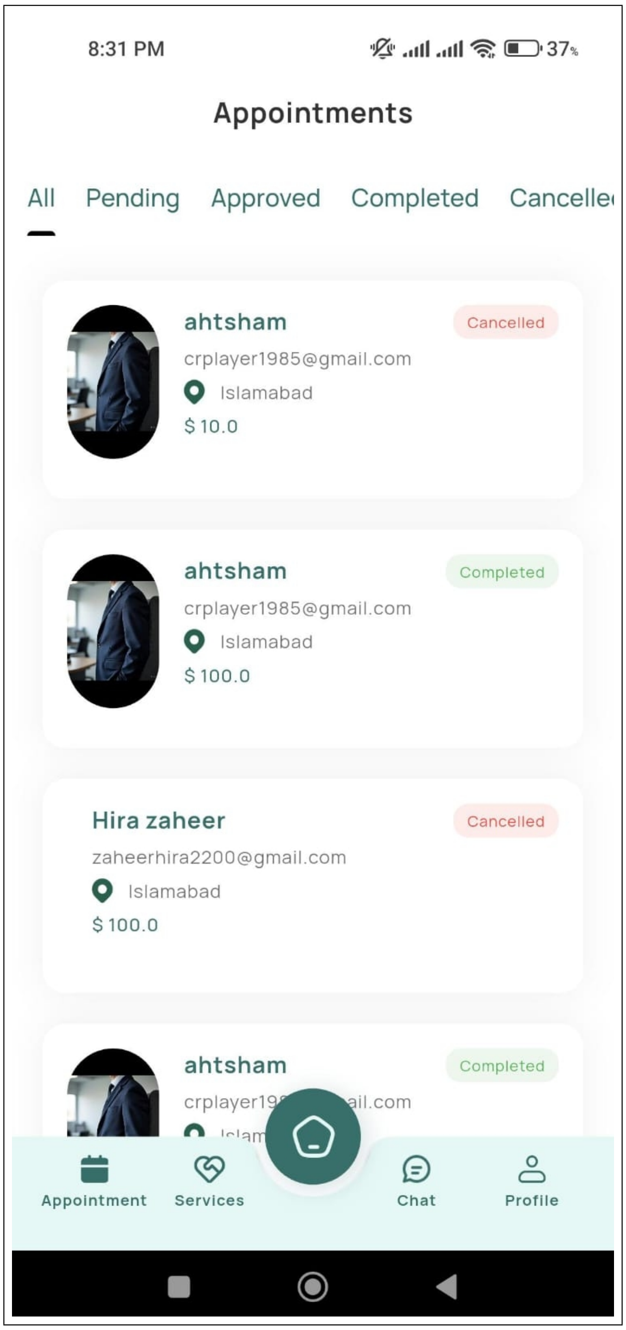

Figure 4.28: Doctor App Appointment Management

The Doctor application appointment management view presents individual patient details along with their appointment information for comprehensive tracking.

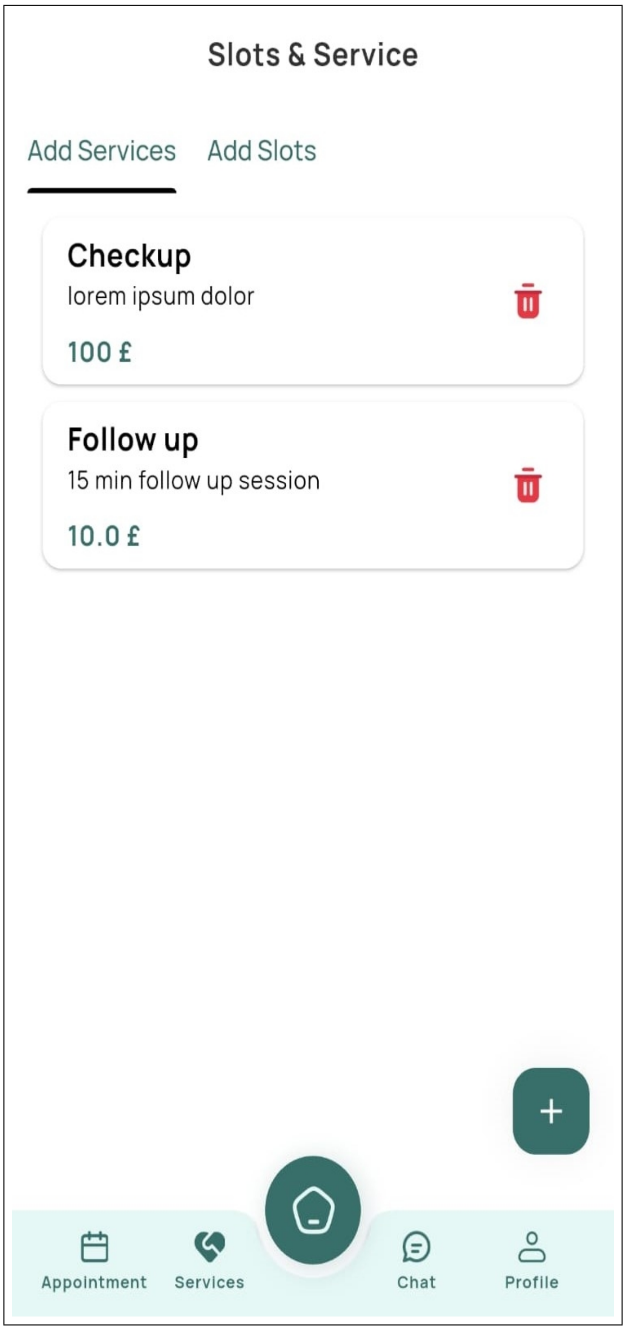

Figure 4.29: Doctor App Services

The Doctor application Services section allows doctors to manage their services and time slots, including options to add or delete services and available appointment slots for improved schedule management.

