GRADUATE CERTIFICATE

ISY 5001

INTELLIGENT REASONING SYSTEM (IRS)

PROJECT MODULE

PROJECT REPORT

GROUP 6

AI-POWERED

HEALTH CONDITIONING MONITORING &

NUTRITION PLANNING

Yatharth Mahesh Sant (A0286001R)
Kristofer Roos (A0285949A)
Zhang Yusen (A0285839H)
Hao Zhenmao (A0285960R)
Chua Kian Yong Kenny (A0056377W)

Table of Contents

1. EXECUTIVE SUMMARY

Traditional healthcare services often adopt a one-size-fits-all approach with limited medical support available only during infrequent appointments scheduled months apart.

Additionally, studies have been done to substantiate the need for good follow-up care with timely access to medical conditions and information could lead to better recoveries.

Thereby, we aim to create an AI-Powered Health Condition Monitoring and Nutrition Planning system to enable a tailored health journey catering uniquely to the User’s needs and provide a wealth of personalised resources to accompany the User throughout the recovery journey.

2. BUSINESS CASE AND MARKET RESEARCH

2.1 Market Needs and Opportunities

In the realm of traditional healthcare, services often adopt a one-size-fits-all approach, where for most cases, generic diagnosis procedures are undertaken, and generic medical advice and prescriptions are given.

The most important factors and typical advice to aid recovery besides medication would be having sufficient rest and proper diet. However, for different patients, the terms “sufficient rest” and “proper diet” would have different meanings.

Additionally, the medical support would only be available during the infrequent appointments usually scheduled months apart. For the most part of the recovery journey, the patient would have to be self-reliant, which could be risky as most patients are also not capable of processing new medical information effectively in the short time of their medical appointments.

Thereby, herein lies the opportunity to develop a personalized companion alongside the patient’s recovery journey, which can encourage and provide patients with the necessary support of health condition monitoring and nutrition planning during their recovery journey.

2.2 Industry Trends

2.2.1 Preventive Healthcare

Many contemporary healthcare systems in developed nations are rooted deeply in classical education paradigms like the Flexner report and philanthropic endeavors, confining contemporary medical practices to rely on laboratory investigations [2]. During their inception, these standards were sufficient, as the medical arena predominantly addressed acute infectious diseases in younger individuals. However, with the elevation of global health benchmarks, the central concern of medicine and healthcare has transitioned towards chronic conditions, given that cardiovascular issues, diabetes, and cancer account for about 70% of U.S. fatalities and 75% of its healthcare spending [3]. Such health trends are increasingly noticeable in the developing world as well, emphasizing the need for a more prevention-oriented medical approach [4]. In essence, the pivot for the medical field should be from merely treating diseases (sick care) to promoting overall well-being (healthcare).

2.2.2 Data Availability and Lack of Utilization

In the context of widespread digitalization, the healthcare sector has become increasingly flooded with data that can be leveraged to improve care. According to a Stanford University study, it was determined that health-related data is expanding at an annual rate of 48%[1], with technological integration contributing to this exponential growth at each phase of the care process. Furthermore, the study found 30% of the world's data generated annually is within the healthcare industry [5]. This surge in information allows for more comprehensive analyses, yielding enhanced accuracy and insights, ultimately facilitating cost reduction and better care quality. While industries such as banking, retail, aviation, and insurance have swiftly evolved their paradigms to become data-centric, optimizing both their operational and strategic dimensions, the healthcare sector has been somewhat hesitant. Many of these industries have reaped the benefits of digital adaptation, witnessing enhanced efficiency, greater return on investment, and superior product offerings. Conversely, healthcare remains conspicuously behind the curve, revealing a sedated evolution in systems and technologies that capitalize on emerging data. Specifically, the healthcare industry was given a score of 28/100 in the McKinsey Digital Quotient, a cross-industry digitalization study, scoring second to last, only to industrials. This slow adaptation hinders the potential to foster a deeper understanding, elevate standards, and improve the overall quality of care [1].

2.3 Quantitative/Qualitative Impacts

2.3.1 Total Addressable Market (TAM):

The Total Addressable Market (TAM) for our AI-Powered Health Condition Monitoring and Nutrition Planning system can be computed by examining the prevalence of chronic diseases in Singapore, considering diseases such as diabetes as a representation. With a 9.4% prevalence rate and the population at 5.9 million (as of 2023), the TAM is estimated at 555,000[2] individuals who could potentially benefit from the system. [9][10]

2.3.2 Serviceable Available Market (SAM):

The Serviceable Available Market (SAM) is a segment of TAM accessible through our system, given the technological infrastructure and user adoption rates. If we assume the system can cater to 50% of the TAM, the SAM would be 277,500 individuals.

[1] Compound Annual Growth Rate, 2013-2020
[2] 5.9 million residents * 9.4% prevalence rate

2.3.3 Serviceable Obtainable Market (SOM):
The Serviceable Obtainable Market (SOM) is the portion of SAM that we realistically can capture, considering the competitive landscape, marketing, and sales capabilities. Capturing 10% of the SAM would translate to a SOM of 27,750 individuals, presenting a more immediate market opportunity for the system.

2.3.4 Implications:
The TAM, SAM, and SOM estimations provide a layered understanding of the market potential for our AI-Powered Health Condition Monitoring and Nutrition Planning system within Singapore. This market sizing analysis serves as a fundamental step in illustrating the prospective reach and impact of the system in addressing the healthcare needs of individuals coping with chronic diseases. By strategically navigating the market, there lies a significant opportunity to foster a tailored healthcare journey for users, contributing to a larger narrative of enhanced healthcare delivery.

2.3.5 Cost Savings and Efficiency Improvements:
The introduction of our system could lead to considerable cost savings and efficiency improvements within the healthcare sector in Singapore. By providing personalized health monitoring and nutritional planning, the system can potentially reduce the frequency of hospital visits and healthcare consultations, thus saving costs for both healthcare providers and patients. Moreover, by facilitating timely intervention and promoting adherence to personalized nutrition and health plans, the system can contribute to better health outcomes, reducing the long-term economic burden of chronic diseases. Furthermore, the system's continuous monitoring and real-time feedback could expedite the identification and management of health issues, thereby improving the efficiency of healthcare delivery. This is particularly relevant in a setting where timely access to healthcare resources is crucial for managing chronic conditions and ensuring better recovery outcomes.

2.4 Competitive Landscape

2.4.1 Competitors
Multiple resources on the internet are available that could provide medical information based on symptoms and illness, and two of the bigger competitors are WebMD and Symptomate. WebMD provides a comprehensive symptom checker application supplemented with health information, health articles, news

and videos. Symptomate boasts of having a doctor-developed symptom checker that advises the probable conditions with care recommendations.

WebMD and Symptomate have positioned themselves as reliable online resources for individuals seeking initial medical guidance based on their symptoms. However, their offerings are largely static and don't extend beyond providing a probable diagnosis and general medical information.

These limitations underscore a significant opportunity for a more integrated and personalized digital health solution. The gap in continuous monitoring, personalized dietary planning, and real-time engagement with users represents a tangible need in the market. Our proposed system is designed to address these specific needs, providing a more holistic and personalized healthcare experience. Through real-time monitoring and personalized meal-planning, it aims to support individuals in managing their health conditions and aiding recovery in a more engaged and proactive manner. Moreover, the envisioned collaboration with healthcare providers and supplement partners could further enhance the value proposition, offering a seamless and holistic health management solution.

2.4.2 SWOT Analysis

To help navigate the complex terrain of healthcare technology, we can examine our positioning across four strategic dimensions: strengths, weaknesses, opportunities, and threats.

Starting with strengths, our system thrives on personalization, setting it apart in a market saturated with one-size-fits-all solutions. By offering customized health monitoring and meal plans, we address a fundamental consumer need for tailored healthcare interventions. The system's ability to provide continuous support fills a critical gap in the healthcare journey, especially during intervals between formal medical consultations. Additionally, our reliance on high-quality data and easily updatable data inflow, ensures that our recommendations are both accurate and relevant. The platform itself is designed for maximum user engagement, with a straightforward interface, easy access via Google Sign-On, and real-time feedback mechanisms. Finally, the inclusion of an end-to-end meal planning system creates a relevant and practical connection to users' recoveries and lives.

However, there are a few areas which we are looking to handle with care. Primary among these is the issue of data privacy. Handling sensitive user health data requires rigorous security measures and compliance with privacy laws, which can be challenging and costly. The advanced technology that powers our system, while a significant asset, also introduces complexity that could be a barrier for less tech-enabled users. Additionally, the system's effectiveness hinges on accurate user input — a variable that introduces an element of unpredictability. Also, the operational demands associated with continuous data management and system updates necessitate a robust allocation of resources.

Looking outward, several market trends work in our favor. The rising tide of health-conscious consumers represents a growing market for our services. This trend, combined with an industry shift toward preventive care, presents a significant opportunity for customer acquisition and retention. Partnerships with established healthcare providers could lend additional credibility to our platform and expand our user base. Moreover, there's substantial potential for geographic expansion and adaptation to local market preferences. Integrating with wearable technology could further solidify our position in the market, offering users a more comprehensive health monitoring solution.

One potential threat the business is faced with is that we operate within a highly regulated industry, and compliance with healthcare regulations will be a constant focus area. We also face stiff competition from both established online health resources and new entrants in the health-tech space. Security risks pose another significant threat; a single data breach could severely damage user trust. Lastly, there's always a risk that users may misinterpret the health information provided, underscoring the need for clear, accessible communication.

