****

**Department of Computer Science and Engineering**

*A Project Report on*

**Multi-Factor based Nutrition Management System and Recipe Recommendation Engine**

*Submitted in partial fulfillment of the requirements for the award of degree in*

**Bachelor of Engineering in Computer Science & Engineering**

*By*

|  |  |
| --- | --- |
| Aravind Shreyas Ramesh | 1MS18CS025 |
| Dheeraj Bhat | 1MS18CS040 |
| Divya | 1MS18CS043 |
| Gaurav V | 1MS18CS046 |

Under the guidance of

Dr. Shilpa Shashikant Chaudhari

Associate Professor

**M S RAMAIAH INSTITUTE OF TECHNOLOGY**

**(Autonomous Institute, Affiliated to VTU)**

**BANGALORE-560054**

**www.msrit.edu**

**July 2022**

****

**Department of Computer Science and Engineering**

**CERTIFICATE**

Certified that the project work entitled “**Multi-Factor based Nutrition Management System and Recipe Recommendation Engine**” carried out by Aravind Shreyas Ramesh – 1MS18CS025, Dheeraj Bhat – 1MS18CS040, Divya – 1MS18CS043, Gaurav V – 1MS18CS046 are bona fide students of M.S. Ramaiah Institute of Technology, Bengaluru in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi during the year 2021-22. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said degree.

**Head of the Department**

**Dr. Annapurna P. Patil**

**Project Guide**

**Dr. Shilpa Shashikant Chaudhari**

**External Examiners: -**

**Signatures with Date**

**Name of the Examiners**

**1.**

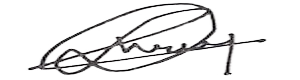
**2.**

****

**Department of Computer Science and Engineering**

**DECLARATION**

We, hereby, declare that the entire work embodied in this project report has been carried out by us at M.S. Ramaiah Institute of Technology, Bengaluru, under the supervision of **Dr. Shilpa Shashikant Chaudhari, Associate Professor,** Dept of CSE. This report has not been submitted in part or full for the award of any diploma or degree of this or to any other university.



Dheeraj Bhat

1MS18CS040

Aravind Shreyas Ramesh

1MS18CS025

Shape, arrow

Description automatically generatedA planet in space

Description automatically generated with low confidence**ACKNOWLEDGEMENT**

Gaurav V

1MS18CS046

Divya

1MS18CS043

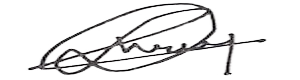
We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. We would like to express our profound gratitude to the Management and **Dr. N.V.R Naidu** Principal, M.S.R.I.T, Bengaluru for providing us with the opportunity to explore our potential.

We extend our heartful gratitude to our beloved **Dr. Annapurna P Patil**, HoD, Computer Science and Engineering, for constant support and guidance.

We whole heartedly thank our project guide **Dr. Shilpa Shashikant Chaudhari,** for providing us with the confidence and strength to overcome every obstacle at each step of the project and inspiring us to the best of our potential. We also thank her for her constant guidance, direction, and insight during the project.

This work would not have been possible without the guidance and help of several individuals who in one way or another contributed their valuable assistance in preparation and completion of this study.

Finally, we would like to express sincere gratitude to all the teaching and non-teaching faculty of CSE Department, our beloved parents, seniors, and my dear friends for their constant support during the course of work.

A picture containing arrow

Description automatically generated

Dheeraj Bhat

1MS18CS040

Aravind Shreyas Ramesh

1MS18CS025

Shape, arrow

Description automatically generatedA planet in space

Description automatically generated with low confidence**ABSTRACT**

Gaurav V

1MS18CS046

Divya

1MS18CS043

Nutrient management in the context of this research aims to quantize the consumption of essential nutrients in an efficient format such that it leads to a healthy and balanced lifestyle. Increased consciousness towards one’s health has recently been in the limelight which creates the need for an intelligent system specially customized for the individual that can analyse your consumption’s quality and suggest options that could essentially fulfil your body’s need to lead a healthy lifestyle. The research’s main goal is to create an intelligent recipe recommender that would aid in the development of a diet that allows all users to make healthy choices in their daily lives while still enjoying food and keeping healthy. The recommender system once implemented as a mobile or web application, can help users who have nutritional deficiencies to maintain a healthy well-balanced diet by suggesting various recipes to the users in video format with additional relevant information which will improve the user’s well-being and quality of life.

#### LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **FIGURE NUMBER** | **TITLE** | **PAGE NUMBER** |
| 1. | Agile software development life cycle | 12 |
| 2. | Recommendation engine architecture | 46 |
| 3. | Mobile application architecture | 47 |
| 4. | Secure user authentication | 48 |
| 5. | Simple error detection and warning system | 49 |
| 6. | User data upload and secure sign out | 49 |
| 7. | Personalized recipes for the user | 50 |
| 8. | Data flow diagram | 51 |
| 9. | Comparing standard nutrition values and user’s profile to detect deficiencies | 57 |
| 10. | Detecting ingredients required for the user | 58 |
| 11. | Obtaining food products with the specific set of ingredients | 60 |
| 12. | Scraping nutritional value of cooked products | 61 |
| 13. | Obtaining best recommendations for user | 62 |
| 14(a). | High Calorie User | 67 |
| 14(b) | Low Protein User | 67 |
| 15. | Results from K-Means Clustering | 68 |
| 16. | Standard vs Recommended values for protein and carbohydrates (in mg) | 69 |
| 17. | Standard vs Recommended values for saturated fat (in mg) | 69 |
| 18. | Published paper certificate | 76 |

#### aa

#### TABLE OF CONTENTS

|  |  |
| --- | --- |
| **Declaration** | 3 |
| **Acknowledgements** | 4 |
| **Abstract** | 5 |
| **List of Figures** | 6 |

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Title** | **Page No.** |
|  |  |  |
| **1** | **INTRODUCTION** |  |
| **1.1** | General Introduction | 9 |
| **1.2** | Problem Statement | 10 |
| **1.3** | Objectives of the project | 10 |
| **1.4** | Project Deliverables | 10 |
| **1.5** | Current Scope | 10 |
| **1.6** | Future Scope | 11 |
|  |  |  |
| **2** | **PROJECT ORGANIZATION** |  |
| **2.1** | Software Process Model | 12 |
| **2.2** | Roles and Responsibilities | 13 |
|  |  |  |
| **3** | **LITERATURE SURVEY** |  |
| **3.1** | Introduction | 14 |
| **3.2** | Related Works with the citation of the References | 14 |
| **3.3** | Conclusion of Survey | 39 |
|  |  |  |
| **4** | **PROJECT MANAGEMENT PLAN** |  |
| **4.1** | Schedule of the Project | 40 |
| **4.2** | Risk Identification | 40 |
|  |  |  |
| **5** | **SOFTWARE REQUIREMENT SPECIFICATIONS** |  |
| **5.1** | Product Overview | 42 |
| **5.2** | External Interface Requirements | 43 |
| **5.3** | Functional Requirements | 44 |
|  |  |  |
| **6** | **DESIGN** |  |
| **6.1** | Introduction | 46 |
| **6.2** | Architecture Design | 46 |
| **6.3** | Graphical User Interface | 48 |
| **6.4** | Data flow diagram | 51 |
| **6.5** | Conclusion | 51 |
| **7** | **IMPLEMENTATION** |  |
| **7.1** | Tools Introduction | 52 |
| **7.2** | Technology Introduction | 53 |
| **7.3** | Overall view of the project in terms of implementation | 54 |
| **7.4** | Explanation of Algorithm and how it is being implemented | 55 |
| **7.5** | Information about the implementation of Modules | 56 |
| **7.6** | Conclusion | 63 |
|  |  |  |
| **8** | **TESTING** | 64 |
| **8.1** | Introduction | 64 |
| **8.2** | Testing Tools and Environment | 65 |
| **8.3** | Test cases |  |
|  |  |  |
| **9** | **RESULTS & PERFORMANCE ANALYSIS** |  |
| **9.1** | Result Snapshots | 67 |
| **9.2** | Comparison results | 69 |
| **9.3** | Result analysis | 70 |
|  |  |  |
| **10** | **CONCLUSION & SCOPE FOR FUTURE WORK** | 72 |
|  |  |  |
| **11** | **REFERENCES** | 73 |
|  |  |  |
| **12** | **APPENDIX** |  |
| **12.1** | Published paper certificate | 76 |

**Chapter 1**

**INTRODUCTION**

**1.1 General Introduction**

Nutrition is defined as the process of providing or obtaining the food necessary for health and growth. Without optimal nutrition, organisms can grow weak, sick and at the very worst can even die. Furthermore, without meeting sufficient nutritional needs, humans can miss developmental milestones and cannot put their bodies through the daily mental and physical tasks.