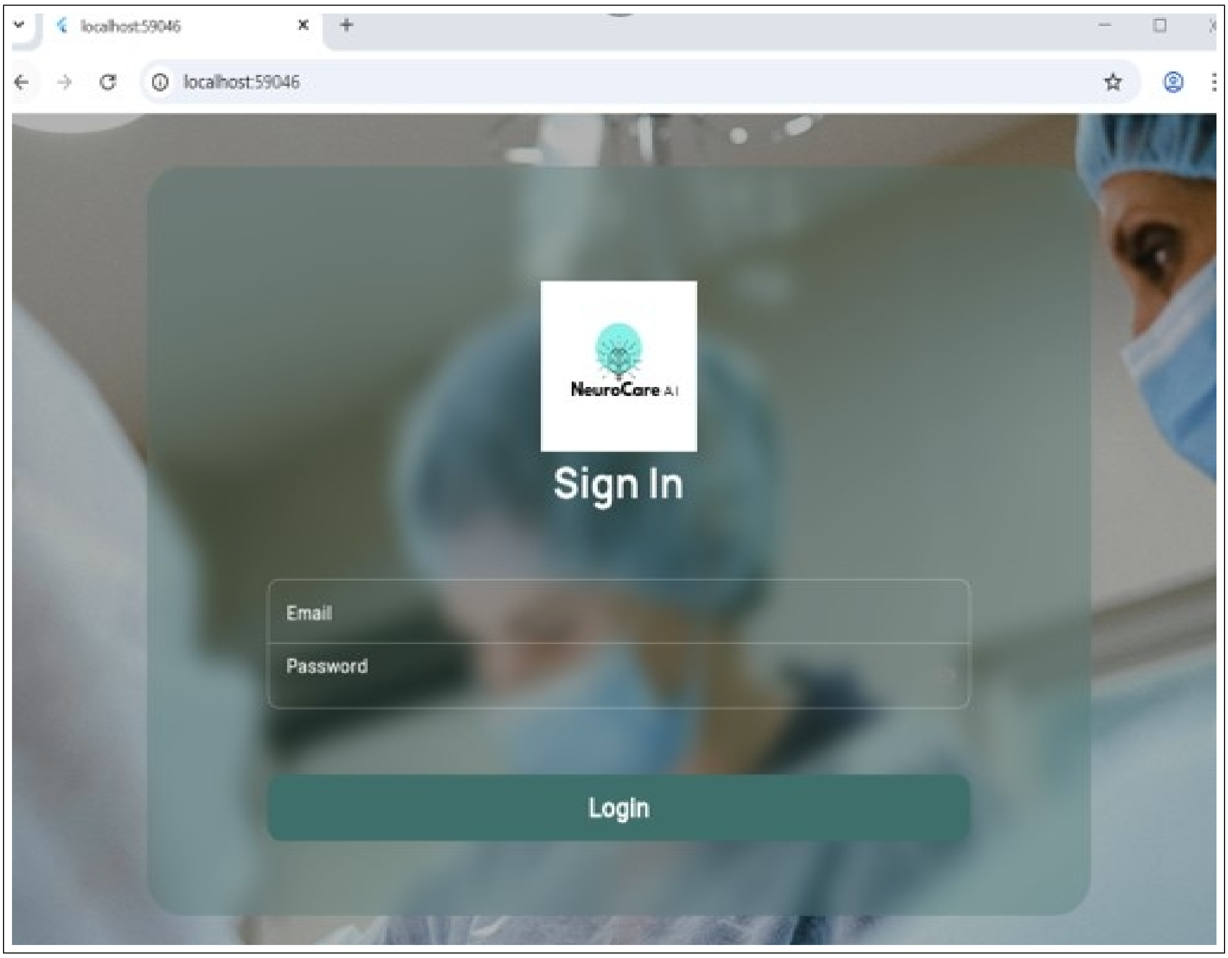

Figure 4.30: Admin Login Page

The Admin Login Page provides a secure authentication interface for administrators to access and manage the system.

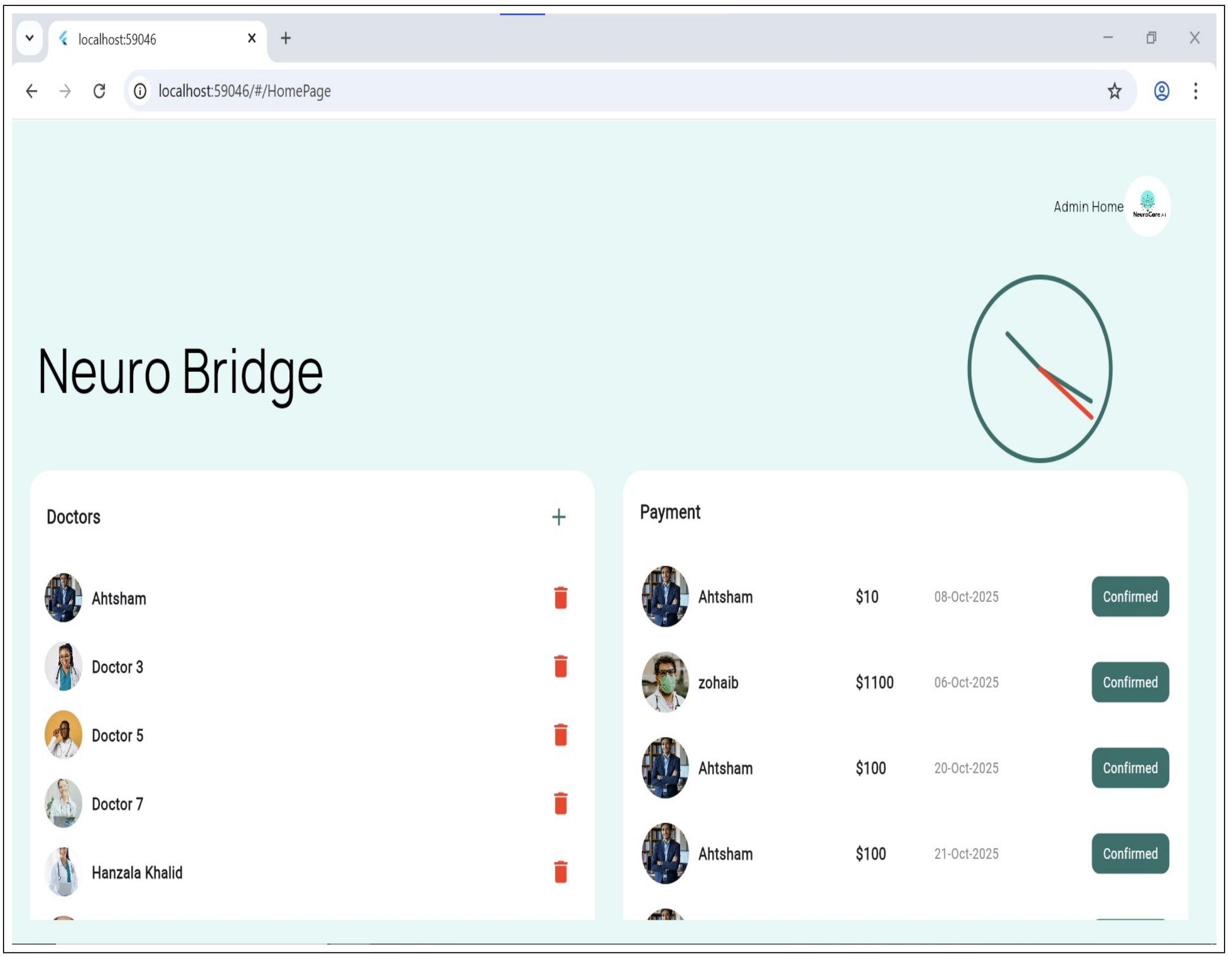

Figure 4.31: Admin Web Page

The Admin Panel provides a centralized dashboard for managing system operations. It allows the administrator to view, add, or delete doctor profiles from the doctor list and manage doctors payment details, enabling transparent monitoring of transactions and ensuring accurate record maintenance.

4.7 Database Design

The Brain Tumor Segmentation and Detection System uses a safe and flexible database design to allow its features to work in real time on patient, doctor and administrative sections. Because it offers real-time synchronization, is friendly with Flutter and is easy to deploy, Firebase, a NoSQL solution in the cloud, has been picked as the database for this system. Managing user profiles, MRI uploads, AI-generated reports, appointment records and payment information works well in this way because it can handle structured and semi-structured data. Because Firebase lets programs talk to each other, information on mobile apps and in the web panel stays consistent and fast.

4.8 Summary

The chapter discusses the architecture and design of the Brain Tumor Segmentation and Detection System with priority on accurate, scalable and real-time benefits in medicine. It brings together technologies for both tumor segmentation and classification and the user interface works well on mobile and web devices. Firebase stores all the data for the application in the cloud and the Flask API helps the AI module and web apps speak together. Detailed attention is paid to factors like performance, keeping data private and compliance with medical laws and regulations. With Agile development, the project focuses on improving the project by parts, getting input early and involving medical experts from the very beginning. The design is built around both main concepts (data arrangements, how users interact, incorporating models) and smaller sections (authentication, picture upload, using AI), so the solution can be relied on in medicine, fits in any healthcare setting and can be developed further if needed.

Chapter 5

System Implementation

Brain tumor detection system will have enhanced MRI segmentation, tumor detection, classification. More efficient models such as CNNs, U-Net, and VGG16 are trained to achieve improved performance. There will be various options tested since the best dataset and the algorithm cannot be predetermined. The adaptive algorithm choice will provide consistent accuracy and reliability enhancement.

