UNIVERSITY OF DAR ES SALAAM

COLLEGE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

IS335 / CS498: Final Year Project Report – End of Semester One

Project Title:	MACHINE LEARNING MODEL FOR DETECTION OF MENTAL HEALTH DISORDERS		
Student Name:	HUGO PHILIP J	2019-04-02512	BSC IN CEIT
Supervisor Name:	MS. LILIAN MKONYI		
Supervisor Signature	………………………………………………………………..		

DECLARATION

I Hugo, Philip Joseph with registration number 2019-04-02512 do hereby declare that this report is my own original work with the views and acknowledged contributions from others which are well cited.

I declare that the work in this report was carried out in accordance with the Rules and Regulations of the University of Dar es Salaam and has not been presented in any other University for examination either within or outside of Tanzania for the award of the same or other degree.

Student's Name: Hugo, Philip J

Signature:………………….

Date: ……………………...

Academic Supervisor's Name: Mrs. Lilian Mkonyi

Signature: ……………….

Date: ……………………...

TABLE OF CONTENTS

Contents

LIST OF ABBREVIATION

UDSM	University Of Dar es saalam
COICT	College Of Information and Communications Technology

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

1.1. General Introduction

Mental health disorders, such as depression, are a major public health concern that can have serious consequences for individuals and society. These conditions can lead to decreased productivity, increased absenteeism from work or school, and an increased risk of other physical and health problems. Despite their importance, mental health disorders can be difficult to detect, especially in individuals who may not seek treatment or who may not be aware that they have a mental health condition. Traditional methods of diagnosis, such as clinical interviews and self-report questionnaires, can be time-consuming and may not always be reliable.

The aim of this project is to develop a machine learning model that can accurately detect mental health disorders, with a focus on depression in individuals. By automating the detection process, it may be possible to identify individuals who are at risk for developing depression more efficiently and accurately than with traditional methods. This could help to improve the quality of life for individuals with mental health disorders and reduce the societal costs associated with these conditions.

1.2. Statement of the Problem

Mental health is a neglected area in health care in Tanzania. With few clinicians and trained researchers in the field, research has been limited both in quantity and quality. Mental health disorders, such as depression, are a major public health concern that can have serious consequences for individuals and society. However, detecting these disorders can be challenging, especially in individuals who may not seek treatment or who may not be aware that they have a mental health condition.

1.3. Objective

1.3.1. Main Objective

The main objective of this project is to develop a machine learning model that can accurately detect mental health disorders, with a focus on depression in individuals.

1.3.2. Specific Objectives

i. To collect and structure dataset.

ii. To develop the model.

iii. To evaluate the performance of the model using metrics such as precision, and accuracy.

iv. To deploy the model in a mobile application.

1.4. Significance of the Project

This project could have significant implications for the detection and treatment of mental health disorders by

- Providing efficient way of performing Mental Disorder Diagnosis
- Creating a mobile app that will be used by health practitioners
- Saving time of carrying out the process manual

Generally it's not about cost reduction but improving the process by saving time and labor that could be used to diagnose mental health disorder.

1.5. Project Scope

This project will focus on developing a machine learning system to detect mental health disorders with a focus on depression. The system will be evaluated on labeled dataset of individuals with and without depression and various algorithms will be tested but the one with the best performance will be used. The dataset of this project is obtained from **https://zindi.africa/competitions/busara-mental-health-prediction-challenge/data** which comes from a 2015 study conducted by the Busara Centre near Lake Victoria. And the created model will be deployed through a mobile application.

1.6. Organization of the report

This report is organized and divided into three main chapters, chapter one which describes and gives a clear statement of the problem, the main and specific objectives of this project, project significance and the project scope.

Chapter two covers literature review which focuses on the existing methods for detecting mental health disorders with a focus on depression. This will include a review of existing machine learning systems for detecting mental health disorders. This chapter will also cover the dataset source to be used on this project.

Chapter three covers the research methodology that is used on the project, **CRISP-DM** which includes business understanding, data understanding, data preparation, modelling, evaluation and lastly deployment.

Chapter four is about the System Analysis. It is composed of Data Collection, Defining the Functional Requirements and use case identification. It also covers system design and analysis on which it comprises of the analysis of the requirement gathered as elaborated in chapter three and the design of the system by providing the system development model, data modeling using different diagrams such as sequence diagram.

CHAPTER TWO

LITERATURE REVIEW

2.1. Introduction

This chapter will provide a detailed review of similar projects to the one that is being proposed. Literature review is an important part in any project as it enables us to conduct research and investigate the existing system or similar projects that have been done around the world as this will enable us to identify the weaknesses of the current projects, this will in turn help us to identify what components can be added to our proposed project in order for it to become more relevant as compared to the ones that are existing currently.

2.2. Previous existing System

Many of the previous existing system in Tanzania relied on traditional approaches to diagnose and come up with treatment of mental health. Traditional approaches to diagnose mental disorder such as clinical interviews and biological markers which are physical or chemical changes in the body. And many of treatments that are given to the clients aren't given by the trained professionals leading to more confusion to the client.

2.3. Disadvantages of existing system

The traditional methods for detecting mental health disorders such as clinical interviews and biological markers can be very subjective and only rely on the individual's ability to accurately report their symptoms and experiences. These methods can be time consuming, not reliable and require trained professionals to administer them.

2.4. Proposed system

Machine learning techniques are a type of artificial intelligence that involve the use of algorithms to analyse large amounts of data and identify patterns. They have the potential to be applied in healthcare to assist with tasks such as diagnosis, treatment planning, and prognosis. Mental health disorders, including depression, are a significant public health issue. According to the World Health Organization (WHO), depression is the leading cause of disability worldwide, and it is estimated that more than 264 million people globally suffer from depression. Previous studies have shown that machine learning techniques can be effective in the detection of mental health disorders, including depression.

So the proposed system will focus on developing a machine learning system to detect mental health disorders with a focus on depression. The system will be evaluated on labeled dataset of individuals with and without depression and various algorithms will be tested but the one with the best performance will be used. Also the created model will be deployed through a mobile application so as to easily link up the client and the trained professional.

The potential benefits of using machine learning for detecting mental health disorders is that machine learning has the potential to revolutionize the field of mental health care by providing a more objective, reliable and accurate means of detecting mental health disorders. Machine learning algorithms can analyse large amounts of data and can identify patterns that may be missed by human analysts.

2.5. Dataset

2.5.1. Dataset source

The dataset was obtained from Zindi, through the following link https://zindi.africa/competitions/busara-mental-health-prediction-challenge/data

The data was collected from 2015 study conducted by a medical Institute, known as Busara found near Lake Victoria. The survey includes more than 70 features including information about household composition, economic activity, financial flows and health. In addition study participants were asked to complete a depression screening tool. The data were hard to collect due to the sensitive content and the need for trained staff and the referral procedures.

2.5.2. About Zindi

Zindi is the first data science competition platform in Africa. Zindi hosts an entire data science ecosystem of scientists, engineers' academics, companies, NGOs, governments and institutions focused on solving Africa's most pressing problems.

Zindi is a social enterprise whose mission is to build the data science ecosystem in Africa. Our vision is for a vibrant community of data scientists across Africa, mobilized towards solving the region's most pressing problems. We are a team for data scientists and creators committed for a better Africa. For this and many other data science projects and competition, they can be found by the following website; https://zindi.africa

2.6. Related Works

There has been a significant amount of research done on detection of mental health disorders over these recent years. Many different types of automated detection systems have been developed including those that use data from the social media posts(Instagram, Facebook) to detect if a person is depressed or not (Liu, 2019). This type of model captures the contents posted on the social media and compares it with its trained data and come up with a conclusion if its mental related post or not.

One study (Zhang, 2019) used a machine learning based approach to predict depression from electronic health records. The electronic health records are the collection of medical information about a person that is stored on a computer including clients health history, medicines etc. Another study (Shah, 2016) just performed a review on the mental health diagnosis using the machine learning techniques.

2.7. Project Gap

The project gap between the one that we are implementing and the ones that have been implemented are as follows:

I. Because in this project machine learning techniques are applied, with the correct amount of data and features the model created on this project will be very accurate which will help on correct detection of the mental disorders.

II. The created model on this project will be deployed through a mobile application so as to not only easily link up the client and the trained professional but also being time effective.

CHAPTER THREE

METHODOLOGY

3.1. CROSS INDUSTRY STANDARD PROCESS FOR DATA MINING (CRISP-DM)

The CRISP-DM is a process model that serves as the base for a data science process. It's the most common methodology for data mining, analytics and data science projects. It has six sequential phases as shown below

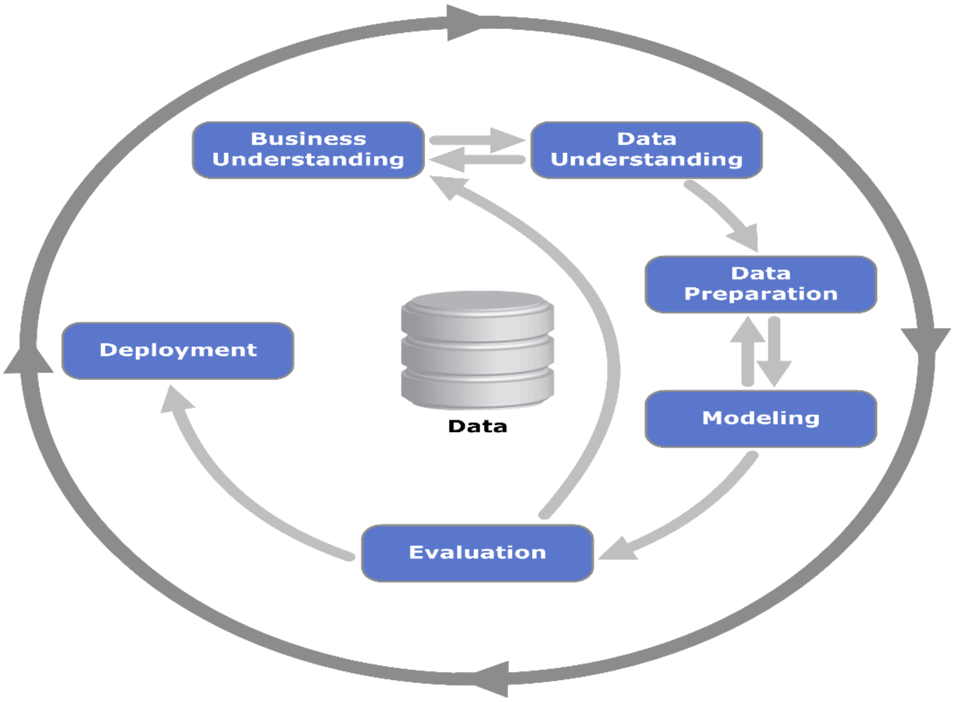


Figure 1 CRISP-DM phases

3.2. Business Understanding

The Business Understanding phase focuses on understanding the objectives and the goals of the project.

