**A Major Project Report On**

**Brain Hemorrhage Detection**

***Submitted to partial fulfillment of the requirements for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**By**

|  |  |
| --- | --- |
| **M. SRI MANISH REDDY** | **(21911A05G7)** |
| **B. NILESH** | **(21911A05D7)** |
| **B. RAKESH KUMAR** | **(21911A05D9)** |
| **B. SRIKANTH** | **(21911A05E2)** |

**Under the Guidance of**

**Mrs. A. LALITHA**

**Assistant. Professor**

****

Department of Computer Science and Engineering

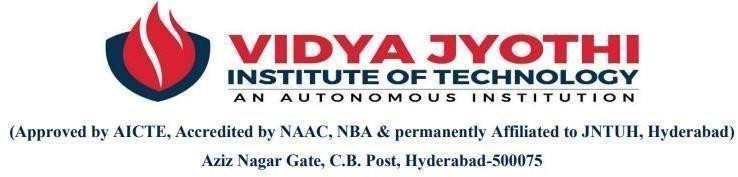
## VIDYA JYOTHI INSTITUTE OF TECHNOLOGY

### (An Autonomous Institution)

**(Approved by AICTE, Accredited by NAAC, NBA & permanently Affiliated to JNTUH, Hyderabad)**

**Aziz Nagar Gate, C.B. Post, Hyderabad-500075**

**2024-2025**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

## CERTIFICATE

This is to certify that the project report titled “ **Brain Hemorrhage Detection**” is being submitted by **M. SRI MANISH (21911A05G7), B. NILESH (21911A05D7), B. RAKESH (21911A05D9), B. SRIKANTH (21911A05E2)** in partial fulfillment for the award of the Degree of Bachelor of Technology in Computer Science and Engineering, is a record of Bonafide work carried out by them under my guidance and supervision. These results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma.

Internal Guide Head of Department

Mrs. A. Lalitha Dr. D. Aruna Kumari

Assistant. Professor Professor

**External Examiner**

## DECLARATION

We, **M. SRI MANISH, B. NILESH, B. RAKESH, B. SRIKANTH** hereby declare that the project entitled, **“Brain Hemorrhage Detection”**  submitted for the degree of Bachelor of Technology in Computer Science and Engineering is original and has been done by us and this work is not copied and submitted anywhere for the award of any degree.

|  |  |  |
| --- | --- | --- |
| **Date:** | **M. SRI MANISH** | **(21911A05G7)** |
| **Place: Hyderabad** | **B. NILESH** | **(21911A05D7)** |
|  | **B. RAKESH** | **(21911A05D9)** |
|  | **B. SRIKANTH** | **(21911A05E2)** |

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to the project guide **Mrs. A. Lalitha**, Assistant Professor, Vidya Jyothi Institute of Technology, Hyderabad for her timely cooperation and valuable suggestions while carrying out this work. It is her kindness that made us learn more from her.

We are grateful to **Dr. D Aruna Kumari**, Professor and HOD, Department of CSE, for her help and support during our academic year.

We would like to take this opportunity to express our gratitude to our Principal

**Dr. A. Srujana** for providing necessary infrastructure to complete this project.

We would thank our parents and all the faculty members who have contributed to our progress through the course to come to this stage.

|  |  |  |
| --- | --- | --- |
|  | **M. SRI MANISH** | **(21911A05G7)** |
|  | **B. NILESH** | **(21911A05D7)** |
|  | **B. RAKESH** | **(21911A05D9)** |
|  | **B. SRIKANTH** | **(21911A05E2)** |

|  |
| --- |
| **CONTENT** |
| **Abstract** |
| **1.INTRODUCTION** |
| 1.1 Objective of the Project |
| 1.2 Problem Statement |
| 1.3 Scope |
| 1.4 Motivation |
| 1.5 Project Introduction |
| **2.** **LITERATURE SURVEY** |
| 2.1 Related Work |
| **3. SYSTEM ANALYSIS** |
| 3.1 Existing System |
| 3.2 Disadvantages |
| 3.3 Proposed System |
| 3.4 Advantages |
| 3.5 work Flow of Proposed system |
| **4. REQUIREMENT ANALYSIS** |
| 4.1Function and non-functional requirements |
| 4.2 Hardware Requirements |
| 4.3 Software Requirements |
| 4.4 Architecture |
| **5. METHODALOGY** |
| 5.1 CNN |
| 5.2 Random Forest |
| 5.3 Support Vector Machine |
| 5.4 Logistic Regression |
| 5.5 K-Nearest Neighbors |
| 5.6 AdaBoost |
| 5.7 CatBoost |
| 5.8 LightGBM |
| 5.9 XGBoost |
| 5.10 Artificial Neural Networks |
| **6. SYSTEM DESIGN** |
| 6.1 Introduction of Input design |
| 6.2 Output Design |
| 6.3 UML Diagram (class, use case, sequence, collaborative, deployment, activity, ER diagram and Component diagram) |
| 6.4 Data Flow Diagram |
| **7. IMPLEMENTATION AND RESULTS** |
| 7.1 Modules |
| 7.2 Algorithms |
| * 1. Output Screens |
| **8. SYSTEM STUDY AND TESTING** |
| 8.1 Feasibility study |
| 8.2 Types of test & Test Cases |
| **9. CONCLUSION** |
| **10. FUTURE ENHANCEMENT** |
| **11. REFERENCES** |

**ABSTRACT:**

Brain hemorrhage, a critical medical condition requiring immediate diagnosis and treatment, poses significant challenges for accurate and timely detection. Leveraging the advancements in deep learning and machine learning, this study explores and evaluates various algorithmic approaches for brain hemorrhage detection, with a particular focus on MobileNet and ResNet architectures. These state-of-the-art convolutional neural networks (CNNs) are known for their efficiency and high performance in image classification tasks. MobileNet, designed for lightweight applications, offers a streamlined architecture suitable for deployment in resource-constrained environments such as mobile devices. ResNet, on the other hand, is recognized for its deep residual learning capabilities, enabling the training of extremely deep networks without the common pitfalls of vanishing gradients. This research involves the application and comparative analysis of these two architectures on a dataset of brain hemorrhage images. By assessing their accuracy, sensitivity, and specificity, we aim to identify the strengths and limitations of each approach. The outcomes of this study are expected to provide valuable insights into the practical applications of MobileNet and ResNet for medical image analysis, ultimately contributing to improved diagnostic tools and patient outcomes in the clinical setting.

**Keywords**: Brain hemorrhage detection, deep learning, machine learning, MobileNet, ResNet, convolutional neural networks, medical image analysis, diagnostic tools, healthcare

1. **INTRODUCTION**

**1.1 Objective of the Project**

The objective of this project is to develop and evaluate advanced deep learning models, specifically MobileNet and ResNet, for the accurate and rapid detection of brain hemorrhages from medical images. The study aims to compare the performance of these architectures in terms of accuracy, sensitivity, specificity, and computational efficiency. By leveraging these state-of-the-art models, the project seeks to create a reliable and automated diagnostic tool that can assist healthcare professionals in making timely and precise diagnoses, thereby improving patient outcomes and reducing the time to treatment for brain hemorrhage patients.

**1.2 Problem Statement:**

The problem addressed in this study is the challenge of accurately and rapidly detecting brain hemorrhages using medical imaging. Current diagnostic practices, relying heavily on manual interpretation by radiologists, are time-consuming and susceptible to errors, leading to potential delays in critical treatment. This research aims to evaluate and compare the effectiveness of advanced deep learning models, specifically MobileNet and ResNet, in automating the detection process. By improving the accuracy and speed of brain hemorrhage diagnosis, the study seeks to enhance clinical decision-making and patient outcomes, ultimately reducing the associated mortality and morbidity rates.

