UNIVERSITY OF ENERGY AND NATURAL RESOURCES

SCHOOL OF SCIENCES,

DEPARTMENT OF INFORMATION TECHNOLOGY AND DECISION SCIENCES

MEDICINE RECOMMENDATION SYSTEM

BY

AGYEI SAMUEL (UEB3221320)

NTI GYIMAH EMMANUEL (UEB3200120)

PRUDENCE AKOSUA ADJAKLO (UEB3215220)

ANANE BUSIA LAWRENCE (UEB3221520)

GYABAAH ABRAHAM (UEB3207620)

A Project work presented to the Department of Information Technology and Decision Sciences, University of Energy and Natural Resources, in partial qualification of the requirement for an award of a degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

October, 2024

Declaration

We, declare that this submission is our work towards the acquisition of a BSC in INFORMATION TECHNOLOGY, at the University of Energy and Natural Resources and that to the best of our knowledge, it contains no material previously published by another person nor material which has been accepted for the awards of any other qualification of the university, except where due acknowledgement has been made in the text.

AGYEI SAMUEL	……………	……………..
UEB3221320	Signature	Date
NTI GYIMAH EMMANUEL	……………	……………..
UEB3200120	Signature	Date
PRUDENCE AKOSUA ADJAKLO	……………	……………..
UEB3215220	Signature	Date
ANANE BUSIA LAWRENCE	……………	……………..
UEB3221520	Signature	Date
GYABAAH ABRAHAM	……………	……………..
UEB3207620	Signature	Date
Mr. EMMANUEL ADJEI DOMFEH	……………	……………..
Supervisor	Signature	Date
PROF. PETER APPIAHENE	……………	……………..
Head of Department	Signature	Date

# Acknowledgement

We express our gratitude for the precious gift of life and extend special thanks to our parents and guardians, who have persevered through challenges to support us throughout our time at this university. We are grateful to our supervisor, Mr. Emmanuel Adjei Domfeh, for sharing his knowledge, imparting skills, providing close guidance, and offering tireless support and advice, all of which have contributed to the success of this project.

Dedication

We dedicate our work to our families. We also dedicate our work to our Information Technology and Decision Sciences department lecturers, who helped us throughout the four-year program. Finally, we dedicate our work to our friends within the department.

Abstract

The Medicine Recommendation System (MRS) is a health/disease prediction mobile app using Artificial Intelligence to look at appropriate medicine regarding user syndromes. The system can predict disease, symptom information, and recommended medication using machine learning algorithms and historical medical datasets. First, we developed a process for data collection, and pre-processing before model training and then system testing. Other challenges included integrating data and preserving data privacy. Still, future development efforts will be around increasing the robustness of models through ensuring security as well as improving its core forecasts. The app is a resource for patients and healthcare providers that is intended to facilitate more educated clinical decision-making and better patient care.

Table of Contents

List of Figures

List Abbreviations

Electronic Health Records – **EHRs**

Computer Based Information System – **CBIS**

Medicine Recommendation System – **MRS**

Health Information Exchanges – **HIEs**

Artificial Intelligent –**AI**

Exploratory Data Analysis – **EDA**

Support Vector Classification – **SVC**

True Positive – **TP**

True Negative – **TN**

False Positive – **FP**

False Negative – **FN**

Receiver Operating Characteristic – **ROC**

Software Development Life Cycle - **SDLC**

Chapter 1: Introduction

1.0 Background

In the age of data in modern life, several key cornerstones for decision-making across different sectors. The healthcare industry is not an exception, but the use of Big Data applications has been more streamlined because they can collect, process, and analyze large health data (Deemed, n.d.). This approach is essential for making informed, data-driven decisions that will significantly enhance both patient care and operational efficiency. Big Data according to Related & Analysis (2021); Big Data refers to data sets that are so large or complex that traditional data processing applications cannot manage them effectively.

Big Data analytics focuses on extracting valuable insights from vast datasets to create opportunities for organizations to improve their operations. For example, companies like Flash use real-time data to optimize pricing strategies and operational logistics (Benarji et al., 2023). In the healthcare sector, platforms like Flash can enhance the delivery of fast medical services to patients and connect them with their respective healthcare providers. Such networking features can efficiently establish the supply-demand relationship in medical services as ride-sharing applications bring together passengers and drivers through the app (Sun et al., 2020).

As the big databased health analysis is cataloguing majority of details about how patients feel and act, it helps to cut through a great detail in learning on what actually put each patient needs. In this post, we describe this continuing research to investigate how daily, monthly and annual patient-level data can be represented as maps that facilitate integrated healthcare services throughout the city of Chicago. The increased availability of on-demand, app-based medical services has long been seen as a natural fit in today's healthcare landscape, offering both flexibility and responsiveness. Conversion tracking at a granular level of service delivery is far more reliable than the traditional healthcare services thus, providing first-hand data to app-based systems and making it an excellent resource for any future comparative analysis (Kumar et al., 2022).

Some platforms like Flash deal with over 20 million data points every single day, which emphasizes the need for real-time analytics to extract insights at scale from such large and diverse datasets. Handling such enormous datasets is difficult because the time lags and data rates to move them are immense, therefore scientists have turned to distributed computation and parallel processing techniques to address these issues. The main applications of these techniques are the real-time big data analytics (Vidhury et al., 2023).

As part of their public health surveillance effort, Flash data science experts have focused on using downstream analysis to identify code-level dynamics and apply machine learning models to identify patients with poor outcomes utilizing information from the medical services across a wide range of areas. Healthcare Big Data applications are discussed by Srinivas et al. High-quality forecasting of healthcare requirements can dramatically improve performance and patient satisfaction with the service provided.

One might say that machine learning is a key point in any model to be using since it's the one component in building predictive models from data in the domain. These techniques apply to models that have been trained previously on the algorithm and then can process new data to solve complex problems without any human intervention. If we look at the recent days, there is a trend of using probabilistic and statistical models in machine learning which suggests that it is gaining importance in healthcare (Kumar et al., 2023).

1.1 Problem Statement

This represents a major bottleneck for both patients and healthcare providers —how can diseases be diagnosed accurately and promptly? Accuracy in the treatment of diseases and taking good care of health requires accurate identification of the ailments that one has (AGoyal et al., 2020). Healthcare providers must diagnose a diverse and accelerated variety of illnesses, with variable levels of information, frequently on the clock (* et al., 2016).

Due to the dynamic and multi-modal nature of symptoms, traditional diagnosis may be slow and incomplete resulting in gaps in treatments on time and misdiagnoses (Deemed, n.d.). These delays in treatment may result in misdiagnosis, leading to suboptimal patient outcomes. As a direct result, many medical data generate overwhelming amounts of information which makes it difficult to discern the right patterns and make accurate predictions (Stark et al., 2019).

In response, we have designed a project that implements machine learning algorithms to create an artificial intelligence model capable of predicting diseases for patients who report their symptoms. Our model learns the essential characteristics of historical clinical data sets through the analysis of symptoms and empirical associations with diseases, so it can learn implicit patterns and factors affecting disease diagnosis.

1.2 Objectives

1.2.0 General Objectives

- To create a machine learning model that predicts diseases based on symptoms reported by patients.

1.2.1 Specific Objectives

- To Assess and establish data for disease predictions.
- To utilize a machine learning model for predicting diseases.
- To assess and evaluate the model.

1.3 Scope of the Proposed System

Medicine Recommendation System a symptom-based predictive model for diagnosing diseases in patients. Using various machine learning approaches, the system intends to deliver accurate diagnoses of diseases and their results in a medical prescription for patients. The system will use a database of historical medical data, which includes various categories like symptom patterns, descriptions of diseases, precautions to be taken regarding food, and exercise plans. Here is what the system will do:

- Machine Learning to Predict the Diseases given the Symptoms of the patient.
- Describe the disease likely predicted.
- Advise on how to take care of the disease.
- Provide personalized diet plans for the anticipated disease.
- Provide recommendations on suitable medications, the dosage, and the kind of side effects that can be expected.

1.4 Significance of the Project

The MRS offers the potential to significantly change how healthcare is delivered with several important advances including:

1. **Smaller Diagnosis Time:** The system can predict many diseases based on the given symptoms using machine learning models, so this whole diagnostic process has been reduced to the maximum. This increase in the speed of diagnosis could result in faster treatment and improve patient outcomes.

2. **Medical recommendations you can trust**: It delivers trustworthy and precise medical instructions based on patient-specific profiles. Such personalized health management strategy based on a full set of medical data including symptoms, disease descriptions, attention items, diet plans, medications & work-out (more incoming) can be realized by the system.