2.5 Core Value Proposition

2.5.1 Strategic Positioning and Competitive Advantage

Our proposed AI-powered health condition monitoring and nutrition planning system aims to enable a tailored health journey catered uniquely to the User's needs and provide personalised meal-planning to accompany the User throughout their recovery journey.

It is envisaged to be a system that would provide necessary information on their probable medical condition, advise meal plans based on medical condition and food preference as well as provide reminders on meal plans.

As a holistic companion application, future development could be developed further to include integration with medical health and healthcare supplements partners. Our system would serve as a post-care medium for clinics and hospitals to track their valued patients' recovery progress through in-app feedback on conditions after clinical consultations and provide the necessary alerts to doctors if there were further complications against the norms of expected recovery. For the healthcare supplements partners, our system would help promote their products targeted specifically for the benefits of patients with the specific illnesses.

2.5.2 Outline Value Proposition

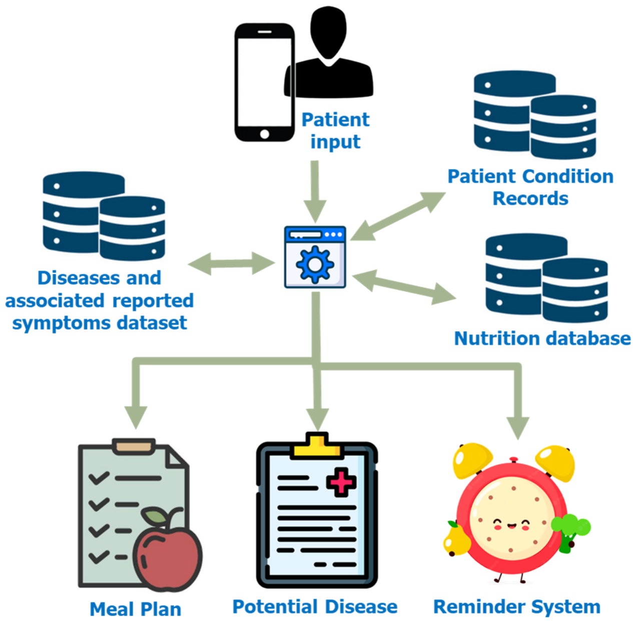
Our system's value proposition is anchored in personalized, continuous healthcare, utilizing advanced technology to provide proactive support. This strategy meets the growing demand for individualized, accessible health solutions beyond traditional settings. This system is strategically developed to go beyond the usual sporadic healthcare model, turning it into a continuous, interactive process tailored to each user's distinct health requirements.

At its heart, the system focuses on personalization. It moves away from broad health solutions, instead paying close attention to individual health conditions, food choices, and daily living patterns. This detailed strategy is key in developing health and nutrition plans that connect with users on a personal level, building trust and encouraging consistent use.

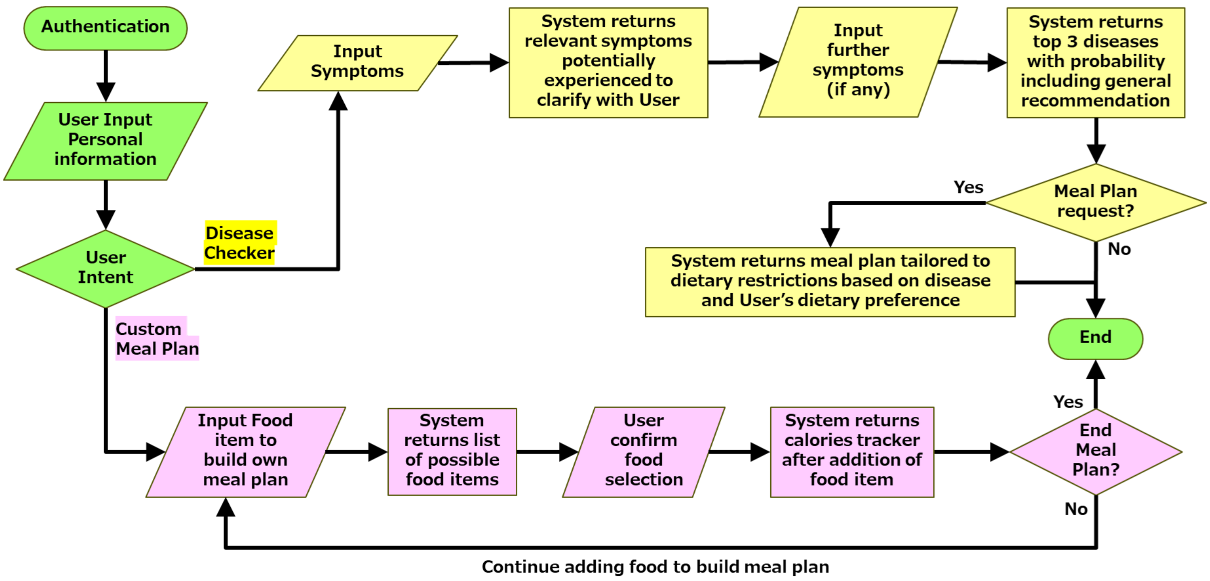
3. SYSTEM DESIGN AND MODEL

3.1 Architecture Overview

The system provides a user-friendly interface for identifying the probable diseases on their symptoms and provides an option to generate meal plans tailored for the diseases. Google Oauth login is used for authentication, thereafter users can access the NLP chatbot which facilitates interaction for symptom analysis and diagnosis. The meal planner operates in two distinct modes: as a standalone application, where users can generate meal plan custom to their requirements and in the post-disease diagnosis mode where the generated meal plan considers diet restrictions associated with the predicted diseases.



Detailed Workflow



1. **User Authentication:**
   - The user logs into the system using Google Sign-On, if user is already logged into their google account, user is automatically redirected to the chatbot page
   - For initial log-in, the user would be prompted to input their personal information and preferences. These information would be used to guide and develop the meal plans.
2. **Symptom Analysis via Chatbot:**
   - Post-login, the user interacts with a chatbot, which gathers information about their symptoms through a series of questions.
   - The chatbot utilizes Natural Language Processing (NLP) and Natural Language Understanding (NLU) to process the user input and extract symptoms from it. If valid symptoms are provided, the chatbot would suggest some relevant symptoms based on the existing knowledge base which the user might be facing.
   - The user can select from the suggested symptoms or continue describing their condition for a more precise diagnosis.
3. **Disease Diagnosis, Recommendation, and Meal Planning:**
   - Upon finalizing the symptoms, the chatbot processes the information and returns a set of potential diseases that the user might be experiencing.
   - The chatbot currently returns the top 3 probable diseases along with the likelihood probability. In addition to this, it also suggests recommendations which imply towards general preventive measures to be taken for the disease.

    - If the user expresses interest in receiving further assistance, the system generates and displays a meal plan tailored to the disease and the user's food preferences, leveraging Genetic Algorithms and a comprehensive nutrition dataset. This meal plan considers dietary restrictions and suggestions related to the diagnosed disease.
4. **Standalone Meal Planner:**
    - Users can access the meal planner independently of the disease diagnosis process.
    - From the particulars provided such as sex, age, height, weight, activity level, weight goal, and meal type, the system computes the Total Daily Energy Expenditure (TDEE) from the Basal Metabolic Rate (BMR). This value will help guide the allocation of calories across the meals of the day.
    - Upon selection of the custom meal planner option, the user could key in their desired food choice. Based on the keyword input, the system will check against the database to return a list of possible food items with the input keyword. Upon confirmation of the food item, nutritional information of the food will be printed. The remaining calories after deducting the selected food will also be presented to the user, where it can help guide the user if he wishes to proceed to add more food items.
    - The process will iterate again if the user requests to add additional food items, where the total calories will be summed to check against the user's target calories count for the meal.

Technical Stack and Architecture

- **Frontend:**
    - **Technologies:** Vue js
    - **Authentication:** Google OAuth 2.0 API for user authentication.
    - **Interaction:** The frontend communicates with the backend via API calls and WebSocket connections.
- **Backend:**
    - **Technologies:** Django
    - **WebSocket:** Utilized for real-time bidirectional communication between the client and server, ensuring instant data transfer and state management.
    - **State Management:** Employed to maintain the user's session data (symptoms, identified diseases, meal plans, etc.) throughout the WebSocket connection.

    - **NLP Engine:** Processes user inputs and manages chatbot interactions.
    - **Disease Prediction Model:** A machine-learning model that predicts potential diseases based on input symptoms.
    - **Meal Plan Generator:** An algorithm that leverages Genetic Algorithms to generate meal plans based on user input and the nutrition dataset.
- **Database:**
    - **Technologies:** SQLite
    - **Purpose:** Storing user data, historical symptoms, disease information, meal plans, and nutrition data.
- **Security:**
    - **API Security:** Utilize tokens and OAuth for secure API calls.

3.2 Data Management

3.2.1 Data Sources

Primarily, two sets of data were required, with one for the symptoms checker, and the other for the meal planning.

The disease-symptoms and symptoms precaution references dataset from Kaggle [6] consists of 41 diseases with 120 entries of reported symptoms for each of the identified diseases. The dataset consists of a total of 128 unique symptoms and each of the reported diseases contains between 3 to 17 symptoms such as itching, cough, chest pain, anxiety, etc. Some examples of the diseases within the dataset are common cold, migraine, pneumonia, AIDS, gastroenteritis and drug reaction.

The food nutrition dataset was retrieved mainly from HPB's Energy and Nutrient Composition of Food [7] and Github [8] comprising a total of 3718 and 3238 food choices respectively with information of key nutritional information of each food including weight, calories, protein, fat, carbohydrates and etc.