5.1 System Architecture

The Brain Tumor Detection System architecture is a well-organized framework where each of the components is distinct but interrelated and works together.The system consists of the following components:

5.1.1 Mobile Application

1. **Flutter, Firebase, Flask API:** The application created with the Flutter development provides a cohesive interface for phones. The application allows patients and medical professionals to upload MRI or CT scans [17] and view the results as real-time brain scan analysis, tumor detection and classification with the help of CNNs and VGG16 and tumor localization to visualize affected areas. Firebase is used to handle user authentication and store MRI data safely, whereas Flask API allows connecting the app and machine learning inference models to give real-time predictions.

2. **Database and Cloud Integration:**Firebase also secures authentication, real-time storage and access to MRI or CT scans, segmented tumor data, classification output using CNNs, U-Net and local tumor maps,thereby ensuring secure cloud integration.

5.2 Dataset Acquisition

The dataset used in this project consist of MRI and CT scan images, selected to improve both the diversity and accuracy of the system's deep learning models. The dataset allows the models to learn from different imaging modalities, ensuring robust tumor segmentation and classification.

5.2.1 Dataset: Brain Tumor Multimodal Image Dataset (CT & MRI)

Dataset Link: https://www.kaggle.com/datasets/murtozalikhon/brain-tumor-multimodal-image-ct-and-mri
This data set includes both CT and MRI scans. This dataset was compiled from multiple reliable open-access sources and refined for improved image consistency and labeling accuracy.

- **Dataset Description:** The Brain Tumor Multimodal Image Dataset is organized into two main directories: Brain Tumor CT scan Images and Brain Tumor MRI images. The CT scan directory contains two subfolders: Healthy with 2300 images and Tumor with 2318 images. The MRI directory also contains two subfolders: Healthy with 2000 images and Tumor with 3000 images).
- **Purpose:** Designed for automatic brain tumor detection, classification, and segmentation tasks using deep learning techniques.
- **Advantages:** The combination of CT and MRI modalities provides a more detailed understanding of structural and functional variations in the brain, improving model performance and diagnostic accuracy.

5.3 AI Algorithms and AI Model

The system is combined with developed deep learning models to allow automated and proper brain tumor analysis of MRI ans CT scans images. It uses VGG16 to extract features and classify tumors in addition to U-Net which is applied on automatizing the image segmentation of tumors. These models collaborate together to conduct tumor detection, classification and localization. A dynamic algorithm selection mechanism is adapted within the selection process to select the most appropriate model in accordance with scan quality and characteristics, with high accuracy and minimal human intervention [18].

5.3.1 Deep Learning Algorithms Used

VGG16 A fully convolutional neural network (CNN) based on feature extraction and classification of brain tumors on Magnetic Resonance Imaging (MRI) images. It assists in identifying and sorting the types of tumors (benign or malignant) according to what has been learned visually.

U-Net A u-net architecture specifically designed for biomedical image segmentation. Follows an encoder-decoder structure where the contracting path captures context and the expansive path enables precise localization.

5.4 Model Training and Evaluation

This section details the implementation of the training, evaluation, and visualization processes used for the deep learning models. The workflow integrates various stages of the analysis, including data preprocessing, model fitting and graphical model evaluation.

5.4.1 Training Script Overview

Each primary model, specifically the VGG16 classifier and the U-Net segmenter, was individually trained. TensorFlow, Keras and NumPy libraries were used for notebook.

Each of the model's predictions was tested using standard metrics for classification (Accuracy, Sensitivity, Specificity, F1-Score) and segmentation (Dice Coefficient, IoU).

5.4.2 Training and Validation Curves

The MRI and CT scans for classification models were trained on 256×256 input images using the VGG16 architecture, while the segmentation models were trained using the U-Net architecture with the same input size.

The stable convergence for both MRI and CT U-Net models was typically achieved after around **100 epochs** without significant signs of overfitting. Early stopping was implemented to halt training when the validation loss failed to improve over several epochs, ensuring optimal training duration and avoiding unnecessary computation.

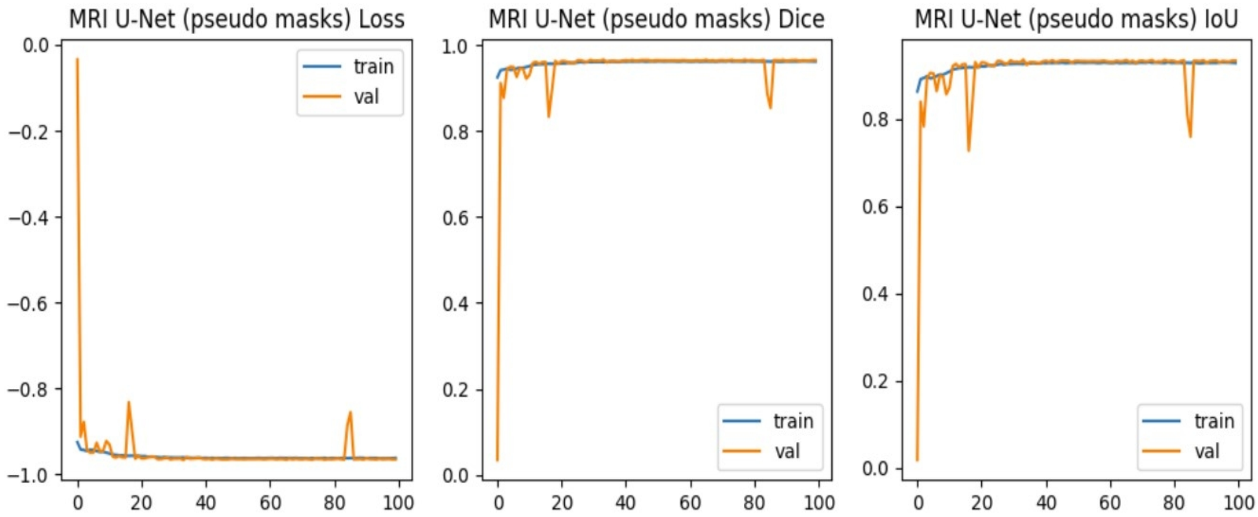

Figure 5.1: Training and Validation Curves of MRI Segmentation Model

This above graph shows the U-Net model's learning progress on MRI scans. The training curve indicates how well the model learned from the MRI data, while the validation curve shows how effectively it generalizes to new, unseen scans. The close alignment between both curves demonstrates the model trained successfully without overfitting.

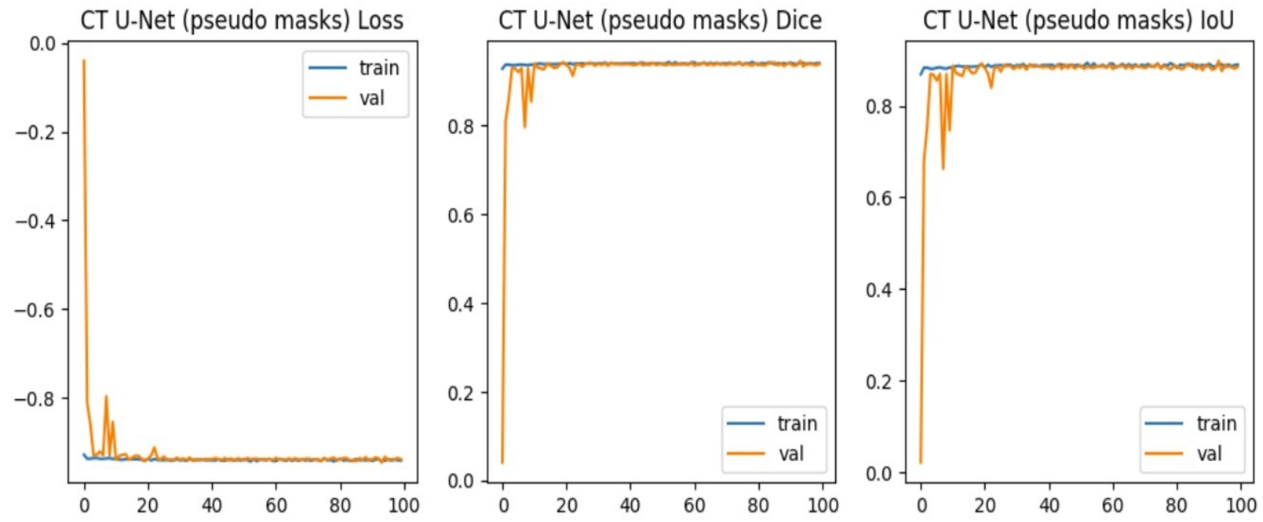

Figure 5.2: Training and Validation Curves of CT Segmentation Model

This above graph shows the U-Net model's learning progress on CT scans.The training line exhibits how precisely the model was able to detect tumors, while the validation line indicates that the model is able to analyze new, unseen CT scans with high reliability.