3. **Assists Informed Decisions**: The system helps healthcare providers in making informed diagnostic choices. Providing insights from data analysis, the system is yet another tool that healthcare professionals can now use to make accurate diagnoses and treatment planning.

4. **Patient Empowerment**: Patients learn many things about diseases and advice on how to take care of and manage the disease. This empowerment allows the proactive management of healthcare while any patient inputs with their treatment plans.

5. **Better Medical Advice**: The system offers timely and individualized medical suggestions to increase health outcomes. This can help to reduce patient health and hospital burden on infrastructure by meeting the healthcare needs promptly & being more accurate.

6. **Bridging Healthcare Accessibility**: Especially in areas with inadequate healthcare resources, this system improves the accessibility of healthcare. It uses technology to provide accurate medical suggestions online which ultimately leads to better healthcare reachability and unbiased deliverance of services.

1.5 Organization of the Project Work.

The work presented in this project is subdivided into five (5) chapters. Chapter 1 provides an overview in terms of the background of the project, the scope, the problem being dealt with, the objectives of the research and why it is important to undertake the study. Chapter two outlines the literature review for the different categories of computer-based systems in medicine recommendation systems, advantages, parts, and problems. Chapter three expounds on the method used in this recommendation system the machine learning life cycle and applying agile in the software development life cycle (SDLC). Chapter 4 discusses all the aspects of the analysis and discussion: system architecture, the analysis of the graphical user interface and testing approaches. Finally, chapter 5 gives an overview of the milestones accomplished, the suggested course of action for enhanced effectiveness in similar projects in the future, and the conclusion to the research.

chapter 2: Literature Review

2.0 Introduction

Healthcare has primarily been influenced technologically through Computer-Based Information Systems (CBIS). Such systems have better healthcare outcomes of hospital services in terms of productivity, accuracy, and range (Venkat Narayana Rao et al., 2020). Applications of CBIS vary from simple Electronic Health Records (EHRs) to advanced diagnostic applications (Gupta et al., 2021). It is a critical aspect for handling a high volume of data, automating workflows, and supporting clinical decisions, leading to better patient outcomes (Benarji et al., 2023). As a main innovation developed in CBIS, Medicine Recommendation Systems (MRS), has been built to analyze patient data using algorithms and AI-based applications to present customized medical paths for treatment (Stark et al. 2019).

This chapter takes a look at where CBIS have come from, what they do, and why they do it, with an emphasis on the MRS and healthcare outcomes.

2.1 Existing System

Al all this is made possible through Medicine Recommendation Systems AI which is currently transforming healthcare. Some of the AI solutions are further discussed as follows: Diagnosis of diseases is possible and advice on what people can do to take certain lines of action can also be gotten from any of the many AI solutions on the internet including IBM Watson Health, Ada Health, Your.MD and several others. (IBM Watson Health, n.d.)., Ram Davan, as an example, uses direct and typical artificial intelligence to analyze overwhelming data and reach therapy for sickness. Ada Health is a firm that provides artificial intelligence solutions, while the application You is a symptom checker application, and when applied it provides potential diseases the user has and what they should do next. Integrated an Intelligent Personal Health Information Service and also a symptom checker.

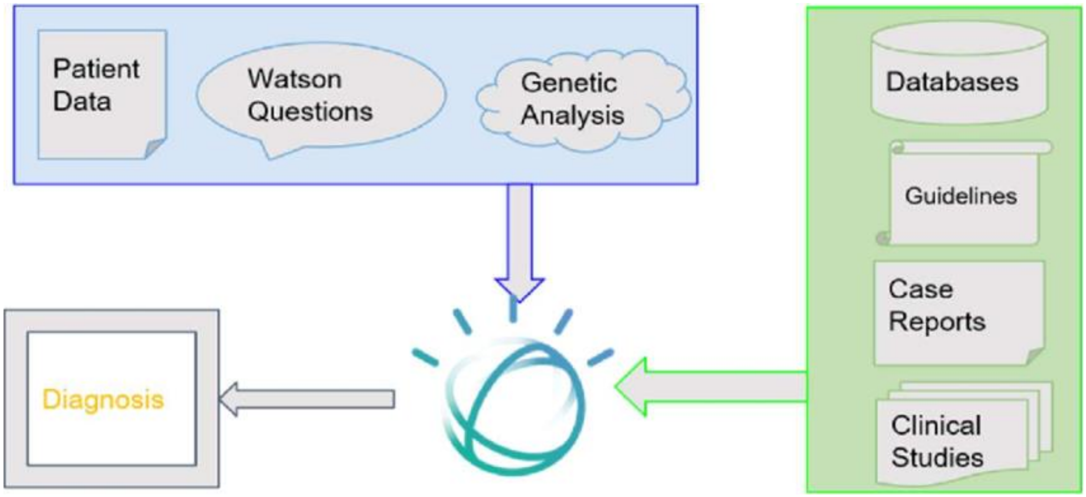

Figure 1 (IBM Watson Health, n.d., 2019)

2.1.1 Problems with the Existing System

1. Limited Accuracy Due to Insufficient Data: The majority of the systems in use today use a generation of patients and diseases that are small and vastly incomprehensive to patients and diseases exhibited in the healthcare system globally. This may lead to the offer of the wrong half-pie half-truth which may not be in the best interest of all users that are exercising the application or the specific system in the business.

2. Lack of Personalization: This being said, all today's systems are rather universal and do not take into account what exactly attributes of the patient themselves are, whether the patient is genetically ready, for instance, or not – this and only this is today possible to define based on the systems offered, HTML or a medical report. He says that if one does not distinguish between every entity

from the other one can abstract a picture where the treatment activities do not help towards matching some problems to solutions.

2.1.2 Advantages of the Existing System

1. Enhanced Diagnostic Support: Given that a practice has declared that smart apps are more preferred in medical diagnosis a note should be taken that they raise the level of accuracy and velocities. For this reason, it can be stated in advance that the lack of sufficient approach in the management of the questions that are connected with the subject of health is manageable in the higher level of potentiality that is in the context of big data of the object.

2. The recommendation systems of IBM Watson Health utilize a vast base of medical data to suggest evidence-based practices and the most effective treatment methods.

2.1.3 Disadvantages of the Existing System

1. Limited Accuracy Due to Insufficient Data: A shortage of data limits the performance of these technologies and consequently reduces the trustworthiness of the guidance that they provide.
2. Lack of Personalization: In general, existing systems struggle to effectively handle critical patient distinctions such as genes, lifestyle, and whole medical background, demonstrating that personalized medical recommendations are infrequently in demand. Integration Challenges: Problems of Integration comprise the breakdown of workflow and insular data that could transpire in the process of marrying AI-oriented systems with conventional EHRs existing in healthcare as well as other IT health establishments.

2.3 Types of Computer-Based Information Systems

CBIS (Computer-Based Information System) is defined by Higgins (1979) as an integrated system of hardware, software, data, and processes that supports operations, decision-making, and problem-solving within an organization. CBIS is vitally important to hospitals for the management of patient data, such as Electronic Health Records to reduce medical errors and improve care continuity (Jimenez & Carrera, 2018). They assist in clinical decision-making, analyze data, and provide evidence-based recommendations (Persson & Persson, 2020). Medical imaging is considered to be advanced by CBIS which assists radiologists in making more accurate diagnoses of conditions (Shambour et al., 2023). They also simplify administrative processes, which gives healthcare providers more time to focus on patient care (Stark et al., 2019).

Health Information Exchanges (HIEs) allow for interoperability, which is the secure sharing of data across major types of RS: According to Location.

Symptom-based systems: Provide recommendations based on the symptoms reported by patients. An example of this would be a patient presenting with fever and cough in whom the diagnosis may in the first instance imply pulmonary-related complaints like pneumonia or COVID-19.

Systems based on diagnosis: provides treatment options specific for certain diagnoses and guarantees that interventions follow these guidelines.

Personalized Medicine Systems: These systems adjust their recommendations based on individual patient data (e.g. genetic information) to improve treatment effectiveness

Institutions that improve care coordination (Shambour et al., 2023).

2.3.1 Benefits of Computer-based systems

CBIS is beneficial when implemented in health care in terms of managing data, providing real-time patient information, and bettering finding diagnoses (Doe et al., 2020; Smith & Johnson, 2019). It eliminates the busy work for staff and lets them concentrate on caring for patients (Brown, 2018). In addition, CBIS decreases human error due to manual data processing and improves resource allocation (Green, 2019).