3.3. Data Understanding

Next is the Data Understanding phase. Adding to the foundation of Business Understanding, it drives the focus to identify, collect, and analyse the data sets that can help you accomplish the project goals. This phase also has four tasks:

1. **Collect initial data:** The dataset of this project will be obtained from https://zindi.africa/competitions/busara-mental-health-prediction-challenge/data .
2. **Describe data:** Examining the data and document its surface properties like data format, number of records, or field identities.
3. **Explore data:** Digging deeper into the data. Query it, visualize it, and identify relationships among the data.
4. **Verify data quality:** How clean/dirty is the data? Document any quality issues.

3.4. Data Preparation

A common rule of thumb is that 80% of the project is data preparation. This phase, which is often referred to as "data munging", prepares the final dataset for modelling. It has five tasks:

1. **Select data:** Determining which data sets features will be used and document reasons for inclusion/exclusion.
2. **Clean data:** Often this is the lengthiest task. Without it, you'll likely fall victim to garbage-in, garbage-out. A common practice during this task is to correct, impute, or remove erroneous values.
3. **Construct data:** Derive new attributes that will be helpful. For example, derive someone's body mass index from height and weight fields.
4. **Integrate data:** Create new data sets by combining data from multiple sources.
5. **Format data:** Re-format data as necessary. For example, you might convert string values that store numbers to numeric values so that you can perform mathematical operations.

3.5. Modelling

What is widely regarded as data science's most exciting work is also often the shortest phase of the project.

Here you'll likely build and assess various models based on several different modelling techniques. This phase has four tasks:

1. **Select modeling techniques:** Determine which algorithms to try (e.g. regression, neural net).
2. **Generate test design:** Pending your modeling approach, you might need to split the data into training, test, and validation sets.

3. **Build model:** Here is where we build our model for this project.
4. **Assess model:** Generally, multiple models are competing against each other, and the data scientist needs to interpret the model results based on domain knowledge, the pre-defined success criteria, and the test design.

3.6. Evaluation

Whereas the Assess Model task of the Modelling phase focuses on technical model assessment, the Evaluation phase looks more broadly at which model best meets the business and what to do next. This phase has three tasks:

1. **Evaluate results:** Do the models meet the business success criteria? Which one(s) should we approve for the business?
2. **Review process:** Review the work accomplished. Was anything overlooked? Were all steps properly executed? Summarize findings and correct anything if needed.
3. **Determine next steps:** Based on the previous three tasks, determine whether to proceed to deployment, iterate further, or initiate new projects.

3.7. Deployment

Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data mining process across the enterprise. Deployment involves creating an interface in which the user will be able to interact with since the user can’t interact with the model.

There are various ways in which a model can be deployed but in this project I shall deploy the model by creating a mobile application in which the user can be able to interact with the model.

CHAPTER FOUR

SYSTEM ANALYSIS AND DESIGN

System Analysis is the collection of notations, methodologies and tools used to gather details and analyze a problem situation prior to information system design and implementation. System analysis is a stage whose sole purpose is to identify the objectives of the project under study, it involves gathering of information so as to capture the functional and non-functional requirements of the system and, identify actors of the system to be built and develop the corresponding use case diagrams.

System Design this is the next stage after System Analysis, where the design strategy is cleared described, the physical design, architecture design, interface design, database specifications and program design all of which are developed. This stage uses the requirements that were gathered during analysis to create a blueprint for the future system.

4.1. Requirements Gathering

The data of this project was obtained from the official website of Zindi with the following link https://zindi.africa/competitions/busara-mental-health-prediction-challenge/data

The data was collected from 2015 study conducted by a medical Institute, known as Busara found near Lake Victoria. The survey includes more than 70 features including information about household composition, economic activity, financial flows and health. In addition study participants were asked to complete a depression screening tool. The data were hard to collect due to the sensitive content and the need for trained staff and the referral procedures.

4.2. Requirements Specification

A requirement is simply a statement of what the system must do or what characteristics it needs to have. System requirements can be divided into categories which are:

i. Functional requirements
ii. Non-Functional requirements

4.2.1. Functional Requirements

These are the requirements that describe descriptively what the system should do. Table 1 shows clearly all the functional requirements in four categories of its major activities.

CORE ACTIVITIES

i. Management of Clients.
ii. Management of Professionals.
iii. Management of prediction process.
iv. Management of Results.

Ref No		Functional Description	Category
F1		Management of Clients	
	F1.1	The system should allow clients to sign up/register.	Evident
	F1.2	The system should allow clients to login.	Evident
	F1.3	The system should be able to validate clients.	Hidden
	F1.4	The system should allow clients to search for the Professionals	Evident
	F1.5	The system should allow clients to download/ view books	Evident
F2		Management of Professionals	
	F2.1	The system should allow professionals to sign up /register	Evident
	F2.2	The system should allow professionals to login	Evident
	F2.3	The system should be able to validate professionals	Hidden
F3		Management of prediction process	
	F3.1	The system should allow professionals to view the questions.	Evident
	F3.2	The system should allow professionals to input the client's data for prediction.	Evident
	F3.3	The system should be able to calculate the degree of the disorder in percentage.	Hidden

	F3.4	The system should allow professionals to view the predicted result.	Evident
F4		Management of Results	
	F4.1	The system should generate the result from the clients information	Evident

Table 1 Functional Requirements

4.2.2. Non-Functional Requirements

These are the requirements that describe the characteristics of what the system should have. Table 2: shows clearly all the non-functional requirements for this application

S/N	ATTRIBUTE	CONSTRAINTS
1	Usability	The system should be accessible by all android mobile phones and should be user friendly.
2	Accuracy	The system should provide accurate and relevant information.
3	Maintainability	The system should be designed in a way that it can be modified to accommodate the changing requirements.
4	Efficiency	The system should be able to perform its functions accordingly and it should handle all information accurately.
5	Ethical	The system should adhere to the ethical behaviors including not exposing the clients' information publicly.

Table 2 Non-Functional Requirements

4.3. Requirements Analysis

In this stage the requirements are analysed and to generate the respective diagrams showing descriptively all of the required use cases and the system actors, the system Architecture/frame work and the entity relationship diagram from where an effective database schema will be generated.

4.3.1. Use Cases

A use case depicts a set of activities performed to produce some output result. Each use case describes how an external user triggers an event to which the system must respond. An actor refers to a person, another software system or a hardware device that interacts with the system to achieve a useful goal.

SYSTEM ACTORS

ACTORS	DESCRIPTION
Client	This is a user who uses the application to predict his mental state well-being and for consulting the mental health professionals.
Professional	This is the user who uses the application to understand and explain thoughts, emotions, feelings and behaviors of the clients and they use techniques to manage treatment and provide a range of therapies for the mental illness
Administrator	This is a super user who is solely responsible for the maintenance of the system and updates

Table 3 System Actors

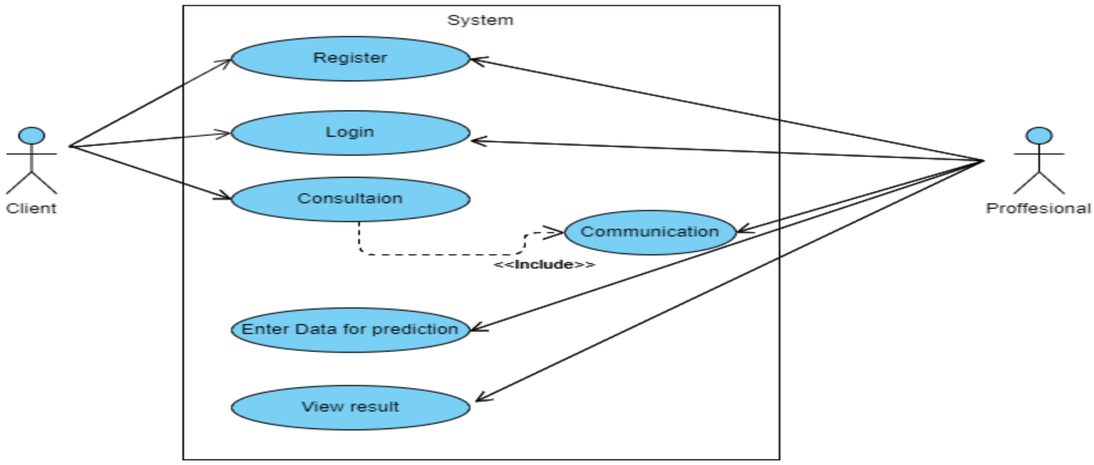


Figure 2 Use case Diagram

USECASE DESCRIPTION

Use case Number	1
Use case Name	Register
Actor	Admin, Client and the Professional
Description	The users interact with the system by registering themselves through a registration form
Pre-condition	User must initially not exist into the system
Post-condition	User needs to verify registration using their filled email address

Table 4 Use case 1 Register

Use case Number	2
Use case Name	Login
Actor	Admin, Client and the Professional
Description	The users must use valid registered credentials to access the system
Pre-condition	User must be registered into the system
Post-condition	User must be able to access the system

Table 5 Use case 2 Login

Use case Number	3
Use case Name	Consultation
Actor	Client and Professional
Description	The client consults the professional seeking for treatment
Pre-condition	The client must be logged on the system
Post-condition	The professional becomes aware of the client

Table 6 Use case 3 Consultation

Use case Number	4
Use case Name	Communication
Actor	Client and Professional
Description	Client and Professional to communicate with each other on how treatment must be done, place, medication etc.
Pre-condition	The client must already consulted the professional
Post-condition	Professional and clients communication becomes possible

Table 7 Use case 4 Communication

Use case Number	5
Use case Name	Prediction
Actor	Professional
Description	The professional enters clients data so the model could predict the mental state
Pre-condition	Client must have already consulted the professional
Post-condition	The model should provide the results of the prediction in terms of percentage.

Table 8 Use case 5 Prediction

Use case Number	6
Use case Name	View Result
Actor	Professional
Description	The model should be able to provide results for prediction
Pre-condition	Client must initially enter his/ her data into the systems modelfor prediction
Post-condition	Professional being able to view the result of their clients predicted mental health state.

Table 9 Use case View result

SYSTEM DESIGN

4.4. Logical Design

Logical design shows the logical organization of data without indicating how data is stored, created or manipulated. This project utilizes the ERD as a means of logical design as depicted in figure 3. Entity relation diagram is a picture which shows the information that is created, stored, and used by a business system.