**1.3 Motivation:**

The motivation behind this study is driven by the critical need for accurate and rapid detection of brain hemorrhages, which can significantly impact patient outcomes. Traditional diagnostic methods are time-consuming and prone to human error, leading to delays in treatment. By harnessing the power of advanced deep learning architectures like MobileNet and ResNet, this research aims to enhance diagnostic precision and efficiency. The ultimate goal is to develop robust, automated systems that can assist healthcare professionals in making timely and accurate diagnoses, thereby improving patient care and reducing the mortality and morbidity associated with brain hemorrhages.

**1.4 Scope:**

The scope of this project encompasses the development, training, and evaluation of deep learning models, specifically MobileNet and ResNet, for brain hemorrhage detection using medical imaging data. This includes preprocessing of image datasets, implementation of the neural network architectures, and rigorous performance evaluation through metrics such as accuracy, sensitivity, and specificity. The study also aims to compare the computational efficiency of the models, highlighting their potential for real-world clinical applications. Additionally, the project explores the feasibility of deploying these models in resource-constrained environments, such as mobile devices, to support widespread and accessible diagnostic solutions.

**1.5 Project Introduction:**

Brain hemorrhage, a life-threatening condition, necessitates immediate and accurate diagnosis to facilitate timely intervention and improve patient outcomes. Traditional diagnostic methods primarily rely on the expertise of radiologists interpreting medical images, a process that is both time-consuming and prone to human error. In recent years, advancements in deep learning and machine learning have shown promising potential in automating and enhancing diagnostic accuracy in medical image analysis. This study focuses on leveraging state-of-the-art convolutional neural network (CNN) architectures, specifically MobileNet and ResNet, for the detection of brain hemorrhages. MobileNet is designed for efficient computation, making it suitable for deployment in resource-constrained environments such as mobile devices, while ResNet is renowned for its deep residual learning capabilities, allowing the training of very deep networks without suffering from vanishing gradients. By applying these advanced models to a dataset of brain hemorrhage images, the research aims to evaluate their performance in terms of accuracy, sensitivity, specificity, and computational efficiency. The ultimate goal is to develop a reliable and automated diagnostic tool that can assist healthcare professionals in making timely and precise diagnoses, thereby improving patient care and reducing the mortality and morbidity associated with brain hemorrhages. This project not only contributes to the field of medical image analysis but also explores the practical applications of deep learning models in clinical settings, paving the way for more accessible and efficient diagnostic solutions.

**2. LITERATURE SURVEY**

**2.1 Related Work:**

**[1] S. Ahmed et al., "Exploring Deep Learning and Machine Learning Approaches for Brain Hemorrhage Detection," in IEEE Access, vol. 12, pp. 45060-45093, 2024**

Brain hemorrhage refers to a potentially fatal medical disorder that affects millions of individuals. The percentage of patients who survive can be significantly raised with the prompt identification of brain hemorrhages, due to image-guided radiography, which has emerged as the predominant treatment modality in clinical practice. A Computed Tomography Image has frequently been employed for the purpose of identifying and diagnosing neurological disorders. The manual identification of anomalies in the brain region from the Computed Tomography Image demands the radiologist to devote a greater amount of time and dedication. In the most recent studies, a variety of techniques rooted in Deep learning and traditional Machine Learning have been introduced with the purpose of promptly and reliably detecting and classifying brain hemorrhage. This overview provides a comprehensive analysis of the surveys that have been conducted by utilizing Machine Learning and Deep Learning. This research focuses on the main stages of brain hemorrhage, which involve preprocessing, feature extraction, and classification, as well as their findings and limitations. Moreover, this in-depth analysis provides a description of the existing benchmark datasets that are utilized for the analysis of the detection process. A detailed comparison of performances is analyzed. Moreover, this paper addresses some aspects of the above-mentioned technique and provides insights into prospective possibilities for future research.

**[2] Y. Xu et al., "Deep Learning-Enhanced Internet of Medical Things to Analyze Brain CT Scans of Hemorrhagic Stroke Patients: A New Approach," in IEEE Sensors Journal, vol. 21, no. 22, pp. 24941-24951, 15 Nov.15, 2021.**

Stroke is among the first pathologies that kill the most in the world, ranking second in deaths from illness. Each year, around 16 million people worldwide are victims of this disease, with approximately 38 % of cases brought to death. Computed Tomography (CT) is a super effective method to aid the medical diagnosis of stroke. But the analysis is subject to variations in the perception of specialists about its characteristics. Faced with the challenge of diagnosing stroke on CT images, this study proposes a fully automatic system based on Health of Things capable of classifying CT images of the skull through deep learning networks, classifying them into (Uninjured or Hemorrhagic stroke). After the image classification, Mask R-CNN segments the stroke through a learning transfer process, combined with machine learning methods. Our innovative method selected excellent results, both for classification with 100 % accuracy, and for segmentation in our best model (Mask + kNN) that reached 99.93% specificity and 99.73 accuracy, with segmentation time of 4.00 seconds, surpassing literature methods based on automatic models.

**[3] M. F. Mushtaq et al., "BHCNet: Neural Network-Based Brain Hemorrhage Classification Using Head CT Scan," in IEEE Access, vol. 9, pp. 113901-113916, 2021.**

Brain Hemorrhage is the eruption of the brain arteries due to high blood pressure or blood clotting that could be a cause of traumatic injury or death. It is the medical emergency in which a doctor also need years of experience to immediately diagnose the region of the internal bleeding before starting the treatment. In this study, the deep learning models Convolutional Neural Network (CNN), hybrid models CNN + LSTM and CNN + GRU are proposed for the Brain Hemorrhage classification. The 200 head CT scan images dataset is used to boost the accuracy rate and computational power of the deep learning models. The major aim of this study is to use the abstraction power of deep learning on a set of fewer images because in most crucial cases extensive datasets are not available on the spot. The image augmentation and imbalancing the dataset methods are adopted with CNN model to design a unique architecture and named as Brain Hemorrhage Classification based on Neural Network (BHCNet). The performance of the proposed approach are analyzed in terms of accuracy, precision, sensitivity, specificity and F1-score. Further, the experimental results are evaluated by comparative analyses of the balanced and imbalanced dataset with CNN, CNN + LSTM and CNN + GRU models. The promising results are achieved with CNN by imbalancing the dataset and gain highest accuracy that outperforms the hybrid CNN + LSTM and CNN + GRU models. The results reveals the effectiveness of the proposed model for accurate prediction to save the life of the patient in the meantime and fast employment in the real life scenario.

**[4] Majid Roohi1, Jalil Mazloum2, Mohammad-Ali Pourmina1, and Behbod Ghalamkari, “Machine Learning Approaches for Automated Stroke Detection, Segmentation, and Classification in Microwave Brain Imaging Systems”, 2021.**

In this paper, an intracranial hemorrhage stroke detection and classification method using microwave imaging system (MIS) based on machine learning approaches is presented. To create a circular array-based MIS, sixteen elements of modified bow-tie antennas around a multilayer head phantom with a spherical target with radius of 1 cm as an intracranial hemorrhage target are simulated in CST simulator. To obtain satisfied radiation characteristics in the desired frequency band of 0.5– 5 GHz a suitable matching medium is designed. Initially, in the processing section, a confocal image reconstructing method based on delay-multiply-and-sum (DMAS) beam-forming algorithms is used. Then, reconstructed images are generated, which shows the applicability of the confocal method in detecting a spherical target in the range of 1 cm. Separating and categorizing targets is a challenging task due to the ambiguity in the extracted target from MIS. Thus, to distinguish between healthy and unhealthy brain tissues, a new compound machine learning technique, including filtering, edge-detection based segmentation, and applying K-Means and fuzzy clustering techniques, which reveal intracranial hemorrhage area from reconstructed images is adopted. Simulated results are presented to validate the proposed method effectiveness for precisely localizing and classifying bleeding targets.