Advantages of Medicine Recommendation Systems

More accurate Diagnosis: MRS helps in enhancing the accuracy of diagnostics and better health care can be provided to patients with a personalized treatment plan

Convenience: Implementation of AI models in MRS accelerates diagnosis and treatment, reduces the pressure on medical infrastructure as well as improves patient access to services

Aiding Patients: Lastly, MRS helps patients better understand their health and takes them on board as partners in treating themselves.

2.3.2 Recommendation System for Medicine

Medicine Recommendation Systems(MRS)Medical CBIS are personalized healthcare's specific CBIS. MRS can help in the prediction of diseases and recommend treatment by analyzing patient data based on AI and machine learning models (Goyal et al., 2020). Those systems connect EHRs, symptoms, and genetic data, featuring precision medicine (Topol, 2019).

MRS improves continuously by including information as it comes in new data, in close touch with the medical research community, and by reducing the cognitive load on healthcare providers

(Esteva et al., 2019). Evidence-based treatment guidelines guide a standard that can answer the question of how a patient will be provided the same care regardless of whether he/she lives in a range directly accessible to specialists (Obermeyer & Emanuel, 2016).

2.4 Medicine Recommendation Systems

MRS can be categorized into:

1. Symptom-Based Systems: These make recommendations based on what symptoms patients report. A patient presenting with fever and cough could be filtered for pulmonary complaints such as pneumonia or COVID-19 (Chin, 2020).

2. Diagnosis-Based Systems: these offer treatment choices based on particular diagnoses to assist in keeping the intervention within clinical guidelines (Smith et al., 2018).

3. Systems explicitly adapting recommendations according to individual patient data such as genetics, thus improving treatment efficacy (Personalized Medicine Systems).

2.5 Components of a Medicine Recommendation System

MRS combines patient information with predictive analytics to provide tailored medical advice. The pipeline is composed of data collection, preprocessing, predictive modeling, and a front-end for healthcare providers and patients (Collins & Varmus, 2015). The use of these systems can improve resource allocation, optimize healthcare delivery, and enhance decision-making (Kass-Hout & Alhinnawi, 2013).

2.6 Benefits of Medicine Recommendation Systems

Enhanced Diagnostic: MRS offers accurate diagnostics which results in improved healthcare delivery and personalized treatment arrangements. These systems are empowering the patients in their own hands, resulting in improved adherence to treatment plans (Goyal et al., 2020). It is also a more efficient and productive use of healthcare resources hence cost-effective at the same time.

2.7 Medicine Recommendation System

Many MRS have already Grasped AI Machine Learning in Healthcare MedWhat — a company that uses an AI chatbot to provide medical information, and Buoy Health, which analyzes symptoms to offer personalized health guidance. Additionally, symptom checker platforms can be offered by Intermedia that enable healthcare organizations to efficiently manage workflow (Persson & Persson, 2020).

In addition to facilitating diagnostics, these systems help empower patients and increase the availability of healthcare. With AI and machine learning on the rise, MRS will bring innovative solutions to healthcare that are both scalable, efficient, and personalized for patients (Jimenez & Carrera, 2018).

2.8 Computer-Based Information Systems in Healthcare

CBIS has transformed the way patient data is structured and decision-making in clinics. According to Higgins (1979), a CBIS is an organized collection of resources that stores and processes data so it can be used as information, which facilitates operations management functions in a knowledge-based organization. Another example would be in healthcare, where CBIS helps keep track of EHRs; which not only reduces the number of mistakes made but also improves care continuation. Automating Workflow and Informing Clinical Decisions directly improved patient care.

CBIS is further strengthened by the functions of Health Information Exchanges (HIEs), which support a secure way to share records among institutions. True interoperability, most obviously in the realm of personalized medicine facilitates superior care coordination.

2.9 Medical Recommendation Systems Related Challenges

Several barriers remain to the widespread implementation and use of MRS, despite these advantages.

Incorrect Diagnosis and Treatment Recommendation: The system could be inaccurate due to poor availability of data which can severely harm the patient

Limited Personalization: The current systems do not take into consideration many of the individual patient characteristics, like genetic propensity or lifestyle factors that should be essential for more personalized care

Integration Issues: One of the limitations associated with MRS is the inability to integrate it with existing electronic health records and healthcare systems, which leads to a lack of continuity in patient data (Chowdhury 2016).

Institutions often struggle to integrate AI-powered systems with legacy IT infrastructure, resulting in soloed workflows

2.10 Use Cases on How it Works

The implementation of MRS in healthcare has already demonstrated value. Here are systems like what MedWhat and Buoy Health provide on personalized health guidance through research chatbot text symptomology analysis using machine learning. The help systems also allow for quicker, more accurate diagnosis and empower patients to take healthcare into their own hands by providing them with convenient advice. MRS bridges gaps in healthcare accessibility, where regions suffer from scarce resources and poor follow-up of patients.

Chapter 3: Methodology

3.0 Introduction

This chapter explains in great detail a plan of the methodology employed while developing the Medicine Recommendation System outlined herein in this chapter. We also provided specific information about other similar systems that are already in existence, with a view of passing comments on them in terms of this new approach. This also presents the machine learning life cycle which was used to design the system for making very good and accurate suggestions to the users. As a result, it involves data collection, data initializations, modeling, and model verifications, the modeling phase, which collectively and systematically steers a less ‘heavy’ trip to a more definite way to achieving a management’s vision but through a proper theoretical project.

3.1 Machine Learning Life Cycle

A Medicine Recommendation System that works effectively is dependent on the standard machine learning life cycle, delivering richer design specifications, advanced educational accuracy, and easy access for its users. During this lifetime, critical milestones accomplish the establishment of reliable and usable systems. You will notice within the framework that you will define business intentions, compile and develop data, execute exploratory data analysis (EDA), design and evaluate models, and ultimately use the model operationally.

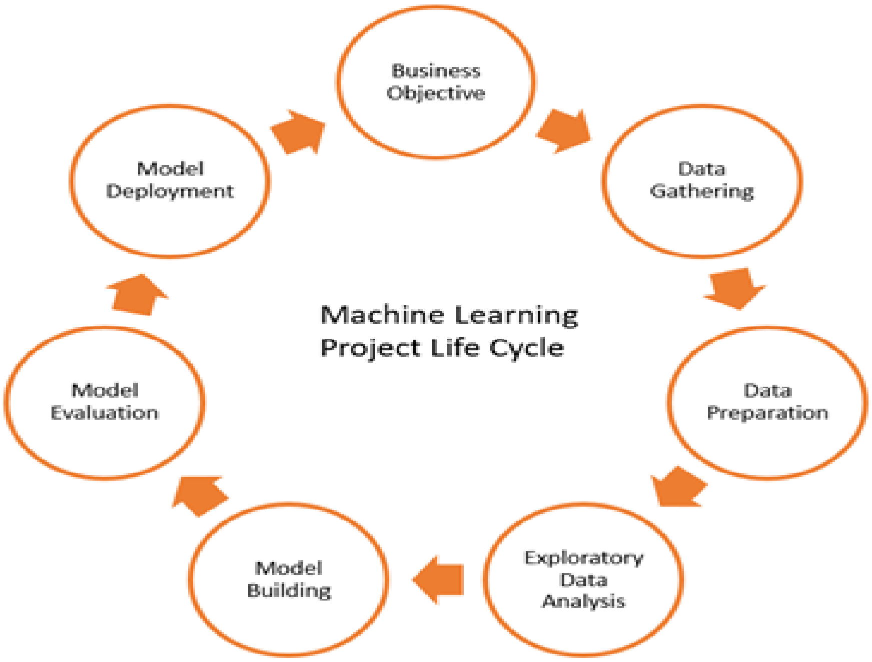

Figure 2 CPMAI methodology diagram (Google Images, 2024)

3.1.1 Business Objective

The objective of the first stage is to precisely understand targets from a business perspective for the Medicine Recommendation System. The leading important functionalities are to increase diagnostic precision, back personalized treatment alternatives, support medical practitioner decisions about healthcare procedures, and amplify the ease of managing healthcare resources.