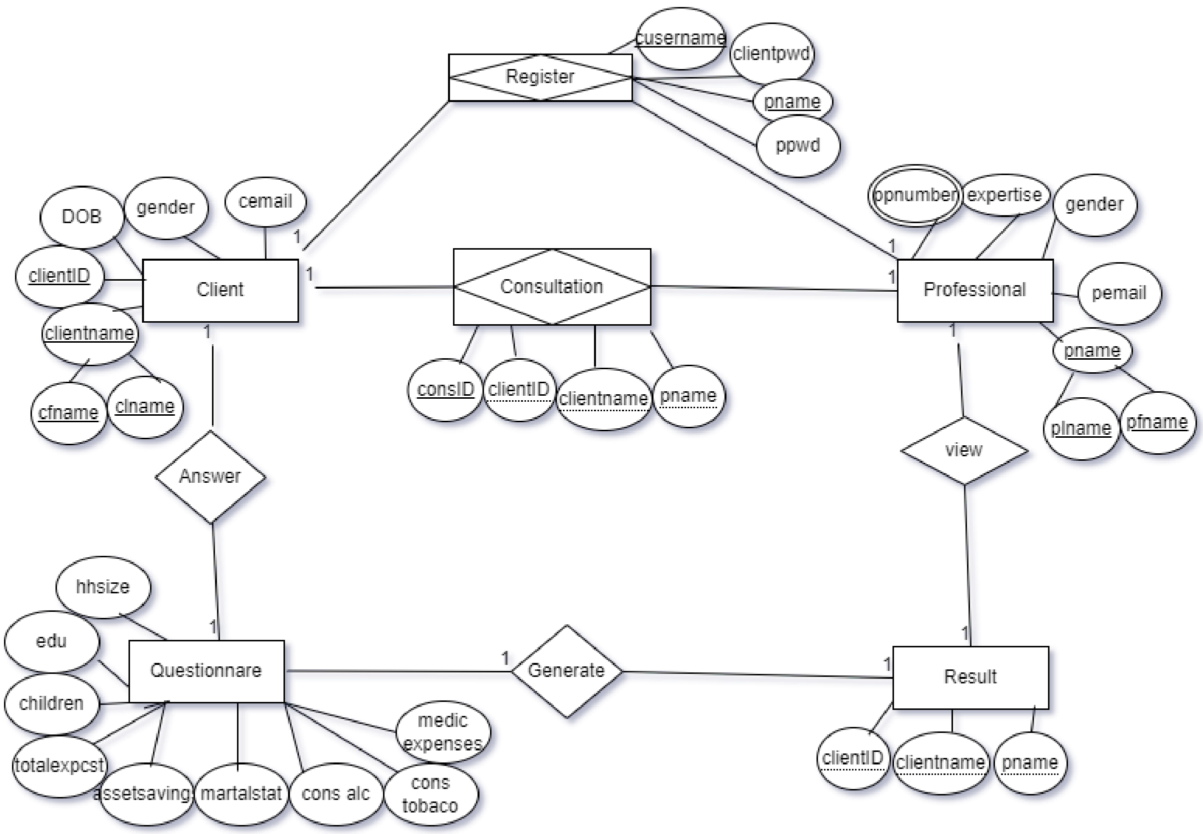


Figure 3 Entity Relationship Diagram

4.5. Sequence Diagram

Sequence diagram is the interaction diagram that shows the time ordering of the messages. These are use case driven diagrams which maps the message passing to the object. It describes the activities that people do. They are developed as the as is system and/or the to-be system. Fig 4 represent the activities done before being logged on the system.

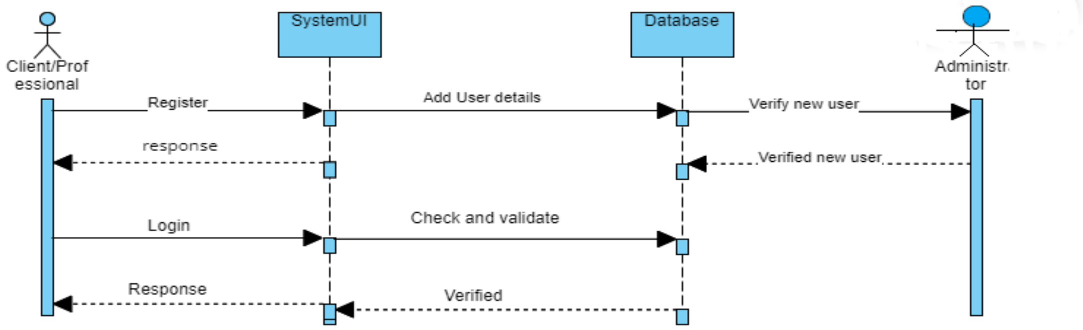
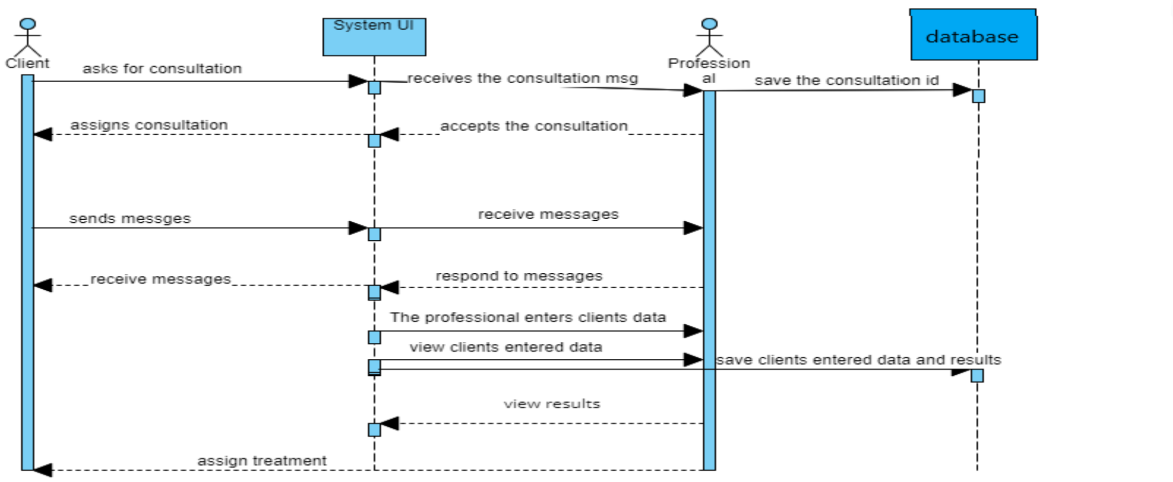


Figure 4 Client/Professional with Administrator sequence diagram

Below is a sequence diagram that contains the activities done by the client and professional after being logged on the system.

Figure 5 Client and Professional sequence diagram



4.6. Model Framework

Model Framework is the essential supporting structure of our model that is being created on the project. On this project after acquiring the data, I divided it into training and testing dataset. The training dataset is responsible for training/ teaching the model that we are creating while the testing dataset is responsible for checking the accuracy of the created model. The training dataset will be used in many classification algorithms but the one with the highest performance out of all will be one used. After creating of the model we test the accuracy by using the testing dataset. If the accuracy is satisfactory then it will be used to predict the data from the clients (production data). These can be summarised by the following below diagram

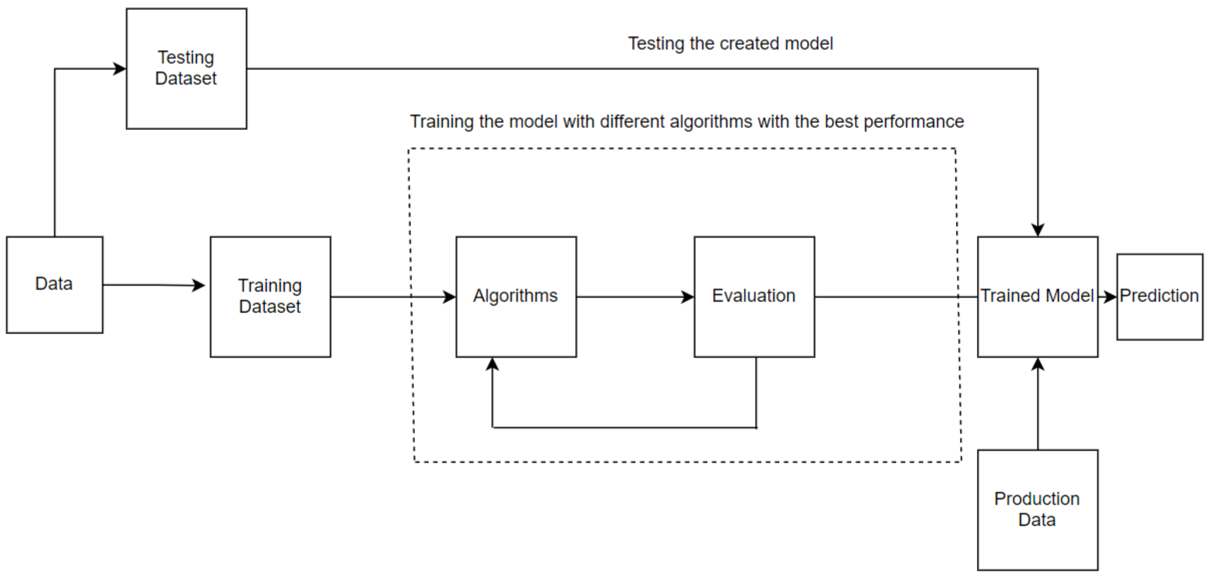


Figure 6 Model Framework

CHAPTER FIVE

IMPLEMENTATION AND TESTING.

The implementation of this project is entirely divided into three parts, creating of the machine learning model, the database and lastly the mobile application. Starting with the most important part, the following were the steps that I took on creating the machine learning model.

5.1 Creating of the machine learning model

5.1.1. Data Sources

The dataset of this project was obtained from Zindi, whereby Zindi is the largest professional network for data scientists in Africa. [https//zindi.africa/competitions/busara-mental-health-prediction-challenge/ data](https//zindi.africa/competitions/busara-mental-health-prediction-challenge/data)

The data was collected from a medical Institute, known as Busara. It contains 75 columns and 1143 rows. The target, depressed was included in the dataset leaving 74 features which will help suggest if an individual is depressed or not.

5.1.2. Loading and viewing our dataset.

After obtaining our dataset, the next step was to load and view the dataset so as to verify the number of the rows and columns as it was stated from the data source.

Load the dataset

In [2]:
```
# Import Data
data = pd.read_csv('train.csv')
```

In [3]:
```
data
```

Out[3]:

	surveyid	village	survey_date	femaleres	age	married	children	hhsize	edu	hh_children	...	given_mpesa	amount_given_mpesa	received_mpesa
0	926	91	23-Nov-61	1	28.0	1	4	6	10	0	...	0	0.0	0
1	747	57	24-Oct-61	1	23.0	1	3	5	8	0	...	0	0.0	1
2	1190	115	05-Oct-61	1	22.0	1	3	5	9	0	...	0	0.0	0
3	1065	97	23-Sep-61	1	27.0	1	2	4	10	2	...	0	0.0	0
4	806	42	12-Sep-61	0	59.0	0	4	6	10	4	...	0	0.0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1138	927	152	27-Dec-61	1	36.0	0	4	5	9	0	...	0	0.0	0
1139	1039	104	13-Sep-61	1	23.0	1	0	2	10	0	...	0	0.0	0
1140	428	54	23-Nov-61	1	28.0	1	5	7	10	5	...	0	0.0	0
1141	23	3	05-Oct-61	1	33.0	0	4	5	9	0	...	0	0.0	0
1142	116	18	24-Sep-61	1	39.0	1	0	4	12	0	...	0	0.0	0

1143 rows × 75 columns

Figure 7 loading the dataset

5.1.3. Exploratory Data Analysis

It is a good practice to understand the dataset first, so as to gather as many insights from it. This will help

- To identify most important variables in the dataset.
- To test hypothesis related to the dataset.
- To verify expected relationships from the dataset.