Recipe recommendation involves ranking and suggesting relevant food products and recipes as outputs by taking various types of inputs such as nutritional values, ingredients, and preferences of the user. This research aims to detect nutritional shortcomings of a user by taking various inputs that can easily be obtained through standard blood test analysis and overcome deficiencies if detected by recommending food and their recipes using an intelligent algorithm.

As computers have become more popular and with the exponential rise in the context of the technical industry, there has been an increase in the number of people who are using computers. As a result of the tremendous development in the use of IT, the landscape around health awareness, living practices, and consumer behaviour’s has changed dramatically.

Food consumption plays a central role in human race survival and there exists a need for recipe recommendation systems that can detect nutrition deficiencies to aid nutrition experts and researchers. This research attempts to develop such a system which may potentially be used by professionals. It also tries to include features for smart chefs in Indian kitchens.

**1.2 Problem Statement**

Development of a smart recipe recommendation engine which provides the user with a list of personalized recipes in video and text format based on the user’s nutritional profile, age, gender and preferences.

**1.3 Objectives of the project**

The project's main goal is to create an intelligent recipe recommender that would aid in the development of a diet that allows all users to make healthy choices in their daily lives while still enjoying food and keeping healthy.

The main objectives of this project are :

* Develop an algorithm that maps the required nutrients tailored for every user to the information put in by them like age, gender, activity levels, diseases and allergies and personal health goals.
* Develop a classification model that can classify and output food groups that are rich in specific groups of nutritional values.
* Develop a ranking system that maps the user inputs explaining their preferences and scrapes the web for recipes for the right diet.

**1.4 Project Deliverables**

A mobile application with a good user interface and design which allows the user to input information such as age, gender, nutritional values, and preferences and provide the user with a list of recipes obtained from a personalized recipe recommendation engine hosted on a server.

**1.5 Current Scope**

The recommender system once implemented as a mobile application, can help users who have nutritional deficiencies to maintain a healthy well balanced diet by suggesting various recipes to the users in video format with additional relevant information which will improve the user’s well-being and quality of life.

**1.6 Future Scope**

In the future, the application can be enhanced in a way where the user can use the camera on their phone to capture the ingredients present at home and obtain recipes containing only those ingredients while having good nutritional value. Along with this, users can also search for general videos based on a particular deficiency to improve their nutritional profile.

**Chapter 2**

**PROJECT ORGANISATION**

**2.1 Software process model**

In order to successfully complete  this project, we have adopted the agile software development model. Agile is an iterative approach to project management and software development that helps teams deliver value to their customers faster. Instead of betting everything on a "big bang" launch, an agile team delivers work in small, but consumable, increments. Requirements, plans, and results are evaluated continuously so teams have a natural mechanism for responding to change quickly.

Logo

Description automatically generated

Fig. 1. Agile software development life cycle

The main advantages of agile software model is as follows:-

1. Project is divided into short and transparent iterations.
2. It minimizes the risk of software development.
3. The correctness of functional requirement is implemented into the development process.

We tested the application out by each functionality consulted with our mentor and made changes if and when required to obtain and verify robust and accurate results.

**2.2 Roles and Responsibilities**

Explained below is the structure of roles and responsibilities to maintain a holistic view of the project while still being able to maintain a fair degree of parallelism to produce a high-quality deliverable on time.

|  |  |  |  |
| --- | --- | --- | --- |
| Sl. No. | USN | Team Member | Roles and Responsibility |
| 1. | 1MS18CS025 | Aravind Shreyas  Ramesh | Recipe Recommendation Approach, Data collection & preparation, Recipe Generation Algorithm, UX approach |
| 2. | 1MS18CS040 | Dheeraj Bhat | Recipe Recommendation Approach, Data collection & preparation, Recipe Generation Algorithm, UX approach |
| 3. | 1MS18CS043 | Divya | Recipe Recommendation Approach, UX approach, UI Design, Flutter application, Backend flask application development & deployment |
| 4. | 1MS18CS046 | Gaurav V | Recipe Recommendation Approach, UX approach, UI Design, Flutter application, Backend flask application development & deployment |

**Chapter 3**

**LITERATURE SURVEY**

**3.1 Introduction**

For literature survey various research papers were obtained from IEEE, Research Gate, and other major publications. Each paper was thoroughly read and analyzed for its pros and cons. The summaries are given below in the form of a table and the references of the paper are also cited.