**3. SYSTEM ANALYSIS**

**3.1 Existing System**

Existing methods for brain hemorrhage detection primarily rely on manual interpretation of medical images, such as CT scans and MRIs, by radiologists. This process, while effective, is time-consuming and subject to human error and variability in diagnostic expertise. Traditional computer-aided detection (CAD) systems have been employed to assist radiologists, using basic image processing and machine learning techniques. However, these systems often lack the sophistication required for high accuracy and are limited in their ability to generalize across diverse datasets. The need for more advanced, automated, and reliable diagnostic tools has prompted the exploration of deep learning models in this domain.

**3.2 Disadvantages:**

• Time-Consuming: Manual interpretation by radiologists is a slow process, leading to delays in diagnosis and treatment.

• Human Error: Diagnostic accuracy is subject to human error and variability, potentially resulting in missed or incorrect diagnoses.

• Limited Availability: Expert radiologists may not always be available, particularly in remote or resource-constrained settings.

• Basic CAD Systems: Traditional computer-aided detection systems often lack the sophistication needed for high accuracy, especially in complex cases.

• Generalization Issues: These systems struggle to generalize across different datasets and imaging conditions, limiting their robustness and reliability.

• Resource Intensive: Manual and traditional methods require significant human and computational resources, making them less efficient.

**3.3 Proposed System**

The proposed system leverages advanced deep learning models, specifically MobileNet and ResNet, to automate and enhance the detection of brain hemorrhages from medical images. MobileNet's lightweight architecture is optimized for deployment in resource-constrained environments, while ResNet's deep residual learning capabilities ensure high accuracy by effectively training very deep networks. This system aims to preprocess the image data, train the models, and evaluate their performance using metrics such as accuracy, sensitivity, and specificity. By providing a reliable and efficient diagnostic tool, the proposed system seeks to support healthcare professionals in making timely and precise diagnoses, ultimately improving patient outcomes.

**3.4 Advantages:**

• Enhanced Accuracy: Deep learning models like MobileNet and ResNet provide high accuracy in detecting brain hemorrhages, reducing the likelihood of missed or incorrect diagnoses.

• Automation: Automated analysis of medical images speeds up the diagnostic process, enabling quicker decision-making and treatment initiation.

• Consistency: Deep learning models provide consistent results, eliminating the variability and potential errors associated with human interpretation.

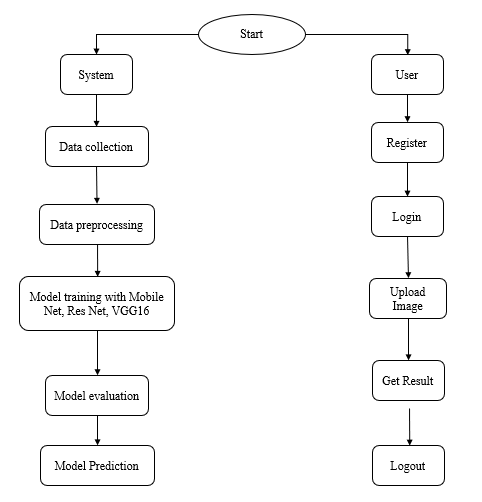
• Resource Efficiency: MobileNet's lightweight architecture allows deployment in resource-constrained environments, such as mobile devices, making advanced diagnostics more accessible.

• Scalability: The system can handle large volumes of data efficiently, making it suitable for use in busy clinical settings.

• Support for Radiologists: By providing a reliable second opinion, the system assists radiologists in making more informed decisions, enhancing overall diagnostic accuracy.

• Real-time Processing: Capable of rapid image analysis, the system supports real-time diagnostic applications, crucial for emergency medical situations.

**3.5 work Flow of Proposed system**

****

**Fig:** Flow Diagram.

**4. REQUIREMENT ANALYSIS**

**4.1 Functional and non-functional requirements**

Requirement’s analysis is very critical process that enables the success of a system or software project to be assessed. Requirements are generally split into two types: Functional and non-functional requirements.

**Functional Requirements**: These are the requirements that the end user specifically demands as basic facilities that the system should offer. All these functionalities need to be necessarily incorporated into the system as a part of the contract. These are represented or stated in the form of input to be given to the system, the operation performed and the output expected. They are basically the requirements stated by the user which one can see directly in the final product, unlike the non-functional requirements.

Examples of functional requirements:

1. Authentication of user whenever he/she logs into the system
2. System shutdown in case of a cyber-attack

**Non-functional requirements**: These are basically the quality constraints that the system must satisfy according to the project contract. The priority or extent to which these factors are implemented varies from one project to other. They are also called non-behavioral requirements.

They basically deal with issues like:

* Portability
* Security
* Maintainability
* Reliability
* Scalability
* Performance
* Reusability
* Flexibility

Examples of non-functional requirements:

1. The processing of each request should be done within 10 seconds
2. The site should load in 3 seconds whenever of simultaneous users are > 10000
   1. **Hardware Requirements**

# Processor - I3/Intel Processor

Hard Disk - 160GB

Key Board - Standard Windows Keyboard

Mouse - Two or Three Button Mouse

Monitor - SVGA

RAM - 8GB

* 1. **Software Requirements:**

Operating System : Windows 7/8/10/11

Server side Script : HTML, CSS, Bootstrap & JS

Programming Language : Python

Libraries : Flask, Pandas, Mysql.connector, Smtplib, Numpy

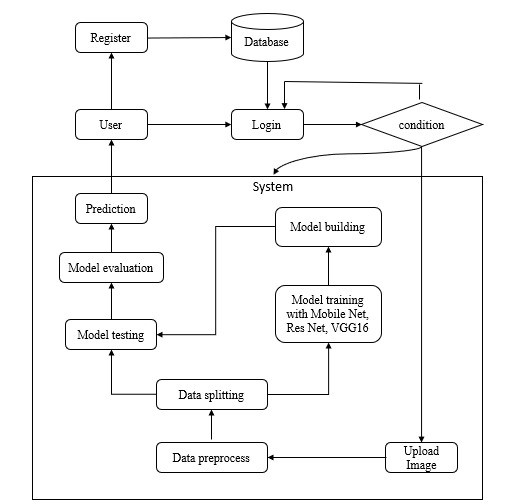
IDE/Workbench : PyCharm or VS Code

Technology : Python 3.6+

Server Deployment : Xampp Server

Database : MySQL

* 1. **Architecture:**



**5. METHODALOGY:**

**5.1 MobileNet**

MobileNet is a class of efficient convolutional neural networks designed specifically for mobile and resource-constrained environments. It uses depthwise separable convolutions, which break down a standard convolution into a depthwise convolution and a pointwise convolution. This approach significantly reduces the number of parameters and computational cost, making MobileNet much lighter than traditional CNNs. Despite its lightweight nature, MobileNet maintains robust performance, making it suitable for various image classification tasks. In our brain hemorrhage detection project, MobileNet is chosen for its efficient architecture. Its reduced size and lower computational requirements allow for deployment on mobile devices and in real-time applications, which is crucial for settings where quick and accurate diagnosis is needed, such as in remote or resource-limited environments. MobileNet provides the capability to perform real-time brain hemorrhage detection, assisting healthcare professionals in making timely decisions and potentially improving patient outcomes.

**5.2 ResNet (Residual Networks):**

ResNet, or Residual Networks, is a groundbreaking deep learning architecture known for its deep residual learning capabilities. It addresses the vanishing gradient problem by introducing shortcut connections that allow gradients to flow more easily through the network. This enables the training of much deeper networks, which can learn more complex patterns and representations. The typical ResNet architecture includes multiple layers of residual blocks, each containing convolutional layers and identity mappings. In our brain hemorrhage detection project, ResNet is employed for its ability to handle very deep networks effectively. Its deep architecture allows for capturing intricate details and patterns in brain hemorrhage images, leading to high diagnostic accuracy. ResNet's robustness makes it a reliable model for detecting subtle signs of brain hemorrhages, thus enhancing the precision of our automated diagnostic tool. The model's performance is critical in providing accurate assessments, which can aid in timely and appropriate medical interventions.

