

**Final Year Project Proposal**

**Title: NextGen AI Healthcare: Symptoms-based Diagnosis and Medical Rentals**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Department of Computer System Engineering**

**University of Engineering and Technology Peshawar**

Submitted By:  
**Muhammad Saad 21pwcse1997**

**Muhammad Umar Jan 21pwcse2000**

**Muhammad Zaid 21pwcse1991**

**Muhammad Ilyas 21pwcse2055**

**Mapping of the Project with Computer System Engineering Subjects**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Course Code** | **Course Name** | **Justification** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Project Title**

**NextGen AI Healthcare: Symptoms-based Diagnosis and Medical Rentals**

# Introduction

The healthcare industry is evolving rapidly, and the integration of AI and technology has the potential to greatly enhance patient care. The **NextGen AI Healthcare** project aims to bridge the gap between individuals seeking medical guidance and those needing access to essential medical equipment. This project has two main components:

1. **Symptoms-based AI Diagnosis System**: Utilizing AI models trained on medical datasets to provide users with insights like potential diagnoses, prescriptions, recommended rest periods, exercise routines, and dietary plans based on their symptoms.
2. **Volunteer Medical Equipment Rental System**: A system where users can list and rent out medical equipment (e.g., diabetes machines, blood pressure monitors) for free, fostering a community-driven initiative for medical supplies.

# Project Objectives

The **NextGen AI Healthcare** project has two major objectives:

**AI-based Diagnosis System**:

1. Leverage a dataset from Kaggle containing symptoms and diagnoses.
2. Build a machine learning model using AI frameworks (PyTorch or TensorFlow) to predict diseases based on input symptoms.
3. Provide users with detailed information such as:
4. Disease diagnosis.
5. Prescription and medication recommendations.
6. Rest and recovery advice.
7. Suggested exercises and dietary guidelines.

**Medical Equipment Rental System**:

1. Create a community-driven platform where users can post medical equipment they no longer need or use.
2. Ensure that people who need these medical devices can rent them for free.
3. Equipment categories include:
4. Diabetes monitoring machines.
5. Blood pressure monitors.
6. Wheelchairs.
7. Crutches.
8. Oxygen concentrators.
9. Thermometers and pulse oximeters.

# Scope of Work

The project will be developed in two phases:

**Phase 1: AI-based Diagnosis System**

1. **Data Collection & Preprocessing**:
2. Utilize a Kaggle dataset of symptoms and diseases.
3. Clean, preprocess, and prepare the dataset for training.

**Model Development**:

1. Experiment with PyTorch and TensorFlow to find the best-fit AI framework.
2. Train the model on the dataset to predict diseases and provide relevant information based on symptoms.
3. Integrate the model with the backend (Node.js) for seamless communication with the frontend.
4. **User Interface**:
5. Design a simple and intuitive user interface using Flutter or React.
6. Allow users to input their symptoms.
7. Display AI-generated recommendations, including diagnosis, medication, rest, exercise, and diet plans.

**Phase 2: Volunteer Medical Equipment Rental System**

**Platform for Listings**:

1. Create a web/mobile interface where users can list their available medical equipment for rent.
2. Include necessary details such as item description, condition, and contact details.

**Search & Filter Functionality**:

1. Implement features that allow users to search for specific medical equipment based on location or category.

**Communication & Rent Requests**:

1. Provide a simple request system where people can send inquiries to owners for equipment rental.
2. Ensure communication between users is secure and straightforward.

# Technology Stack

**Frontend**:

1. **Flutter** or **React**: The choice of Flutter or React depends on performance and UI requirements.
2. **Flutter**: Offers cross-platform development, enabling a single codebase for both Android and iOS.
3. **React**: A robust framework with extensive libraries and a large community, ideal for web-first interfaces.

**Backend**:

1. **Node.js**: A powerful and efficient backend technology that allows for scalability and fast processing of data. It is ideal for handling the AI logic and managing interactions between the frontend and database.

**AI Framework**:

1. **PyTorch** or **TensorFlow**: The decision between PyTorch and TensorFlow will be based on integration capabilities with Flutter/React and performance:
2. **PyTorch**: Known for its dynamic computational graph and easier debugging.
3. **TensorFlow**: Offers extensive deployment options, especially for mobile and web applications.

**Database**:

1. **MongoDB**: A NoSQL database that provides scalability and flexibility, ideal for handling both structured and unstructured data (e.g., user records, equipment listings, and AI model predictions).

# Functional Requirements

**AI-based Diagnosis System:**

1. **User Input**: Users enter their symptoms via the frontend.
2. **AI Prediction**: The AI model predicts the likely disease and recommends a treatment plan.
3. **Results Display**: Display the diagnosis, medication, rest, exercise, and diet advice on the frontend.

**Medical Equipment Rental System:**

1. **Post Listings**: Users post medical equipment with details like name, description, and availability.
2. **Browse & Search**: Allow users to browse listings and filter based on equipment type or location.
3. **Request System**: Users can request to borrow equipment, and the owners can accept or deny these requests.

# Non-Functional Requirements

1. **Security**: Ensure the platform is secure by encrypting personal data and implementing authentication for users.
2. **Scalability**: Design the system to handle a large number of users and data as the platform grows.
3. **Performance**: Optimize both the AI model and rental platform for quick response times and smooth user experience.
4. **Usability**: The system should be intuitive and easy to navigate for all age groups, especially given the nature of medical users.

# Project Timeline

|  |
| --- |
|  |

|  |  |
| --- | --- |
| Requirements gathering and design | 2 weeks |
| AI model development | 4 weeks |
| Backend development | 8 weeks |
| Frontend development | 10 weeks |
| Integration and testing | 10 weeks |
| Final Presentation | 1 weeks |

# Challenges and Risks

1. **AI Model Accuracy**: Achieving high accuracy in the AI model may require extensive training and tuning.
2. **Data Privacy**: Protecting sensitive medical data and ensuring compliance with data protection regulations is critical.
3. **User Adoption**: Encouraging users to post medical equipment for rent and ensuring a steady supply of listings may take time.

# Conclusion

The **NextGen AI Healthcare** project aims to revolutionize healthcare access by providing users with an AI-driven diagnosis tool and a platform for free medical equipment rentals. With the right use of technology and community involvement, this project has the potential to positively impact numerous individuals by making healthcare guidance and resources more accessible.

# Team Structure

1. **Project Lead**: Oversees the development and ensures milestones are met.
2. **AI Developer**: Responsible for model development and training.
3. **Backend Developer**: Implements the Node.js backend, handling API creation and database management.
4. **Frontend Developer**: Builds the Flutter or React frontend, ensuring responsive design and seamless user experience.
5. **Tester**: Ensures that the system meets functional and non-functional requirements.

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