**3.2 Related Works with the citation of the References**

|  |  |  |  |
| --- | --- | --- | --- |
| **Reference** | **Year** | **Title** | **Results / Observations** |
| **[1]** | 2013 | Personalized Ubiquitous Diet Plan Service Based on  Ontology and Web Services | As computers have become more popular and with the exponential rise in the context of the technical industry, there has been an increase in the number of people who are using computers. As a result of the tremendous development in the use of IT, the landscape around health awareness, living practices, and consumer behaviors has changed dramatically. In the meantime, demand continues to grow for information- and knowledge-based healthcare services. People have begun to pay attention to their health and well-being because of this. The goal of this study was to create and test an ontology-based system for dietary recommendations based on standard health-level-seven (HL7) data from medical screening. The system makes intelligent recommendations leveraging a web-based platform for its users. The system was built to generate diet plans by eliminating unsuitable food groups based on the user’s priorities. |
| **[2]** | 2021 | Food recommendation with graph convolutional network | Web-based services have revolutionized the way information is consumed. Dietary recommendation systems have attracted not just end consumers but also several food-related applications and services. Said systems aim to match the preferences of a user to a recipe. Several systems have been utilized ranging from content-based and collaborative filtering to evolved and sophisticated methods. A novel system that utilizes Graph Convolutional Network (GCN) that exploits ingredient-ingredient-recipe-user cross-relations deeply. It leverages an information propagation mechanism and adopts multiple embedding propagation layers to model high-order connectivity across multiple food-based relations to enhance representation. 3 unique propagation systems from a permutation of ingredient-user-recipe states. In-depth analysis shows that the GCN-based model could reduce the scarcity of elite systems in the context of food recommendations. |
| **[3]** | 2014 | The nature and evolution of online food  preferences | Food consumption plays a central role in human race survival. Food preferences have manifested themselves as social, cultural, and economic forces. Historically, recorded data for food preparation, preferences, and consumption patterns collected from households and individuals have played a huge role in realizing systems to recommend recipes and diets to users. The publication in consideration aims to scrape useful data from the world wide web and analyze its usefulness in food recommendations and patterns. The important conclusions of the experiment are the correlation between recipes and ingredients, differences between regional ingredients, and the variation of preferences based on the time frame. |
| **[4]** | 2015 | Temporal Patterns in Online Food Innovation | The role of innovation in food preferences and preparation has in the context of food-based careers, be it recommender engines, chefs to even packed food producers has stood the wrath of time. The experiment aims to explore the dimension of innovation in a virtual format. To be more precise, it talks about the processed results from a large-scale German online food community platform to explore the use case of online food recipe production. The results bring about the findings and temporal patterns in how online food recipe innovation takes place. |
| **[5]** | 2018 | Automated and Personalized Nutrition Health Assessment,  Recommendation, and Progress Evaluation using Fuzzy Reasoning | People's lives have grown increasingly focused on living a healthy lifestyle. The latter necessitates maintaining a healthy diet while considering the types and amounts of foods consumed. It also calls for leading an active lifestyle that includes adequate physical activity to control calorie and nutrient intake and consumption. As a result, people seek out nutrition specialists to conduct health assessments, which are costly, time-consuming, and difficult to come by. Even though a variety of e-nutrition solutions have been developed, most of them focus on meal planning rather than health assessment or evaluation (traditionally provided by human experts). This research aims to provide an automated solution for performing nutritional health assessments, recommendations, and progress evaluations. A novel framework titled PIN was tested which is based on the fuzzy logic paradigm to simulate human expert health assessment capabilities including weight, calorie consumption, age, exercise patterns, and height. The results quantify PIN’s assessment and recommendations on par with human nutritionists. |
| **[6]** | 2011 | Hierarchical Attention Network for Visually aware Food Recommendation | Food recommender systems are useful in guiding users in identifying the foods they want to eat. Deciding what to eat is a complicated and multi-faceted process that is influenced by a variety of elements including the ingredients, the appearance of the recipe, the user's personal food preferences, and multiple contexts such as previous meals. The authors formulate the meal recommendation problem in this paper as predicting user preference for recipes based on three major aspects that influence a user's food choice: 1) the user's (and other users') history; 2) the contents of a dish, and 3) the recipe's descriptive image. To tackle this difficult problem, the authors created Hierarchical Attention-based Food Recommendation (HAFR), a dedicated neural network-based solution capable of 1) capturing the collaborative filtering effect, such as what similar users eat; 2) inferring a user's preference for the ingredient level; and 3) learning user preference from the recipe's visual images. The authors create a large-scale dataset with millions of reviews from AllRecipes.com to test our proposed technique. Extensive tests demonstrate that our system outperforms many competing recommender solutions, such as Factorization Machine and Visual Bayesian Personalized Ranking, by an average of 12%, indicating that it can accurately predict user food preferences. After acceptance, the codes and dataset will be made public. |
| **[7]** | 2016 | Artificial Bee Colony – Based for Dietary  Recommendation in Daily Nutrition Requirements | A healthy lifestyle is a must for people, which may be attained through balanced nutrition. Unbalanced nutrition raises the chance of health issues. The disparity between nutrition required and nutrition consumed must be as little as feasible to achieve balanced nutrition. In Indonesia, many people consume high-carbohydrate foods, even though nutrition consists of protein, carbohydrate, and fat. That nourishment comes from five types of foods: main dish (MP), vegetable side dish (LN), meat (LH), vegetable (SY), and fruit (F) (BH). As a result, it requires a system that can make recommendations for healthy eating. The Artificial Bee Colony (ABC) was employed in this study to obtain optimal nutrition, which has five dimensions (MP, LH, LN, SY, BH). Food sources are represented by these dimensions and variables, which will be optimized by bees. The study's key contribution is to determine the best portion and type of food to reach the best solution, which is to make food recommendations for Indonesian cuisine. The best solution comes from fitness function, which is the difference between the nutrition required and the nutrition recommended. This study proposes an artificial bee colony algorithm for determining the amount and type of food required for daily nourishment. ABC can give acceptable portions and types of foods with an error tolerance of 80 percent to 110 percent and an average accuracy of 99.90 percent. |
| **[8]** | 2013 | You are what you eat: learning user tastes for  rating prediction | Poor nutrition is one of the leading causes of illness and death in the Western world, and it is driven by several factors such as a lack of nutritional knowledge and a preference for convenience meals. We want to create systems that can offer healthy meal plans to consumers, but one of the most important requirements is the ability to recommend dishes that people would enjoy. By analyzing the findings of a longitudinal study (n=124), we analyze critical aspects that influence how recipes are assessed to better understand how to tackle the suggestion challenge. Several critical contextual aspects that can influence the rating decision are identified. We build numerous recipe suggestion models based on this study that can utilize information of user preferences in terms of ingredients and combinations of ingredients, as well as nutritional content. We can show that these models can dramatically outperform a few competing baselines using an experiment on our dataset. |
| **[9]** | 2018 | Enhancing multi-label classification for food truck recommendation | Food trucks are a popular fast food restaurant alternative that are distinguished by their proximity to customers. Their success has prompted a growth of available options, which currently comprise several distinct types of cuisines, making consumer selection a difficult task. This work focuses about food truck suggestion using a multi-label technique, based on data acquired from market research in which hundreds of participants submitted their food truck preferences.  It focuses on how to improve the recommendation task in the context of a previous study in which some labels were never predicted. Various options were considered to address this issue. One of these options, the Ensemble of Single Label proposed in this study, was successful in lowering it. When they were applied in the researched task, they produced good predictive results despite their simplicity. All labels were successfully predicted on at least a few occasions, among other advantages. |
| **[10]** | 2019 | The Effectiveness of Nutrition Education  about Local Specific Food-based Balanced  Nutrition Recommendation on Dietary Intake  Level and Anemia Status in Female  Adolescents | Anemia is the most serious dietary issue among adolescents. Poor diet is one of the key risk factors. Nutrition education has been found to increase healthy lifestyles and academic achievement in school-aged children when used as an intervention technique. The goal of this study was to see how effective nutrition education about local food-based balanced nutrition recommendations affected female adolescent knowledge, dietary intake, and anemia status. This was a quasi-experimental study that included a pre and post-test design. Eighty-three female pupils took part in the study for five months. Nutrition instruction in the form of classroom counselling on balanced dietary guidelines and healthy snacks was provided. Before and after the intervention, dietary consumption, knowledge, and hemoglobin levels were examined. The paired t-test and Wilcoxon signed-rank test were used to analyze the data, with p<0.05 considered significant. There was a significant difference in protein (p=0.029) and iron consumption (p=0.021) intake levels before and after the intervention, but not in vitamin C or folic acid intake. Increased knowledge, dietary intake (protein and iron), and hemoglobin levels in female teenagers can be achieved through nutrition education regarding local specific food-based balanced nutrition recommendations. |
| **[11]** | 2021 | Food recommendation system for the elderly | When the elderly sits down to eat, they frequently struggle to select nutritious foods. Normally, they choose their own food or have a caregiver assist them in finding a menu or arranging the dishes they want to consume. However, food preparation for the elderly differs from that for other ages because it necessitates careful consideration of health and proportions appropriate for their age. It also implies that more caution is required. The creation of a meal recommendation system for the elderly has as its goal the introduction of healthy diets and the promotion of good health among the elderly. This system employs approaches that aid in menu recommendations, allowing the elderly to make more informed eating choices. As a result, a suitable food recommendation system is being developed by using Clustering Algorithm analysis techniques to divide the elderly into groups based on their behaviors, eating habits, and food preferences, and using the Slope One Algorithm, which can predict menu-preference scores, as a technique to provide food suggestions. Furthermore, the calculating procedure has been enhanced to make it more useful to improve the application's quality and the accuracy of food recommendations: The Root Mean Square Error (RMSE) is utilized in forecasting and recommending appropriate menus for each elderly person, and it can be used to develop the application system to satisfy the users' needs so that they can choose healthy diets for their loved ones. |