3.2.2 Data Preparation

Additional information to supplement the dataset is required to achieve better personalisation. An important aspect was to understand the recommended food intake and food to avoid for the respective diseases. The information is collated through extensive research to supplement the disease dataset. Correspondingly, the assessment of nutritional value with respect to the diet restrictions imposed by the respective diseases was required to be updated for the nutrition dataset.

Additional labels to categorise breakfast choices, lunch/dinner choices, drinks and fruits were added so that appropriate meal selection could be achieved. Uncommon food and raw ingredients from the dataset were also excluded as we deliberately designed for the meal planner to recommend food that is readily available, for the convenience of the Users.

Furthermore, there was a need to enrich the finalized nutrition and meal dataset with additional categorical values, which are leveraged to define optimal meal recommendations. When a User is diagnosed with a condition, there are certain medical recommendations given, for example low-fat diet, which needed to be generated in the dataset. This was done by calculating the concentration of the component in focus, relative to the weight. Therefore, the diet recommendation from the disease diagnosis can be added as an additional constraint to the meal plan generation.

Finally, the disease dataset was enriched with additional context on the diseases via research from WebMD, which allowed recovery-times, general recommendations, and dietary precautions to be integrated into our database. This is used another point of reference for the Users, when they are provided their diagnosis.

4. SYSTEM DEVELOPMENT AND IMPLEMENTATION

4.1 Tools / Techniques

4.1.1 Multi-label classification using machine learning

Overview:

Multi-label classification refers to the technique where data records can belong to more than one class at the same time depending on the context being viewed in. For instance, many diseases have mild fever, high fever as a common symptom and hence if someone is having fever that means possibilities of many diseases. The disease prediction using the symptoms is an example of multi-label classification. Some of the techniques employed for multi-label classification are: OneVsRest Classifier, Binary Relevance, and Classifier Chain.

OneVsRest Classifier:

Train N binary classifiers for each class N.

The binary classification predicts positive outcome for that class and negative for all the other classes. For instance for class Ci, we have the classifier Ni which treats Ci as positive class and all other than Ci as negative class.

While prediction we use all the N classifiers and take their positive outcomes. Only those classifiers are considered giving a positive outcome and the labels corresponding to the classifier are extracted.

Classifier Chain:

During the training phase, there are N binary classifiers for each label. However, the original input features are fed only to the first classifier. The output of the first classifier is used as a feature and appended to all the other features and fed to the second classifier. This process is then repeated.

While doing predictions the same pattern is followed, that is first classifier predicts based on input, second based on the output of first and the input features, and the process continues.

Selected Strategy:

Since diseases can have associations and this is taken care by classifier chain, that approach is used for disease predictions. SVM (Support Vector Machine) is used alongside the classifier chain strategy.

4.1.2 Disease prediction Chatbot

Overview:

The chatbot is designed to diagnose potential diseases based on user-provided symptoms. It employs advanced natural language processing techniques, fuzzy string matching, and machine learning models to achieve this. The process can be broken down into three main steps: symptom extraction, symptom suggestion, and disease prediction.

Symptom Extraction:

The chatbot begins by processing the user's input to extract potential symptoms. This involves tokenizing the input text and filtering out common stopwords and punctuation. The tokenized words are then refined by grouping related terms together, ensuring that multi-word symptoms are captured accurately. For instance, if the user mentions "chest pain," the system recognizes it as a single symptom rather than two separate words.

To ensure the accuracy of symptom detection, the system employs a combination of fuzzy string matching and semantic similarity calculations. Fuzzy string matching quantifies the similarity between the user's input and a predefined list of symptoms. If a potential match is found, its semantic similarity with the user's input is computed. This is achieved using vector representations of words, where the cosine similarity between vectors represents the semantic similarity between terms.

Symptom Suggestion:

Once the user's symptoms are identified, the chatbot suggests additional relevant symptoms. This is based on conditional probabilities derived from historical symptom-disease associations. The conditional probability of a symptom given the user's symptoms is computed by analyzing the frequency of co-occurrence of symptoms in the historical data. The system then suggests the top n symptoms that have the highest conditional probabilities, ensuring that these suggestions are relevant to the user's condition.

Disease Prediction:

After gathering a comprehensive list of symptoms from the user and the suggested symptoms, the chatbot predicts potential diseases. It first transforms the symptoms into a multi-label binary format, where each symptom corresponds to a column in a dataframe, and the presence or absence of a symptom is represented by 1 or 0, respectively.

Using this binary representation, the system employs a trained machine learning classifier to predict the probabilities of various diseases. The classifier used in this case is the Support Vector machine (SVM). The classifier's output is a list of probabilities corresponding to different diseases. To provide the user with the most relevant information, the system selects the top N diseases with the highest probabilities that exceed a predefined threshold.

For each predicted disease, the system also provides associated precautions. These precautions are retrieved from a separate dataset that maps diseases to their recommended precautions.

Mathematical formulas:

- **Conditional Probability**: $P(A|B) = \frac{P(A \cap B)}{P(B)}$
  - Used to compute the probability of each symptom given the user's symptoms.
- **Semantic Similarity**:

$$similarity(A, B) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

- Used to compute the cosine similarity between two vectors, representing the semantic similarity between terms.

4.1.3 Meal Planner

Overview:

Two distinct applications were built for the meal planner. The first application acts as a follow up to the disease prediction chatbot, recommending appropriate meals in consideration of dietary restrictions associated with the predicted illness over the expected recovery period. The second application operates as a standalone meal planner providing Users with the flexibility to customize their own meal plans according to their calories intake goal for each meal.

Determining Calorie Goals - Total Daily Energy Expenditure (TDEE):

The calorie goals play an important role in shaping the meal plans, aiming to establish a well-rounded diet that aligns with the User's unique profile, activity levels, and weight goals. The calculations are based on the Basal Metabolic Rate (BMR) formulas derived from the Harris-Benedict equations, refined by Miffin and St Jeor in 1990. These BMR values are then multiplied by activity factors to account for the User's day-to-day energy expenditure as illustrated in the Table below to derive the Total Daily Energy Expenditure (TDEE). However, if the User's weight goals involve gaining or losing weight, an additional 500 kcal is either added or subtracted from the TDEE for a more tailored approach.

Men	BMR = (10 × weight in kg) + (6.25 × height in cm) – (5 × age in years) + 5
Women	BMR = (10 × weight in kg) + (6.25 × height in cm) – (5 × age in years) – 161

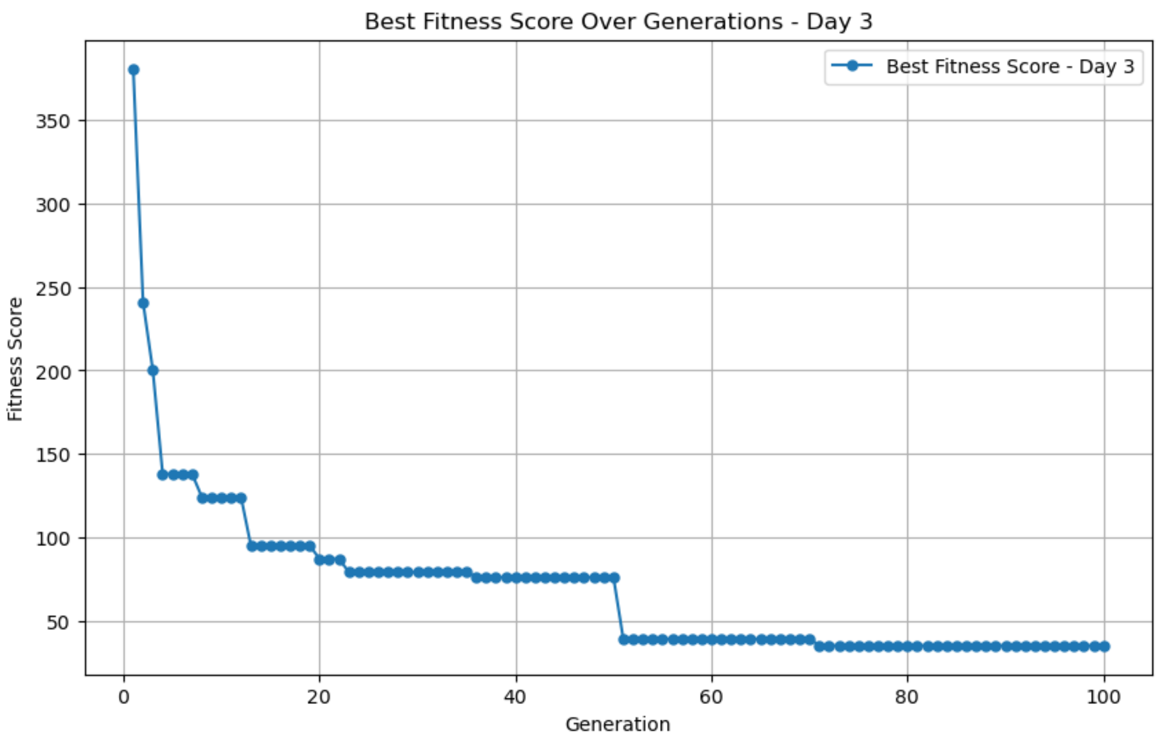
Amount of Exercise/Activity	Description	TDEE (BMR x Activity Factor)
Sedentary	Little or no exercise	BMR x 1.2
Lightly Active	Light exercise (1-3 days per week)	BMR x 1.375
Moderately Active	Moderate exercise (3-5 days per week)	BMR x 1.55
Very Active	Heavy exercise (6-7 days per week)	BMR x 1.725
Extremely Active	Extremely active (twice per day, extra heavy workouts)	BMR x 1.9

Determining Allowable Range of Food for Meal Planning Selection:

User preferences on diet restrictions as well as illness-related diet restrictions were built into the model to retain only the appropriate list of food to be recommended. User preferences encompass choices of vegetarian, halal and non-beef options to allow personalisation. Additionally, illness-related diet restrictions are integrated, encompassing considerations for appropriate oil/fat content, carbohydrate levels, protein content, fiber content and ease of digestibility. These constraints are implemented at the outset to filter the dataset, ensuring that the subsequent meal planning recommendations adhere to the User's dietary needs.