From the dataset, the following hypothesis were generated

1. Married people are likely to suffer from depression.
2. Female respondents are likely to suffer from depression.
3. Children are unlikely to suffer from depression.
4. Most people who do not have savings are likely to be depressed.
5. The most age group to be surveyed is youth.
6. Females were more depressed than males.
7. Most people surveyed are depressed.

Exploratory Data Analysis is classified into:

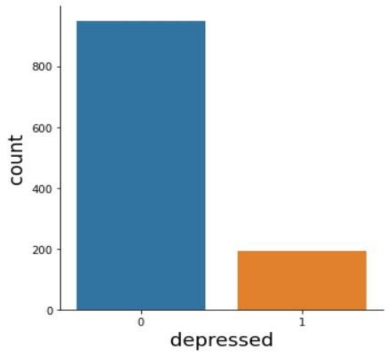
- Univariate Analysis
- Bivariate Analysis

5.1.4. Univariate Analysis.

Here, the visualization is done with each feature individually. Python packages such as seaborn and matplotlib are used to plot the graphs. For categorical features, bar plots were used while for numerical features we used histograms.

The distribution exploration starts with the target feature:

```
In [9]:  # explore target distribution
         sns.catplot(x="depressed", kind="count", data= data)
Out[9]: <seaborn.axisgrid.FacetGrid at 0x20f50c86160>
```



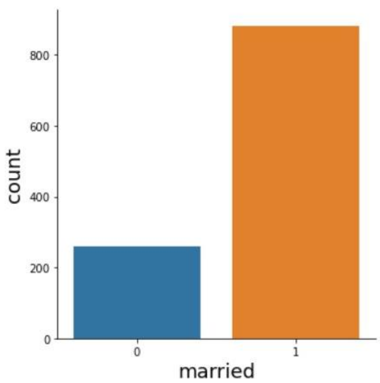
where: 1- number of people depressed 0- number of people not depressed

count -number of surved people

Most people surved are not depressed

Figure 8 Target Distribution

```
In [10]:  # explore married distribution
          sns.catplot(x="married", kind="count", data=data)
Out[10]: <seaborn.axisgrid.FacetGrid at 0x20f3c1e8730>
```



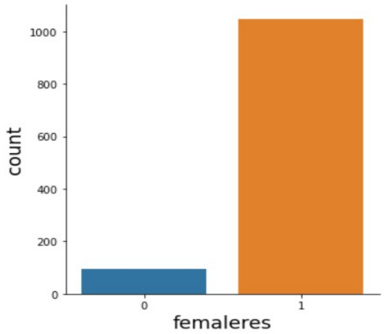
where: 1- number of people married 0- number of people not married

count -number of surved people

Most married people were surveyed.

Figure 9 Married distribution

```
In [12]:  # Explore femaleres distribution
          sns.catplot(x="femaleres", kind="count", data=data)
Out[12]: <seaborn.axisgrid.FacetGrid at 0x20f53fefe20>
```



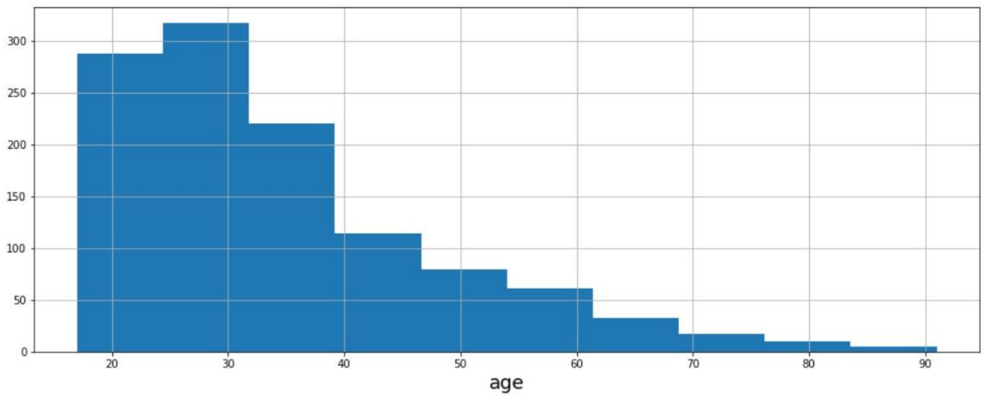
where: 1- number of female respondents 0- number of people not male respondents

```
count -number of surved people
```

Females responded more than males.

Figure 10 Female Distribution

```
In [25]:  # age distribution
          plt.figure(figsize=(16, 6))
          data.age.hist()
          plt.xlabel('age')
Out[25]: Text(0.5, 0, 'age')
```



Most of the age-group surveyed is between 25 to 32 years.

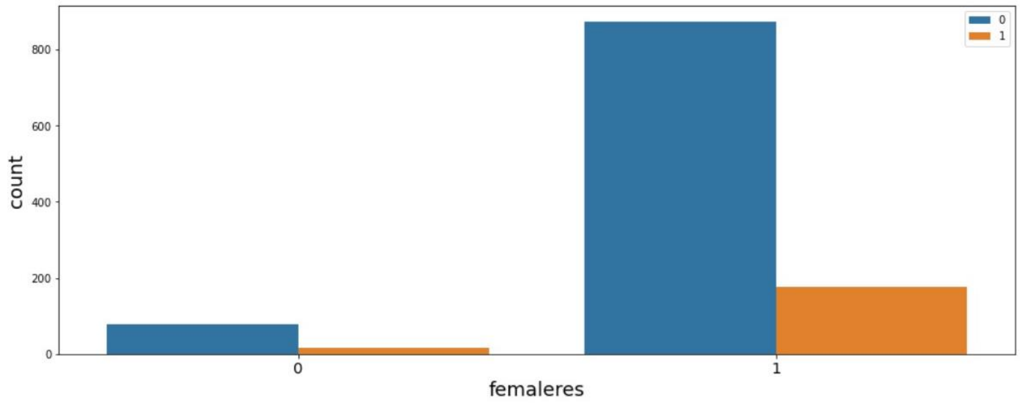
Figure 11 Age distribution

5.1.5. BIVARIATE ANALYSIS

Here, we visualize the relationship of each feature with the target feature.

```
    fontsize='x-large'
)
plt.legend()
```

Out[28]: <matplotlib.legend.Legend at 0x20f541ad3a0>



where: 0-maleres

1-femaleres

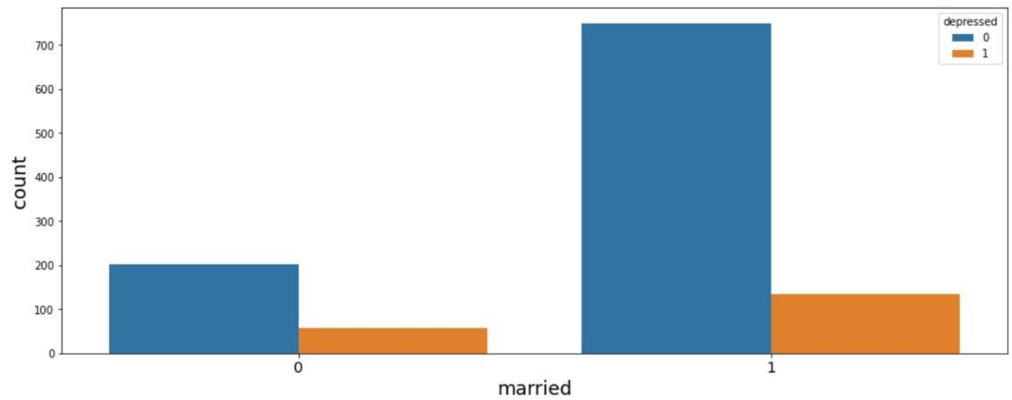
Females were more depressed than males.

Females who were not depressed were more than males.

Figure 12 Females vs Depressed Distribution

```
    fontweight='light',
    fontsize='x-large'
)
```

Out[29]: (array([0, 1]), [Text(0, 0, '0'), Text(1, 0, '1')])



where: 0-not married

1-married

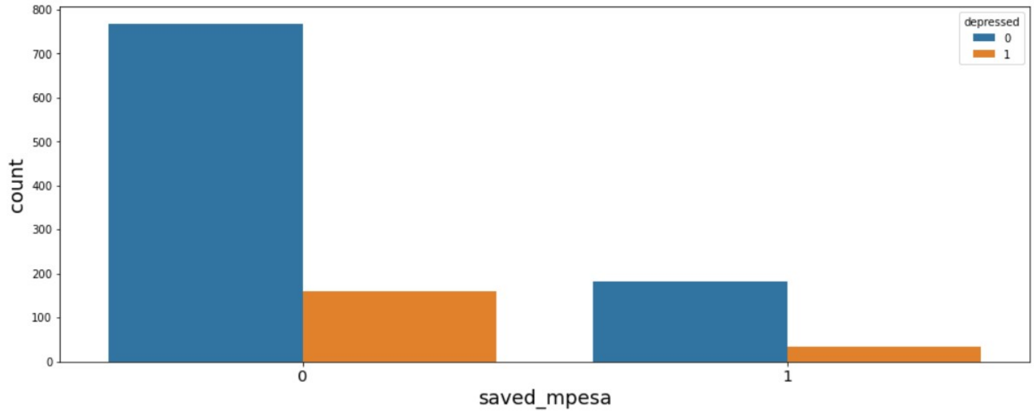
Married are more depressed than not married.

Married who were not depressed were more than not married.

Figure 13 Married vs Depressed Distribution

```
sns.countplot(x='saved_mpesa', hue= 'depressed', data=data)
plt.xticks(
    fontweight='light',
    fontsize='x-large'
)
```

Out[32]: (array([0, 1]), [Text(0, 0, '0'), Text(1, 0, '1')])



where: 0-no saved_mpesa

1-saved_mpesa

Most people who were depressed had no saved_mpesa.

Figure 14 Savings vs Depressed Distribution

1 .Married people are likely to suffer from depression-True

2 .Female respondents are likely to suffer from depression-True

3 .Children are unlikely to suffer from depression-False

4 .Most people who do not have savings are likely to be depressed-True

5 .Most of the age group interviewed were youth-True

6 .Females were more depressed than males-True

7 .Most people surveyed are depressed- False

5.1.6. Cleaning of the Dataset

Data cleaning involves a process of making sure the dataset has no missing values, handling duplicates and unique values that have no meaning in the dataset.