| **[12]** | 2019 | Patient Diet Recommendation System Using K Clique and Deep learning | There are a few frameworks intended to suggest this. The suggesting framework has acquired its conspicuousness even in the clinical business for recommending the weight control plans for the patient's, meds to be taken, medicines to be taken  and so on. The suggestion framework for the most part upgrades the vigor, broadens insurance against the numerous illnesses and works on the nature of living. So, to naturally recommend the food sources in light of their medical issue and the degree of sugar, pulse, protein, fat, cholesterol, age and so on, the paper advances the k-clique embedded deep learning classifier recommendation system for suggesting diets for the patients. The K-clique embedded deep learning classifier in the proposal framework in a work of getting a further developed accuracy and expanded the precision of the profound learning classifier (gated repetitive units). The dataset for the observational examination of the created framework was performed with the informational index of the patients gathered over the web as well as emergency clinics; data of around 50 patients were gathered with thirteen elements of different infection and thousand items with eight lists of capabilities. This large number of elements were encoded and assembled into a few bunches prior to applying them into the profound learning classifiers.  The better accuracy and the exactness noticed for the created framework tentatively is contrasted and the AI  methods, for example, strategic relapse and Naïve Bayes and other profound learning classifiers, for example, the MLP and RNN to illustrate the capability of the K-Clique deep learning classifier-based proposal framework (K-DLRS). |
| **[13]** | 2019 | The Cholesterol Factor: Balancing Accuracy and Health in Recipe Recommendation Through a Nutrient-Specific Metric | Many food recommender systems optimize for users’ preferences, health is another but often overlooked objective. This paper aims to recommend relevant recipes that avoid nutrients that contribute to high levels of cholesterol, such as saturated fat and sugar.  The author introduced a novel metric called ‘The Cholesterol Factor’, based on nutritional guidelines from the Norwegian Directorate of Health, that can balance accuracy and health through linear re-weighing. The author tested popular recommender approaches by evaluating a recipe dataset from AllRecipes.com, in which a CF-based SVD method outperformed content-based and hybrid methods. Although the author found that increasing the healthiness of a recommended recipe set came at the cost of Precision and Recall metrics, only putting little weight (10-15%) on our Cholesterol Factor can significantly improve the healthiness of a recommendation set with minimal accuracy losses. |
| **[14]** | 2017 | A Cross-Sectional Survey in Rural Bihar, India, Indicates That Nutritional Status, Diet, and Stimulation Are Associated with Motor and Mental Development in Young Children | The goals of this study were to look at the nutritional, psychological, environmental, and household determinants of child development in Bihar, India, as well as to discover mediators between dietary diversity and mental development. They surveyed 4360 households with children aged 6–18 months in Bihar's West Champaran area using two stages of cluster randomized sampling. One of the most important findings was that gross motor development and fine motor development were major mediators in the relationship between dietary diversity and mental development. |
| **[15]** | 2011 | A Personalized Recipe Advice System to Promote Healthful Choices | The article presents a prototype of a personalized recipe guidance system that assists users in making health-conscious meal choices based on previous choices. A goal setting mechanism is used in conjunction with tailored recipe ideas to encourage the adoption of a healthy lifestyle. The app's major focus is on selecting appropriate recipes for future meals, with no feedback on previous choices provided. |
| **[16]** | 2021 | Deriving a Recipe Similarity Measure for Recommending  Healthful Meals | In the past decades, the process of urbanization has shaped general socio-economic aspects of cities with different population sizes.  Among them, food consumption is a good indicator to read the quality of life. In this paper, the authors study the impact of city size on food preferences, as shown by users of a large German food sharing community. The authors quantitatively and qualitatively analyze differences in dietary choices made by users who indicate they live in cities of different sizes, from metropolises and big cities to medium and small towns. Further, the authors demonstrate that the city size of the creators. In the past decades, the process of urbanization has shaped general socio-economic aspects of cities with different population sizes.  Among them, food consumption is a good indicator to read the quality of life. In this paper, the authors study the impact of city size on  food preferences, as shown by users of a large German food sharing community. the authors quantitatively and qualitatively analyze differences in dietary choices made by users who indicate to live in cities of different sizes, from metropolises and big cities to medium and small  towns. Further, the authors demonstrate that the city size of the creators of online recipes can be predicted with a good accuracy of 86%, using predictors based on recipe authors’ roles, recipe popularity, season, and recipe complexity and contents. Endings indicate that city size is a useful feature to consider in various other domains of online recipes that can be predicted with a good accuracy of 86%, using predictors based on recipe authors’ roles, recipe popularity, season, and recipe complexity and contents. endings indicate that city size is a useful feature to consider in various other domains. |
| **[17]** | 2012 | A Food Recipe Sourcing and Recommendation System to Minimize Food Miles | Supportable Recipes is an instrument that (1) interfaces food plans fixing records with the nearest natural suppliers  to limit the distance that food goes from homestead to food readiness site and (2) suggests plans given a GPS direction to limit food miles. Maintainable Recipes gives purchasers, business visionaries,  cooking lovers, and waiters in the United States and somewhere else with a simple to utilize point of interaction to  help them associate with natural fixing makers to source fixings to deliver food plans, limiting food miles and suggesting plans utilizing privately developed food. The super scholarly commitment of Sustainable Recipes is to overcome any issues between two floods of writing in information study of food plans, investigations of food plans and investigations of food supply chains. The results of the interphase are:  (1) a guide perception that features the area of the makers that can supply the elements for a food formula alongside a ticket consisting of their contact addresses and the food miles used to create a formula and  (2) A rundown of plans that limit food miles for a given GPS coordinate in which the formula will be delivered. |
| **[18]** | 2018 | Recipe recommendation using ingredient networks | Sustainable Recipes is a tool that connects food recipe ingredient lists with the closest organic providers to minimize the distance that food travels from farm to food preparation site and recommends recipes given a GPS coordinate to minimize food miles. Sustainable Recipes provides consumers, entrepreneurs,  cooking enthusiasts, and food chains in the United States and elsewhere with an easy-to-use interface to help the connect with organic ingredient producers to source ingredients to produce food recipes  minimizing food miles and recommend recipes using locally grown food. The main academic contribution of Sustainable Recipes is to bridge the gap between two streams of literature in data science of food recipes: studies of food recipes and studies of food supply chains. The outcomes of the interphase are a map  visualization that highlights the location of the producers that can supply the ingredients for a food recipe along with a ticket consisting of their contact addresses and the food miles used to produce a recipe a list of recipes that minimize food miles for a given GPS coordinate in which the recipe is going to be produced. |
| **[19]** | 2021 | Personalized Food Recommendation as Constrained Question Answering over a Large-scale Food Knowledge Graph | The research introduces a unique problem formulation for food recommendation, describing the task as constrained question responding across a large-scale food knowledge base/graph (KBQA). Addressing shortcomings such as i) failing to consider users' explicit requests, ii) ignoring critical health concerns (e.g., allergies and nutrition needs), and iii) failing to employ rich food knowledge for recommending healthy recipes, they develop a dataset in the QA manner for personalized food suggestions. The benchmark results reveal that the technique beats non-personalized counterparts (average 59.7 percent absolute improvement across multiple evaluation parameters) and can offer more relevant and healthier recipes. |
| **[20]** | 2011 | The Influence of City Size on Dietary Choices | In the previous many years, the course of urbanization has molded general financial parts of urban communities with different population sizes. Among them, food utilization is a decent marker to read personal satisfaction. In this paper, the authors concentrate on the effect of city size on food inclinations, as shown by clients of a huge German food sharing local area. The authors quantitatively and subjectively dissect differences in dietary decisions made by clients who demonstrate living in urban communities of different sizes, from cities and huge urban communities to medium and little towns. Further, the authors show that the city size of the makers of online plans can be anticipated with a decent precision of 86%, utilizing indicators considering formula creators' roles, formula prevalence, season and formula intricacy and items. Endings show that city size is a valuable element to consider in different other areas. |
| **[21]** | 2015 | Food Recommendation System Using Clustering  Analysis for Diabetic Patients | Food and nutrition are key to having good health. They are important for everyone to maintain a healthy diet especially for diabetic patients who have several limitations. Nutrition therapy is a major solution to prevent, manage and control diabetes by managing the nutrition based on the belief that food provides vital medicine and maintains good health.  Typically, diabetic patients need to avoid additional sugar and fat, so the food pyramid is recommended to the patients for finding the substitution from the same food group. However, there is still  a dietary diversity within food groups that can affect the diabetic patients. In this study, the authors proposed Food Recommendation System (FRS) by using food clustering analysis for diabetic  patients. Our system will recommend the proper substitute foods in the context of nutrition and food characteristics. the authors used  Self-Organizing Map (SOM) and K-mean clustering for food clustering analysis which is based on the similarity of eight  significant nutrients for diabetic patients. In the end, the FRS was evaluated by nutritionists, and it has performed very well and is useful for the nutrition space. |
| **[22]** | 2015 | Using Tags and Latent Factors in a Food Recommender  System | The offered suggestions are derived by using a data collection of users' preferences expressed in the form of ratings and tags, which signify the food's components or aspects that the consumers enjoy. Our empirical evaluation demonstrates that the suggested recommendation approach greatly outperforms state-of-the-art algorithms. the authors discovered that adding tags in meal recommendation algorithms may greatly improve prediction accuracy, i.e., the match of anticipated preferences with the genuine user's chosen dishes. Furthermore, our user survey demonstrates that our system prototype is highly usable. |
| **[23]** | 2017 | Investigating the Healthiness of Internet-Sourced Recipes | Researchers demonstrate how algorithmic solutions relate to the healthiness of the underlying recipe collection by focusing on two techniques from the literature (single item and daily meal plan suggestion) and leveraging a huge Internet derived dataset from Allrecipes.com. First, they assess the nutritional value of Allrecipes.com recipes using World Health Organization and United Kingdom Food Criteria Agency nutritional standards. Second, they look into user engagement patterns and how they relate to the nutritional value of dishes. Third, they test both recommendation techniques. The results show that, on average, the recipes in the collection are highly harmful, but this differs across the website's categories. |
| **[24]** | 2019 | Food Recommendation: Framework, Existing  Solutions and Challenges | This article provides a unified framework for food recommendation and discusses key difficulties impacting food recommendation, such as merging different context and domain information, developing a personal model, and examining unique food attributes.  The author then goes through known solutions to these problems before elaborating on research obstacles and prospects in this subject. According to the author, this is the first survey that focuses on the study of food suggestion in the multimedia sector, and it provides a collection of research papers and technologies to help researchers in this field. |
| **[25]** | 2010 | Intelligent Food Planning: Personalized Recipe Recommendation | The author offers early research into the design of a recipe recommender aiming at educating and sustaining user engagement and making personalized recommendations of healthy dishes. They focus on the first two dimensions of food recommendations: data gathering and food-recipe interactions and offer research on the applicability of several recommender algorithms for recipe suggestion. |