3.1.2 Data Gathering

Successfully building a predictive analytics model entails a complete database. The data set in use for this study originates from Kaggle's Medicine Recommendation System Dataset, which includes files such as: Within the file, Precautions.csv are precautions for a comprehensive selection of ailments; Training.csv transmits vital information for creating easily remembered models; Medications.csv contains all drugs relevant to disease management; and Diets.csv present suggested meals suited to multiple health issues.

```
In [14]: dataset
Out[14]:
```

	itching	skin_rash	nodal_skin_eruptions	continuous_sneezing	shivering	chills	joint_pain	stomach_pain	acidity	ulcers_on_tongue	...	blackheads	scu
0	1	1	1	0	0	0	0	0	0	0	...	0	
1	0	1	1	0	0	0	0	0	0	0	...	0	
2	1	0	1	0	0	0	0	0	0	0	...	0	
3	1	1	0	0	0	0	0	0	0	0	...	0	
4	1	1	1	0	0	0	0	0	0	0	...	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	
4915	0	0	0	0	0	0	0	0	0	0	...	0	
4916	0	1	0	0	0	0	0	0	0	0	...	1	
4917	0	0	0	0	0	0	0	0	0	0	...	0	
4918	0	1	0	0	0	0	1	0	0	0	...	0	
4919	0	1	0	0	0	0	0	0	0	0	...	0	

4920 rows × 133 columns

Figure 3: Training dataset description

3.1.3 Data Preparation

For data science, the skip on data preprocessing in the end-to-end process makes it difficult to trust and make reliable inferences through analysis, modeling, and decision-making. The two reasons why they preferred only quality assurance while ensuring the correctness of data for evaluation, one more time after the data collection, it will be preprocessed again.

- **Data Cleaning**

Nevertheless, the data set used for future analysis and model building should be taken care of with a few notes while considering the following; Exploratory or explanatory models are the two most recommended but dependent on what type of data is used and which problem is to be solved, there might be different modes.

- **Normalization using Z-Score**

Regarding its literal meaning, we have some confusion too so we appeal this as the Z-score or Standardizing of variables where for mean value is equal to zero & for standard deviation the value is equal to one. It is the most useful for us in applied practice when we try to test all factors and want each of them to scale and your covariate to be selected also has an equal effect.

```
In [25]: # Standardize the data
         scaler = StandardScaler()
         X_scaled = scaler.fit_transform(X)
```

Figure 4: z-score standardization

3.1.4 Exploratory Data Analysis (EDA)

Conducting an Exploratory Data Analysis (EDA) is essential to understand a dataset's attributes. Another name for this process is data visualization, herein, graphical means are used to summarize the main characteristics of the data, as well as reveal any underlying patterns and relationships.

- **Data Visualization:** The creation of ethnic groups and their relative conditions in the parish database to receive relevant information for reference groups is now a crucial and primary task that should be accomplished.

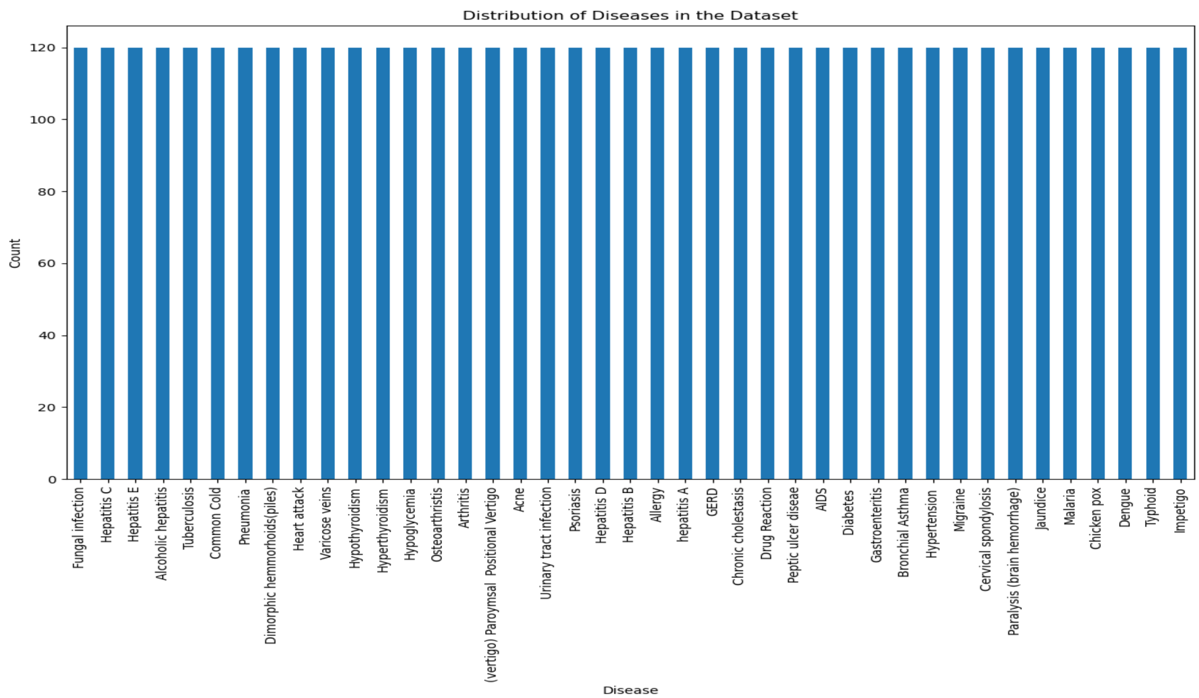

Figure 5: Distribution of Diseases

.

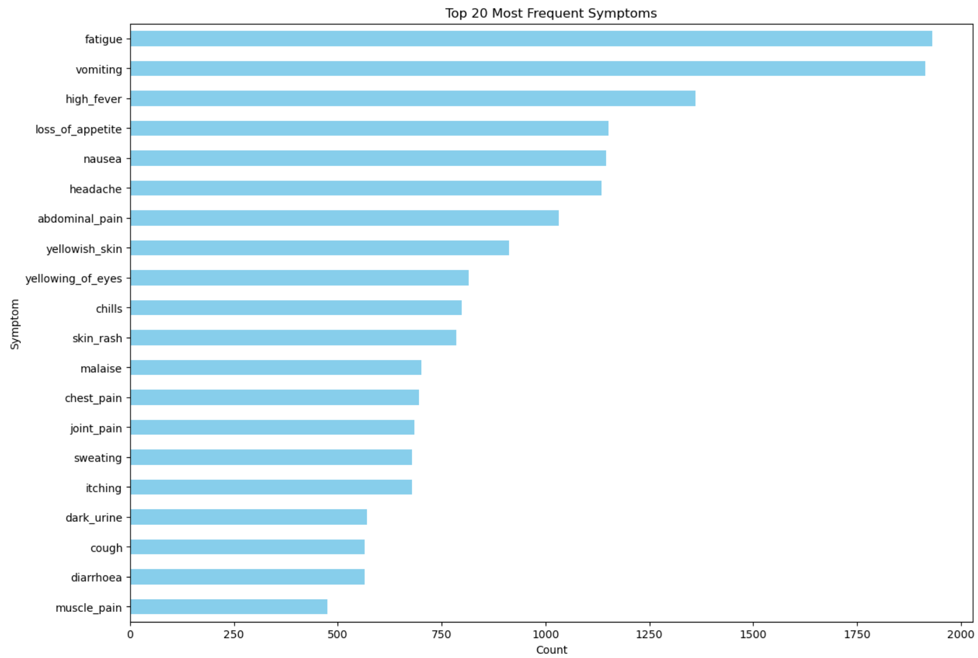

Figure 6: Top 20 Most Frequent Symptoms

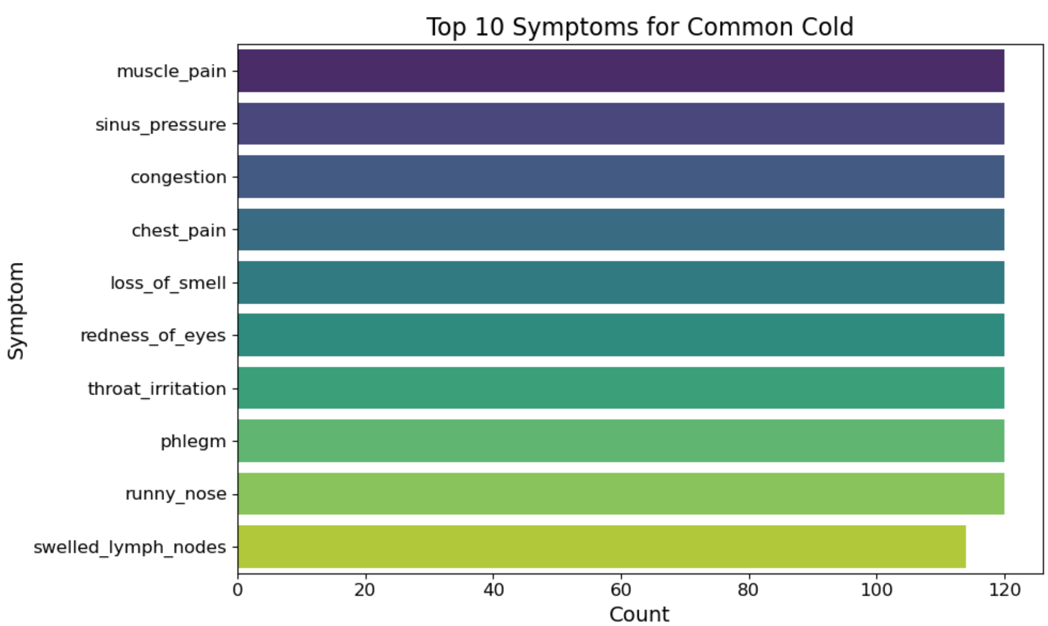

Figure 7 Top 10 Symptoms for Common cold

- **Correlation Analysis:**

  Some relationships between different features are investigated regarding strong connections of respective features with each other. This would facilitate a better understanding of the association between variables and determine their importance regarding statistical modeling.

