**Final Year Project Proposal**

+

**Title**

**Stroke Prediction App**

**Supervisor**

**Usman Aziz**

**Submitted By**

|  |  |
| --- | --- |
| Name | Roll Number |
| Marwah Ejaz | 22201003-001 |
| Mamoon Mukhtar | 24020103-013 |
| Joshina | 24020103-020 |
|  |  |

**APPROVAL**

I **Mr. Usman Aziz** am willing to guide these students in all phases of project titled “**Stroke Prediction App**” as supervisor. I have carefully seen the title and description of the project and believe that it is of an appropriate difficulty level for the number of students named above.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date Signature of Supervisor

Submission Date: \_\_\_\_\_\_\_\_\_\_\_\_\_

**Table of Contents**

[1. Project Title 5](#_Toc128424069)

[1.1 Introduction 5](#_Toc128424070)

[1.2 Problems in Existing System 5](#_Toc128424071)

[1.3 Proposed System 5](#_Toc128424072)

[1.4 Main Modules 5](#_Toc128424073)

[1.5 Expected Outcomes 6](#_Toc128424074)

[1.6 Tools & Technology 6](#_Toc128424075)

[1.7 Activity Index 6](#_Toc128424076)

[1.8 References 8](#_Toc128424077)

**List of Figures**

[**Figure 1.1 Stroke Prediction App Logo** 5](#_Toc128423900)

**List of Tables**

[**Table 1.1 Activity Index** 6](#_Toc128423914)

1. Project Title

Stroke Prediction App

1.1 Introduction

Stroke prediction applications leverage machine learning to analyse individual risk factors such as age, gender, medical history, lifestyle, and genetics to estimate the likelihood of a stroke. This allows for early identification of high-risk individuals, enabling proactive preventive measures like lifestyle changes or medical interventions. While offering personalized risk assessments and potentially improving patient management these applications are not a replacement for professional medical advice and should be used as a supplementary tool in conjunction with consultations with healthcare professionals. Stroke is the world's second-biggest cause of death and one of the most life-threatening diseases for people over 65 years. It causes brain damage like how a "heart attack" causes heart damage. It damages the brain in the same way that a "heart attack" damages the heart. When a stroke strikes, it not only costs much money in hospital bills and causes lasting disability but it can also lead to death. Every 4 minutes someone dies from a stroke however up to 80% of strokes can be avoided if we can detect or forecast stroke onset early on. Stroke is the fifth-leading cause of death in the United States. According to the Centers for Disease Control and Prevention, stroke is a non-communicable disease that accounts for roughly 11% of deaths. Over 795,000 people in the United States suffer from a stroke daily.

1.2 Problems in Existing System

* Accuracy and Generalizability: Existing risk prediction models often struggle to accurately predict stroke risk in diverse populations. They may be very accurate for one group but perform poorly on others.
* Data Issues: The lack of comprehensive, high-quality data makes it difficult to train effective ML models. Data is often fragmented, incomplete, or inconsistently recorded.
* Lack of Transparency ("Black Box" Problem): Many ML models are complex and difficult to interpret. Understanding why a model makes a specific prediction is crucial for trust and acceptance by clinicians.
* Data Integration: Combining data from various sources (electronic health records, wearables, imaging) is challenging due to inconsistencies in formats and standards.
* Ethical Concerns: Using ML in healthcare raises ethical issues around data privacy, potential biases in the data leading to biased predictions, and the unexpected consequences of relying on algorithms for critical medical decisions.

1.3 Proposed System

The proposed system is a mobile application integrated with machine learning and AI-based recommendation features to predict the likelihood of a user suffering from a stroke and to provide personalized health recommendations. The mobile app will collect key health-related inputs from the user, such as age, gender, BMI, hypertension status, heart disease history, smoking habits, and glucose levels. This data will be sent to a backend server hosting a trained machine learning model, which will predict the risk of stroke. Based on the user's risk factors and profile, an AI-powered recommendation engine will generate tailored suggestions to promote a healthier lifestyle or prompt medical consultation. The system will utilize a Python-based backend (Flask/Fast API) for model hosting and API management and a cross-platform mobile development framework such as Flutter or React Native for the frontend. The goal of the proposed system is to provide a fast, accessible, and intelligent tool for early stroke risk detection and health improvement guidance, ultimately enhancing preventive healthcare through technology.



Figure 1.1 Stroke Prediction Logo

1.4 Main Modules

* User Authentication
* User Data Entry
* Stroke Prediction Engine (ML Model Integration)
* Result Display
* Personal AI Recommendation
* Data Storage
* Notification & Reminders
* Admin Panel

1.5 Expected Outcomes

The expected outcomes of this project include a fully functional mobile application capable of accurately predicting a user's risk of stroke based on their personal and medical information. The system will also provide personalized health recommendations generated by an AI-based recommendation engine helping users take preventive actions. The backend machine learning model will achieve a high level of prediction accuracy while the mobile app will deliver a user-friendly, responsive experience with real-time stroke risk assessments. Additionally the project will result in a deployed and accessible cloud-based API service complete project documentation and a comprehensive presentation. Over all the system aims to contribute to early detection and prevention strategies for stroke by empowering users with actionable health insights through technology.