**5.3 VGG16:**

VGG16 is a convolutional neural network known for its simplicity and depth, consisting of 16 layers. Developed by the Visual Geometry Group at Oxford, VGG16 uses small 3x3 convolutional filters throughout the network, stacked on top of each other, followed by max-pooling layers. The network culminates in fully connected layers and a softmax classification layer. VGG16's structured and deep architecture enables it to learn complex features and perform exceptionally well in image classification tasks. In our brain hemorrhage detection project, VGG16 is utilized for its proven performance in image classification. Its deep architecture allows for capturing detailed features of brain hemorrhage images, making it a strong candidate for accurate diagnosis. VGG16 serves as a robust baseline model for comparison with other architectures. Its implementation helps in validating the effectiveness of more complex or efficient models like MobileNet and ResNet. By leveraging VGG16, we ensure that our project incorporates a well-established and reliable network, contributing to the overall robustness and accuracy of our brain hemorrhage detection system.

**6. SYSTEM DESIGN**

**6.1 Introduction of Input Design:**

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices such as PC, MICR, OMR, etc.

Therefore, the quality of system input determines the quality of system output. Well-designed input forms and screens have following properties −

* It should serve specific purpose effectively such as storing, recording, and retrieving the information.
* It ensures proper completion with accuracy.
* It should be easy to fill and straightforward.
* It should focus on user’s attention, consistency, and simplicity.
* All these objectives are obtained using the knowledge of basic design principles regarding −
  + What are the inputs needed for the system?
  + How end users respond to different elements of forms and screens.

### **Objectives for Input Design:**

The objectives of input design are −

* To design data entry and input procedures
* To reduce input volume
* To design source documents for data capture or devise other data capture methods
* To design input data records, data entry screens, user interface screens, etc.
* To use validation checks and develop effective input controls.

**6.2 Output Design:**

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

### **Objectives of Output Design:**

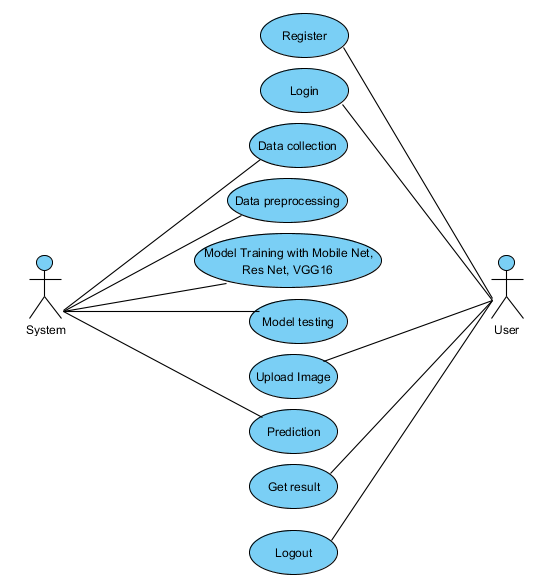
The objectives of input design are:

* To develop output design that serves the intended purpose and eliminates the production of unwanted output.
* To develop the output design that meets the end user’s requirements.
* To deliver the appropriate quantity of output.
* To form the output in appropriate format and direct it to the right person.
* To make the output available on time for making good decisions.

**6.3 UML Diagrams:**

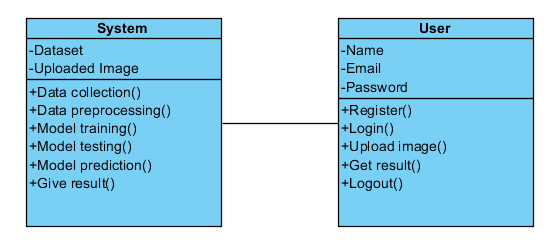
**6.3.1 Use Case Diagram:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.



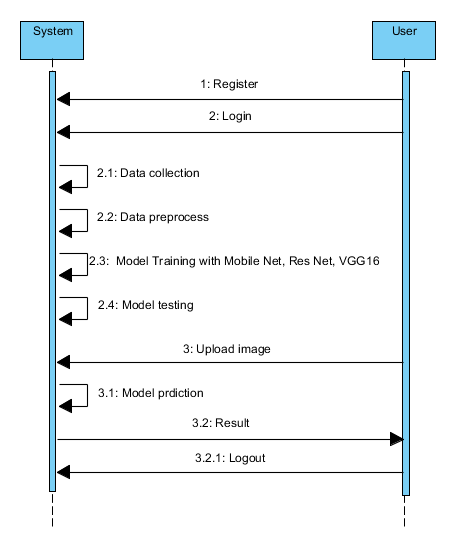
**6.3.2 Class Diagram:**

In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



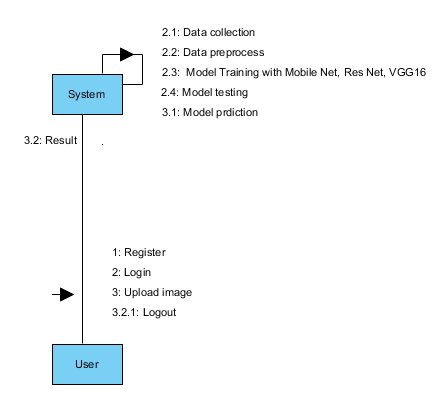
**6.3.3 Sequence Diagram:**

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



**6.3.4 Collaboration Diagram:**

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization whereas the collaboration diagram shows the object organization.



**6.3.5 Deployment Diagram**

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.



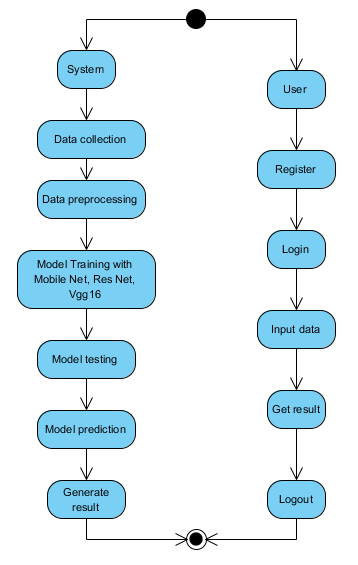
**6.3.6 Component Diagram**:

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical **c**omponents in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.



**6.3.7 Activity Diagram:**

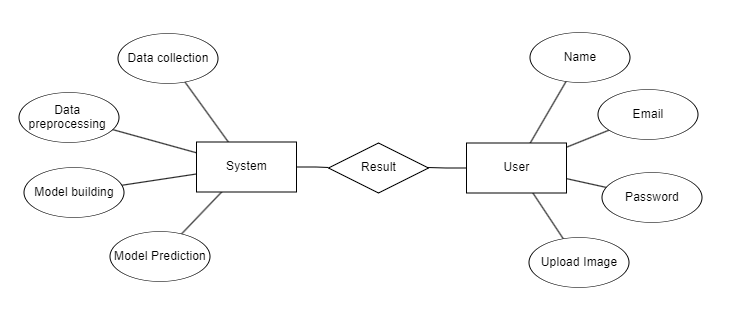
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



**6.3.8 ER Diagram:**

An Entity–relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as Entity Relationship Diagram (ER Diagram). An ER model is a design or blueprint of a database that can later be implemented as a database. The main components of E-R model are: entity set and relationship set.

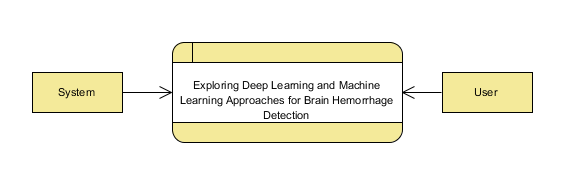
An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database. Let’s have a look at a simple ER diagram to understand this concept.