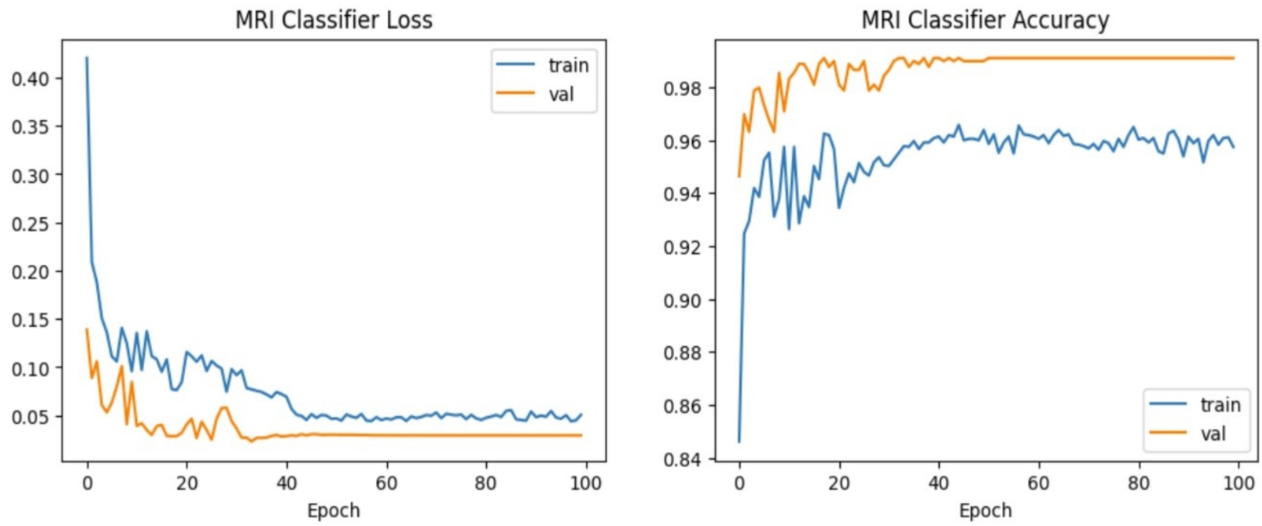

Figure 5.3: Training and Validation Curves of MRI classification Model

The above-mentioned figure illustrates the effectiveness of the MRI classification model Vgg16. The training curve signifies the extent to which the model was able to distinguish brain tumors from the given MRI data, whereas the validation curve indicates the accuracy of the model for classifying new, unseen MRI scans.

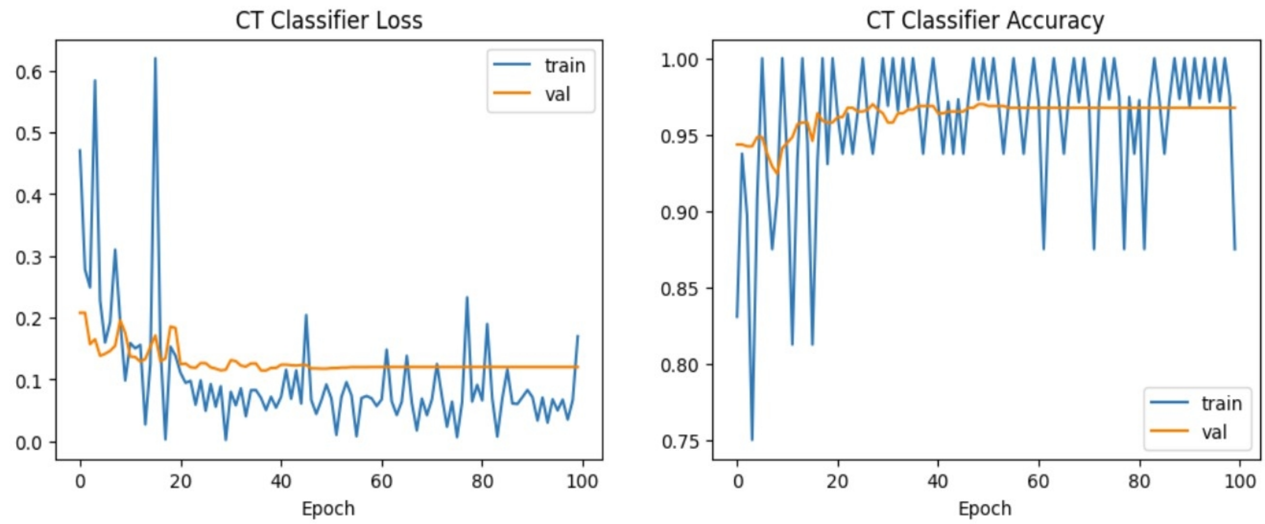

Figure 5.4: Training and Validation Curves of CT classification Model

The above graph illustrates the learning journey of the VGG16 model which was trained on the CT scan data to classify brain tumors. The training curve keeps track of how well the model learnt to discriminate between and classify the tumors, and the validation curve measures its accuracy on new, unseen CT scans.

5.4.3 Model Comparison

The validation dataset analysis results have made it very clear that both the classification and segmentation models are effective. The results have revealed MRI and CT model predictions' consistency, hence their reliability and overall performance have been established.

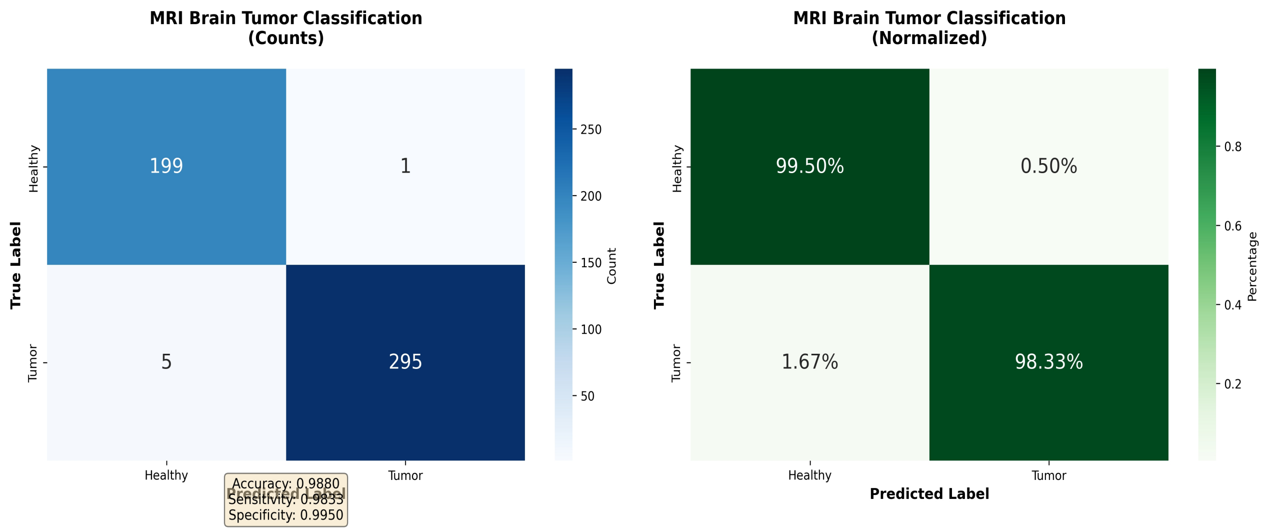

Figure 5.5: Confusion matrix result of MRI classification Model

This above confusion matrix evaluates the VGG16 model's performance in classifying brain tumors from MRI scans. It shows correct predictions: true positives/negatives alongside errors: false negatives/false positives.