Meal Plan Recommendation for Illness Recovery:

Genetic algorithm was used to determine the most suitable meal plan based on the respective food data and with the goal to minimize a fitness function to meet the targeted calorie goals. To ensure that a sensible meal plan is recommended such that the calorie is proportionally distributed between the mains, drinks and fruit choices, a high penalty was enforced within the fitness function to ensure that at least 50% of the meal's allowable calories is allocated to the mains. An overview of the fitness score of the meal plan across generations running the GA is shown in the below plot.



Custom Meal Plan by User:

The customizable meal plan leverages on the existing food database to enable users to craft their own meal setups based on their specific calories requirements. Users begin by inputting keywords to search for desired foods and the system responds with a list of matching items from the database. The user can then review this list and choose the appropriate food item. Upon selection, the system will display the nutritional values of the chosen food and calculate the remaining calories for the meal.

Subsequently, users have the option to build up their meal plan by adding further items, where the process repeats to prompt users for keywords of desired addition. The count of total calories remaining will be updated after each addition.

Recognizing the challenges of strictly adhering to recommended food items, this iterative and interactive approach acknowledges that users may want flexibility. It allows users to design their meal plans to meet specific dietary preferences and caloric targets, offering a balanced and personalized approach to meal planning.

4.1.3 Backend

Web Sockets:

Web sockets have been used to facilitate real-time communication between the client and the server. Web sockets are a communication protocol that enables bidirectional, full-duplex communication channels over a single, long-lived connection. This ensures that both ends can send and receive data at any time, eliminating the need for re-establishing connections.

Advantages:

1. **Real-time Interaction**: The use of web sockets ensures that our chatbot interacts with users in real-time, providing immediate feedback and responses.
2. **Efficiency**: Unlike traditional HTTP requests, which require a new connection for each request-response cycle, web sockets maintain a single connection, reducing overhead and latency.

State Management:

State management is an integral part of our system. It pertains to the practice of managing and preserving the state or context of an application. By keeping track of the user's journey and interactions, our system can provide contextually relevant responses and ensure a coherent user experience.

Advantages:

1. **Contextual Responses**: By understanding the current state of the conversation, our system can understand the context of the conversation and hence it becomes easy to track the progress of user’s request at any point of time.

2. **Error Handling**: Due to the robust nature of state management, we can introduce fallback paths for error handling and can have states that exclusively represent any errors that can arise due to the user input.

Implementation of the above concepts in the code:

Web sockets and asynchronous programming forms the backbone of the chatbot used for disease diagnosis and the meal planner. For a chatbot based system, a real-time interaction is of utmost importance, and hence we chose to use the web sockets approach due to it's full duplex bi-directional nature. Further, the state management combined with the web sockets, serves as a guide to map the user request into a set of tasks and iteratively complete them.

4.2 Challenges and Solutions

4.2.1 Interactive Communication through Web Sockets

Leveraging web sockets for real-time communication and asynchronous operations presents its set of challenges. A primary concern is synchronizing conversation states with user interactions. To address this, we adopted an inheritance-driven method to define states, mirroring the principles of the factory design pattern.

4.2.2 Parsing user input for the NLP Chatbot

Given the dependency of NLP on user intent, it became imperative to refine user input to extract pertinent details (in our scenario, symptoms). Based on this refined input, the system identifies known symptoms for subsequent predictions. This identification process harnesses both syntactic (through fuzzy string comparisons) and semantic (via cosine similarity scores) approaches.

4.2.3 Meal Planner Algorithm

In the development of the recommended meal plan from a pool of over 6000 potential food items, the choice of algorithm is critical. It is essential to recognize that for the meal planner, achieving the absolute optimal solution (global optimal) is not paramount. This is particularly so as the algorithm could be required to generate multiple solutions catering to different meals and days. Computational efficiency on the other hand is crucial to ensure seamless user experience, avoiding prolonged waiting times for generation of meal recommendations.

The backtracking algorithm, with its exhaustive search through the solution space, was deemed unacceptable due to its prolonged computation time. Given the need for a quick and efficient model, the Genetic Algorithm (GA) emerged as a more viable alternative. GA allows for quick iteration as it selects only the best-performing solutions for reproduction. This approach strikes a balance between computational speed and the generation of diverse and suitable food combinations for the recommended meal plan.

5. IMPACT AND RESULTS

5.1 Quantitative

XX

5.2 Qualitative

5.2.1 Feedback from users, interview few people

xxx

5.2.2 General impact on patient care, healthcare and industry

xxx

5.3 Risk Analysis

In our order to reduce risk in launching our solution, we have streamlined our risk analysis into three primary categories. This is mainly to prepare for, anticipate, and mitigate risks which are facing the solution.

5.3.1 Technical and Compliance Risks

Data security stands at the forefront, necessitating stringent measures to protect sensitive health information and ensure compliance with global data protection regulations. The system's reliability and accuracy are equally vital, demanding flawless performance to maintain user trust and provide precise health assessments. The ability to scale effectively while maintaining this accuracy and compliance is a challenge that requires advanced, adaptable solutions.

5.3.2 Market and Operational Risks

Our market environment is highly competitive, emphasizing the need for relentless innovation and differentiation. User adoption hinges on trust in AI, making transparency and education pivotal. Operationally, the focus is on efficiency and talent retention, ensuring that our system evolves with technological advancements and that our team maintains the expertise to innovate continually.

5.3.3 Financial and Reputational Risks

Financial health hinges on sustained funding and user adoption, both influenced by market perception and the actual value provided. Our reputation is tied directly to both data integrity and the accuracy of health advice; any misstep can result in user attrition and diminished trust. Proactive financial planning and a commitment to user education and transparent communication are crucial in mitigating these risks.

5.4 Roadmap for Future

5.4.1 Future Considerations

As the system developed is positioned as an MVP, further enhancements listed below are recommended to improve the system.

- Expanding list of diseases and symptoms. Expanding our database of diseases and symptoms is imperative for a more comprehensive health platform. Recognizing potential limitations in our current list, we propose forging collaborations with clinics and hospitals to enrich the dataset. By tapping into their clinical expertise, a more exhaustive resource could be created that better serves users. This collaborative approach ensures a broader spectrum of health conditions is covered, empowering users with enhanced knowledge for improved health management.

- Expanding food choices (including health supplements). While our current dataset is comprehensive, there's an opportunity to augment it by incorporating suggested health supplements within the meal planning process. This inclusion can prove advantageous for patients undergoing recovery, providing them with a more holistic approach to nutrition. Furthermore, greater personalization and flexibility could be introduced into our meal recommendations. Unlike the current standard suggestion of a typical meal comprising mains, a fruit, and a drink, we propose exploring options that allow users to customize their meal types. This could involve breaking down a meal into smaller mains spread throughout the day, providing users with the flexibility to adapt their eating patterns according to their preferences and dietary requirements. This approach not only adds a layer of customization but also aligns with evolving preferences for varied and adaptable meal plans.

- Creating an integrated health management system designed for collaboration with clinics or hospitals. This post-care solution could be developed in partnerships with healthcare facilities, fostering a collaborative effort to create an unique and personalized experience for patients. The system could be enhanced to go beyond clinical appointments by consistently checking in on patients' well-being during recovery. This extended monitoring allows healthcare providers to stand out from others by displaying their commitment to patient well-being. Additionally, the system can be optimized to include medication information and reminders, offering patients convenient access to medical details and prescribed treatments from their consultations.

6. CONCLUSIONS

XXX

APPENDIX A – PROJECT PROPOSAL

GRADUATE CERTIFICATE: Intelligent Reasoning Systems (IRS)

PRACTICE MODULE: Project Proposal

Date of proposal:

21 September 2023

Project Title:

ISS Project – **AI-Powered Health Condition Monitoring and Nutrition Planning**

Sponsor/Client: *(Name, Address, Telephone No. and Contact Name)*

Institute of Systems Science (ISS) at 25 Heng Mui Keng Terrace, Singapore

NATIONAL UNIVERSITY OF SINGAPORE (NUS)

Contact: Mr. GU ZHAN / Lecturer & Consultant

Telephone No.: 65-6516 8021

Email: zhan.gu@nus.edu.sg

Background/Aims/Objectives:

Background: In the realm of traditional healthcare, services often adopt a one-size-fits-all approach, with limited medical support available only during infrequent appointments scheduled months apart. But what if healthcare could be a personalized and constant companion? Imagine having an AI-Powered Health Condition Monitoring and Nutrition Planning that caters uniquely to your needs, available whenever you require assistance.

Aim: Our mission is to lead a healthcare revolution by creating an AI-Powered Health Condition Monitoring and Nutrition Planning system. This is more than just another health solution; it's a transformative endeavor. Our goal is to empower individuals with a tailored health journey, arm them with tools for physical and mental well-being, and provide a wealth of personalized resources.