****

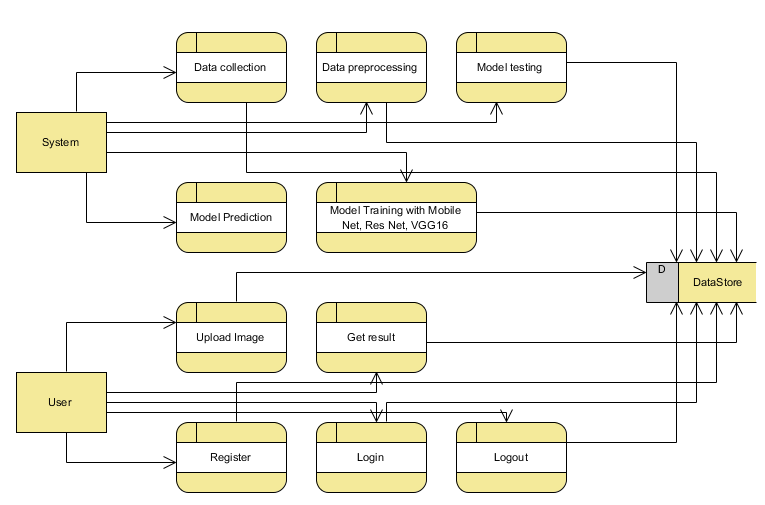
**6.4 DFD Diagram:**

A Data Flow Diagram (DFD) is a traditional way to visualize the information flows within a system. A neat and clear DFD can depict a good amount of the system requirements graphically. It can be manual, automated, or a combination of both. It shows how information enters and leaves the system, what changes the information and where information is stored. The purpose of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communications tool between a systems analyst and any person who plays a part in the system that acts as the starting point for redesigning a system.

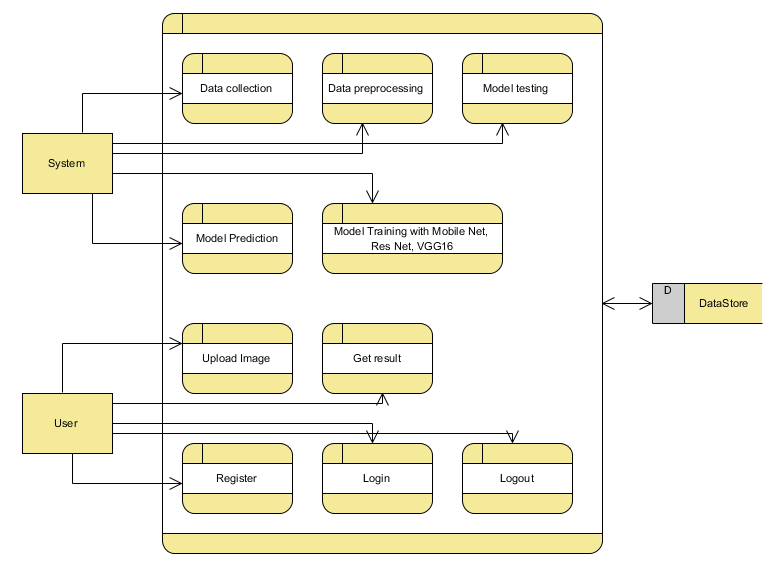
**Context Level Diagram**



**Level 1 Diagram:**

****

**Level 2 Diagram:**

****

**7. IMPLEMENTATION AND RESULTS**

**7.1 Modules:**

**1. System:**

**1.1 Data Collection:** In this module, the dataset containing images for stroke identification is divided into two subsets: the training dataset and the testing dataset. This split is typically done with a test size of 20%. The training dataset is used to teach the model, while the testing dataset is used to evaluate its performance.

**1.2 Data Splitting:** The pre-processed dataset is split into two subsets – Model training, Model testing.

**1.3 Model Training:** The training process involves using 80% of the dataset to teach the model. The model parameters are fine-tuned to minimize reconstruction errors through iterative optimization techniques, such as gradient descent.

**1.4 Model Testing:** The remaining 20% of the dataset is used for testing. The trained model predicts the segmentation of ischemic stroke lesions, and its performance is evaluated to determine the model's accuracy.

**1.5 Model Saving:** Once trained, the model is saved in a .pt format, preserving its learned weights and biases.

**1.6 Model Prediction:** Finally, we can input new images into the trained model to predict stroke.

**2. User:**

**2.1 Register:** Users should first register with their credentials to create an account in the system.

**2.2 Login:** Users can log in with their registered credentials to access the system.

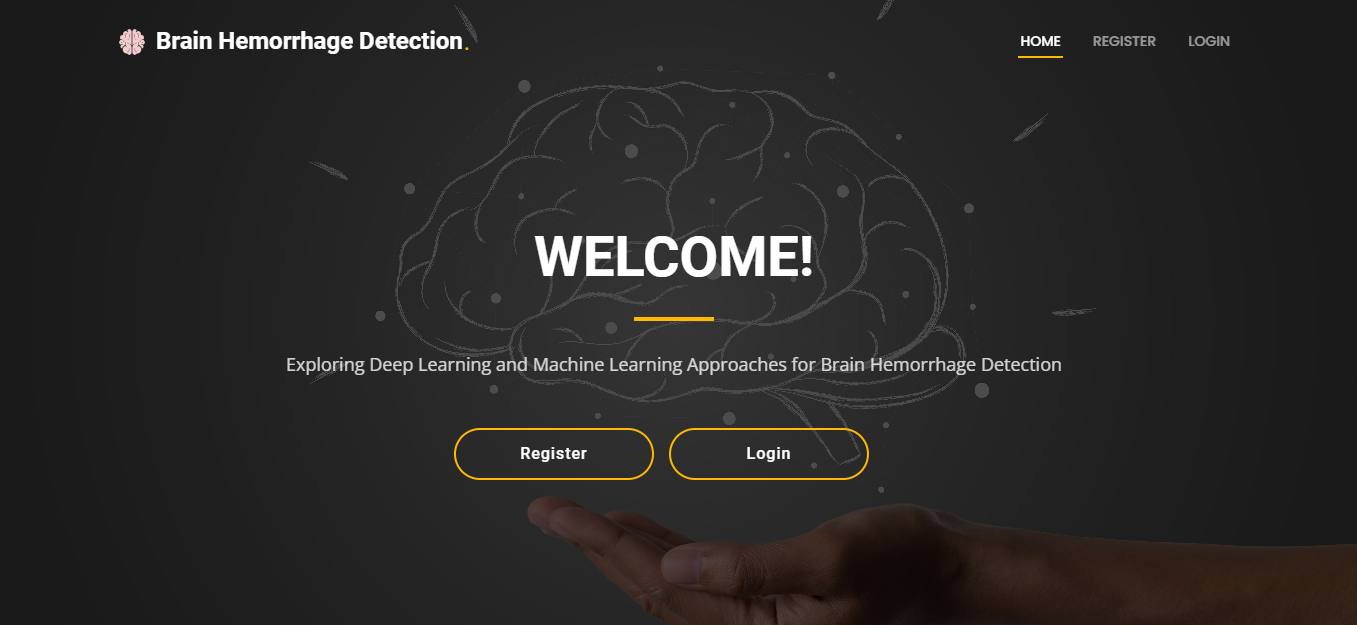
**2.3 Upload Data:** Users can upload their images to predict whether it is stroke or normal.

**2.4 Viewing Results:** That uploaded image will going to the model part to predict and it will give the prediction and user can view the result.