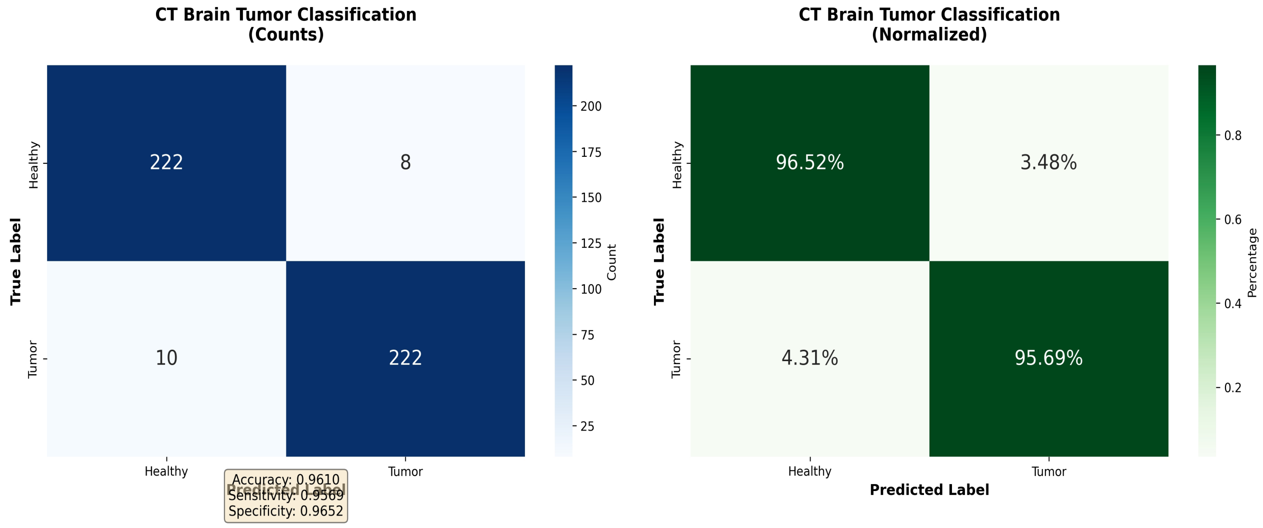

Figure 5.6: Confusion matrix result of CT classification Model

This above confusion matrix details the VGG16 model's CT scan classification accuracy. It breaks down performance into four categories: correctly identified tumors and healthy tissue, missed tumors and false alarms.

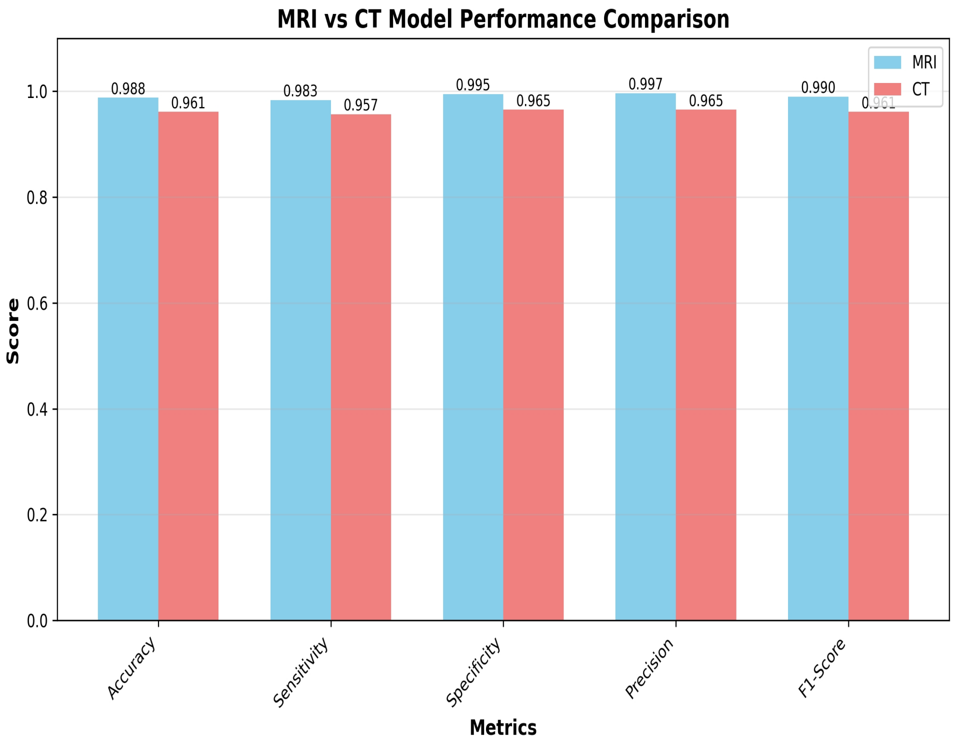

Figure 5.7: MRI vs CT model performance comparison

The bar chart illustrates the performance comparison between MRI and CT models across various classification metrics.The MRI model consistently demonstrates higher performance than the CT model in all evaluated metrics.

Table 5.1: Performance Metrics of Classification Models on Validation Data

Model	Val Accuracy	Val Loss	Precision	F1-Score
VGG16 MRI Classifier	98.80	0.0836	0.98	0.99
VGG16 CT Classifier	96.75	0.1223	0.95	0.96

Table 5.2: Performance Metrics of Segmentation Models on Validation Data

Model	Val Loss	Dice Score	IoU	Binary Accuracy
U-Net MRI Segmenter	-0.9651	0.9652	0.9328	0.9564
U-Net CT Segmenter	-0.9389	0.9389	0.8859	0.9280

5.5 Component Functionality

The Brain Tumor Detection System has all the components to complete a particular task:

- **Mobile App:**
  1. Offers a single place where patients and medical practitioners can post Magnetic Resonance Imaging (MRI) images.
  2. Shows the outcome of real-time tumor analysis of the brain by automated segmentation of the images using U-Net and classification of the results using VGG16.
  3. Raises the awareness of the users of the existence of tumors.The system relies on Firebase to store historical medical records, provide real-time synchronization, and ensure secure authentication.
  4. Application of Firebase to perform safe user authentication and archive the MRI data, and get a report about the diagnoses.
  5. It is used to make rapid correct inferences by using Flask API in order to integrate the application with AI models.
- **Backend and AI Processing Engine:**
  1. Brings the task of the basic processing unit of the mobile application and web portal with the AI models.
  2. Train deep learning network (VGG16 and U-Net) to find, distinguish, partition, phase and localize tumors.
  3. An autonomous also takes adaptive selection of algorithms to select the best model based on quality of scan.
  4. Trains AI models on a periodic basis causing them to become more accurate when used on new training data.

5.6 Communication Between Components:

The mobile app securely transfers Magnetic Resonance Imaging (MRI) data to an AI engine, which segments, detects, classifies and localizes tumors, then sends results back in real time for accurate diagnosis.