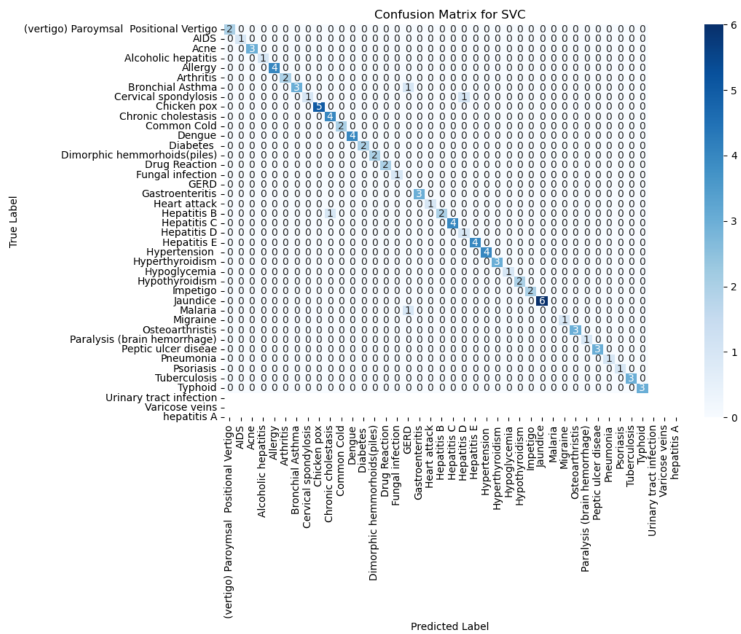

Figure 8: Confusion Matrix for SVC

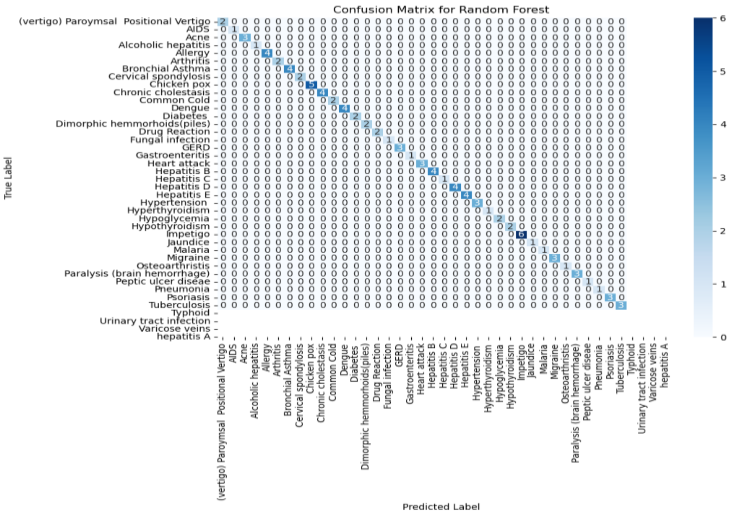

Figure 9: Confusion Matrix for Random Forest

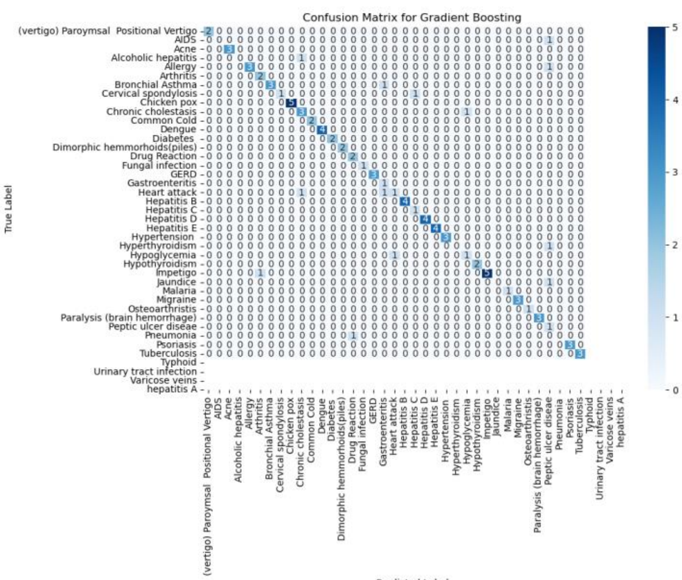

Figure 10: Confusion Matrix for Gradient Boosting

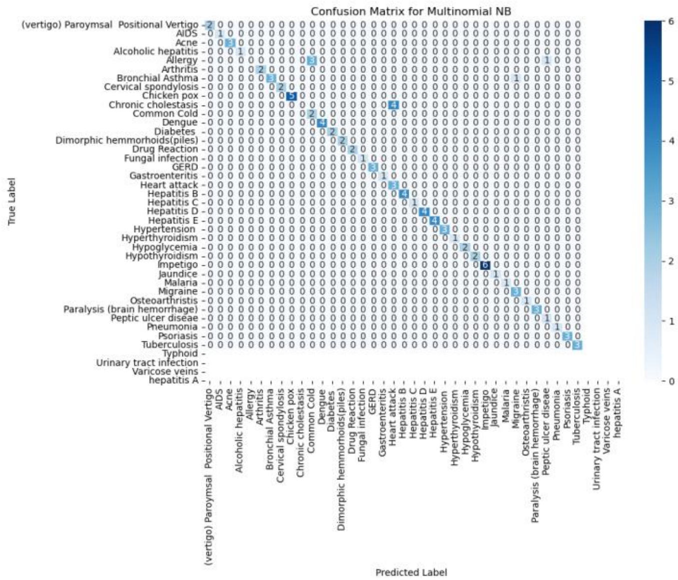

Figure 11: confusion Matrix for Multinomial NB

3.1.5 Model Building

We used a subset of these machine learning models that we trained to predict the best medication for each patient given their data and medical conditions, and built a medicine recommendation system.

3.1.6 Machine Learning Models Used

- **Random Forest Classifier:** An ensemble approach: an alternative to the way of obtaining classification accuracy and robustness to recommend medications
- **Gradient Boosting Classifier:** Ensemble method in which a queue of models — one for each quirk corrects mistakes made by previous ones to improve the medicine recommendation accuracy.

- **Multinomial NB**: Bayes classifier that is attribute-sensitive performs a comprehensive task by addressing categorical data where predictions are made regarding drug categories based on some variables that originate from multinomial distributions.
- **SVC (Support Vector Classification):** Classification can be understood as finding the optimal hyperplane that effectively separates different types of medications, ensuring the correct prescriptions are administered.

3.1.7 Performance Measure

The specified metrics were utilized to assess the performance of the implemented machine learning models.

Accuracy Score

Accuracy refers to how effectively a model predicts outcomes for new data. The accuracy score is calculated based on the proportion of correct predictions to the total number of cases in the dataset.

Accuracy is calculated by dividing the number of correct predictions by the total number of predictions.

Usage: Calculation: Calculation with accuracy_score from sklearn. On the two metrics, you will measure the predicted labels (y_pred) of the model and compare them with true labels (y_test)

Confusion Matrix

A confusion matrix is a representation of classification results, displaying true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

Structure:

True Positives (TP): Positive cases that are correctly classified as positive.

False Negatives (FN): The number of negative cases incorrectly identified as positive

False Positives (FP)The number of false positives is the total of negative cases you misclassified by concluding that they are positive.