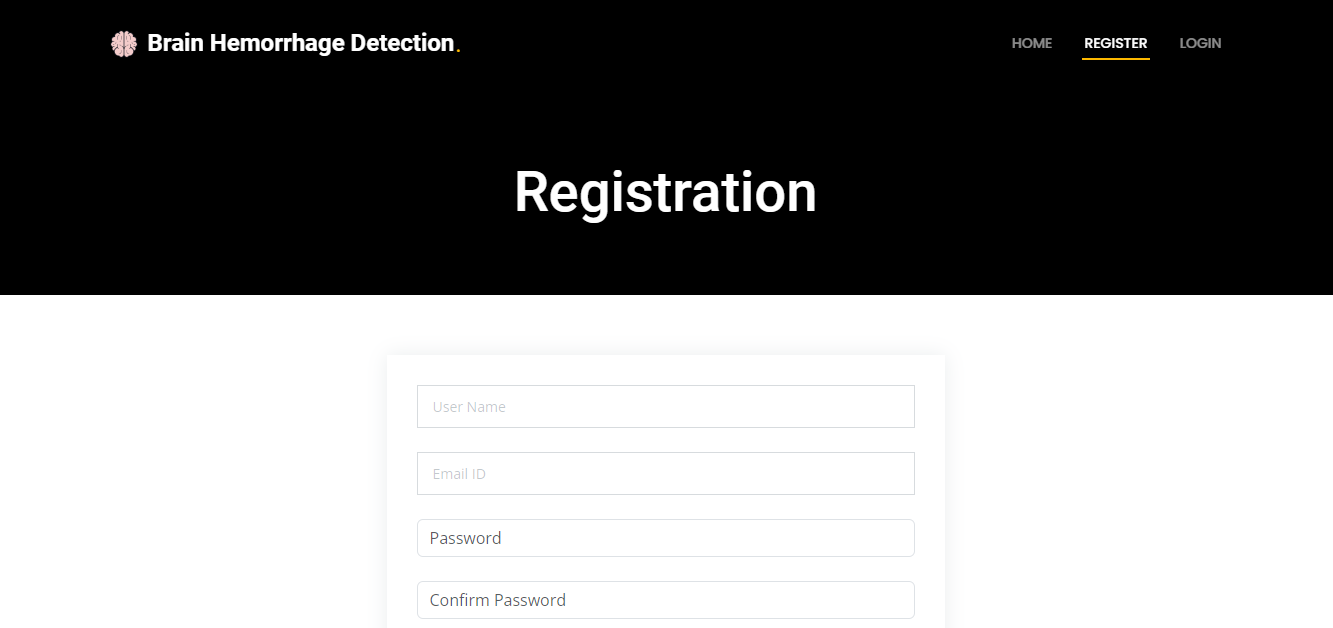
**2.5 Logout:** Finally, users can log out of the system to secure their session and personal data.

**7.2 OUTPUT SCREENS:**

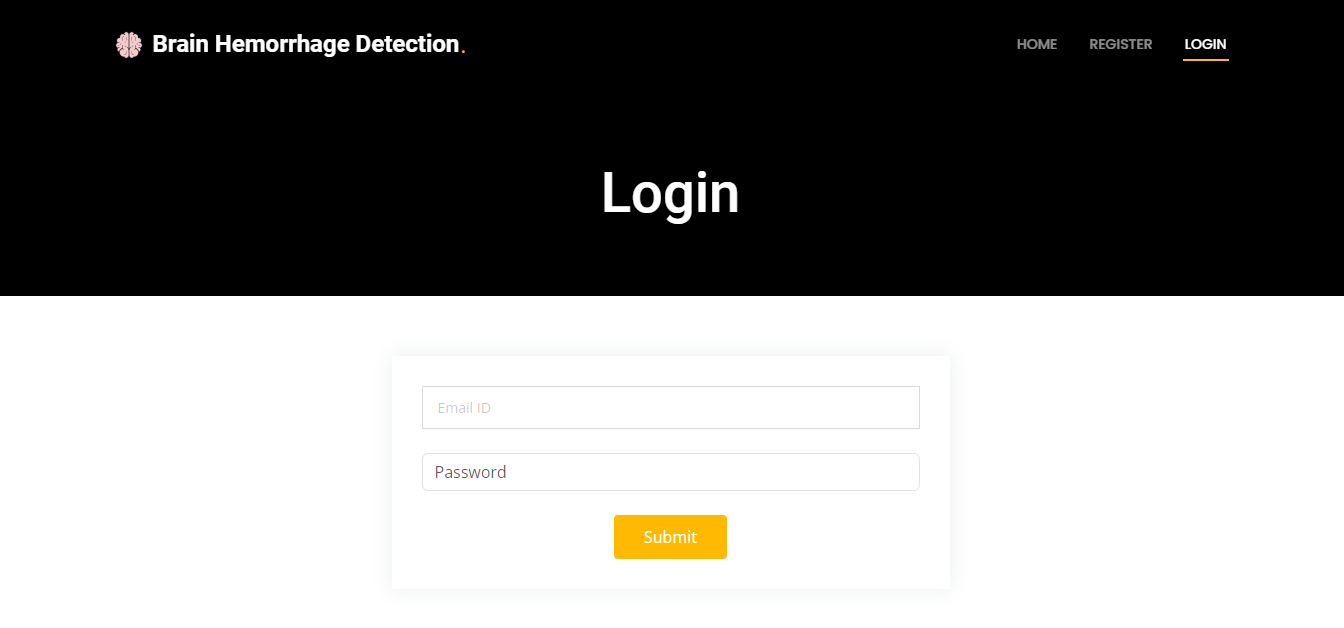
**Home page:** This is the index Page of our website.

****

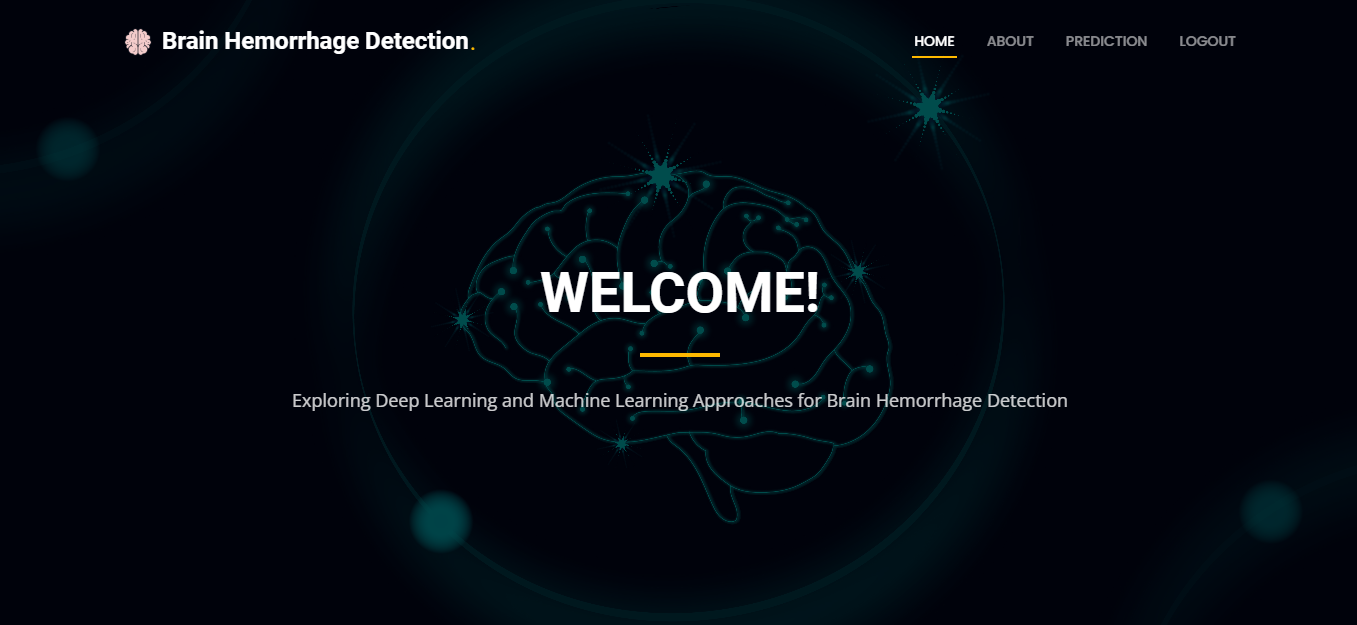
**Registration Page**: User can register with their credentials.