5.7 Technology Stack

The development of the brain tumor detection system leverages the following core technologies:

- **Mobile App:** Flutter
- **Backend Development:** Flask
- **Database and Cloud Services:** Firebase

Additional Technologies

- **Machine Learning and AI Models**: Python with VGG16 and U-Net
- **Version Control:** Git and GitHub
- **Integrated Development Environment (IDE):** Visual Studio Code (VSCode) and Android Studio
- **Hardware**:A device with a minimum of 8 GB RAM and active internet connectivity.
- **Cloud Tunnel Service**: Ngrok

5.8 Security Measures

The Brain Tumor Detection System gives much emphasis on the security of user data when working with sensitive Magnetic Resonance Imaging (MRI) data. The measures employed are as follows:

- **Secure Zones and Firewalls:** Earmarked zones and firewalls are used to protect the scan data of the patients. Security checks are carried out on a regular basis to enhance defenses as well as deal with emerging threats.
- **Data Encryption:** The encrypted information includes all MRI scans and other diagnostic outputs so that they cannot be read without decryption keys, and hence ensures confidentiality.
- **Authentication and RBAC:** Role-Based Access Control (RBAC) will guarantee that only authorized users (ex: radiologists, administrators) will be allowed to access or edit sensitive data. Unauthorized access is blocked by the use of multi-level authentication and login credentials.
- **Auditing and Logging:** Every user activity is recorded and examined to identify and examine suspicious or inappropriate system activity.

- **Firebase Security:** This system takes advantage of Firebase security capabilities, such as intensive authentication, authorization restrictions, limited access on the Internet and audit trails to safeguard the database.

5.9 Summary

This chapter discussed the key components of the Brain Tumor Detection System, including its architecture, AI models, technology stack and security framework to explain the working principles of this system. To improve the speed of the diagnostic process and guarantee the accuracy and reliability of the outcomes, the system combines deep learning-based methods so that it could fulfill five key goals: automatic tumor regions segmentation on Magnetic Resonance Imaging (MRI) images and CT scans, tumor detection and classification with the use of VGG16, U-Net, and Convolutional Neural Networks (CNNs) and the ability to locate tumors on the images obtained by the MRI with high precision, with the help of such adaptive AI models together with a secure back-end, cloud-based storage and mobile/web platform would guarantee an effective medical image processing, convenient functionality among medical practitioners and enhanced levels of transparency and trust in diagnostic results.

Chapter 6

System Testing and Evaluation

6.1 Importance of System Testing

System testing plays an important role in the construction of the Brain Tumor Detection System in order to identify potential issues. The quality of the systems and their ability to provide the correct diagnosis results to the users is vital. It identifies a root cause of mistakes during the initial phases to avoid mistakes, and makes sure that the product is in line with user and clinical needs.

6.2 Graphical User Interface (GUI) Testing

The testing in GUI aims at analyzing the user interface elements of the Brain Tumor Detection System in order to make the interaction with users streamline. It verifies the screens containing such elements as buttons, menus, icons, and forms. Under this system, the GUI testing will ensure that the patient app, doctor app and web portal are showing all the pages, MRI upload screens, and analysis result dashboards properly. The work is aimed at having a simple user-friendly interface facilitate navigation when performing tumor segmentation, detection and classification.

6.3 Usability Testing

Usability testing will be done to ascertain the ease with which the end users will be able to work with the Brain Tumor Detection System. Such testing guarantees that the patients and doctors will not have problems navigating through the system. The system has an easy to use and simple interface with which the user can upload Magnetic Resonance Imaging (MRI) scans, view detection results. Its design is user-friendly making the system more attractive and more accessible to the users.

6.4 Software Performance Testing

Software performance testing is carried out to identify the ability to handle massive loads of Magnetic Resonance Imaging (MRI) images and intensive calculations by the Brain Tumor Detection System. This testing will ensure that the system can run deep learning architectures like VGG16, U-Net and CNN without delays and crashes. It verifies timely and accurate execution of the segmentation, detection and classification.It is aimed at maintaining a high speed, stability, and responsiveness even in case of a heavy workload [19].

6.5 Compatibility Testing

The key role of compatibility testing involves the smooth running of the Brain Tumor Detection System in various hardware configurations, operating systems and software environments. It confirms that the mobile and web applications are effective on different devices among doctors and patients. The system is tested across multiple browsers and platforms to ensure that it can perform its task without performance problems across the board so that users can post MRI and CT scans and get reliable results with no technical problems in the process.

6.6 Exception Handling

Exception handling testing tests the way the Brain Tumor Detection System reacts to the unlikely or invalid input. In case of the uploading of corrupted Magnetic Resonance Imaging (MRI) scans, unsupported formats and empty obligatory fields, the system shows clear error messages and directs the user to rectify the input. This guarantees a reliable functionality and a steady user interaction in the process of tumor segmentation, detection and classification.

6.7 Load Testing

The load testing is an evaluation of the Brain Tumor Detection System under peak usage conditions such as when multiple patients upload MRI scans and doctors simultaneously view results in real time. The system undergoes testing that allows it to smoothly segment, detect and localize tumors devoid of delays. Adaptive algorithm choice (CNN, VGG16, U-Net) is confirmed to allow upholding accuracy and efficiency even with high server load with a seamless and reliable user experience.

6.8 Test Cases

This section gives the explanation in tabular form for the test cases we performed. In these explanatory tables we provided brief description of the test cases.

6.8.1 Patient Login Test Case

Table 6.1: Patient Login Test Case

Test Case ID	plogin_TC_01
Description	Verify that patient can log in successfully
Applicable for	Patient Mobile App
Preconditions	Patient account exists
Test Steps	1. Open app. 2. Enter valid credentials. 3. Click on the login button.
Expected Result	Patient should be logged in successfully
Actual Result	Patient is logged in successfully
Status	Pass

This test case validates the main authentication functionality for the Patient Mobile Application. It ensures that valid credentials provided by registered patients allow them to successfully access their accounts via the login process, securely and reliably entering into the medical services of the app. This test follows a simple sequence: start up the application, insert appropriate login credentials, perform the login command, and the system correctly grants access as intended.

6.8.2 Patient Sign up Test Case:

Table 6.2: Patient Sign up Test Case

Test Case ID	PSignup_TC_01
Description	Verify that a new patient can sign-up
Applicable for	Patient Mobile App
Preconditions	Patient does not have an account
Test Steps	1. Open app. 2. Enter personal details and email. 3. Click on the signup button.
Expected Result	Patient should be created.
Actual Result	Account created successfully
Status	Pass

This test case validates the patient registration functionality, ensuring new users can create an account in the mobile application. This will confirm that the system processes personal information and email correctly to make new patient profiles and gives access to the healthcare services on the platform.

6.8.3 Patient Payment Test Case:

Table 6.3: Patient Payment Test Case

Test Case ID	PPay_TC_01
Description	Verify that patient can make payment through Stripe.
Applicable for	Patient Mobile App
Preconditions	Patient is logged in
Test Steps	1. Open doctor booking. 2. Select doctor and time slot. 3. Click on "Pay Now" and complete Stripe payment.
Expected Result	Payment should be processed successfully.
Actual Result	Payment processed successfully.
Status	Pass

This test case validates the Stripe integrated payment processing system within the doctor booking work-flow.The test case ensures that authenticated patients can complete a financial transaction of appointment scheduling successfully, ensuring a seamless payment experience from selection to confirmation.

6.8.4 Doctor Login Test Case:

Table 6.4: Doctor Login Test Case

Test Case ID	DLogin_TC_01
Description	Verify that doctor can log in successfully
Applicable for	Doctor Mobile Application
Preconditions	Doctor account exists
Test Steps	1. Open app. 2. Enter valid credentials. 3. Click on login button.
Expected Result	Doctor should be logged in successfully
Actual Result	Doctor is logged in successfully
Status	Pass

This test case covers the authentication of medical professionals through the Doctor Mobile Application.It checks the validity of a registered doctor with credentials to securely access their portal for appointment management and consultation with patients.

6.8.5 Doctor Sign up Test Case:

Table 6.5: Doctor Sign up Test Case

Test Case ID	DSign up_TC_01
Description	Verify that new doctor can sign up.
Applicable for	Doctor Mobile Application
Preconditions	Doctor does not have an account.
Test Steps	1. Open app. 2. Enter doctor details, qualification and email. 3. Click on the signup button.
Expected Result	Doctor account should be created
Actual Result	Account created successfully
Status	Pass

This test case verifies the registration of doctors, confirming that new medical professionals can create an account by providing their details, qualifications, and email. It ensures that the system properly processes and stores doctor credentials for platform access.