False Negatives (FN): The number of positive cases wrongly marked as negative.

Usage:

Functional: The confusion matrix of sklearn is used for visualization. Actual computation of the matrix and seaborn. Heatmap to visualize it.

Heatmap: This gives better visibility as to across how many classes the model predicts

Cross-Validation Score

Cross-validation is a method to verify the performance of a model on an independent dataset. As such, it requires you to split your dataset into different subsets (folds) and train/test your model on various combinations of these folds.

Process:

K-Fold Cross-Validation: The available dataset is divided into K segments, with K being a positive integer that can vary. In this method, the model is trained using K-1 segments and validated on the remaining segment. This process is repeated K times, with each segment taking turns as the test set while the other segments are used for training the model.

Verification: Performance measures (e.g., accuracy) from all the folds are averaged to get an overall performance number.

Usage:

Calculation: For computation, you will use the cross validation score function from sklearn. Cross-validation procedures and model selection. This returns an array of scores for each fold, that you can then average to get the final score.

3.1.8 Model Performance Evaluation

This analysis aimed to review the performance of different Machine learning algorithms on our data set. We first evaluated with 80% Train data and 20% Test data split. In a later stage, we then split the dataset into 70% training and 30% testing data respectively, and refitted the models.

Split: 80% Training Data — 20% Testing Data

We calculated the accuracy scores of every model in 80/20 split form. Below is the summarization of Accuracy for diverse models: SVC: 0.95%, Random Forest Classifier: 1%, Gradient Boosting Classifier 85%, and Multinomial NB: 0.90%.

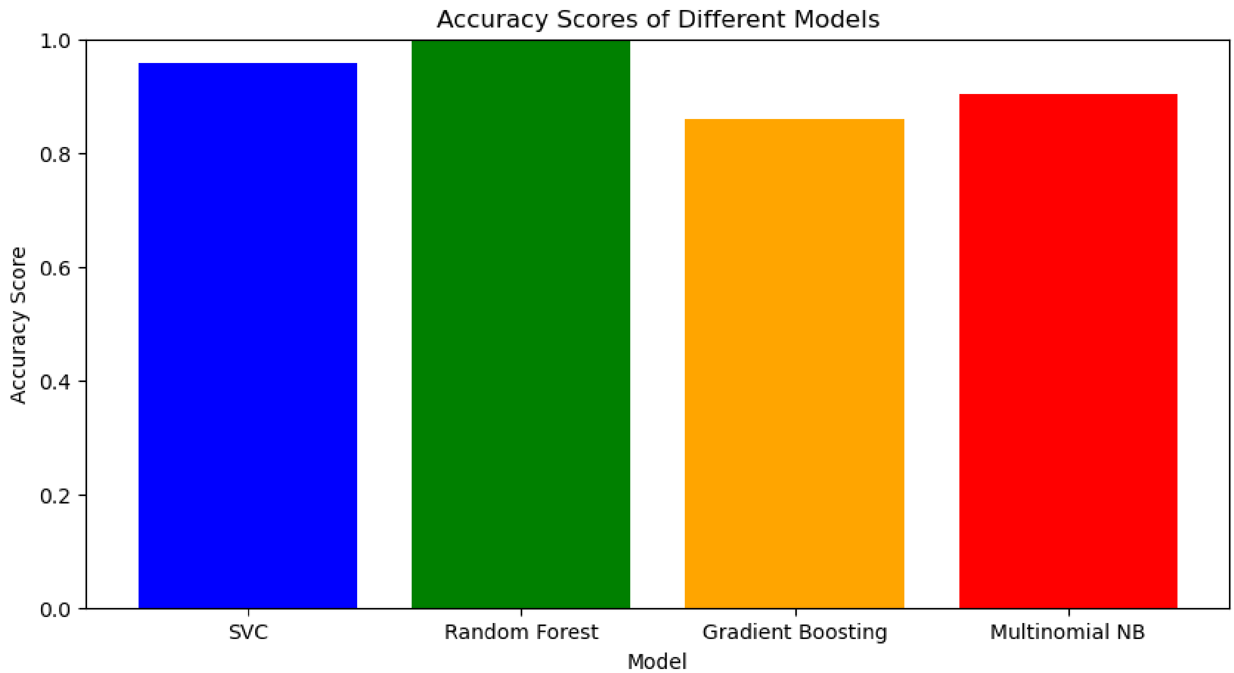

Figure 12 Accuracy Scores for Models

3.1.9 Model Evaluation

Model evaluation helps to analyze how good our model is and if it can be generalized effectively on unseen data. This phase includes:

Performance Metrics: metrics such as accuracy, precision, recall, F1 score, and Area under the ROC Curve to evaluate the model

K-Fold Cross-Validation: To ensure that the model works well in unseen data, it is historical data to an extent that is consistent with k under a fold cross-validation.

Error Analysis: Breaking down misclassifications and errors to focus on areas of improvement.

Extensive model evaluation ensures that the system is robust, and performs well in a variety of different scenarios giving you confidence it will work out in the real world.

3.1.10 Model Deployment

The model, once trained and evaluated, is deployed into a production environment that can predict the interests of new users in real-time. This phase involves:

Interface: Embedding the model with the current healthcare IT infrastructure in a way that can smoothly interact with EHR systems and other data sources.

Developing a user interface for healthcare providers and patients along with algorithm input data they can gather by accessing this technology.

Monitoring and Maintenance: It continually monitors the model performance, determining its retraining on new data or leveraging recent medical advances.

By ensuring that the models are deployed successfully, it will leverage Machine Intelligence to effectively work in hospitals providing a valuable support tool for the healthcare staff; thus improving patient outcomes.

3.2 Software Development Life-Cycle

Utilization of the Software Development Life Cycle (SDLC) is indispensable for the project, in which proper planning, elaboration, and sustenance of building and maintaining a mobile app to predict disease is carried out. While all this may spread up, agile should only be used for such a project purely because some changes are inevitable in terms of agility and customer collaboration supported by an iterative approach.

3.3 Rational Behind Agile Software Development Life Cycle (SDLC)

Well, the Agile Software Development Life Cycle (SDLC) molds the path in the plan for responding to drawbacks of other software development approaches but also becomes more productive by adjusting quickly and effectively producing quicker responses. Agile, therefore stresses a customer-centric approach that contributes to early value delivery, with continuous customer engagement which eventually makes sure that the final product meets the expectation of them. It can embrace the best available knowledge even after interventions are deployed and respond to changing conditions alongside uncertainty.

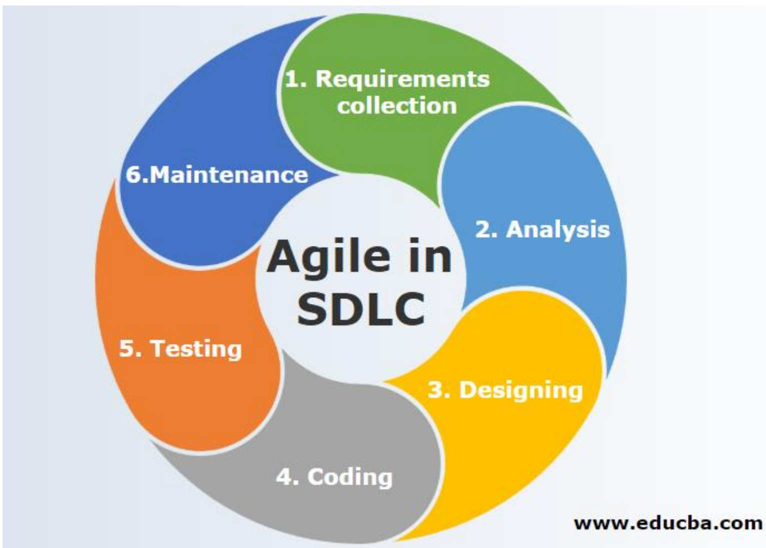

Figure 13 Agile Software Development Life Cycle (source:https://images.app.goo.gl/dsg94KVEEKgcjbN16)

- **Requirement Collection:**

  Collect the requirements of the project and understand it This is where the development teams work with stakeholders to gather detailed requirements. It involved catering dealing alongside breaking down what the clients need, what those association objectives and whatever other limitations, etc. They could be structured as interviews, surveys, reviews of existing records, brochures or workshops, and other means that may be used to accomplish these purposes.

- **Analysis:**

  Review the requirements for feasibility and create a plan. Once the requirements are gathered, those needs(developer) to be analyzed and see, if they need to meet the standard of Clear Complete, The Developer can understand what is coming in flow. It includes the following aspects — Requirement analysis, breakdown of requirements into sub-parts, risk mitigation, and detailed project plan.