1.6 Tools & Technology

* **Frontend:** React native (using JavaScript and React).
* **Backend:** Java
* **Database:** Firebase
* **AI:** Node.js/Flask API
* **Model Training:** Python
* **Model Deployment:** TensorFlow Lite
* **Machine Learning Algorithms:**
* **Support Vector Machine (SVM)**
* **Logistic Regression**
* **Random Forest**
* **Decision Tree**

1.7 Activity Index

Table 1.1 Activity Index

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Activity** | **Duration** | **Deliverables** | **Responsibilities division** |
|  | 1)Requirement gathering  1.1) Defining the scope of the project  1,2) Gathering all data related to the project from the most advanced university management system  1.3) listing all the functional and nonfunctional requirements for the scope of the project | 1.1) 7 days (29 April to 5 May) | 1) Defining the requirements roles, collecting data, deciding deadlines and division of project according to the individual strengths in the field  Role | 1) Partner1: 33.33%  2) Partner2: 33.33%  3) Partner3: 33.33% |
|  | 1)Starting Implementation  1.1) Defining roles and responsibilities of each actor in the project  1.2) Making ER diagram  1.3) Implementation of models based on the ER diagram | 1.2) 7 days (6 May to 12 May) | 2) Defining the roles and make ER diagram for the admin, faculty and student in an iterative model which can be changed on future developments Start writing the report | 1) Partner1: 15.50% (Report) + 15.50% (Development)  2) Partner2: 15.50% (Report) + 15.50% (Development)  3) Partner3: 15.50% (Report) + 15.50% (Development) |
|  | 1)Implementing roles of the actors (students, Faculty and admin)  1.1) Implementing the decided roles  1.2) Revising the roles if required  2) Modifying models if required  2.1) Modifying models if we required any more fields or tables to support roles  2.2) Modifying data types of the models if required | 1.3) 7 days (13 May to 19 May) | 3) Implementing the roles of the actors and match them with our high level and low-level requirements in incremental way and go to revision 2 of models if we required any change  3.1) Update the report as well with our recent development | 1)Partner1: 15.50% (Report) + 15.50% (Development)  2) Partner2: 15.50% (Report) + 15.50% (Development)  3) Partner3: 15.50% (Report) + 15.50% (Development) |
|  | 1) Implementing the views for the controllers and models  1.1) Developing the GUI (User Interface and User interface)  1.2) Implementation of GUI in web application  2) Revision of roles if required by front end  3) Revision of models if required by front end | 1.4) 7 days (20 May to 26 May) | 4) Developing a GUI on paper and transform it on the web application. Connect it with the roles and make all the things functional  4.1) Update the report as well with our recent development  4.2) Revise the model or control if required by iterative approach | 1) Partner1: 15.50% (Report) + 15.50% (Development)  2) Partner2: 15.50% (Report) + 15.50% (Development)  3) Partner3: 15.50% (Report) + 15.50% (Development) |
|  | 1) Revising the GUI and do the required changes  2) Complete the report  3) Finalize the web application and add revisions if necessary | 1.5) 7 days (27 May to 2 June) | 5) Updating the GUI if required and finalizing the application by adding the revision to models and roles (If necessary)  5.1) Completing the report with all the S.E components  5.2) Match the application with the requirements provided above so that nothing is left behind | 1) Partner1: 15.50% (Report) + 15.50% (Development)  2) Partner2: 15.50% (Report) + 15.50% (Development)  3) Partner3: 15.50% (Report) + 15.50% (Development) |
|  | 1) Implementing the high-level requirements like security, authentication and robustness  1.1) Mock Testing  1.2) Low level testing  1.3) High level testing | 1.6) 7 days (3 June to 9 June) | 6) Implementing components like role authentication security, data integrity and robustness  6.1) Doing Black box testing  6.2) Doing in depth testing like white box, testing fields testing etc.  6.3) Doing mock testing with actual people involved | 1) Partner1: 20% (Testing) + 10% (Development)  2) Partner2: 20% (Testing) + 10% (Development)  3) Partner3: 20% (Testing) + 10% (Development) |

* 2. References

**[1]** [(PDF) Stroke Prediction and Contributing Factors Using Machine Learning](https://www.researchgate.net/publication/380424935_Stroke_Prediction_and_Contributing_Factors_Using_Machine_Learning)

**[2]** [stroke prediction app existing problems - Google Scholar](https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=stroke+prediction+app+existing+problems+&btnG=#d=gs_qabs&t=1745827187833&u=%23p%3Dg0pekrkxkNEJ)

**[3]**  [stroke prediction app existing problems - Google Scholar](https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=stroke+prediction+app+existing+problems+&btnG=#d=gs_qabs&t=1745827215372&u=%23p%3DQRgp1GXK6UUJ)

**[4]**  [stroke prediction app existing problems - Google Scholar](https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=stroke+prediction+app+existing+problems+&btnG=#d=gs_qabs&t=1745827215372&u=%23p%3DQRgp1GXK6UUJ)

**[5]** <http://ww1.w3scool.com/>