- Checking for missing values on the dataset if present

```
In [4]:  # checking missing values
         data.isnull().sum().sum()

Out[4]:  10262
```

Figure 15 Checking for missing values

- Dropping of the columns that were 60% empty and the ones that were not useful on the dataset and also checking the new amount of the missing values.

```
In [7]:  #remove unique features
         data = data.drop('surveyid', axis=1)
         data = data.drop('village',axis = 1)
         data = data.drop('survey_date',axis = 1)
         data = data.drop('day_of_week',axis = 1)
         data = data.drop('early_survey',axis = 1)
         data = data.drop('med_u5_deaths',axis = 1)
         data = data.drop('med_child_check', axis = 1)
         data = data.drop('med_vacc_newborns', axis = 1)
         data = data.drop('med_expenses_sp_ep',axis = 1) #dropping because more than 60% of column is empty
         data = data.drop('med_expenses_hh_ep',axis = 1) #dropping because more than 60% of column is empty

         data.shape

Out[7]:  (1143, 66)

In [8]:  # remaining missing values
         data.isnull().sum().sum()

Out[8]:  7607
```

Figure 16 Dropping of columns

- Handling of missing values

The dataset had a mixture of categorical and numerical features. Handling of these features is different from one another. Categorical features demands the use of frequency to fill the missing values while Numerical features demands the use of either mean or median to fill the missing values.

```
In [42]:  # Handling missing values
          data['hh_totalmembers'].fillna(data['hh_totalmembers'].value_counts().idxmax(), inplace = True)
          data['cons_alcohol'].fillna(data['cons_alcohol'].value_counts().idxmax(), inplace = True)
          data['cons_tobacco'].fillna(data['cons_tobacco'].value_counts().idxmax(), inplace = True)
          data['cons_med_total'].fillna(data['cons_med_total'].value_counts().idxmax(), inplace = True)
          data['fs_chwholed_often'].fillna(data['fs_chwholed_often'].value_counts().idxmax(), inplace = True)
          data['fs_meat'].fillna(data['fs_meat'].value_counts().idxmax(), inplace = True)
          data['fs_enoughtom'].fillna(data['fs_enoughtom'].value_counts().idxmax(), inplace = True)
          data['fs_sleephun'].fillna(data['fs_sleephun'].value_counts().idxmax(), inplace = True)
          data['fs_chskipm_often'].fillna(data['fs_chskipm_often'].mean(), inplace = True)
          data['med_expenses_child_ep'].fillna(data['med_expenses_child_ep'].mean(), inplace = True)
          data['med_portion_sickinjured'].fillna(data['med_portion_sickinjured'].mean(), inplace = True)
          data['med_port_sick_child'].fillna(data['med_port_sick_child'].mean(), inplace = True)
          data['med_afford_port'].fillna(data['med_afford_port'].mean(), inplace = True)
          data['med_sickdays_hhave'].fillna(data['med_sickdays_hhave'].mean(), inplace = True)
          data['med_healthconsult'].fillna(data['med_healthconsult'].mean(), inplace = True)
          data['ed_expenses'].fillna(data['ed_expenses'].mean(), inplace = True)
          data['ed_expenses_perkid'].fillna(data['ed_expenses_perkid'].mean(), inplace = True)
          data['ed_sch_missedpc'].fillna(data['ed_sch_missedpc'].mean(), inplace = True)
          data['ed_work_act_pc'].fillna(data['ed_work_act_pc'].mean(), inplace = True)
          data['cons_med_children'].fillna(data['cons_med_children'].mean(), inplace = True)
          data['ed_schoolattend'].fillna(data['ed_schoolattend'].mean(), inplace = True)

          # viewing missing values
          data.isnull().sum().sum()

Out[42]:  0
```

Figure 17 Handling missing values

5.1.7. Feature Engineering

The main objective of feature engineering is to prepare a proper input dataset, compatible with the machine learning algorithm requirements.

Depending on the type of data (Categorical and Numerical), each one has its method for performing Feature engineering. Starting with Numerical data,

```
In [12]: # import MinMaxScaler
         from sklearn.preprocessing import MinMaxScaler
```

```
In [13]:  #feature scalling by using minmaxscaler method
          # converting numerical data into a range
         scaler = MinMaxScaler(feature_range=(0, 1))

         feat_col =[ 'age', 'children', 'hhsize', 'edu',
                'hh_children', 'hh_totalmembers', 'cons_nondurable', 'asset_livestock',
                'asset_durable', 'asset_phone', 'asset_savings',
                'asset_land_owned_total', 'cons_allfood',
                'cons_ownfood', 'cons_alcohol', 'cons_tobacco', 'cons_med_total',
                'cons_med_children', 'cons_ed', 'cons_social', 'cons_other',
                'ent_nonag_revenue', 'ent_nonag_flowcost',
                'ent_farmrevenue', 'ent_farmexpenses', 'ent_animalstockrev',
                'ent_total_cost', 'fs_adskipm_often', 'fs_adwholed_often',
                'fs_chskipm_often', 'fs_chwholed_often', 'fs_meat', 'med_expenses_child_ep', 'med_portion_sickinjured',
                'med_port_sick_child', 'med_afford_port', 'med_sickdays_hhave',
                'med_healthconsult', 'wage_expenditures', 'ent_employees',
                'ed_expenses', 'ed_expenses_perkid', 'ed_schoolattend',
                'ed_sch_missedpc', 'ed_work_act_pc', 'durable_investment', 'nondurable_investment',
                 'amount_given_mpesa',
                'amount_received_mpesa', 'net_mpesa',
                'amount_saved_mpesa']

         for col in feat_col:
           data[col] = scaler.fit_transform(data[col].values.reshape(-1,1))
```

Figure 18 Feature engineering numerical data

- Conversion of categorical data using label encoder method shown below

```
In [16]: # import preprocessing module
         from sklearn.preprocessing import LabelEncoder
```

```
In [17]: # converting categorical data
         le = LabelEncoder()

         data['femaleres'] = le.fit_transform(data['femaleres'])
         data['married'] = le.fit_transform(data['married'])
         data['asset_niceroof'] = le.fit_transform(data['asset_niceroof'])
         data['ent_wagelabor'] = le.fit_transform(data['ent_wagelabor'])
         data['ent_ownfarm'] = le.fit_transform(data['ent_ownfarm'])
         data['ent_business'] = le.fit_transform(data['ent_business'])
         data['ent_nonagbusiness'] = le.fit_transform(data['ent_nonagbusiness'])
         data['fs_enoughtom'] = le.fit_transform(data['fs_enoughtom'])
         data['labor_primary'] = le.fit_transform(data['labor_primary'])
         data['given_mpesa'] = le.fit_transform(data['given_mpesa'])
         data['received_mpesa'] = le.fit_transform(data['received_mpesa'])
         data['saved_mpesa'] = le.fit_transform(data['saved_mpesa'])
         data['day_of_week'] = le.fit_transform(data['day_of_week'])
         data['depressed'] = le.fit_transform(data['depressed'])
         #data['med_vacc_newborns'] = le.fit_transform(data['med_vacc_newborns'])
         #data['med_child_check'] = le.fit_transform(data['med_child_check'])
         data

  Out[17]:
```

Figure 19 Feature engineering categorical data

- The dataset before performing feature engineering

```
In [11]: data
```

Out[11]:

	femaleres	age	married	children	hhsize	edu	hh_children	hh_totalmembers	cons_nondurable	asset_livestock	...	durable_investment	nondurable
0	1	28.0	1	4	6	10	0	4.0	0.000000	0.000000	...	0.00000	
1	1	23.0	1	3	5	8	0	4.0	0.000000	0.000000	...	0.00000	
2	1	22.0	1	3	5	9	0	4.0	0.000000	0.000000	...	0.00000	
3	1	27.0	1	2	4	10	2	4.0	378.832920	526.671080	...	778.11230	
4	0	59.0	0	4	6	10	4	6.0	258.671880	8.260629	...	201.00562	
...	...	...	...	...	...	...	...	...	...	...	...	...	
1138	1	36.0	0	4	5	9	0	4.0	0.000000	0.000000	...	0.00000	
1139	1	23.0	1	0	2	10	0	2.0	93.073326	8.260629	...	367.00491	
1140	1	28.0	1	5	7	10	5	7.0	387.758820	8.260629	...	490.32327	
1141	1	33.0	0	4	5	9	0	4.0	0.000000	0.000000	...	0.00000	
1142	1	39.0	1	0	4	12	0	4.0	178.984730	111.518490	...	612.79956	

1143 rows × 65 columns

Figure 20 Dataset before feature engineering

- The dataset after performing feature engineering

Out[17]:

	femaleres	age	married	children	hhsize	edu	hh_children	hh_totalmembers	cons_nondurable	asset_livestock	...	nondurable_investme
0	1	0.148649	1	0.4	0.454545	0.500000	0.0	0.272727	0.000000	0.000000	...	0.0000
1	1	0.081081	1	0.3	0.363636	0.388889	0.0	0.272727	0.000000	0.000000	...	0.0000
2	1	0.067568	1	0.3	0.363636	0.444444	0.0	0.272727	0.000000	0.000000	...	0.0000
3	1	0.135135	1	0.2	0.272727	0.500000	0.2	0.272727	0.264620	0.191202	...	0.0304
4	0	0.567568	0	0.4	0.454545	0.500000	0.4	0.454545	0.180686	0.002999	...	0.0019
...	...	...	...	...	...	...	...	...	...	...	...	
1138	1	0.256757	0	0.4	0.363636	0.444444	0.0	0.272727	0.000000	0.000000	...	0.0000
1139	1	0.081081	1	0.0	0.090909	0.500000	0.0	0.090909	0.065013	0.002999	...	0.0000
1140	1	0.148649	1	0.5	0.545455	0.500000	0.5	0.545455	0.270855	0.002999	...	0.0650
1141	1	0.216216	0	0.4	0.363636	0.444444	0.0	0.272727	0.000000	0.000000	...	0.0000
1142	1	0.297297	1	0.0	0.272727	0.611111	0.0	0.272727	0.125023	0.040485	...	0.0194

1143 rows × 66 columns

Figure 21 Dataset after feature engineering

5.1.8. Feature Selection

The aim of feature selection is to realize the important features that contribute most to the prediction variable which we are most interested with. Hence, we would be able to filter the non-important features and train our model faster with a good accuracy.