**3.3 Conclusion of survey**

A majority of the research done on the topic either focuses on a specific deficiency of the user and provides recipes based on that or fails to provide a custom personalized experience to the user based on their nutritional values and preferences. Our research aims to use standard nutritional values related to gender and age along with nutritional information including deficiencies of the user while also infusing the user’s personal preferences which makes the system unique to existing models. The nutritional profile analysis and recipe recommendation along with result ranking based on user’s needs is bundled into one functional system making it the novel feature of this research.

**Chapter 4**

**PROJECT MANAGEMENT PLAN**

**4.1 Schedule of the Project**

W1 W2 W3 W4 W5 W6 W7 W8 W9

PROJECT TITLE X

AND ABSTRACT

SUBMISSION

ZEROTH X X---

REVIEW

PROJECT XXX-----

PLAN

LITERATURE --------

SURVEY

SYSTEM --------

DESIGN

IMPLEMENTATION ------------- ---------

AND TESTING

REPORT **--------------**

**4.2 Risk Identification**

While this application will eventually help manage dietary requirements as per set standards, it is vital to consult a doctor/nutrition expert for an opinion. AI in healthcare is a boon to society but the lack of accountability of AI in a production environment poses a challenge to the potential users of this application.

Changes made to the users’ diet after multiple iterations of inputs should ideally reflect in the form of a relatively healthier individual. But this output cannot practically be reinforced in the model immediately since it takes time for the recommended diets to take effect on the users’ health.

With an increase in the number of users, the API server used for this application will see more hits. In such an event, the plan used for this API must be altered to handle the user traffic in a production environment.

**Chapter 5**

**SOFTWARE REQUIREMENT SPECIFICATIONS**

**5.1 Product Overview**

The product is designed to be open source, under the MIT public license. It is a mobile application system implementing a client-server model. It enables users to input their nutritional profile into an interactive user interface, which then takes in the values and passes it on to the backend. An algorithm maps the required nutrients tailored for every user to the information put in by them like age, gender, activity levels, diseases and allergies and personal health goals. At the backend, we make use of a clustering algorithm that can outputs food groups that are rich in specific groups of nutritional values. A ranking system maps the user inputs explaining their preferences and scrapes the web for recipes for the right diet.

1. **User Account:** The system allows the users to create their accounts. The users have an option to input their health-related values and store it with us on the database which can be modified at any point, triggering the suggestion retrieval from the recommendation system with state change.
2. **Ranking of Recipes:** The user profile will be processed, and a list of food ingredients will be generated such that they cover the nutritional requirements of the user.
3. **Application Output:** The user gets to view selected recipe videos from the internet recommended by our app as a list of hyperlinked videos based on our ranking system on an interactive UI.

**5.2 External Interface Requirements**

**5.2.1 User interfaces**

The main users of this system are casual users and researchers.

**Casual users:** All casual users can access the project through the flutter mobile app on both iOS and Android.

**Researchers:** All researchers interested in learning more about the algorithm and the output can access the project by making REST API calls to the Flask backend server endpoint.

**5.2.2 Hardware Interfaces**

Any mobile or personal computing device which has the following basic specifications.

* CPU: Any modern 64-bit processor ARM processor
* RAM: 3 GB or more
* Storage: 1 GB minimum, Additional space recommended
* GPU: integrated GPUs or higher
* Internet access and support

**5.2.3 Software Interfaces**

Works with any of the latest mobile operating systems like iOS 15 and Android 11.

**5.2.4 Communication Interfaces**

The communication standard in the model mostly includes the use of HTTP and HTTPS for REST API calls between the mobile application and the backend server.

**5.3 Functional Requirements**

**5.3.1 Functional Requirements**

The functional requirements for our application are as follows:

**Authentication:** secure authentication of a user when they login / register on the mobile app.

**Valid Recommendation:** A fully functional system capable of suggesting recipes based on users’ nutritional requirements and preferences.