6.8.6 Doctor Payment Receiving Test Case:

Table 6.6: Doctor Payment Receiving Test Case

Test Case ID	DPayR_TC_01
Description	Verify that the doctor can receive and view payment details successfully.
Applicable for	Doctor Mobile Application
Preconditions	Doctor is logged into the mobile application and has completed a patient consultation.
Test Steps	1. Navigate to the "Payments" section in the doctor dashboard. 2. View the list of recent transactions. 3. Select a specific payment record. 4. Verify payment details such as amount, date, and patient name.
Expected Result	The system should display accurate and complete payment details for the selected transaction.
Actual Result	The system successfully retrieves and displays all payment details for the selected transaction.
Status	Pass

The test case verifies the doctor’s ability to view his payment records correctly after consultations.It confirms the payment tracking system correctly displays transaction details, including amount, date, and patient information in the doctor dashboard.

6.8.7 Admin login Test Case

Table 6.7: Admin Login Test Case

Test Case ID	Alogin_TC_01
Description	Verify that admin can log in successfully.
Applicable for	Web admin Panel
Preconditions	Admin account exist. User is logged in
Test Steps	1. Open admin panel. 2. Enter Valid credentials. 3. Click on log in button. 4. Click on the Submit button.
Expected Result	Admin should be logged successfully.
Actual Result	Admin is logged in successfully.
Status	Pass

This test case validates the web panel administrator authentication system: the authorized admin user should be able to access the administrative dashboard and management features with valid credentials.

6.8.8 Admin add Doctor Test Case:

Table 6.8: Admin add Doctor Test Case

Test Case ID	AaddDoc_TC_01
Description	Verify that admin can add new doctor profile.
Appliable for	Web admin Panel.
Preconditions	Admin is logged in.
Test Steps	1. Open doctors managemet. 2. Click on add doctor. 3. Fill doctor details and submit.
Expected Result	Doctor should be added succcessfully.
Actual Result	Doctor added successfully.
Status	Pass

This test case confirms that administrators can add new doctor profiles to the system via the web panel. It verifies the complete workflow from accessing doctor management to submitting and storing new doctor information.

6.9 Summary

These test cases discuss all the aspects of Brain Tumor Detection System. They provide that automatic segmentation, tumor detection and classification on MRI scans are all functional and reliable. To test the system to have different scenarios, we provided normal and edge cases, and adaptive algorithm to apply CNNs, VGG16, or U-Net models, depending on the quality of the MRI scan. The objective is to ensure the proper functionality and high accuracy because the specific dataset and algorithm will be determined once a number of alternatives have been tested. These tests will be necessary to ensure that the Brain Tumor Detection System functions as expected and offers effective, efficient, and convenient diagnostic assistance.

Chapter 7

Conclusion

7.1 Summary of Work

The brain tumor detection system has successfully resulted in the development of a sophisticated and powerful application that helps one in early diagnosis and planning treatment. Using automatic image segmentation, CNNs, VGG16 and U-Net to identify and classify tumors, the system has been successful in achieving its fundamental goals of detecting and classifying tumors with accuracy and reliability [20].

The architecture is created to ensure scalability, mobile and web application integration, and multi-language to both patients and doctors. The system enables patients to upload MRI or CT scans through its user-friendly interface while allowing them to view their results and access specialist doctors. The system delivers exact AI-based medical insights to doctors who can use them for their clinical choices. The system achieves outstanding results through its tumor segmentation accuracy (Dice Score: 0.9652 for MRI) and classification precision (Accuracy: 98.80% for MRI) which operate within a simple interface that supports real-time processing and multiple languages

7.2 Project Outcomes and Significance

This direct system development comes to grips with the major issues that modern neuro-oncology faces, which are subjectivity, time-consuming, and expensive manual radiological analysis. The outcomes are mainly and their significance are:

- **Improved Diagnostic Ability:** This system analyzes the medical images in a quantitative and consistent manner in a very short time thus relying less on the subjective assessments and possibly cutting down the diagnostic delay.
- **Incorporation of Clinical Workflow:** The use of these AI models in different mobile and web applications is a practical guide to the AI tools' integration in the existing clinical workflows, a process that gives power to both the patients and the providers.
- **Scalable and Accessible Design:** The architectural choices, including the use of cloud processing and multi-tier application design, have made the system capable of handling a larger number of

users and at the same time, being available on different types of devices around the world in different languages.

- **Validation of Hybrid AI Approach:** The piece of work confirms that it is really efficient to combine different neural network topologies, that is, VGG16 for feature-based classification and U-net pipeline for pixel-wise segmentation in solving a really complicated medical imaging problem to a very high degree of precision.

In summary, it has been the major contribution of the present research to provide a more than substantial and very functional tool for the AI-aided medical diagnosis; its usage is already showing the advantages of increasing the precision of diagnosis, hastening the treatment plan preparation, and consequently, bettering the overall outcome of patients undergoing brain tumor treatment.

7.3 Limitations and Challenges

Despite the positive outcomes, the project went through a number of limitations, which not only show its current status but also provide hints for future development:

- **Data Dependency:** The performance and the generalization of deep learning models are unavoidably dependent on the quality, quantity, and diversity of the training data. The establishment of more extensive and clinically varied datasets would be a prerequisite for its global and widespread adoption.
- **Computational Requirements:** The high computational resources required for training and fine-tuning such complex models as U-Net and VGG16 may limit some research or clinical settings.
- **Clinical Validation Scope:** The system has been validated through extensive testing on several validation datasets; however, a large-scale trial in the hospital environment is the next step for the testing of efficacy, reliability, and impact on clinical decisions to be comprehensive.

7.4 Concluding Remarks

The development process of this project from initial idea to complete system implementation shows how artificial intelligence technology solves major healthcare problems. The project shows how different fields of computer science and software engineering and medical diagnostics work together to achieve success. The Brain Tumor Detection and Analysis System represents a technological advancement that goes beyond its technical capabilities. The system demonstrates a forward-thinking approach to healthcare integration of AI-based tools into medical practice. The system enables doctors to perform earlier interventions and deliver more accurate treatments which could lead to better life quality for patients throughout the world. The established base of this research provides a solid starting point for future development which includes system optimization and real-world application expansion.

7.5 Future Work

The Brain Tumor Detection System will be further developed in the future, aiming at better automatic segmentation and increased accuracy of tumor classification. The one thing that will be central to this future research is the recognition of the power for the prognosis and treatment planning that comes along with the ability to detect and classify the stages of tumors accurately. A continual testing of the best dataset and algorithms (CNNs, U-Net, VGG16) will be done, and the selection of algorithms because of their adaptability will be a major factor in obtaining trustworthy results. The system will be more advanced than before since it will have the capability to process in real-time, use sophisticated models, and also have user-friendly interfaces, all of which will contribute to more accurate diagnoses and thereby better patient outcomes.

On top of that, the future development will also extend support to mobile and web applications that will allow doctors and patients to work together in real-time. The system, with features like AI-generated reports, multilingual support, and safe cloud storage, will be more usable in healthcare settings and will, thus, promote quick, evidence-based decision-making.

References

[1] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, and K. Van Leemput. *The multimodal brain tumor image segmentation benchmark (BRATS)*. *IEEE Transactions on Medical Imaging*, 34(10):1993–2024, 2015. Cited on pp. 1 and 7.

[2] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, and J. A. van der Laak. *A survey on deep learning in medical image analysis*. *Medical Image Analysis*, 42:60–88, 2017. Cited on p. 1.

[3] K. Kamnitsas, C. Ledig, V. F. Newcombe, J. P. Simpson, A. D. Kane, D. K. Menon, and B. Glocker. *Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation*. *Medical Image Analysis*, 36:61–78, 2017. Cited on p. 5.