Objectives:

1. **Personalized Health Monitoring and Diagnosis Enhancement:** Create a personalized health monitoring and diagnosis system meticulously crafted for each user. Empower individuals to not only track their health goals and celebrate milestones but also provide them with tailored health recommendations that align precisely with their unique needs and health conditions. Bid farewell to generic advice, and usher in a system that seamlessly connects users to a wealth of health knowledge, uniquely customized to enhance their well-being.
2. **Harness the Power of AI in Nutrition Planning:** Picture an AI-powered nutrition planner that offers personalized meal plans, considering your health conditions, dietary preferences, and allergies. It's like having a personal nutritionist who understands your unique needs, guiding you toward healthier dietary choices seamlessly.

Features overview:

1. Health Condition Monitoring:

- **Building Blocks:**
  - Data Integration
  - Alerts and Notifications
- **Algorithms:**
  - Time Series Analysis: For monitoring of health status (duration of symptoms) and recovery progress.
  - Decision Trees: Disease/Illness identification based on user input symptoms predicted through illness and associated reported symptoms datafile. For risk assessment and recommending appropriate actions based on health data.

2. AI-Powered Nutrition Planning:

- **Building Blocks:**
  - Nutrition Database
  - User Dietary Profile
  - Meal Plan Generation
- **Algorithms:**
  - Collaborative Filtering: To suggest meal plans based on similar user dietary profiles.
  - Genetic Algorithms: For optimizing meal plans based on dietary preferences and health conditions.
  - Food Recommender System: Using item-based collaborative filtering to recommend foods based on dietary restrictions.

3. Personalized Medication Reminders:

- **Building Blocks:**
  - Medication Database
  - User Medication Schedule
  - Notifications System
- **Algorithms:**
  - Rule-based Systems: For scheduling medication reminders based on user prescriptions.
  - Bayesian Networks: To estimate the likelihood of medication adherence and optimize reminder timing.

Requirements Overview:

❖ **<u>Research ability</u>**

❖ **Importance:** Research ability is crucial for staying updated with the latest advancements in healthcare, artificial intelligence, and data science. It enables the team to make informed decisions, select appropriate algorithms, and adapt to evolving medical guidelines.

❖ **Key Responsibilities:**

➢ **Literature Review:** Regularly review academic papers, articles, and industry reports related to healthcare, AI, and decision support systems.

➢ **Clinical Understanding:** Develop an understanding of medical concepts, terminologies, and healthcare regulations to design a system that aligns with healthcare standards.

➢ **Data Analysis:** Analyze healthcare data to identify trends, patterns, and correlations that can enhance decision support.

❖ **<u>Programming ability</u>**

❖ **Importance:** Programming ability is the foundation for developing and maintaining the AI-Driven Personal Health Assistant. It's essential for implementing algorithms, building user interfaces, and ensuring system reliability and scalability. ❖ **Key Responsibilities:** ➢ **Software Development:** Write clean, efficient, and maintainable code for the various components of the system, including the front-end and back-end. ➢ **Algorithm Implementation:** Implement machine learning and AI algorithms for health monitoring, recommendation engines, and data analysis. ➢ **Testing and Debugging:** Perform rigorous testing and debugging to ensure the system's functionality and security. ❖ **<u>System integration ability</u>** ❖ **Importance:** System integration ability is crucial for ensuring that the AI-Driven Personal Health Assistant can seamlessly connect with various data sources, third-party services, and healthcare systems. ❖ **Key Responsibilities:** ➢ **Data Integration:** Integrate data from wearables, medical records, and other sources into a unified system. ➢ **API Integration:** Connect with external APIs for services like telehealth, content delivery, and medication databases. ➢ **Security Integration:** Implement robust security protocols to protect user data during transmission and storage.
Resource Requirements (please list Hardware, Software and any other resources) Hardware proposed for consideration: ● GPU Software proposed for consideration: ● Web GUI ● NLP tools ● **Machine Learning Models:** For vision, speech, and NLP tasks. ● Databases, e.g SQL, NoSQL ● Chat-bots, e.g. ChatterBot ● Cloud computing/server, e.g. Amazon, Google, IBM, Azure, etc. ● Application container, e.g. Docker
Number of Learner Interns required: (Please specify their tasks if possible) A team of four to six project members.

Methods and Standards:

Procedures	Objective	Key Activities
Requirement Gathering and Analysis	The team should meet with ISS to scope the details of the project and ensure the achievement of business objectives.	1. Gather & Analyze Requirements 2. Define internal and External Design 3. Prioritize & Consolidate Requirements 4. Establish Functional Baseline
Technical Construction	· To develop the source code in accordance with the design. · To perform unit testing to ensure the quality before the components are integrated as a whole project	1. Setup Development Environment 2. Understand the System Context, Design 3. Perform Coding 4. Conduct Unit Testing
Integration Testing and acceptance testing	To ensure interface compatibility and confirm that the integrated system hardware and system software meets requirements and is ready for acceptance testing.	1. Prepare System Test Specifications 2. Prepare for Test Execution 3. Conduct System Integration Testing 4. Evaluate Testing 5. Establish Product Baseline
Acceptance Testing	To obtain ISS user acceptance that the system meets the requirements.	1. Plan for Acceptance Testing 2. Conduct Training for Acceptance Testing 3. Prepare for Acceptance Test Execution 4. ISS Evaluate Testing 5. Obtain Customer Acceptance Sign-off
Delivery	To deploy the system into the production (ISS standalone server) environment.	1. Software must be packed by following ISS's standard 2. Deployment guideline must be provided in ISS production (ISS standalone server) format 3. Production (ISS standalone server) support and troubleshooting process must be defined.

Team Formation & Registration

Team Name: **AI-Powered Health Condition Monitoring and Nutrition Planning**
Project Title (repeated): **AI-Powered Health Condition Monitoring and Nutrition Planning**
System Name (if decided): **AI-Powered Health Condition Monitoring and Nutrition Planning**
Team Member 1 Name: YATHARTH MAHESH SANT
Team Member 1 Matriculation Number: A0286001R
Team Member 1 Contact (Mobile/Email): e1221813@u.nus.edu
Team Member 2 Name: KRISTOFER ROOS
Team Member 2 Matriculation Number: A0285949A
Team Member 2 Contact (Mobile/Email): e1221761@u.nus.edu
Team Member 3 Name: CHUA KIAN YONG KENNY
Team Member 3 Matriculation Number: A0056377W
Team Member 3 Contact (Mobile/Email): e1216683@u.nus.edu
Team Member 4 Name: ZHANG YUSEN
Team Member 4 Matriculation Number: A0285839H
Team Member 4 Contact (Mobile/Email): e1221651@u.nus.edu
Team Member 5 Name: HAO ZHENMAO
Team Member 5 Matriculation Number: A0285960R
Team Member 5 Contact (Mobile/Email): e1221772@u.nus.edu

APPENDIX B – MAPPED SYSTEMS FUNCTIONALITIES AGAINST KNOWLEDGE, TECHNIQUES AND SKILLS OF MODULE COURSES

MODULE COURSE	KNOWLEDGE/TECHNIQUES/SKILLS APPLIED
Machine Reasoning (MR)	Decision Automation techniques: - Abductive Reasoning (conditional probabilities): Symptom suggestions based on symptom-disease associations - Knowledge Elicitation (rule based system): Providing associated precautions with diseases
Reasoning Systems (RS)	Knowledge Discovery techniques: - Analytic Tasks (retrieval reasoning system) - Symptom checker Business Resource Optimization techniques: - Synthetic Tasks (generative reasoning system) - Meal plan generation utilizing evolutionary computing techniques
Cognitive Systems (CS)	Cognitive Techniques: - NLP - semantic similarity - Chatbot interface

APPENDIX C – INSTALLATION AND USER GUIDE

Installation of Vue.js Front end:

1) Download Node.js from the official website.
2) Check the installation using command npm –v
3) In the frontend directory, run the command: “npm i”
4) To start the development server, run command: “npm run dev”. This will start the development server at localhost:3000

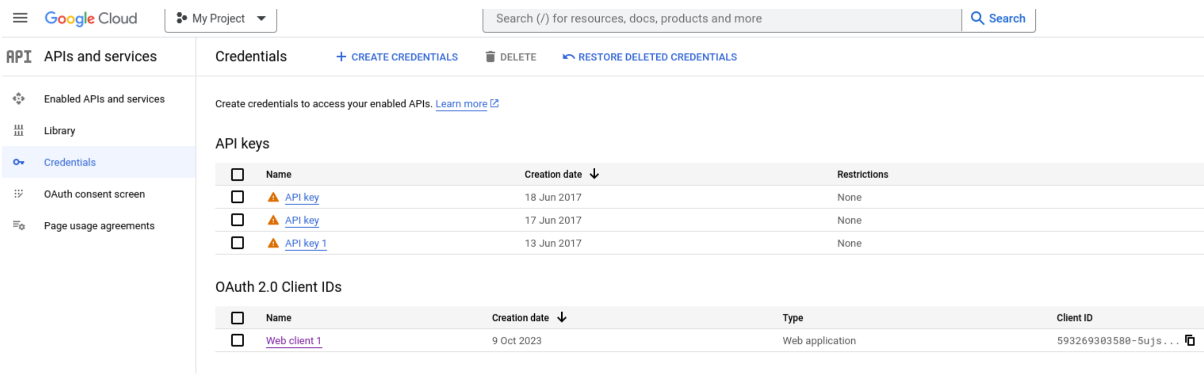
Installation of Django backend:

1) Download SQLite and Anaconda from the respective websites.
2) (Optional) Create a new virtual environment using the command: “conda create ENVNAME python=PYTHON-VERSION”, here ENVNAME is your environment name and PYTHON-VERSION is the desired python version (for ex: conda create myenv python=3.8).
3) To optionally clone an existing environment use the command: “conda create ENVNAME python=PYTHON-VERSION –clone EXISTING”.
4) Activate the virtual environment as: conda activate ENVNAME
5) Install pip if not installed already using command: “conda install pip”
6) In the backend directory, install all the required packages using the command: “pip install -r requirements.txt”
7) Install the spacy language model using the command: “python -m spacy en_core_web_md”.
8) Run the following commands to do the database migrations:

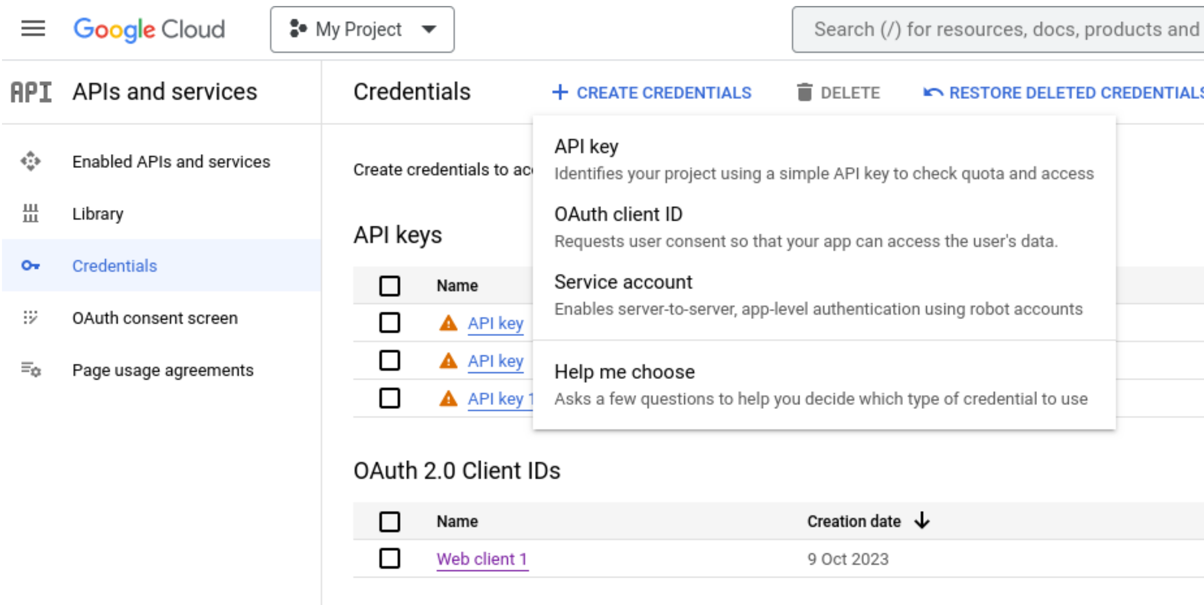
   python manage.py makemigrations

   python manage.py migrate

9) Create a superuser using the command: python manage.py createsuperuser and follow the prompts ahead.
10) Setup the Google Oauth locally as follows:
    a. Go to the URL: https://console.cloud.google.com/apis/ and create a new project.
    b. Under the APIs and Services tab click on Credentials:



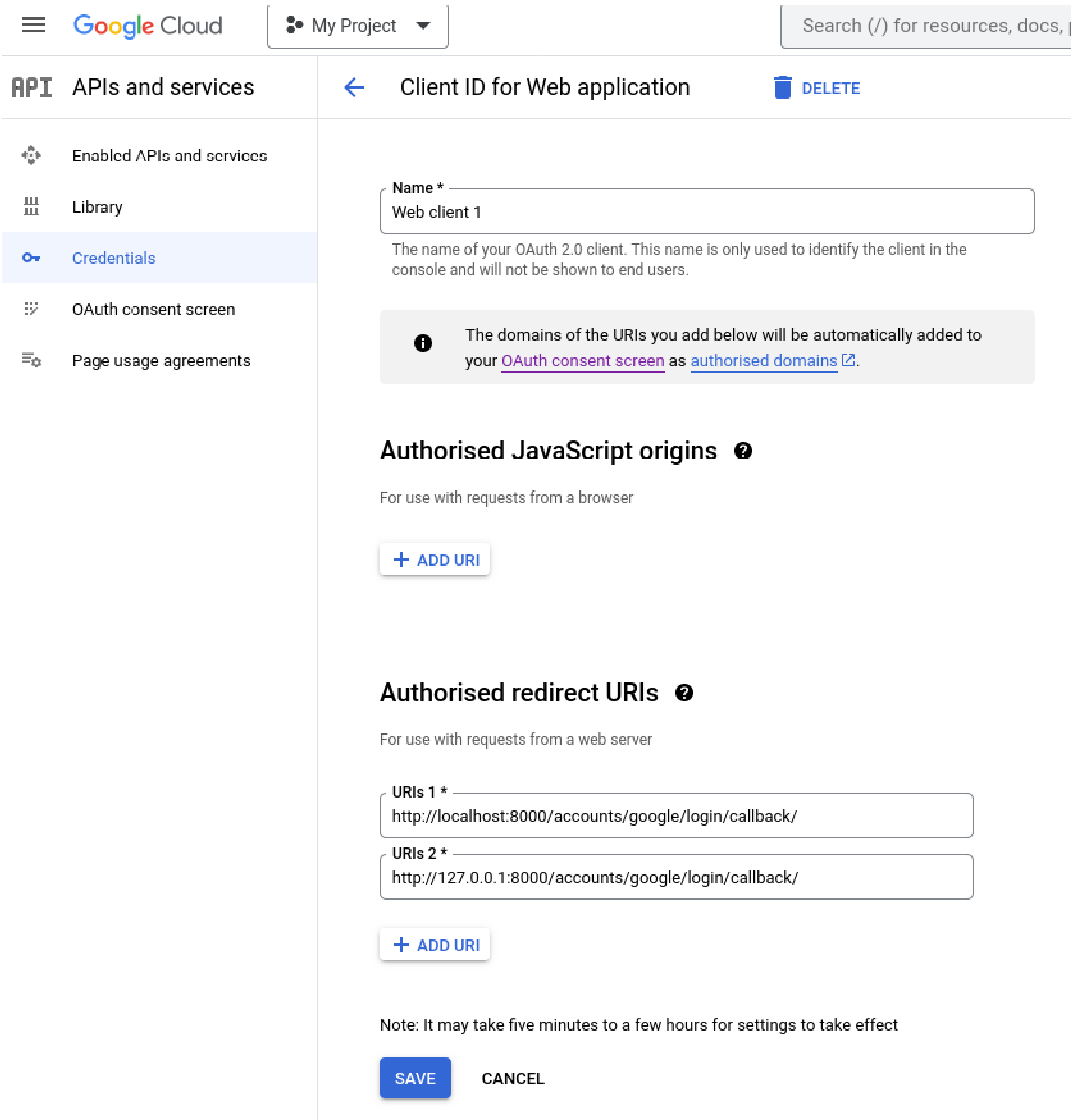
c. Then click the Create Credentials button at the top and select the Oauth Client ID:



d. Select the application type as web application, give a suitable name and in the Authorised redirect URIs, add the following URLs:

http://127.0.0.1:8000/accounts/google/login/callback/

http://localhost:8000/accounts/google/login/callback/



Take a note of the OAuth of the Client ID and Client Secret (would be on the right of the page)