****

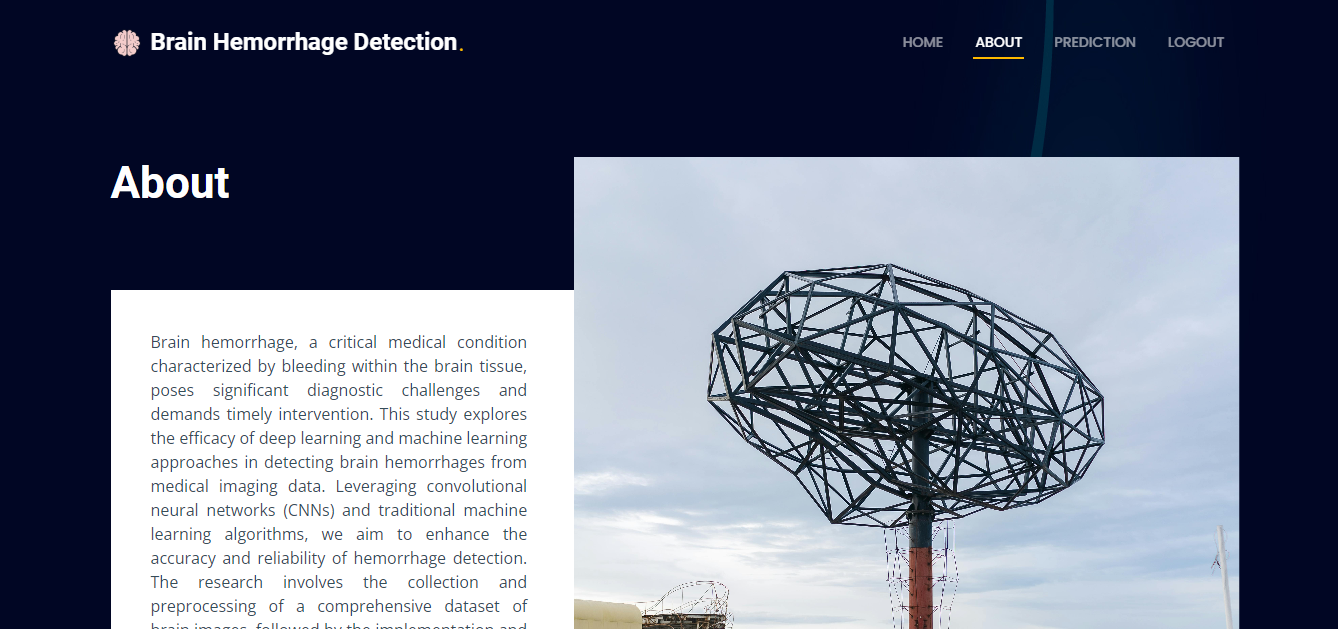
**Login Page:** User can login with their registered credentials.

****

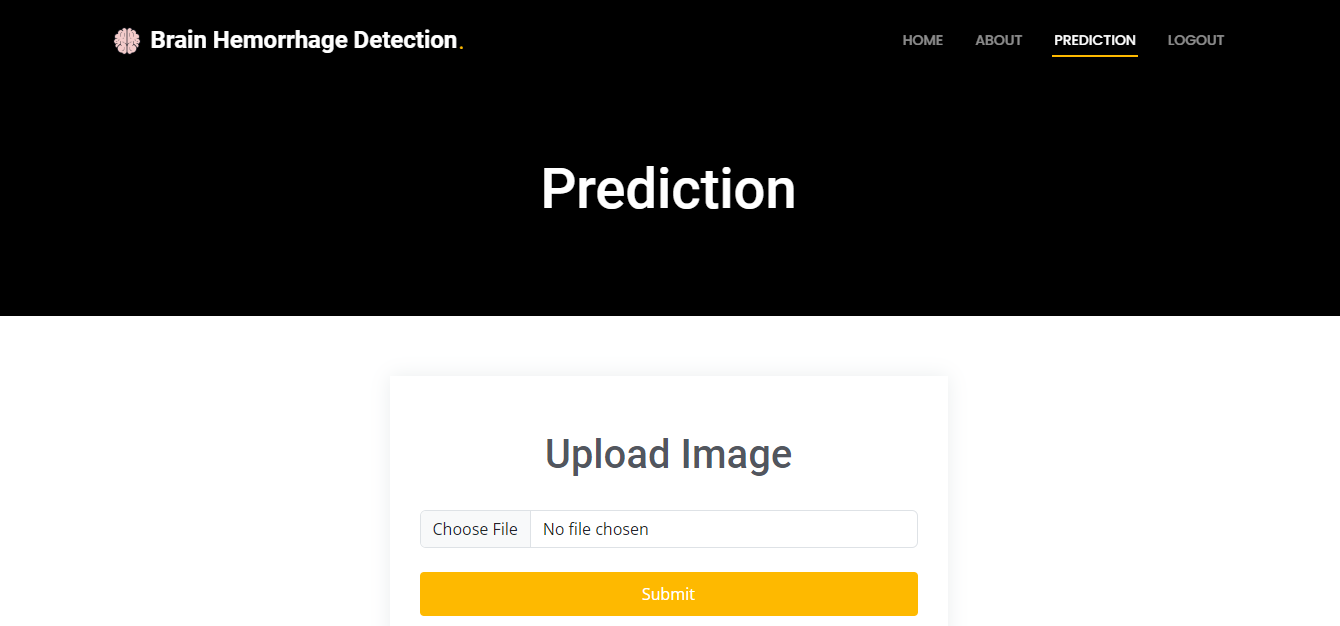
**Home Page:** After user login this page will be come.

****

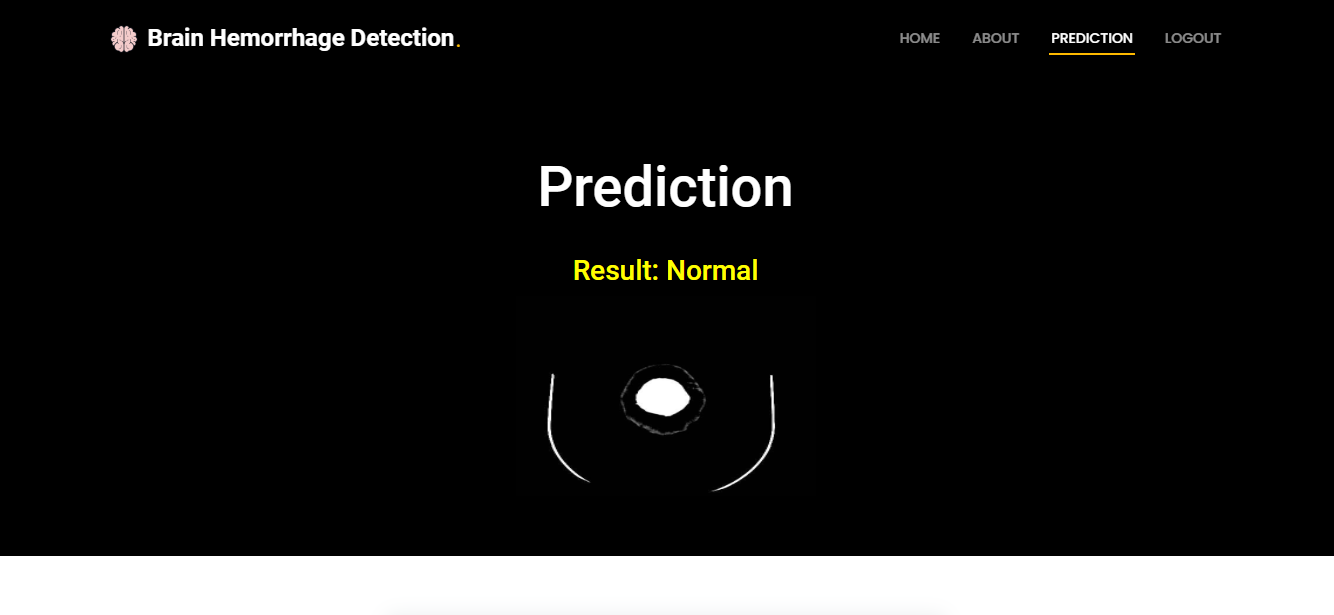
**About Page:** This page contains about our website.