It involves three approaches;

- SelectKBest
- Extra Trees Classifier
- Correlation

The following are the best features using SelectKBest method

```
#train to find best features
fit = bestfeatures.fit(features,target)

#save in the dataframe
dfscores = pd.DataFrame(fit.scores_)
dfcolumns = pd.DataFrame(features.columns)

#concat two dataframes for better visualization
featureScores = pd.concat([dfcolumns,dfscores],axis=1)

#naming the dataframe columns
featureScores.columns = ['Specs','Score']

#print 20 best features
print(featureScores.nlargest(18,'Score'))
```

```
                    Specs     Score
36       fs_adwholed_often  8.689845
38       fs_chwholed_often  1.857910
1                      age  1.667093
2                  married  1.603855
5                      edu  1.007411
17            cons_alcohol  0.761584
37        fs_chskipm_often  0.497763
49      ed_expenses_perkid  0.461801
26            ent_business  0.444182
57            given_mpesa   0.427641
30      ent_nonag_flowcost  0.409831
40            fs_enoughtom  0.389946
34          ent_total_cost  0.376028
64             day_of_week  0.374598
11             asset_phone  0.329214
56   nondurable_investment  0.323834
28           ent_employees  0.315866
46       med_sickdays_hhave  0.307217
```

Figure 22 SelectKBest features

The following are the best features using Extra Trees Classifier method

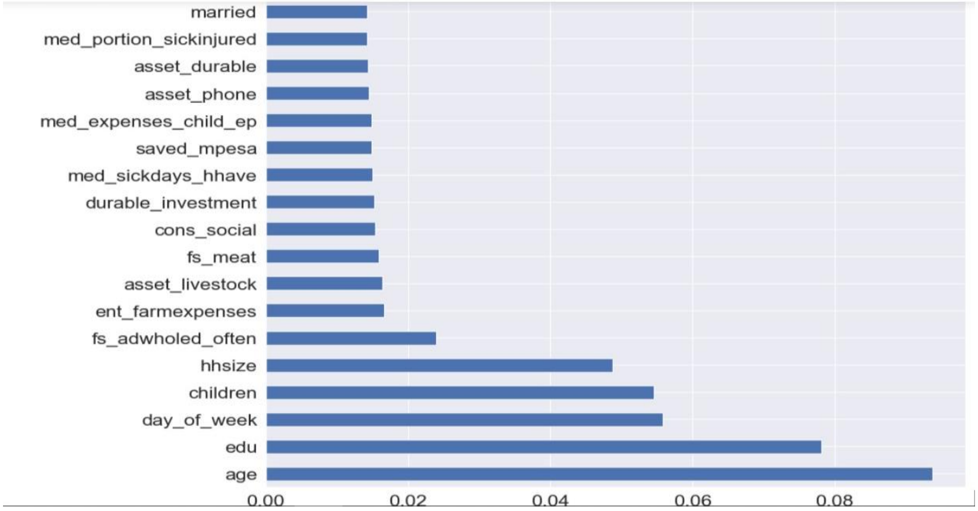


Figure 23 Extra Trees features

The following are the best features using Correlation between target feature and other features

```
# Create a new DataFrame with the top 20 features and their correlation values
features_corr = pd.DataFrame({'feature': top_features.index, 'corr': top_features.values})

print(features_corr)

                     feature      corr
0                  depressed  1.000000
1           fs_adwholed_often  0.143458
2                        edu  0.122236
3                        age  0.100159
4                    married  0.078541
5           fs_chwholed_often  0.070507
6    med_portion_sickinjured  0.058253
7          ed_expenses_perkid  0.048736
8              asset_durable  0.047191
9           ed_schoolattend  0.045790
10          hh_totalmembers  0.045394
11         med_sickdays_hhave  0.045199
12              cons_alcohol  0.042513
13           ed_work_act_pc  0.040210
14         med_healthconsult  0.038582
15          fs_chskipm_often  0.037803
16            ent_total_cost  0.037660
17               asset_phone  0.037200
18         ent_nonag_flowcost  0.035660
19     nondurable_investment  0.035176
```

Figure 24 Correlation features

So from the above pictures the following columns were the ones I used on the training of the m odel

S/N	NAME OF THE COLUMN	NUMBER OF ROWS	DATA TYPE
0	femaleres	1143	non-null int64
1	age	1143	non-null float64
2	married	1143	non-null int64
3	edu	1143	non-null float64
4	hh_totalmembers	1143	non-null float64
5	asset_durable	1143	non-null float64
6	asset_phone	1143	non-null float64
7	cons_alcohol	1143	non-null float64
8	ent_business	1143	non-null int64
9	ent_employees	1143	non-null float64
10	ent_nonag_flowcost	1143	non-null float64
11	ent_total_cost	1143	non-null float64
12	fs_adwholed_often	1143	non-null float64
13	fs_chskipm_often	1143	non-null float64
14	fs_chwholed_often	1143	non-null float64
15	fs_enoughtom	1143	non-null int64
16	med_portion_sickinjured	1143	non-null float64
17	med_healthconsult	1143	non-null float64
18	ed_expenses_perkid	1143	non-null float64
19	nondurable_investment	1143	non-null float64
20	given_mpesa	1143	non-null int64
21	amount_saved_mpesa	1143	non-null float64
22	depressed	1143	non-null int64

Table 10 Used features

5.1.9. Model Creation and Training

The most important part of this project lied on this part because we had to create an accurate machine learning model through using of different algorithms and monitoring their performances. The one that had the better performing performance was the one that was used on the prediction.

The following is a table showing the algorithms used, the number of columns which were used, and their respective performances;

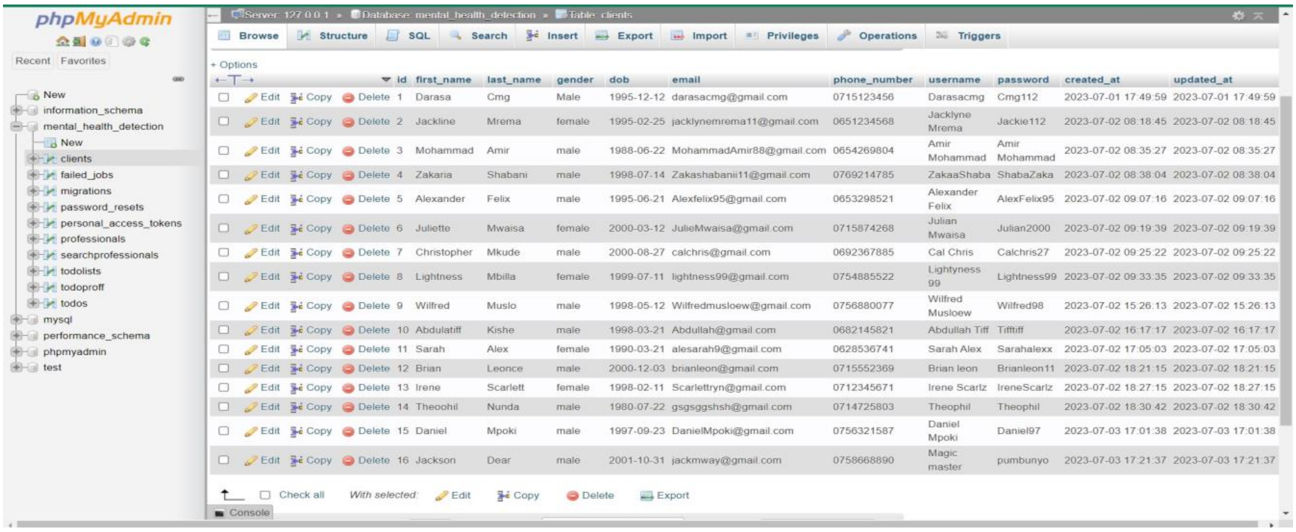
S/N	Algorithm Used	Nature of the Dataset (Number of features)	Performance (Accuracy %)
1	Random Forest Classifier	22	83.16%
2	XGBoost	22	83.53%
3	Logistic Regression	22	57.68%
4	CatBoost	22	82.11%
5	Voting Classifier	22	60.21%
6	**Histogram Gradient Boosting**	22	**84.42%**

Table 11 Algorithms and their performances

From the above table, the Histogram Gradient Boosting had the best performance and it was the one that was used on prediction.

5.1.10. Database Creation

The Database that was created and used on this project was MySQL database and laravel was used to connect the database and the front end of the mobile application.



Server: 127.0.0.1 » Database: mental_health_detection » Table: clients

id	first_name	last_name	gender	dob	email	phone_number	username	password	created_at	updated_at
1	Darasa	Cmg	Male	1995-12-12	darasacmg@gmail.com	0715123456	Darasacmg	Cmg112	2023-07-01 17:49:59	2023-07-01 17:49:59
2	Jackline	Mrema	female	1995-02-25	jacklynemrema11@gmail.com	0651234568	Jacklyne Mrema	Jackie112	2023-07-02 08:18:45	2023-07-02 08:18:45
3	Mohammad	Amir	male	1988-06-22	MohammadAmir88@gmail.com	0654269804	Amir Mohammad	Amir Mohammad	2023-07-02 08:35:27	2023-07-02 08:35:27
4	Zakaria	Shabani	male	1998-07-14	Zakashabanii11@gmail.com	0769214785	ZakaaShaba	ShabaZaka	2023-07-02 08:38:04	2023-07-02 08:38:04
5	Alexander	Felix	male	1995-06-21	Alexfelix95@gmail.com	0653298521	Alexander Felix	AlexFelix95	2023-07-02 09:07:16	2023-07-02 09:07:16
6	Juliette	Mwaisa	female	2000-03-12	JulieMwaisa@gmail.com	0715874268	Julian Mwaisa	Julian2000	2023-07-02 09:19:39	2023-07-02 09:19:39
7	Christopher	Mkude	male	2000-08-27	calchris@gmail.com	0692367885	Cal Chris	Calchris27	2023-07-02 09:25:22	2023-07-02 09:25:22
8	Lightness	Mbilla	female	1999-07-11	lightness99@gmail.com	0754885522	Lightyness 99	Lightness99	2023-07-02 09:33:35	2023-07-02 09:33:35
9	Wilfred	Muslo	male	1998-05-12	Wilfredmusloew@gmail.com	0756880077	Wilfred Musloew	Wilfred98	2023-07-02 15:26:13	2023-07-02 15:26:13
10	Abdulatiff	Kishe	male	1998-03-21	Abdullah@gmail.com	0682145821	Abdullah Tiff	Tifftiff	2023-07-02 16:17:17	2023-07-02 16:17:17
11	Sarah	Alex	female	1990-03-21	alesarah9@gmail.com	0628536741	Sarah Alex	Sarahalexx	2023-07-02 17:05:03	2023-07-02 17:05:03
12	Brian	Leonce	male	2000-12-03	brianleon@gmail.com	0715552369	Brian leon	Brianleon11	2023-07-02 18:21:15	2023-07-02 18:21:15
13	Irene	Scarlett	female	1998-02-11	Scarlettryn@gmail.com	0712345671	Irene Scarlz	IreneScarlz	2023-07-02 18:27:15	2023-07-02 18:27:15
14	Theoohil	Nunda	male	1980-07-22	gsgsggshsh@gmail.com	0714725803	Theophil	Theophil	2023-07-02 18:30:42	2023-07-02 18:30:42
15	Daniel	Mpoki	male	1997-09-23	DanielMpoki@gmail.com	0756321587	Daniel Mpoki	Daniel97	2023-07-03 17:01:38	2023-07-03 17:01:38
16	Jackson	Dear	male	2001-10-31	jackmway@gmail.com	0758668890	Magic master	pumbunyo	2023-07-03 17:21:37	2023-07-03 17:21:37

