

## HealthAI Screen

The recent surge in the cost of living, particularly in the healthcare sector, has created an impetus to drive down costs through the use of standard AI-trained models and more specifically generative AI in the form of Large Language Models (LLMs).

Furthermore, a societal transformation is taking place whereby healthcare professionals, and patients alike, are shifting from reactive to predictive HealthAI care management. Rather than waiting for health problems to be manifested as life-threatening conditions, AI technology enables the prediction of future health problems using datasets of patient information.

This transformation in healthcare management has created an opportunity for your software development company to develop applications for this lucrative medical sector. The objective of your start-up is to develop a bespoke system which predicts the likelihood of someone contracting the serious medical conditions of **heart disease, lung cancer and colon cancer**. As a method of fundraising your start-up, the team will charge subscribers a monthly fee. Payments will be made using a payment processor such as PayPal Sandbox or Stripe.

The system consists of the key features outlined below.

### FEATURES:

To develop the system, several applications are required:

- An android app, HealthAI-App, enables patients and doctors, to calculate the risk of illness based on lifestyle and prior medical history
- A portal website, HealthAI-Web, for medical professionals.
- Machine learning models, based on internationally available medical datasets, will be developed to implement the backend AI (HealthAI-Predict).
- Integration of LLM technology, HealthAI-LLM

### HealthAI-App basic features should include:

- **Register/login:** email and/or social media login option and accounts.
- **Record details of patients' GP**
- **Record details of the insurance company**
- **Payment of subscription premium using Paypal, Stripe or another suitable payment processor**
- **Medical History:** detailed forms capturing the information required by the machine learning models in HealthAI-Predict.
- **HealthAI-Predict Interface** based on the user's profile, HealthAI-Predict may be called to determine the patient's risk of contracting a range of illnesses.
- **Request the professional** ability to call the GP, or insurance company on record, from within the app
- **Support** forms to contact the insurance company or medical professional

- **Ratings & reviews**      Capability of rating the quality of the app and leaving reviews of the app's performance

#### **HealthAI-Web basic features should include:**

- **Registration** of medical professionals using email and/or social media login
- **Access Restrictions:** Professionals should only be able to see the details of their own patients
- **HealthAI-predict Interface** allows professionals to view their patient risk profiles
- **User Profiling** enables medical professionals to run reports establishing levels of risk for all categories
- **Aggregation of new patient data.** Using new patient information, the administrator should have the option to create and export new datasets by extending those used by HealthAI-Predict.

#### **HealthAI-Predict basic features should include:**

- Simple API linking HealthAI-Predict backend with the HealthAI-App and HealthAI-Web applications
- Models, using publicly available datasets, which determine patient risk of **heart disease, lung cancer and colon cancer**
- Reporting that displays the accuracy of each of these models

#### **HealthAI-LLM**

In parallel to HealthAI-Predict, the power of generative AI will also be leveraged through the use of LLMs to achieve the following functionality:

- API Integration of an LLM such as OpenAI's chatGPT or Google's Bard into both the web and Android applications
- **Doc-bot:** A symptom-checker to detect the likelihood of a user having any of the conditions currently being checked i.e. **heart disease, lung cancer or colon cancer**
- **Doc-bot:** Appropriate advice to the user by the LLM (with all the necessary qualifications and disclaimers)
- Few-shot learning of the chosen LLM to reduce hallucinations and improve the quality of the LLM responses.
- Incorporation of machine translation technology to create a multilingual application which facilitates the rapid global rollout of this technology
- System help provided by chatBot

**Each project group should also incorporate a custom feature which is unique to the group.** This custom feature should be implemented in one (or more) of the key components of the project i.e. HealthAI-App, HealthAI-Web, HealthAI-Predict or HealthAI-LLM.

## **Technology:**

The first application, the patient app (HealthAI-App), involves the development of an Android app to interface with the machine learning models developed as part of the backend AI. Development may be carried out using Android Studio or a similar development environment.

The second application, the website (HealthAI-Web), involves developing a website for medical professionals. Programming languages for HealthAI-Web may include Node.js, HTML, Python and PHP. To facilitate rapid development of the site, a CMS such as WordPress or Bootstrap may be used.

Both HealthAI-App and HealthAI-Web will store their data in a centralised database such as Firebase. The machine learning models used in HealthAI-Predict will be developed using Python and Keras.

Github must be used for code management and marks will be allocated for its use. A suitable project management tool such as JIRA, Teamwork etc should be used.

## **Reference:**

It's extremely important to go through these resources since some of this project involves self-directed learning. The links below outline the datasets and sample code which may be used for the AI components of the project.

### **Simple Python machine learning implementation:**

<https://machinelearningmastery.com/machine-learning-in-python-step-by-step/>

<https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/>

### **Important post illustrating how new unseen data can be entered into previously built models:**

<https://machinelearningmastery.com/how-to-connect-model-input-data-with-predictions-for-machine-learning/>

## **Datasets**

<https://www.kaggle.com>

## **LLMs**

<https://www.geeksforgeeks.org/project-ideas-using-large-language-models/>

<https://platform.openai.com/docs/quickstart>