**User Interface:** Attractive and simple UI/UX for interaction.

**Inputs:** valid inputs from the user about their nutritional profile, gender, age and their personal preferences.

**Modern Processors:** User systems capable of handling and running complex mathematical tasks on- premises.

**Transaction correctness:** Making the right API calls in the backend for running the model and CRUD operations.

**Bug Tracking:** A simple method to find and resolve issues caused during runtime.

**5.3.2 Non-Functional Requirements**

**Performance Requirements**

* The product must have reasonable processing times.
* The product must be scalable.
* The product must be maintainable

**Safety Requirements**

* If configured to use a login technique, the database should store the user credentials in an encrypted format in order to avoid any breach of data associated with the user. This is achievable by using a third-party authentication service like Firebase Authentication.
* Care must be taken to not overload the database where user data is stored, in order to prevent incurring additional costs of storage or potential lapse in security to a larger repository.

**Security Requirements**

* Every unsuccessful attempt to login by the user should be logged in the backend.
* Videos must be checked to ensure they do not contain explicit content or derogatory materials of any nature.
* The audio within the video must be checked to ensure that there are no derogatory comments being made

**Software Quality Attributes**

* **Availability:** The recommendation engine must be available whenever the user uploads/updates a nutritional profile for processing or requests for recipe videos.
* **Accuracy:** The recommendation engine must be able to recommend recipes with great accuracy matching the user’s nutritional requirements.
* **Maintainability & scalability:** The engine and its components must be easy to maintain and update, if necessary, as well as scale well with increasing load levels.
* **Accessibility:** The website/mobile application housing the engine must be accessible and have an intuitive UI that is usable by a large audience, much like popular platforms today.

**Chapter 6**

**DESIGN**

**6.1 Introduction**

The purpose of this chapter is to give a detailed description of the system design for the project. It will illustrate the purpose and complete declaration for the design of system. It will also explain Architecture Design, Dataflow Diagram and User Interface Design.

**6.2 Architecture Design**

**A picture containing diagram

Description automatically generated**

Fig. 2. Recommendation engine architecture

Fig. 2. represents the high-level system design used to develop a working scalable and deployable application as a product of this research. The general flow of the application is as follows, user is prompted to input their nutritional values, gender and age with the help of a simple file upload. The user’s data is then analysed by performing a difference calculation of the user’s values with the standard nutritional values from the NIN dataset, from this we obtain the deficiencies of the user and run a K-Means clustering algorithm to obtain ingredients with appropriate nutritional values which will help in bettering the user’s nutritional profile, with these ingredients food recipes are found using the Spoonacular API[27]. The recipes obtained are then ranked based on the user’s nutritional profile and then the system scrapes for recipes from the internet using the YouTube API.

Diagram

Description automatically generated

Fig. 3. Mobile application architecture

The mobile application developed follows a client-server architecture, wherein the recommender system is made accessible by a Flask backend which acts as a standalone REST API server. The frontend is developed with the help of Flutter and performs REST API calls to the backend with the relevant inputs obtained from the user. Firebase is also used as a Backend as a Service, services like Firebase Authentication and Firebase Firestore are used to authenticate and store user data securely.

**6.2 Graphical User Interface**

This project is implemented as a mobile application which can also be extended to a web and desktop class application, this is one of the main benefits of using Flutter for development. It consists of multiple unique screens with which the user can interact with ease.

The graphical user interface was designed keeping in mind the main objectives of user interface design which are as follows: -

* Consistency
* Usability and Accessibility
* Navigation
* Visual appeal
* Performance

Graphical user interface, application

Description automatically generatedFig. 4. Secure user authentication

On opening the app, the user is presented with an elegant graphical user interface wherein they can either sign in or register to create an account by entering a unique email and password pair.

Graphical user interface, text, application, chat or text message

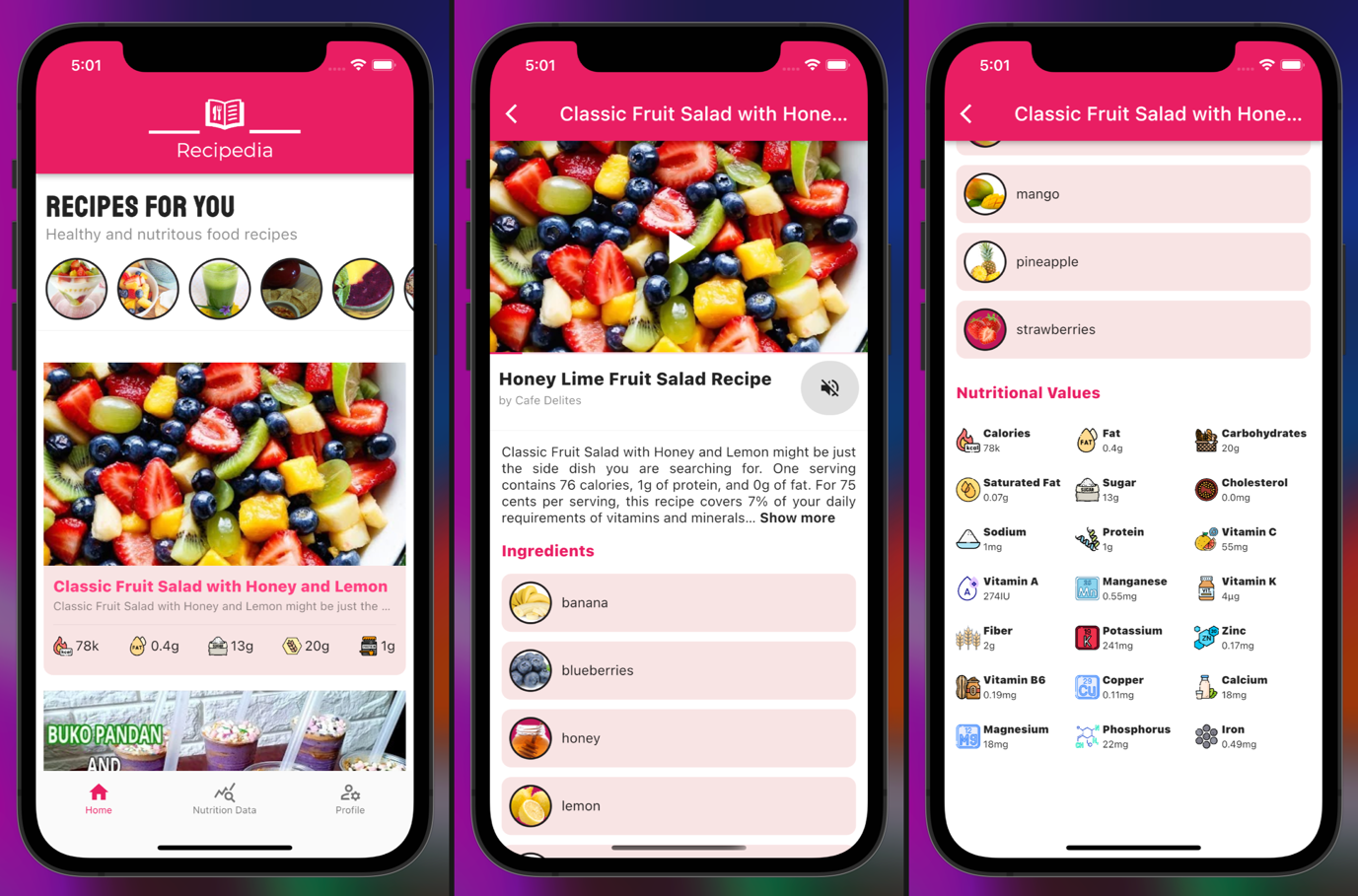
Description automatically generatedFig. 5. Simple error detection and warning system

The mobile application is built with a robust error detection and warning system. Whenever a user comes across a runtime error, the user is promptly notified about the error and a solution is prompted so as to overcome it.

Graphical user interface, application

Description automatically generatedFig. 6. User data upload and secure sign out

The user is provided with a simple file upload option to provide their personal nutritional profile, age, gender and preferences in the app, this data can be edited as it’s saved in the cloud.