Figure 25 Database

5.1.11. Creating the Mobile app User Interfaces

As per to the requirements, the mobile app was created through react native JavaScript (programming language) and the following pictures represent the interfaces of the app

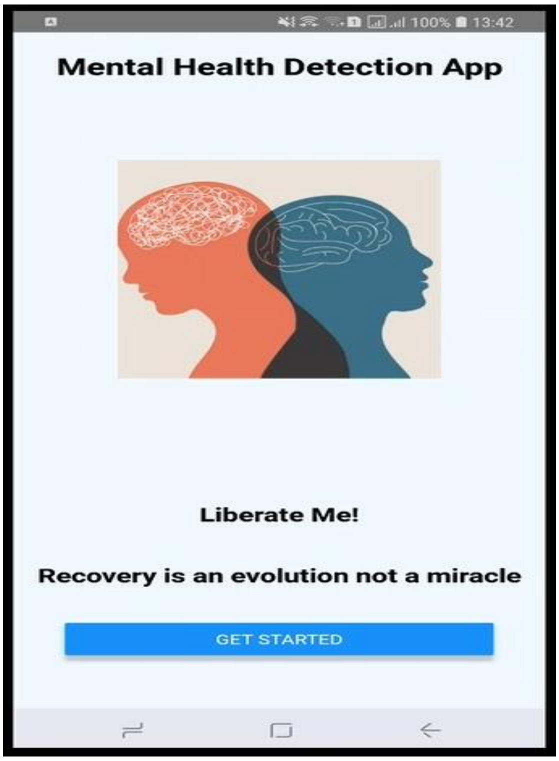


Figure 26 Getting started

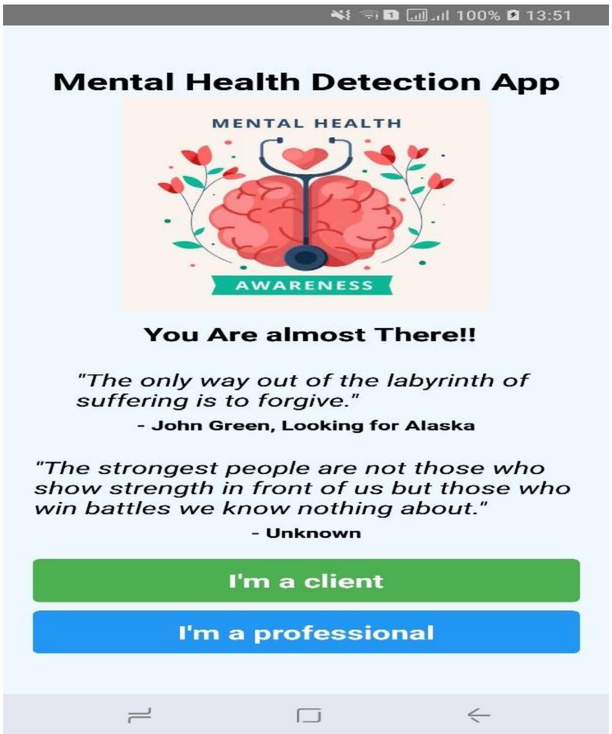


Figure 27 Choosing user

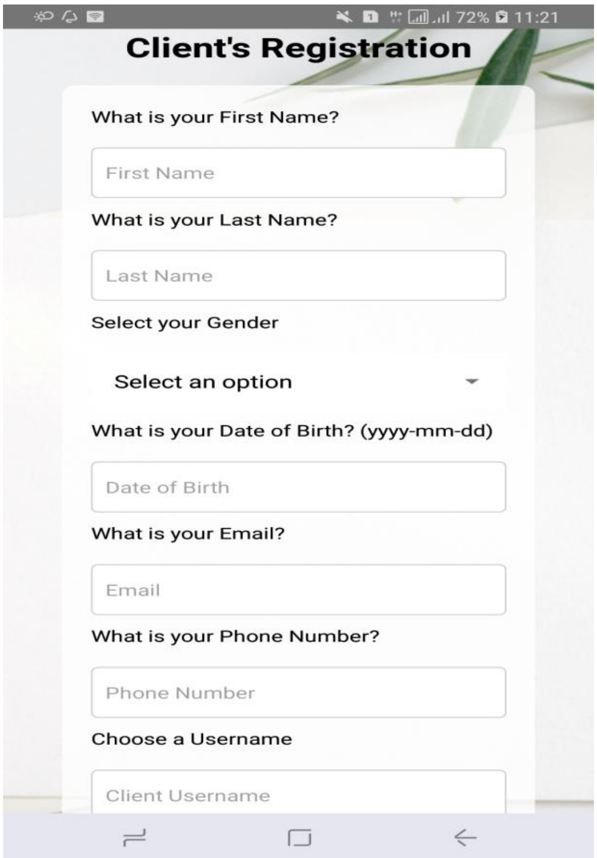


Figure 28 Client Registration

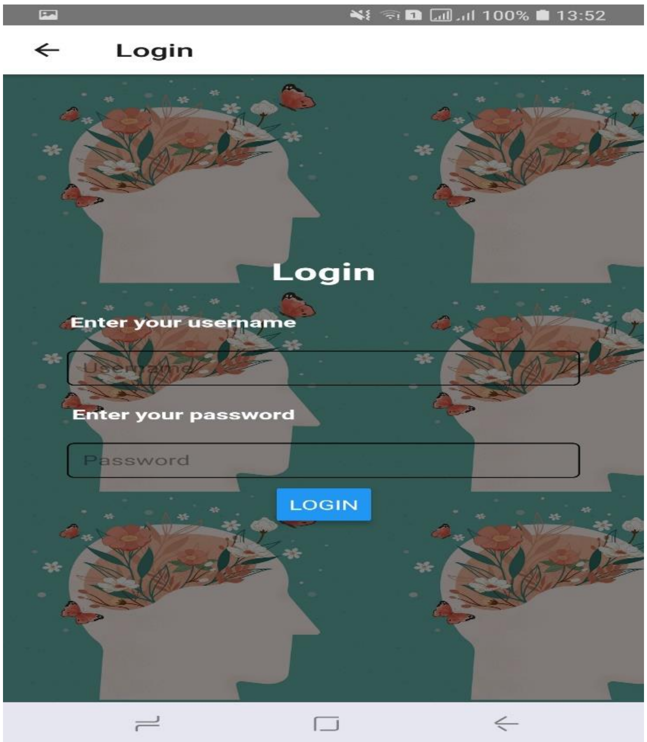


Figure 29 Login

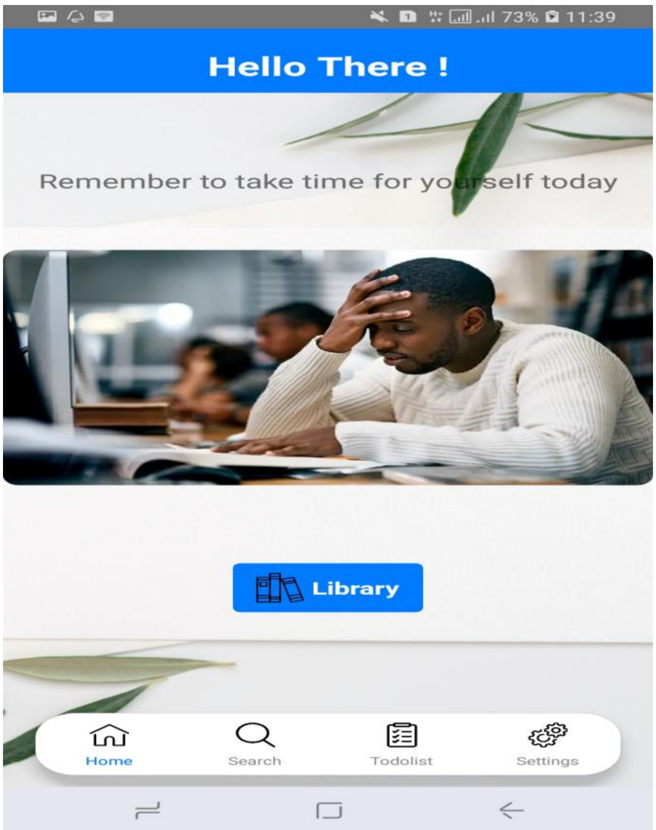


Figure 30 Clients Home

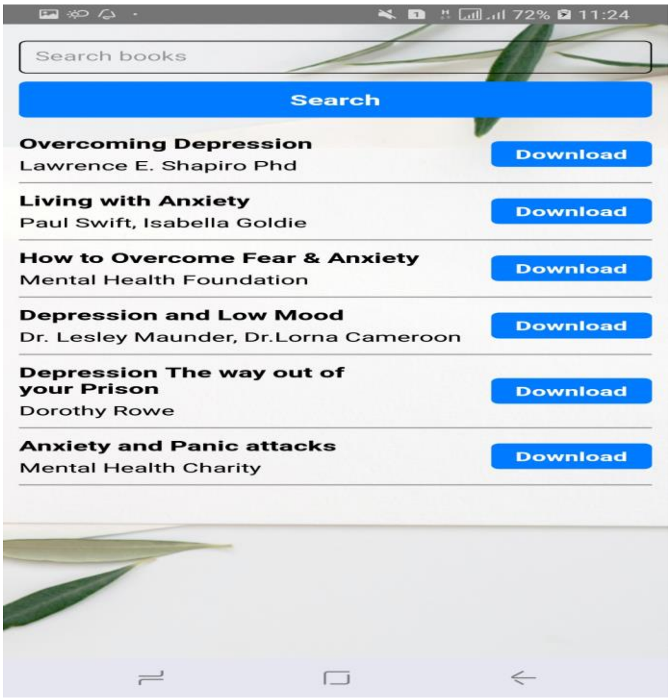


Figure 31 When Library Button is clicked

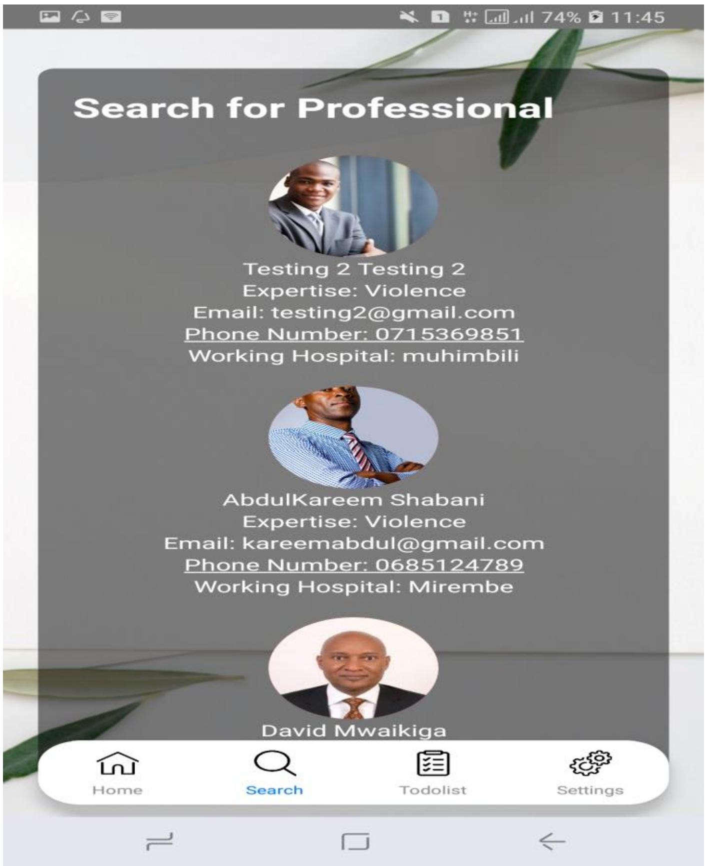


Figure 32 Search for professionals

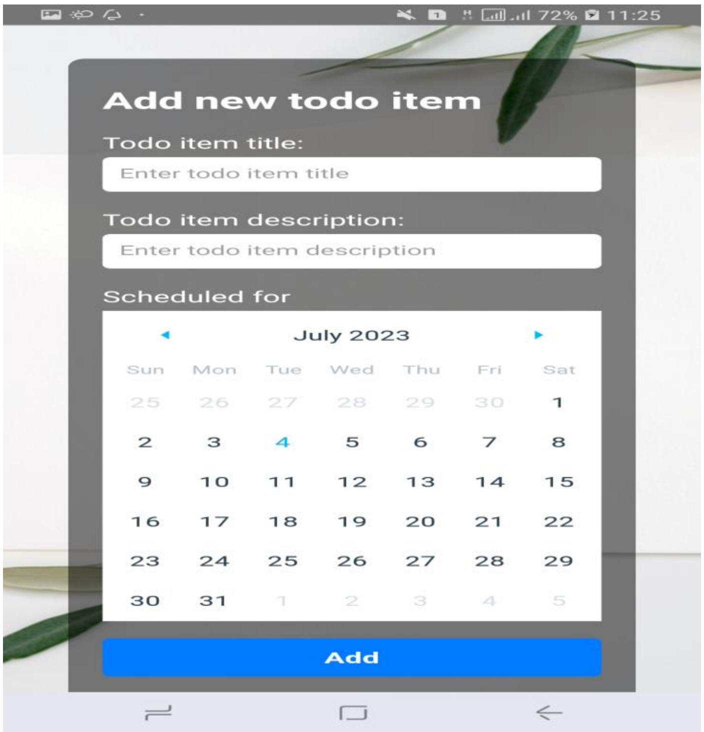


Figure 33 Add to-do

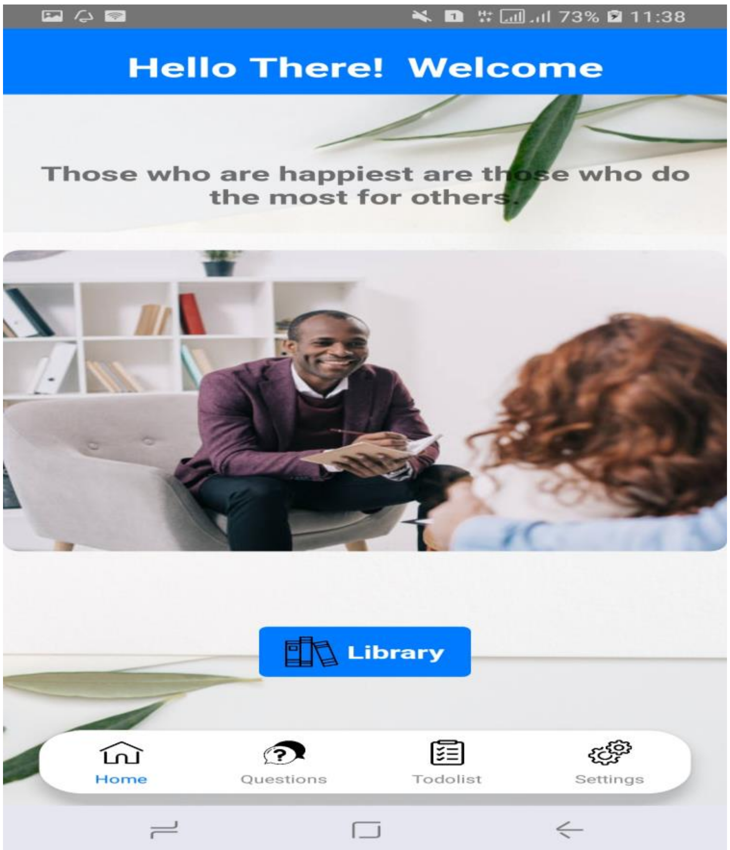


Figure 34 Professionals Home screen

And the most important interface was the one containing the form with questions which the clients will be asked, this is also the page which will be used to interact with the machine learning model I created to give out the predictions if a person is depressed or not.

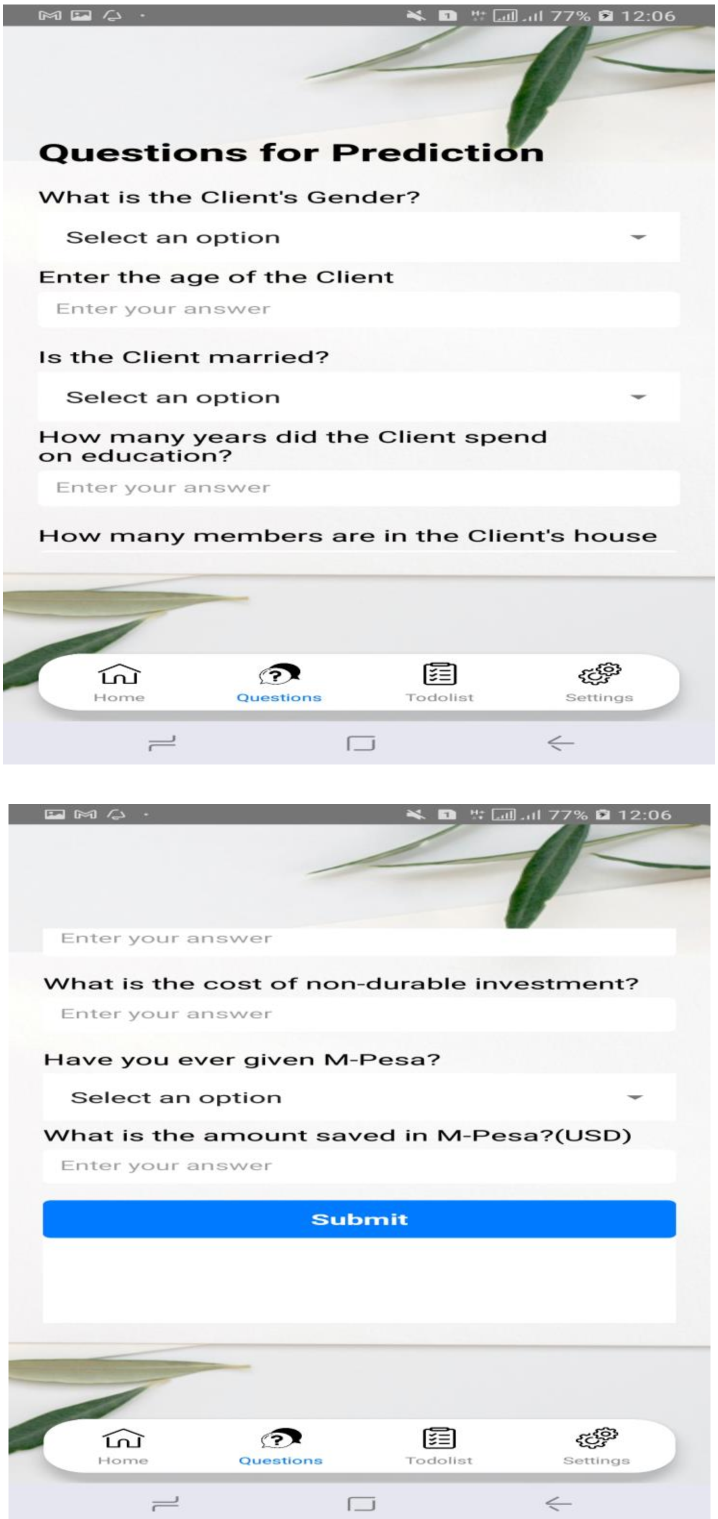


Figure 35 Prediction

CHAPTER SIX: CONCLUSION AND RECOMMENDATION

6.1. CONCLUSION

The Mental Health Detection App is designed to provide support and assistance to individuals facing mental health challenges. Leveraging the power of Machine Learning, the app aims to detect potential mental health issues focusing on depression and provide valuable insights for users.

The app focuses on helping people by analysing their responses to a series of questions and utilizing a trained Machine Learning model to identify possible signs of mental health conditions. It aims to raise awareness, encourage early intervention, and promote overall well-being.