- **Designing:**

  System architecture and detailed design All analyzed requirements transformed into system architecture and afterward detailed the design documents through the design phase. This captured the full system-level architecture, detailed design diagrams and technologies to be used.

- **Coding (Development):**

  Write the code for developing diagnostics according to the design specification. Development: In this step, the actual developers write the code to fulfill the requirements of a project. In this phase all the modules are coded, unit-tested, and then integrated. To build an operating software that satisfies the design requirement is the aim.

- **Testing:**

Perform functional and defect testing for the software Different types of testing including unit, integration, system, and user acceptance are part of the testing phase. The objective is to discover and sort out issues present in a piece of software. It Works Correctly Meets its Requirements Reliable Secure Make Sure That Testing happens to test “Whether or not” the software, works.

- **Maintenance:**

However, once it is deployed, you will need people to support and extend the software as it is said once the software is deployed, then comes the maintenance phase. This phase includes software evaluation, bug fixes, user support, and updating or adding new features.

Chapter 4: Analysis and Discussion

4.0 Introduction

In this chapter, the implementation of the Medicine Recommendation app is explained. It is the app that by entering the symptoms, recommends the possible diseases with machine learning research. The chapter discusses the architecture, design or UX, and implementation steps of an app

4.1 System Architecture

Structure The Dust Prediction app is designed to have the system flowing data and capabilities uninterruptedly. Specifically, it includes the following components.

User Interface (UI): It serves as a user interface for users to input symptoms, get results/contact information, etc.

Backend Server: This server will hold the ML model and process data coming in.

4.2 User Interface Design

UI is designed, keeping simplicity, usability, and visual appeal in mind. The main screen designs are Home, About, and Contact.

4.2.1 Home Screen

The home screen allows the user to input symptoms they are feeling which enables them to manage their health. Its design is deliberately curated for an easy-to-use interface to be able to record the symptoms and predict diseases correctly. A super-intuitive set-up divides symptoms into intuitive buttons and makes entering quick. The app processes this data as soon as symptoms enter the system allowing users an informed choice on their health. Better navigation and interactive elements improve the user experience, making the prediction process smooth and open to anyone.

Figure 14 Home Screen

4.2.2 About Screen

The About screen tells you the basic functionality of the app and some details on how the team is working dedicatedly to offer a better user experience for both patients as well Doctors by following a continuous development life cycle. The app can predict diseases by your symptoms and give you suggestions of medicines suitable for your situation. If you want to know more about your health or need professional advice this is your trusted app.

.

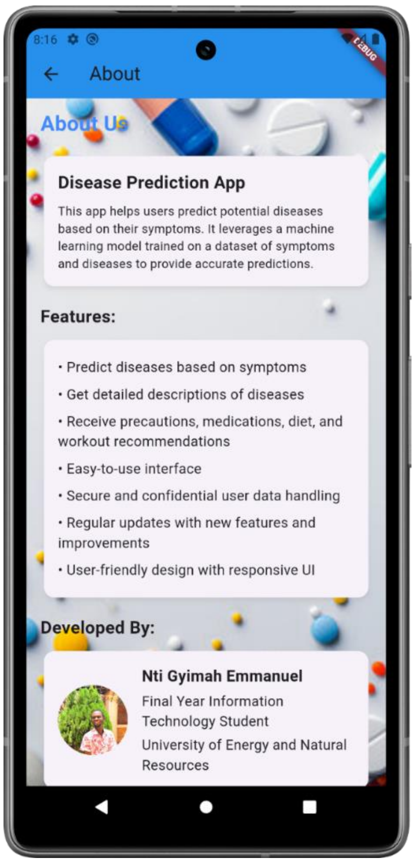

Figure 15 About Screen

4.2.3 Contact Screen

The Contact screen is the most delicate bridge between you and the developer, helping your support message to reach him. Designed for ease of use, you can easily reach out via the web to ask a question make a suggestion or voice a concern. If that user interface connects you to an expert in real-time for help, feedback or just to network with. Just tell us what you need please, and we will get back to you as quickly as possible, to make your experience with this application better.

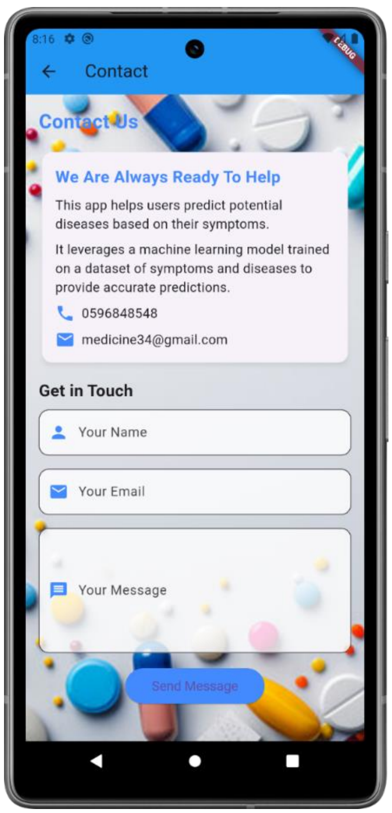

Figure 16 Contact Screen

4.3 Testing

- Testing: Finding errors Testing is essential for ensuring software quality and reliability. This makes testing results for maintenance later.
- One of the psychology of testing: Testing to show that a program works by showing errors. The primary goal of the testing phase is to identify any errors in the program. In other words, you cannot test to prove that a program works, but instead, you can only test to reveal that a program does not work. Testing is just the process of running a program to find errors.
- Purpose of testing: The primary goal of the testing phase is to identify any errors in the program.
- Testing is running a program to uncover an error.
- The undetected bug surfaces when the test is passed successfully.
- It has a good chance to find out the error in case it finds space.
- It cannot catch errors that might already exist if any

4.4 Levels of Testing

- Testing and Validation Strategy:
- Multi-layered testing strategy inspired by the Testing pyramid. The intention was to ensure that the system meets all functional and non-functional requirements affiliated with web-scale systems. This is where we are choosing the ratio between different types of tests, such as white-box and black-box test methods for different levels of the system. All the stages

of testing were important as they spoke to how well the system worked functionally, in integration, and as a whole.

4.4.1. Unit The Building Block of Development

- At the lowest level of the pyramid, Unit/Component Testing had a clear focus on validating that individual parts of the system worked as they were intended. Essentially, at this stage were used the methods of white-box testing which involved an in-depth study of the internal structure and logic behind the code.
- Machine Learning Models — Every step of the machine learning pipeline, such as data pre-processing, model training, and prediction logic was tested in isolation for correctness and consistency.
- Flask APIs: Unit-tested endpoints to check the implementation with correct error handling, input validation, and output correctness.
- Testing Flutter Interface: The completion of this stage marked success in individually testing each Flutter component to respond correctly and look the same after the user gave input, giving a strong foundation for the interface.
- You have been using these tests to catch bugs early in the development process, which enabled quicker iterations and lowered the cost of fixing issues later.

4.4.2. Integration Tests - How They Fit Together

- Further up the pyramid, Integration Testing was performed to ensure Every sector of the system talks to each other. This was still a white-box testing stage but also adding black-box techniques where components communicated with each other, noting that it did not traverse into the internal logic.
- The communication from frontend-backend — We tested whether the data was properly transferred between the Flutter Frontend and Flask Backend, i.e., Analysis of API calls to be valid and expected.
- Database Integration: These are integration tests that appeared to be covering the feature of how well the system talks to Firebase; Testing data fetching, storage, and update operations across different components.
- Integration of Models: The model integration, if it works properly with the API endpoint or not If it provides correct predictions and data flows correctly from input to output.
- Because there were many functions (modules) that work correctly, but do they also work together that was more dependent on the system integration testing?

4.4.3 System QA Testing: End-to-End System Functionality Testing

- We tested the whole system at once, to test this we reached in System Testing phase. Black-box techniques were used to test at this level of testing, the system was treated as a whole and tests were performed just on its outputs without taking into consideration internal codes.

- Functional Testing: We executed several test case scenarios on the system to confirm whether it was working as intended, such as user authentication, data processing, and real-time interaction.
- Behavior Testing: Conducted load testing of the system under various average loads to see how quickly it could serve some requests.
- Security Testing: We tested to make sure that user data was safe and communication would be secure by performing some vulnerability assessments.
- System Testing: It was to verify that the system can satisfy requirements under realistic conditions.