Fig. 7. Personalized recipes for the user

After a successful authentication and account creation, the user is presented with a beautifully designed personalised home screen wherein the user can view the list of recipes generated for them by the recipe recommendation engine based on the user’s input. The recipes are in a scrollable view, hence the user can view multiple recipes by simply scrolling up or down. Each recipe on the home screen provides concise yet important information regarding itself, it consists of the recipe title and quick nutrition stats which help the user to quickly judge if the recipe interests them or not. On clicking on a particular recipe, the user is provided with a new information packed screen which consists of the recipe video which is scraped from YouTube and ranked based on the user’s nutritional profile and preferences. Along with the video the user is provided with certain video controls like mute/unmute option etc. so as to make the video viewing experience smooth and simple. The user is also provided with a short yet informative summary of what the recipe is and how long it will take to cook it. Following the summary the user is presented with a list of ingredients required to prepare the recipe, the ingredients are displayed both in the form of an image and text. Finally as the user scrolls down the page, they are provided with complete nutritional data for the selected recipe.

**6.3 Data flow Diagram**

**Diagram

Description automatically generated**

Fig. 8. Data flow diagram

The diagram represents the flow of data through various functional components in the system for the recipe recommendation engine. Each functional component has defined and unique inputs and outputs. It also depicts the necessary data stores to hold intermediate outputs.

**6.4 Conclusion**

In this section we have provided the architecture design for both the frontend mobile application as well as the backend Flask server which runs the recipe recommendation engine. We have also illustrated the elegant and easy to use graphical user interface along with the data flow diagram which mentions how the data is passed between multiple functional components.

**Chapter 7**

**IMPLEMENTATION**

**7.1 Tools Introduction**

**The Jupyter Notebook** is a web-based application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. Jupyter notebook was used to verify the sanity of the solution provided by building modular components based on the functional requirements to create an end-to-end flow in a real-time scenario.

**XCode** is Apple's integrated development environment (IDE) for macOS, used to develop software for macOS, iOS, iPadOS, watchOS, and tvOS. XCode was one of the platforms used to develop the flutter application. It was specifically used to develop and test the iOS version of the application.

**Android Studio** is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. Android Studio was one of the platforms on which the Flutter application was built as it provides multiple Flutter plugins and typical IDE features for easy development. It was specifically used to develop and test the android version of the application.

**Visual Studio Code**, also commonly referred to as VS Code, is a source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. VS Code was one of the platforms used to develop the flutter application. It was specifically used to write and debug the application.

**7.2 Technology Introduction**

**Flask** is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. Flask was used to build the backend that would run the backend scripts which consists of components built by optimizing the notebooks which contained the initial implementation of the solution provided.

**Flutter** is an open-source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase. Dart is the official language to write scripts in the Flutter framework. Flutter was used to build the front-end application which acts as the single point of entry for the users.

**Dart** is a programming language designed for client development, such as for the web and mobile apps. It is developed by Google and can also be used to build server and desktop applications. It is an object-oriented, class-based, garbage-collected language with C-style syntax.

**Python** is a high-level, interpreted, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected.

**Machine learning** is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificial intelligence.

**REST API** also known as RESTful API is an application programming interface (API or web API) that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services. REST stands for representational state transfer.

**7.3 Overall view of the project in terms of implementation**

The implementation took place as follows. Discussions were carried out to design a solution that would tackle the existing problem statement and design a solution base. Primary research was conducted on existing tools and technologies that could be used to develop a tangible module that could showcase the sanity and verify the solution provided. Each of the functional components was developed on Jupyter Notebook using python scripts in sequence and iteratively optimized. Several comparisons took place to determine the best set of APIs necessary to bridge the gap between the functional components. Spoonacular and YouTube APIs were finally picked for their functionality, pricing,  and easy-to-use procedure. Once the initial solution was verified, an application was built using flask as the backend and Flutter(Dart) as the front end so that we could welcome the initial set of users and enable testing.

The mobile application implementation consists of:

* Login screen
* Register screen
* Home screen listing personalised recipes
* Nutrition data screen showing the user’s profile
* Data upload screen
* User account screen

The server implementation consists of

* A route which accepts user nutritional profile, age, gender and preferences as input and provides personalised recipes as the output in JSON format
* A route to test the recipe recommendation engine with default data
* A flask application which accommodates multiple routes and returns the correct route and data for every request

**7.4 Explanation of Algorithm and how it is being implemented**

Three primary components, namely deficiency detection, ingredient identification, and food product ranking form the base of the algorithm. Each component provides an intermediary output which feeds into the next component to finally list a set of ranked food product and their recipes for the users to consume.

**Deficiency detection**: The user’s consumption and biological profile are the primary input in this component. The input is a set of nutritional values that depict the general consumption pattern of the said user. It consists of 25 unique nutritional values i.e.., Protein, Ash, Fat, Dietary-Fibre, Carbohydrate, Energy, Thiamine, Riboflavin, Niacin, Pantac, Vitamin B6, Vitamin B7, Vitamin B9, Vitamin C, Aluminium, Calcium, Copper, Iron, Magnesium, Manganese, Nickel, Phosphor, Potassium, Sodium, and Zinc. The user’s age range and gender are also received in this stage. These values are compared against the recommended standard values for the age range and gender. The shortcomings of any nutrient are calculated by finding the arithmetic difference between the two sets (user’s input and the standard values). The shortcoming depicts the nutritional deficiency/overload in the user’s standard consumption scenario. This deficiency/overload set is then adjusted to derive the required nutrition by performic arithmetic addition with the recommended standard values for the given age range and gender. This final set represents the nutritional requirements of the user for a particular consumption by the user. This result acts as the input to the next component i.e.., ingredient identification.

**Ingredient identification:** The second major component in the solution, ingredient identification uses the nutritional requirements of the user as input. It refers to a dataset that contains 299 food ingredients with each ingredient depicted by a set of values with nutrient information similar to the user’s input. The principal aim of this component is to identify a subset of the ingredients from the dataset which fulfil the user’s nutritional requirements. This is achieved by using a clustering algorithm called K-Means which is an unsupervised method to quantize vectors by partitioning x observations into y clusters where each observation is in a cluster with minimal spatial distance to the points that behave like the mean of each cluster. Traditional K-Means produce clusters with uneven sizes but as per the solution design, the cluster centres are rearranged to guarantee a number of ingredients returned each time that is most similar to the user’s nutritional requirement. This component outputs a set of ingredients that contain the necessary nutrients which can be used to identify whole food products that can be presented to the user with their recipe and basic nutritional information.

**Food product ranking:** The final component before the results are provided to the user is the food product ranking process. The input to this component is a set of food products with their nutritional information. The web is scraped for each of  the nutrients that the user’s nutritional information set contains. These food products are ranked by finding the Euclidian distance between each of the food products and the nutritional requirement of the user. Euclidian distance is calculated as the distance between the two sets when represented as a point mapped on a 25 dimension map. The food products are ranked in the order of the sets that are closest to the user’s nutritional requirements in this spatial mapping. The web is then scraped for recipes for each of these food products and presented to the user.

The above-mentioned sequence of components forms the base of the solution which is built upon to verify the sanity of the solution designed.

**7.5 Information about the implementation of Modules**

Implementation to verify the sanity of the solution provided is carried out on Jupyter Notebook as python scripts. The following section contains a detailed description regarding the implementation of all functional components in the solution provided.