[4] K. Kamnitsas, E. Ferrante, S. Parisot, C. Ledig, A. Nori, A. Criminisi, D. Rueckert, and B. Glocker, "DeepMedic for Brain Tumor Segmentation," *Biomedical Image Analysis Group, Imperial College London, and Microsoft Research, Cambridge, UK*. Cited on pp. 6 and 9.

[5] S. Bauer, R. Wiest, L. P. Nolte, and M. Reyes. *A survey of MRI-based medical image analysis for brain tumor studies*. *Physics in Medicine and Biology*, 58(13):R97–R129, 2013. Cited on pp. 6 and 10.

[6] O. Ronneberger, P. Fischer, and T. Brox. *U-Net: Convolutional networks for biomedical image segmentation*. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 234–241, 2015. Cited on p. 6.

[7] F. Sénée, P. Kickingereder, W. Wick, M. Bendszus, and K. H. Maier-Hein. *Brain tumor segmentation and radiomics survival prediction: Contribution to the BRATS 2017 challenge*. *International Journal of Computer Assisted Radiology and Surgery*, 14(4):551–561, 2019. Cited on pp. 7, 10, and 12.

[8] Cardoso, M. J., Li, W., Brown, R., Ma, N., Kerfoot, E., Wang, Y., Murrey, B., Myronenko, A., Zhao, C., Yang, D., Nath, V., He, Y., Xu, Z., Hatamizadeh, A., Zhu, W., Liu, Y., Zheng, M., Tang, Y., Yang, I., Zephyr, M., Hashemian, B., Alle, S., Zalbagi Darestani, M., Budd, C., Modat, M., Vercauteren, T., Wang, G., Li, Y., Hu, Y., Fu, Y., Gorman, B., Johnson, H., Genereaux, B., Erdal, B. S., Gupta, V., Diaz-Pinto, A., Dourson, A., Maier-Hein, L., Jaeger, P. F., Baumgartner, M., Kalpathy-Cramer, J., Flores, M., Kirby, J., Cooper, L. A. D., Roth, H. R., Xu, D., Bericat, D., Floca, R., Zhou, S. K., Shuaib, H., Farahani, K., Maier-Hein, K. H., Aylward, S., Dogra, P., Ourselin, S., and Feng, A. *MONAI: An open-source framework for deep learning in healthcare*. *arXiv preprint arXiv:2211.02701*, 2022. Available at: https://doi.org/10.48550/arXiv.2211.02701 Cited on pp. 7 and 10.

[9] Gibson, E., Li, W., Sudre, C., Fidon, L., Shakir, D. I., Wang, G., Eaton-Rosen, Z., Gray, R., Doel, T., Hu, Y., Whyntie, T., Nachev, P., Modat, M., Barratt, D. C., Ourselin, S., Cardoso, M. J., and Vercauteren, T. *NiftyNet: A deep-learning platform for medical imaging*. *arXiv preprint arXiv:1709.03485*, 2017. Available at: https://doi.org/10.48550/arXiv.1709.03485 and https://doi.org/10.1016/j.cmpb.2018.01.025 Cited on pp. 7 and 10.

[10] Zavaleta-Monestel, E., Quesada-Villaseñor, R., Arguedas-Chacón, S., García-Montero, J., Barrantes-López, M., Salas-Segura, J., Anchía-Alfaro, A., Nieto-Bernal, D., and Diaz-Juan, D. E. *Revolutionizing Healthcare: Qure.AI's Innovations in Medical Diagnosis and Treatment. Cureus*, 16(6):e61585, 2024. DOI: https://doi.org/10.7759/cureus.61585. Cited on pp. 8 and 10.

[11] Yusufzai, N. *AI in Medical Imaging: How Aidoc Detects Pulmonary Embolisms and a Beginner's Guide to Python-Based Imaging Algorithms*. Published in 2024. Available at: https://www.aidoc.com/ Cited on pp. 8 and 11.

[12] Faiyazuddin, M., Rahman, S. J. Q., Anand, G., Siddiqui, R. K., Mehta, R., Khatib, M. N., Gaidhane, S., Zahiruddin, Q. S., Hussain, A., and Sah, R. *The Impact of Artificial Intelligence on Healthcare: A Comprehensive Review of Advancements in Diagnostics, Treatment, and Operational Efficiency. Health Sci Rep*, 8(1):e70312, 2025. DOI: https://doi.org/10.1002/hsr2.70312. Cited on pp. 9 and 11.

[13] Microsoft Research. *Project InnerEye Open-Source Software for Medical Imaging AI - Microsoft Research*. [online] Available at: https://www.microsoft.com/en-us/research/project/project-innereye-open-source-software-for-medical-imaging-ai/ [Accessed 9 Nov. 2025]. Cited on pp. 9 and 11.

[14] Siemens Healthineers. *AI-Rad Companion*. [online] Available at: https://www.siemens-healthineers.com/en-pk/digital-health-solutions/ai-rad-companion [Accessed 9 Nov. 2025]. Cited on pp. 8 and 11.

[15] M. Abadi, A. Agarwal, P. Barham, et al. *TensorFlow: A system for large-scale machine learning*. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 265–283, 2016. Cited on p. 17.

[16] M. Buda, A. Saha, and M. A. Mazurowski. *Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. Computers in Biology and Medicine*, 109:218–225, 2019. Cited on p. 26.

[17] Ronneberger, O., Fischer, P., and Brox, T. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015. Cited on p. 55.

[18] K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition*. In *International Conference on Learning Representations (ICLR)*, 2015. Cited on p. 56.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, and N. Houlsby. *An image is worth 16x16 words: Transformers for image recognition at scale*. In *International Conference on Learning Representations (ICLR)*, 2021. Cited on p. 65.

[20] K. He, X. Zhang, S. Ren, and J. Sun. *Deep residual learning for image recognition*. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. Cited on p. 71.

[21] Shah, N.Z. et al. (2023) "Factors affecting well-being in brain tumor patients: An LMIC perspective", *Frontiers in Psychology*. Available at: https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2023.1117967/full No Citations.

[22] N. Tajbakhsh et al. "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning", *IEEE Journals & Magazine*. Available at: https://ieeexplore.ieee.org/document/7426826/ No Citations.

[23] Wang, G., Li, W., Ourselin, S., and Vercauteren, T. (2017). “Automatic Brain Tumor Segmentation using Cascaded Anisotropic Convolutional Neural Networks”. *arXiv:1709.00382v2*. No Citations.

[24] Pawan Kumar Mall et al. (2023) “A Comprehensive Review of Deep Neural Networks for Medical Image Processing: Recent Developments and Future Opportunities”, *Healthcare Analytics*. Available at: https://www.sciencedirect.com/science/article/pii/S2772442523000837#d1e1290 No Citations.

[25] Havaei, M., Davy, A., Warde-Farley, D., et al. (2016). “Brain Tumor Segmentation with Deep Neural Networks”. *arXiv:1505.03540v3*. No Citations.

[26] Dosovitskiy, A. et al. (2021) “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”, *arXiv.org*. Available at: https://arxiv.org/abs/2010.11929 No Citations.

[27] Chougale, P., Yadav, V., and Gaikwad, D. A. *Firebase – Overview and Usage*. *International Research Journal of Modernization in Engineering Technology and Science*, 3(12):1178–1183, 2021. No Citations.

[28] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., and Liang, J. *UNet++: A nested U-Net architecture for medical image segmentation*. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, Vol. 11045, pp. 3–11, 2018. No Citations.

[29] Isensee, F., Petersen, J., Kohl, S. A. A., Jäger, P. F., and Maier-Hein, K. H. *nnU-Net: A self-adapting framework for U-Net-based medical image segmentation*. *Nature Methods*, 18:203–211, 2021. No Citations.