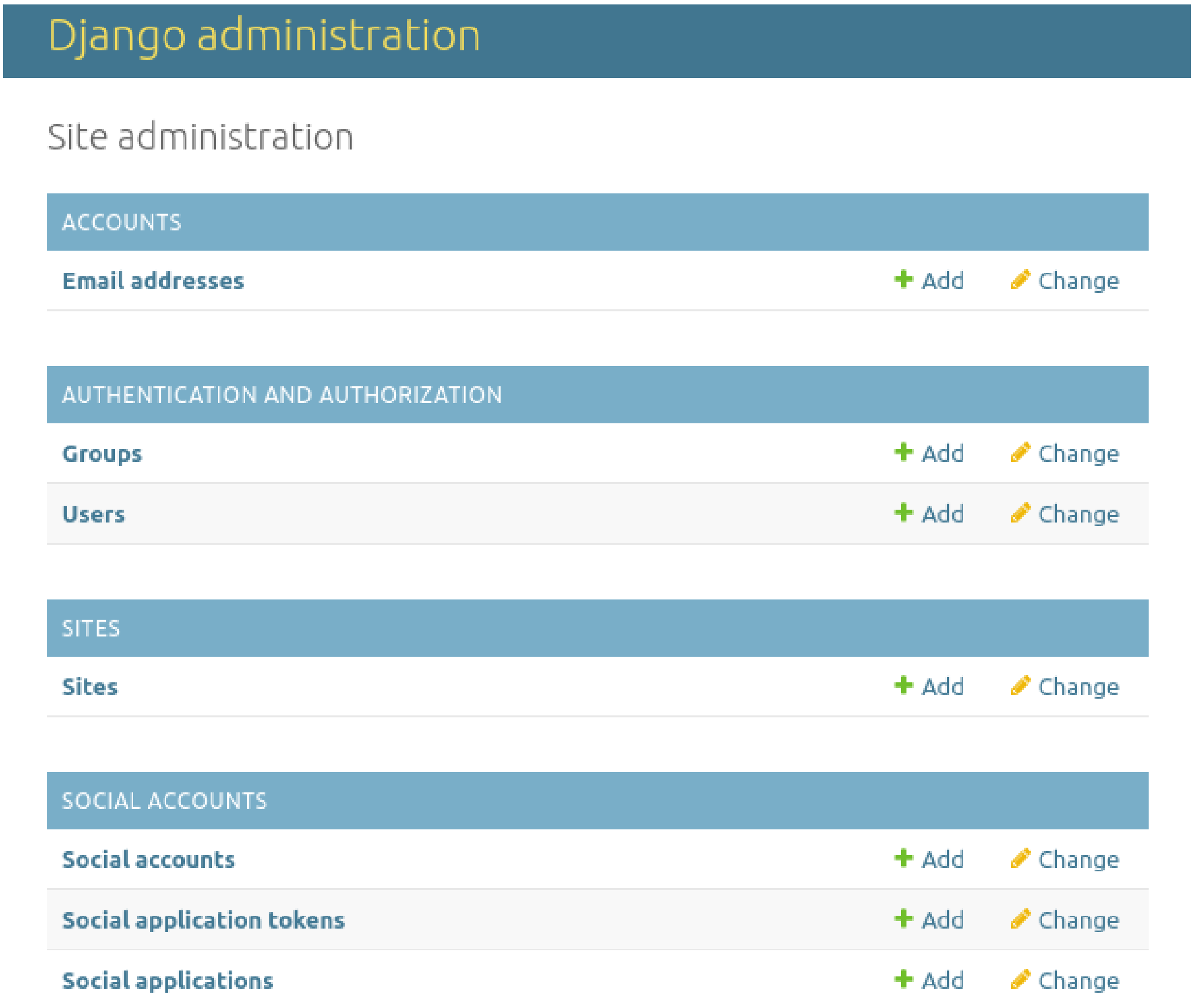
11)Add the client ID and the Client Secret in the settings.py file inside ai_health_monitoring directory as follows:

```
SOCIALACCOUNT_PROVIDERS = {
    'google': {
        'APP': {
            'client_id': '<<YOUR-CLIENT-ID>>',
            'secret': '<<YOUR-SECRET-KEY>>',
        }
    }
}
```

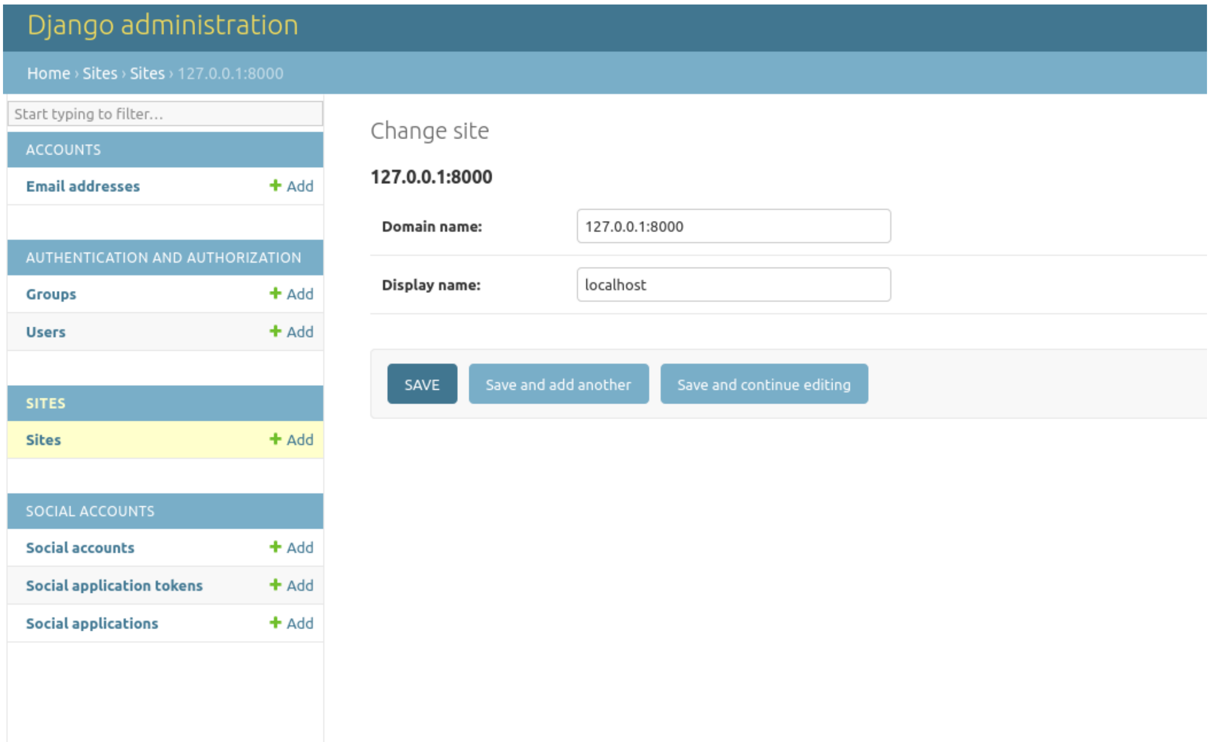
12) Launch the local development server for django using the command: “python manage.py runserver”

12) Navigate to the admin url: localhost:8000/admin OR 127.0.0.1:8000/admin

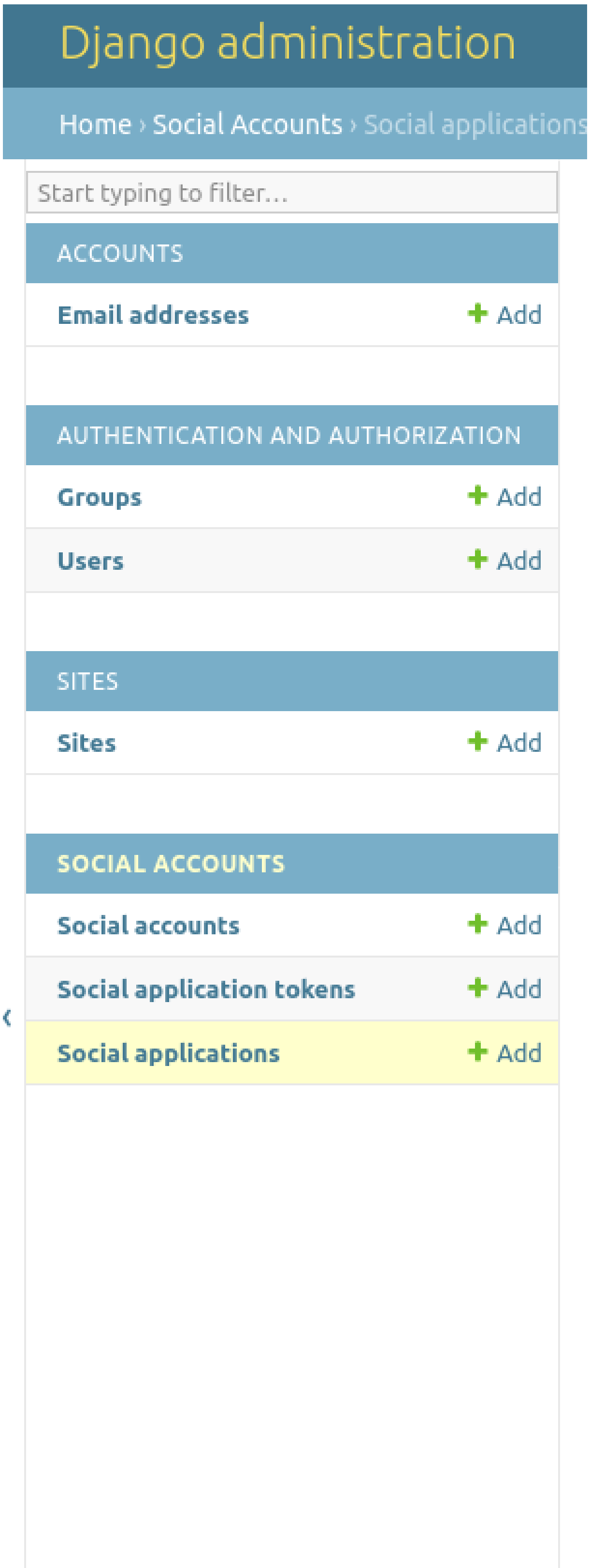
13) Use the credentials created in step 9 above.

