

Fin-AI

The global crisis of the Covid-19 pandemic has created widespread disruption. Apart from a crisis in the healthcare sector, there has also been serious economic consequences. Financial institutions, in particular, are exposed to severe economic downturns since a large volume of their business involves lending money which needs to be repaid. Rather than relying on low skilled and poorly paid banking officials to make loan decisions, new technology enables more accurate predictions through the use of AI and appropriate datasets of client information.

This transformation in financial services, coupled with new machine learning models and frameworks have created an opportunity for your start-up software development company to develop applications to address a potential tsunami of loan defaults on new business. Your company has been approached by a US bank and tasked with the development of a bespoke **prototype** system which predicts the likelihood of its customers defaulting on new loans taken out. As part of the negotiations with the company, you have retained the IP and may use it in future projects.

The system consists of the following key features:

FEATURES:

To develop the Fin-AI system, several applications with different functionality are required:

- From the client's perspective, the Fin-AI app enables members to calculate whether or not they will qualify for a loan based on their income and personal situations. Customers may also use the Fin-AI app to determine how much they should be paying for a particular house.
- For financial professionals, a portal website (Fin-AI Web), stores **all** client records. These professionals will also have the same functionality as the client i.e. they can make decisions on loan applications and advise clients on the maximum loan amount which may be taken out. Similar to the clients, financial professionals will also be able to see how much should be paid for a particular house.
- Machine learning models, based on internationally available datasets, will be developed to implement the backend AI (i.e. Fin-AI Engine).

Fin-AI App basic features should include:

- **Register/login of the client:** email and/or social media login option and accounts.
- **Record details of the loan officer** i.e. who is dealing with the loan application
- **Record details of financial institution (bank or broker)**
- **Payment of broker premium (if a financial broker is being used)**
- **House specification:** detailed forms capturing the information required by the Fin-AI Engine
- **Financial History:** detailed forms capturing the information required by the Fin-AI Engine
- **Fin-AI Engine Interface** based on client profile, Fin-AI may be called determining client's loan suitability
- **Request professional** ability to call the loan officer, from within the app
- **Support** forms to contact the bank
- **Ratings & reviews.** Capability of rating the app and leaving reviews of the app's performance

Fin-AI Web basic features should include:

- **Registration / Login** of banking professionals or brokers using email and/or social media login
- **Access Restrictions:** professionals should only be able to see the details of their own clients
- **Fin-AI Engine** allows professionals view their clients' ability to purchase and the type of houses being considered
- **User Profiling** should enable 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 the Fin-AI Engine.

Fin-AI Engine basic features should include:

- Simple API linking Fin-AI Engine backend with the client Fin-Ai app and Fin-Ai Web applications
- A model, using a publicly available dataset, which determines loan suitability given the client's financial profile.
- A model, using a publicly available dataset, which determines the type of house the client can purchase.
- Reporting which displays the accuracy of the loan suitability and house type models
- A chatbot which provides automated customer support on the Fin-AI Web .

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. the Fin-AI App, Fin-AI Web or Fin-AI Engine.

As this is a prototype system, it will be developed using the publicly available Kings County dataset which predicts the prices of property in the Seattle area. Loan suitability is determined using the US loan prediction problem dataset available on Kaggle. On successful completion of the prototype, it is envisaged that Fin-AI will be developed for each state within the US by applying datasets unique to these areas.

Technology:

The first application, the Fin-AI 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 (Fin-AI Web), involves developing a web site for banking and financial brokers. Programming languages for Fin-AI Web may include HTML, Python and PHP. To facilitate rapid development of the site, a Firebase Web App may be developed. In-app payments via credit cards may use Paypal.

Both Fin-AI App and Fin-AI Web will store their data in a centralised database such as Firebase.

The machine learning models used in Fin-AI Engine will be developed using python, Keras and Tensorflow.

Github or BitBucket must be used for code management and marks will be allocated for their use.

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.

Basic concepts of Machine learning (very simple implementation):

<https://machinelearningmastery.com/how-to-run-your-first-classifier-in-weka/>

Simple python machine learning implementation:

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

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/>

Resources on Kings County Datasets (Kaggle notebooks):

<https://www.kaggle.com/harlfoxem/housesalesprediction/notebooks>

Resources on loan prediction (Kaggle notebooks):

<https://www.kaggle.com/altruistdelhite04/loan-prediction-problem-dataset>

Chatbot resources:

<https://data-flair.training/blogs/python-chatbot-project/>