Medi-Check

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

The transformation in health care management, coupled with new machine learning models have 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, diabetes and Alzheimer's.

Utility tokens are now frequently used to fund start-ups. As a method of fundraising your start-up, the team will use an ICO (Initial Coin Offering) to develop a crypto utility token, Medi-Coin. This new crypto will be built on top of the Ethereum blockchain and will be used to purchase services offered by Medi-Check.

The system consists of the key features outlined below.

FEATURES:

To develop the Medi-Check system, several applications are required:

- An android app, Medi-App, that enables patients and doctors, to calculate risk of illness based on lifestyle and prior medical history
- A portal website, Medi-Web, for medical professionals.
- Machine learning models, based on internationally available medical datasets, will be developed to implement the backend AI (Medi-predict).
- A simple blockchain and ICO

Medi-App basic features should include:

- **Register/login**: email and/or social media login option and accounts.
- **Record details of patients GP**
- **Record details of insurance company**
- **Payment of insurance premium using Medi-Coin**
- **Medical History**: detailed forms capturing the information requirement by the machine learning models in Medi-Predict.
- **Medi-AI Interface** based on the user's profile, Medi-Predict may be called to determine the patient's risk of contracting a range of illnesses.
- **Request 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

Medi-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
- **Medi-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 an option to create and export new datasets by extending those used by Medi-AI.

Medi-Predict basic features should include:

- Simple API linking Medi-Predict backend with the Medi-App and Medi-Web applications
- Models, using publicly available datasets, which determine patient risk of heart disease, diabetes and Alzheimer's
- Reporting which displays the accuracy of each of these models

Blockchain and Crypto Payments

In addition to storing patient data in a database (such as Firebase), a blockchain will also be used for storing patient data. The blockchain will serve as a secure, immutable and distributed record of patient data.

Your company may decide the exact tokenomics of the project i.e. how many coins will be issued, how many are retained by you as developers and what, if any, mining rewards will be paid out.

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. Medi-App, Medi-Web, Medi-Predict, the blockchain or in Medi-Coin.

Technology:

The first application, the patient app (Medi-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 (Medi-Web), involves developing a web site for medical professionals. Programming languages for Medi-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 Medi-App and Medi-Web will store their data in a centralised database such as Firebase. The machine learning models used in Medi-Predict will be developed using python, Keras and Tensorflow.

Github must be used for code management and marks will be allocated for their 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, Crypto and Blockchain 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 to previously built models:

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

Datasets

Diabetes using Pima Indians diabetes dataset:

https://cainvas.ai-tech.systems/use-cases/diabetes-prediction-app/

Heart Disease:

https://towardsdatascience.com/heart-disease-prediction-73468d630cfc

https://github.com/ShubhankarRawat/Heart-Disease-Prediction

https://www.kaggle.com/jboysen/mri-and-alzheimers/activity

Crypto currency

https://mlsdev.com/blog/how-to-create-your-own-cryptocurrency

Blockchain

How does a blockchain work - Simply Explained

https://www.youtube.com/watch?v=SSo_EIwHSd4

Creating a blockchain with Javascript (Blockchain, part 1)

https://www.youtube.com/watch?v=zVqczFZr124

Implementing Proof-of-Work in Javascript (Blockchain, part 2)

https://www.youtube.com/watch?v=HneatE69814