****

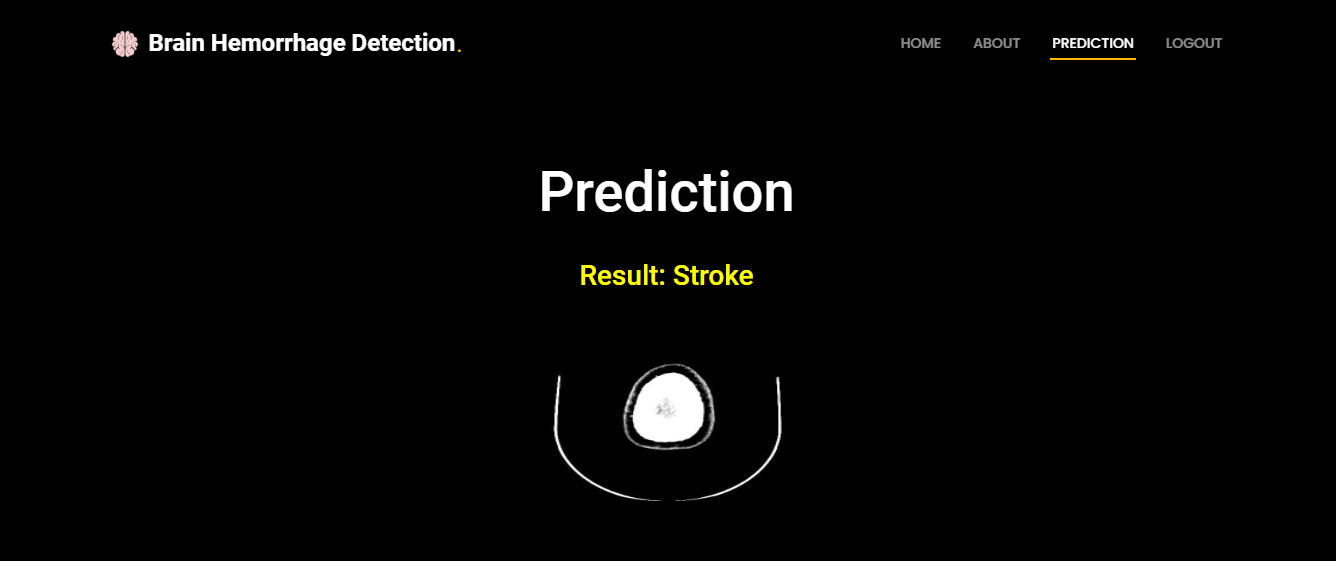
**Prediction Page:** This is the prediction page. User can give upload image here and get prediction result in result page.

****

**Result page:** This is the result page.



Prediction - Normal



Prediction - Stroke

**8. SYSTEM STUDY AND TESTING**

**8.1 Feasibility Study**

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

* Economic feasibility
* Technical feasibility
* Social feasibility

**Economic Feasibility**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### **Technical Feasibility**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

**Social Feasibility**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**System Testing**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

**8.2 Types of Tests**

**8.2.1 Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

**8.2.2 Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**8.2.3 Functional testing**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**8.2.4 White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**8.2.5 Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

**9. CONCLUSION**

This study evaluated the efficacy of deep learning models—MobileNet, ResNet, and VGG16—in detecting brain hemorrhages from medical images. Each model demonstrated high accuracy and specific advantages: MobileNet's lightweight architecture is ideal for resource-constrained environments, ResNet excels in handling deep networks with high accuracy, and VGG16 offers a simple yet effective structure for image classification. The comparative analysis highlighted that while all models perform well, their suitability depends on deployment requirements. MobileNet is excellent for mobile applications, whereas ResNet and VGG16 are better for settings with ample computational resources. The findings underscore the potential of deep learning to automate and enhance brain hemorrhage detection, reducing reliance on manual radiologist interpretation. This automation facilitates quicker diagnosis and treatment, crucial for improving patient outcomes. In conclusion, integrating deep learning models like MobileNet, ResNet, and VGG16 in medical diagnostics can significantly enhance accuracy and efficiency, supporting healthcare professionals in making timely decisions. This research contributes to the growing adoption of AI in healthcare, aiming to improve patient care and reduce the morbidity and mortality associated with brain hemorrhages. Future work could focus on optimizing these models and exploring their application in other critical medical conditions.

**10. FUTURE ENHANCEMENT**

Future enhancements for brain hemorrhage detection using deep learning models could focus on optimizing MobileNet, ResNet, and VGG16 for higher accuracy and faster inference. Advanced data augmentation techniques and synthetic data generation using GANs can improve model robustness. Implementing ensemble methods can enhance diagnostic accuracy by combining multiple model predictions. Transfer learning from larger medical image datasets can boost performance and reduce training time. Integrating these models into clinical workflows with user-friendly interfaces will facilitate real-time diagnostic support. Improving model transparency through explainability techniques can build clinician trust and ensure informed decision-making. Additionally, combining imaging data with patient history and genetic information can create a comprehensive diagnostic tool. Extensive clinical trials and real-world testing are essential to validate model effectiveness and ensure practical applicability. By addressing these areas, future work can significantly advance the capabilities of deep learning models in brain hemorrhage detection and broader medical diagnostics.

**11. REFERANCES**

1. Howard, A. G., et al. (2017). "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." arXiv preprint arXiv:1704.04861.
2. He, K., et al. (2016). "Deep Residual Learning for Image Recognition." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 770-778).
3. Simonyan, K., & Zisserman, A. (2015). "Very Deep Convolutional Networks for Large-Scale Image Recognition." arXiv preprint arXiv:1409.1556.
4. Litjens, G., et al. (2017). "A Survey on Deep Learning in Medical Image Analysis." Medical Image Analysis, 42, 60-88.
5. Ronneberger, O., Fischer, P., & Brox, T. (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation." In International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 234-241).
6. Springer, Cham. Shen, D., Wu, G., & Suk, H. I. (2017). "Deep Learning in Medical Image Analysis." Annual Review of Biomedical Engineering, 19, 221-248.
7. Lakhani, P., & Sundaram, B. (2017). "Deep Learning at Chest Radiography: Automated Classification of Pulmonary Tuberculosis by Using Convolutional Neural Networks." Radiology, 284(2), 574-582.
8. Rajpurkar, P., et al. (2018). "Deep Learning for Chest Radiograph Diagnosis: A Retrospective Comparison of the CheXNeXt Algorithm to Practicing Radiologists." PLoS Medicine, 15(11), e1002686.
9. Esteva, A., et al. (2017). "Dermatologist-level classification of skin cancer with deep neural networks." Nature, 542(7639), 115-118.
10. Hwang, E. J., et al. (2019). "Development and Validation of a Deep Learning–based Automated Detection Algorithm for Major Thoracic Diseases on Chest Radiographs." JAMA Network Open, 2(3), e191095.
11. LeCun, Y., Bengio, Y., & Hinton, G. (2015). "Deep Learning." Nature, 521(7553), 436-444.
12. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). "ImageNet Classification with Deep Convolutional Neural Networks." In Advances in Neural Information Processing Systems (pp. 1097-1105).
13. Zhang, J., et al. (2019). "Whole-brain segmentation and identification of brain structure in mice using U-Net convolutional network." BMC Bioinformatics, 20(1), 377.
14. Gulshan, V., et al. (2016). "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs." JAMA, 316(22), 2402-2410.
15. Sudre, C. H., et al. (2017). "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations." In Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support (pp. 240-248).