4.4.4 System Integration Testing —Integration Across the Board

- System Integration Testing broadened the scope to include interactions among different subsystems and modules in the entire system environment. This phase made sure that all the modules that we were integrating, worked together fine.
- Integration 4A- Cross-module interactions — Test that checks to make sure the parts of the system interact with each other properly, Related data sharing and dependency are proper.
- End-to-end Testing: This testing ensured the workflows in the system from user inputs to final outputs were working fine through multiple integrated modules.
- System Integration Testing was very important because it helped ensure the system as a whole that is, when followed up with individual subsystems worked well in operation.

4.4.5 Solution Testing: End-to-End Solution Review

- Solution Testing: It was more about validating the solution as a whole and taking care of it from a holistic view. This phase included end-to-end testing of the complete system for overall solution integrity and user scenarios.

- User Scenario Testing: Real-world scenarios were being used to test the system and ensure a harmonious user experience.
- Regression Testing: Both automated and manual tests were executed to verify that new updates or changes introduced did not corrupt existing logic, preserving the whole solution.
- Testing the entire solution allowed us to be certain that all parts of our complete system were requirements-matching and prepared for deployment.

4.4.6 Acceptance Testing: Assuring the Acceptance of a User

- Acceptance Testing (Top of the Pyramid): The system was validated from the end user's perspective. This step was performed using black-box testing techniques, aimed at meeting the users' requirements and guaranteeing that this system would be useful in real-world situations.
- Final Validation: The Product Owner completed acceptance testing on the final system to ensure it met the objectives of the project and the expectations of users, and voted whether that system was ready or not for production.

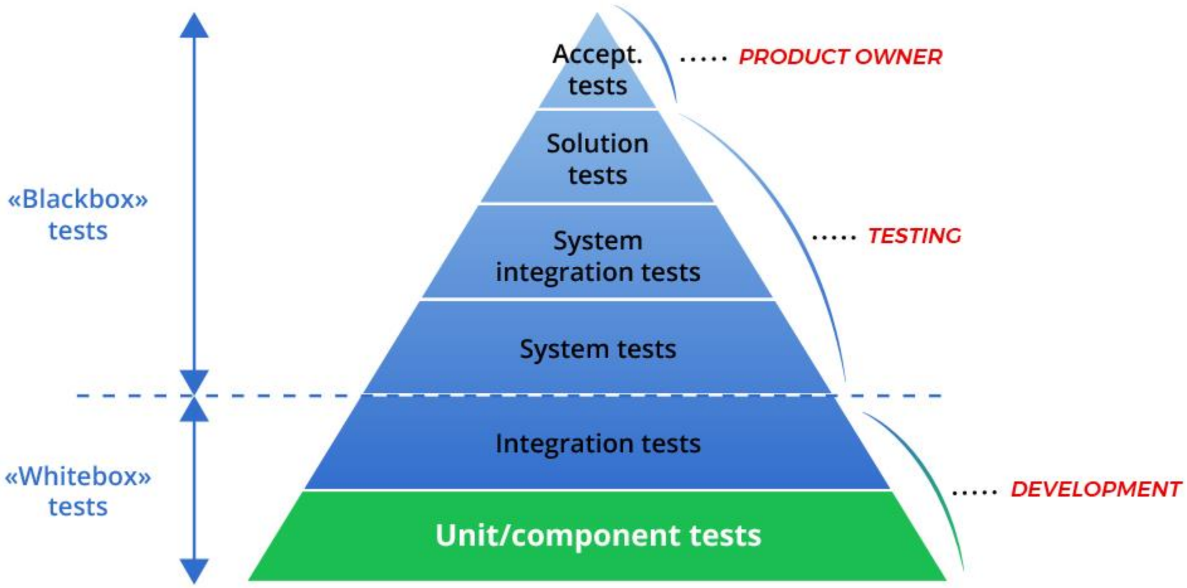

Figure 17 Google. (n.d.,2019). Testing Pyramid

Chapter Five:

Summary, Recommendations, and Conclusion

5.0 Summary

In this chapter, the authors present an overview of the topic of the development of the Medicine Recommendation app, including key aspects related to the development cycle, architecture, interface, and testing. It presents the main conclusions and outcomes in developing important and stable applications for enhancing healthcare with the help of Artificial Intelligence disease prediction and suggestion.

5.1 Key Achievements

Throughout the project, significant achievements include:

- Development of a Machine Learning Model: This was able to develop an accurate model in the diagnosis of diseases since the user can input symptoms.
- System Architecture Design: Standard operational procedures for modularity by modifiability and system measurability by architecture for that movement and analysis of the chatter data.
- User Interface Design: I have made the layout with an emphasis on simplicity and minimalism, because it is a phone, with only slight changes from the first page Home, About Us Contact Us, etc.

- Testing and Quality Assurance: The following testing measures were followed on the real feel measures, to include/ to design the application to run faster and be reliable the following testing measures were followed.
- Integration of Feedback: Compiled the feedback of several different users to enhance as well as create new features in the application as well as the graphical user interface.

5.2 Recommendations

Based on the project's implementation and testing phases, the following recommendations are suggested for further enhancement:

Enhanced Data Integration: They may be on how to apply one kind of information including; a new outlook in health status or genomics into increasing disease prediction.

Continuous Model Improvement: The subsequent is the way how the reserved ML model will be retrained when there is new medical information when the need of the user changes:

Security Enhancements: The other necessity is also the enhancement of safeguarding the user data which is among others; the laws and regulations of healthcare securities.

User Experience Enhancement: All that should add to increasing the comprehensiveness of the choice option even higher, and to making the site more convenient for new users with every subsequent usage of Navigation displays on any test.

Collaboration with Healthcare Providers:

Thus, the usability of the designed iOS application is something that should be discussed with many health stakeholders as such the iOS should be designed.

5.3 Conclusion

However, Medicine Recommendation continues to be one of many largest successes of using AI within the sphere of medicine. This is why the app directly addresses the disease prediction issue and gives tips and nothing more to help increase the chances of getting as many people to make the right decision as regards any matter affecting their health. In this project, there are several cases to apply the presented ML application for medical diagnostics and there are some prospects for its future improvement.

References

1. Benarji, P. S., Bharadwaj, P. S., Neeha, B., Srikanth, D., & Ankitha, V. (2023). Taxi demand prediction using machine learning. *International Research Journal of Engineering and Technology*, May, 1527–1531.

   Deemed, G. E. (n.d.). Medicine gaurav-1.

2. Goyal, V. A., Parmar, D. J., Joshi, N. I., & Champanerkar, P. K. (2020). Medicine recommendation system. 1658–1662.
3. Goyal, V., Parmar, D. J., Joshi, N. I., & Champanerkar, K. (2020). Medicine recommendation system. *International Research Journal of Engineering and Technology*, 1658–1662. www.irjet.net
4. Gupta, J. P., Singh, A., Kumar, R. K., & Pradesh, A. (2021). A computer-based disease prediction and medicine recommendation system. *International Journal of Advanced Research in Engineering and Technology*, 12(3), 673–683. https://doi.org/10.34218/IJARET.12.3.2021.062
5. Jimenez, C., & Carrera, I. (2018). Defining architectures for recommended systems for medical treatment: A systematic literature review. *American Journal of Computing-Lajc*, V(2), 33.
6. Nandal, S., & Dahiya, S. (2016). *International Journal of Scientific Research in Science, Engineering and Technology, IJSRSET*, 4(4), 687–692.
7. Persson, F., & Persson, F. (2020). Building a medical recommendation system: A case study on digitalizing evidence-based radiology.

8. Shambour, Q. Y., Al-Zyoud, M., Abu-Shareha, A. A., & Abualhaj, M. (2023). Medicine recommender system based on semantic and multi-criteria filtering. *Interdisciplinary Journal of Information, Knowledge, and Management*, 18, 435–457. https://doi.org/10.28945/5172
9. Stark, B., Knahl, C., Aydin, M., & Elish, K. (2019). A literature review on medicine recommender systems. *International Journal of Advanced Computer Science and Applications*, 10(8), 6–13. https://doi.org/10.14569/ijacsa.2019.0100802
10. Suryadevara, C. K., & Engineer, S. (n.d.). Towards personalized healthcare: An intelligent medication system. *International Journal of Engineering Science and Computing*, 5(9), 1–16.
11. Venkat Narayana Rao, T., Unnisa, A., & Sreni, K. (2020). Medicine recommendation system based on patient reviews. *International Journal of Scientific and Technology Research*, 9(2), 3308–3312.