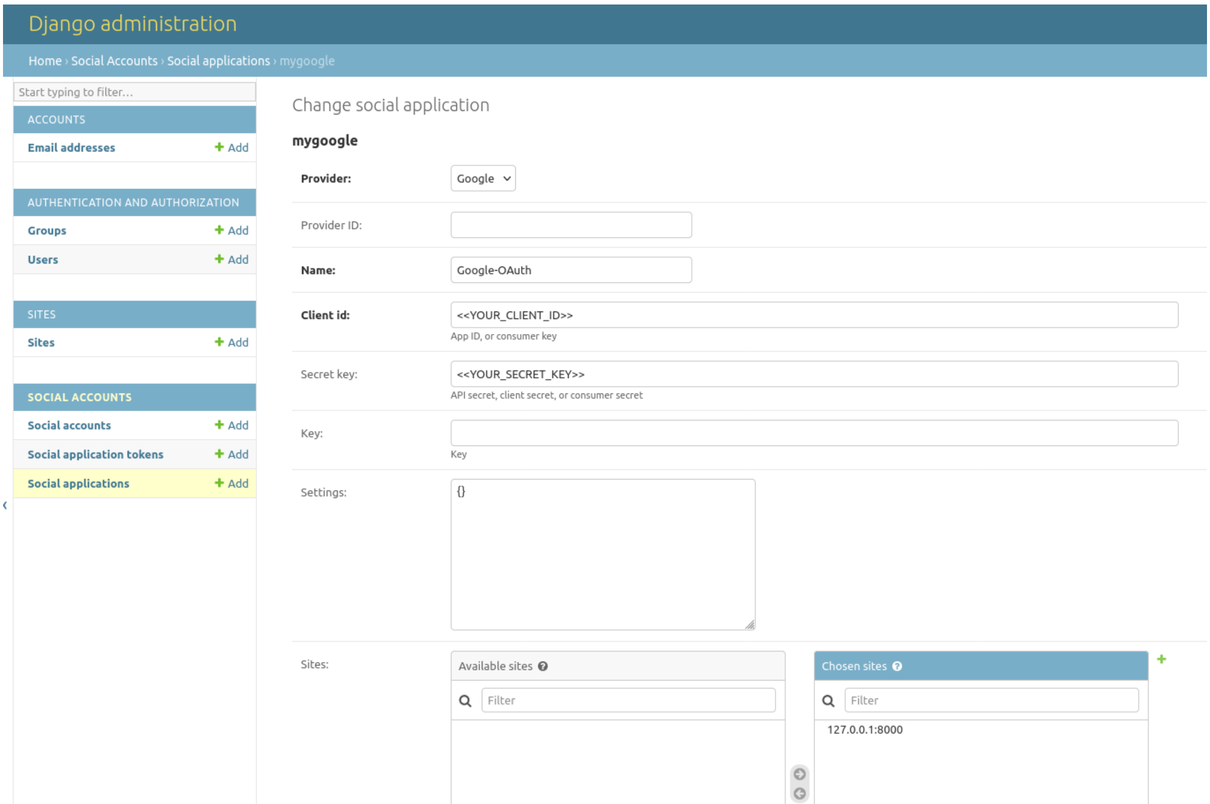


14) Go to sites tab and create a site (display name can be anything here and then click on save):



15) Go to the social applications tab and create a new social application and fill the details (Make sure to use the same client ID and client secret retrieved as part of step 10, choose the created site from list of sites and click on save) :


Django administration
Home › Social Accounts › Social applications
Start typing to filter…
ACCOUNTS
Email addresses
+ Add
AUTHENTICATION AND AUTHORIZATION
Groups
+ Add
Users
+ Add
SITES
Sites
+ Add
SOCIAL ACCOUNTS
Social accounts
+ Add
Social application tokens
+ Add
Social applications
+ Add



16) Logout from the django admin console, restart the server using command: "python manage.py runserver". The google authentication should work now.

Note: Make sure the django server is currently running on the port 8000, else run it as "python manage.py runserver 8000". This must be the same port number added to redirection URIs in the google cloud API.

APPENDIX D – INDIVIDUAL PROJECT REPORT

YATHARTH MAHESH SANT (A0286001R)

Personal Contribution

Primarily involved in the development of disease diagnosis part of our project. Developed the multi label classification and tried out different models (namely: Decision Tree, Random Forest, SVM, Logistic Regression, Gaussian Naive Bayes). Tried two main different techniques for multi-label classification: OneVsRest and ClassifierChain Approach.

Also developed the NLP Chatbot using spacy and nltk. Gone through multiple revisions of the same and introduced features like suggesting relevant/likely symptoms to user based on current symptoms, introducing the additional fuzzy string matching (for approximate string matching) and the semantic similarity (cosine similarity).

Developed the backend for the diagnosis part using asynchronous web scoket approach in Django. Implemented a state based approach for smooth transitions as and when the conversation between the bot and the user proceeds further and also integrated the Google Authentication portion. Developed UI components for the response returned by the bot primarily for the disease diagnosis part.

Learning Journey

Learned a lot about classification in general and got introduced to the new concept of multi label classification. Initially, tried out using a simple multi class approach and soon realized the mistake by having a look at the high accuracy of all models (even on test dataset), but non agreement of metrics like ROC AUC, log loss. Later on realized that symptoms can be mapped to multiple diseases and hence need a multi label method here. Also learned a lot on how to handle high dimensional data and still try to maintain the interpretability.

Gained a lot of understanding in the cognitive systems part especially when it comes to NLP and chatbot development. Understood that even a simple chatbot implementation needs lot of input pre-processing to understand the user intents and identify slots. Got to learn about how to handle textual data and how to focus on the named entities we are interested in. Due to some of the aspects involved in this chatbot, it also enhanced the mathematical foundations too, for instance, to suggest relevant symptoms the approach was to make use of conditional probabilities, so overall got a deeper understanding of mathematical concepts applied to real life.

Got a deeper understanding of the web development standards and was able to develop a truly scalable system using asynchronous programming and the web sockets standard. Learnt some design patterns too and was able to incorporate a loose version of the factory design pattern for the state transition process. Also got to explore on how to facilitate good interaction with the web sockets using javascript and client side interaction.

Application of Knowledge and Skills

An inquisitive mindset to dig deeper into the classification part. Explore all possible models that I could in less time and given the nature of data. Was able to use lots of learnings from the course related to the classification models and added more information to my own knowledge base as well. Further, was able to also demonstrate some of the software engineering skills learned in the past for developing the interactive chatbot platform. Was able to manage well with the NLP part too, given the knowledge learnt from the cognitive systems course.

KRISTOFER ROOS (A0285949A)

Personal Contribution

xxx

Learning Journey

xxx

Application of Knowledge and Skills

xxx

ZHANG YUSEN (A0285839H)

Personal Contribution

xxx

Learning Journey

xxx

Application of Knowledge and Skills

xxx

HAO ZHENMAO (A0285960R)

Personal Contribution

xxx

Learning Journey

xxx

Application of Knowledge and Skills

xxx

CHUA KIAN YONG KENNY (A0056377W)

Personal Contribution

In the development of this project, my primary contributions spans across data acquisition, research into relevant equations for deriving suitable measurements of meal plans, and algorithmic model development for the meal planner.

The initial phase involved meticulous curation and review of the dataset to categorize and include only sensible food items to be recommended.

The development of the meal planner model included the establishment of food choices filters to allow customizability in accordance with dietary preferences and requirements. Additionally, the formulation of fitness function was crucial for evaluating and optimizing the meal plans.

Learning Journey

The learning journey throughout this project was enriching. One key takeaway was the understanding of selecting appropriate resource optimization techniques based on the specific problem. Understanding the strengths and limitations of each algorithm played a pivotal role in crafting an efficient and effective meal planner.

Furthermore, the significance of data pre-processing is evident in influencing the success of the meal planner as could be observed from initial runs of the application where impractical food items were recommended. This realization emphasized the need for a meticulous pre-processing phase to ensure the dataset's integrity and, consequently, the practicality of the meal plans generated.

Another important lesson learned was that an envisaged idea might not be inherently user-centric. This recognition prompted a continuous loop of development to enhance the system. Specifically, the transition from a rigid meal planner to developing a more customizable system emerged as a key improvement. Acknowledging the practical challenges users face in strictly adhering to recommended meal plans led to the incorporation of this customizability. This shift aligns more closely with the realities of available meal choices to the user.

Application of Knowledge and Skills

An adaptive mindset to improve the system continuously and iteratively is important and could be applied to future projects. This would allow us to adopt a practical approach where a user-centric design could be achieved.

Additionally, the emphasis on meticulous data preparation underscores the importance of ensuring data quality in any data-driven project. A good understanding of the data is critical and thereby for future projects, the involvement of domain-experts to provide the necessary guidance and clarifications is of utmost importance in ensuring a successful project.

REFERENCES

1. Kasem, M., & Hassanein, E. E. (2014). Cloud Business Intelligence Survey. International Journal of Computer Applications, 90(1), 23–28. https://doi.org/10.5120/15540-4266
2. Fleming, D. G. (1954). William H. Welch and the Rise of Modern Medicine. Journal of the American Medical Association, 155(15), 1377. https://doi.org/10.1001/jama.1954.03690330081034
3. Healthy People 2020. Washington, DC: U.S. Department of Health and Human Services; 2010. (Accessed September 24, 2023 at http://www.healthypeople.gov/2020/default.aspx)
4. Fani Marvasti F, Stafford RS. From sick care to health care--reengineering prevention into the U.S. system. N Engl J Med. 2012 Sep 6;367(10):889-91. doi: 10.1056/NEJMp1206230. PMID: 22931257; PMCID: PMC4339086.
5. Stanford University (Stanford Medicine). (2020, September 24). Projected growth in global healthcare data volume 2020. https://www.statista.com/statistics/1037970/global-healthcare-data-volume/?locale=en
6. Pranay Patil (2020). Disease Symptom Prediction. Kaggle https://www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset
7. HPB Energy & Nutrient Composition of Food. https://focos.hpb.gov.sg/eservices/ENCF/foodsearch.aspx
8. Leo Lee (2021). Diet Generation as Sequence. https://github.com/Leo-Lee92/Diet-Generation-As-Sequence/tree/master
9. Population and population structure. (n.d.). Base. https://www.singstat.gov.sg/find-data/search-by-theme/population/population-and-population-structure/latest-data
10. Hajat C, Stein E. The global burden of multiple chronic conditions: A narrative review. Prev Med Rep. 2018 Oct 19;12:284-293. doi: 10.1016/j.pmedr.2018.10.008. PMID: 30406006; PMCID: PMC6214883.