6.2. CHALLENGES

The knowledge required to develop both the machine learning model and the mobile application was at some point beyond what we are taught in class, so intensive more hours had to be invested in order to bring this project to life.

6.3. RECOMMENDATIONS

This project could take a great turn not just for the University of Dar es salaam, but also some Mental health institutions who would prefer a more technological, reliable and accurate measures for determining person's mental state.

Also the skills of system analysis, project documentation, appropriate use of A.I and Programming should actually be placed to more practice in the university, because it plays a major role in the development of the final year project. With this, students could have ease in their projects and better establishments.

REFERENCES

- Alberto, Túlio C, Johannes V Lochter, and Tiago A Almeida. "Tubespam: comment spam filtering on YouTube." In Machine Learning and Applications (Icmla), Ieee 14th International Conference on, 138–43. IEEE. (2015).
- Fisher, Aaron, Cynthia Rudin, and Francesca Dominici. "All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously." http://arxiv.org/abs/1801.01489 (2018).
- Grömping, Ulrike. "Model-Agnostic Effects Plots for Interpreting Machine Learning Models." Reports in Mathematics, Physics and Chemistry: Department II, Beuth University of Applied Sciences Berlin. Report 1/2020 (2020).
- Dataset source https://zindi.africa/competitions/busara-mental-health-prediction-challenge/data
- Scroggins, R. (2014). SDLC and development methodologies. Global Journal of Computer Science and Technology.