**Deficiency detection and nutritional requirement analysis**:-

This component receives defined inputs from the user regarding their age range (one of 11-14, 15-18, 19-24, 25-50, 51+), their gender (either male or female), and a nutrition consumption profile with 25 well-defined nutrients (Protein, Ash, Fat, Dietary-Fibre, Carbohydrate, Energy, Thiamine, Riboflavin, Niacin, Pantac, Vitamin B6, Vitamin B7, Vitamin B9, Vitamin C, Aluminium, Calcium, Copper, Iron, Magnesium, Manganese, Nickel, Phosphor, Potassium, Sodium, and Zinc). It also refers to the recommended standard for each of the nutrients across age range and gender. The user’s nutritional profile is arithmetically subtracted from the defined set of recommended standard nutrient values to identify nutrient overload/deficiency in the user’s nutrient consumption. This derived set is then added to the standard set to identify the nutritional requirement for the user’s single consumption. This critical component sets the base for ingredient identification.

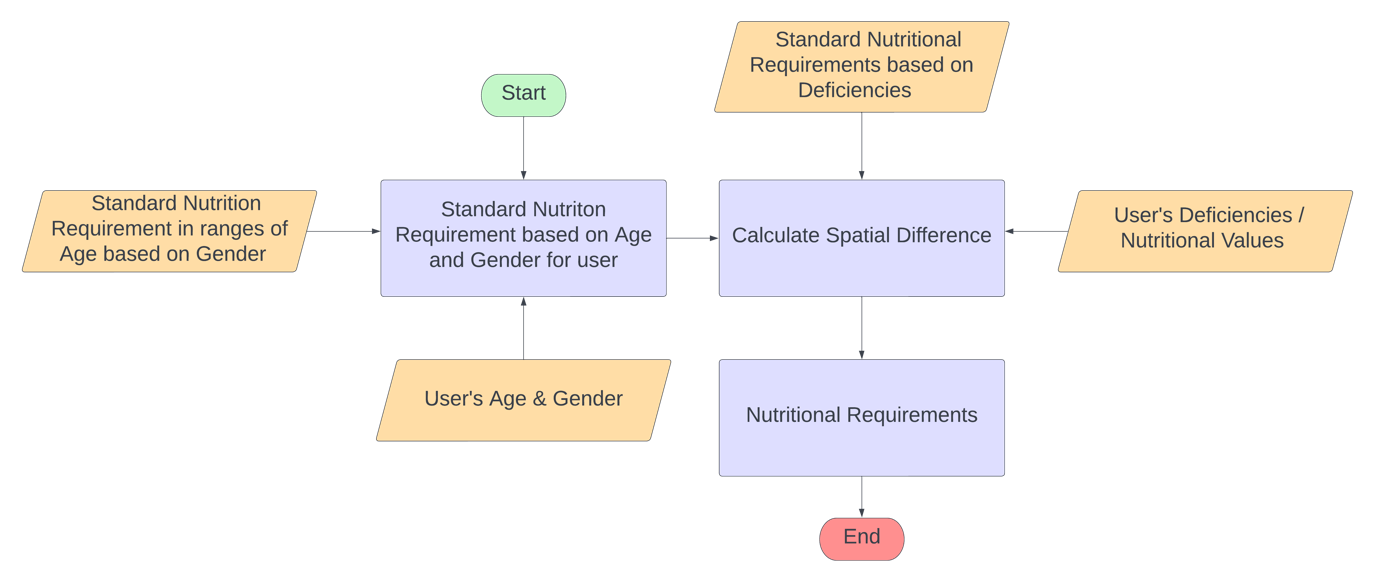


Fig. 9. Comparing standard nutrition values and user’s profile to detect deficiencies

*Deficiency Detection procedure***:**

1: **procedure** nutrition-requirements (user-nutrition-values, age, gender)

2: ms 🡨 male-standard-nutritional-values[age]

3: fs 🡨 female-standard-nutritional-values[age]

4: **if** gender = male **then**

5:difference 🡨 ms - user-nutrition-values

6: nutritional-requirements ms + (- difference)

7: e**lse**

8:difference 🡨 fs - user-nutrition-values

9: nutritional-requirements fs + (- difference)

10: return nutritional-requirements

11: **end**

**Ingredient identification**

This component identifies the set of ingredients that fulfil the user’s nutritional requirements by finding ingredients most similar to the user’s requirements. This task is achieved by running a K-Means clustering algorithm on 299 unique ingredients with the same ser of nutrients the user’s input contains to guarantee that each of the nutrients has an impact on the final result. K-Means is a widely known unsupervised clustering algorithm that partitions a set of n observations into y clusters where y n. It does so by randomly picking y cluster centres. Assign each of the remaining n - y observations into one of the y clusters. Find the distance between the observation and each of the y cluster centres. Reassign the observation to a cluster in such a way that the distance is minimal. This iteration continues until each of the observations is assigned to a cluster with minimum distance between the observation and the cluster centre. To ensure that a set number of ingredients are always returned irrespective of the input, the clusters derived from K-Means are recentred to pick the 9 most similar ingredients compared to the user’s nutritional requirements. This component returns a set of ingredients to satisfy the user’s nutritional requirement.

Diagram

Description automatically generated

Fig. 10. Detecting ingredients required for the user

*Ingredient detection procedure:*

1: **procedure** identify-ingredients (nutritional-requirements)

2: import ingredient-list

3: ingredient-list 🡨 ingredient-list + nutritional-requirements

4: n-clusters 🡨 ingredient-list-size/cluster-size

5: k-means-object 🡨 K-Means(n-clusters)

6: rearranged-cluster 🡨 Rearrange-Cluster(k-means-object)

7: required-cluster 🡨 Cluster-Number(rearranged-cluster, nutritional-requirements)

8: ingredients 🡨 []

9: **while** (ingredient in ingredient-list) **do**

10: ingredient-cluster 🡨 Cluster-Number(rearranged-cluster, ingredient)

11: **if** ingredient-cluster = required-cluster **then**

12: ingredients 🡨 ingredients + ingredient

13: **done**

14: return ingredients

15: **end**

**Get food products**

A vital component in the solution design flow, this module ensures to get food products. At this stage, the user could optionally reject some of the suggested ingredients due to various reasons such as allergic reactions, unavailability of certain ingredients, or just personal choice. Once the user’s favourable list of ingredients is received, an upstream call is made to Spoonacular API. It returns a set of food products along with their nutritional information, the and ingredients used. The ingredients used in the final food product are generally a subset of the ingredients sent as a query as the API returns meaningful and practical food products back.

Diagram

Description automatically generated

Fig. 11. Obtaining food products with the specific set of ingredients

*Get Food Products procedure:*

1: **procedure** get-food-products(ingredients)

2: user-modified-ingredients 🡨 PersonalizeIngredients(ingredients)

3: food-products 🡨 CallSpoonacular(user-modified-ingredients)

4: parsed-response 🡨 ParseAPIResponse(food-products)

5: return parsed-response

6: **end**

**Rank food products**

The final component that concludes the implementation of the base algorithm. It receives parsed API responses from the previous stage and the user’s nutritional requirement set as input. It uses Euclidean distance as the method to rank the food products in comparison with the user’s nutritional requirements.

The formula for Euclidean distance is as follows:-

where,

p, q are points in an n-dimensional plane

n is the number of dimensions (25 in our case)

pi and qi are co-ordinates of p and q in dimension i.

The user’s nutritional requirement set is taken as vector q and the nutritional information is considered as vector p. The Euclidean distance is calculated for each of the food products by calculating the arithmetic difference between the two vectors. The food vectors are then ranked in the order of the least difference.

Diagram, text, chat or text message

Description automatically generated

Fig. 12. Scraping nutritional value of cooked products

*Rank Food Products procedure:*

1: **procedure** rank-food-procedure(food-products, nutritional-requirements)

2: distance 🡨 []

3: **while** (food-product in food-products) **do**

4: dist 🡨 EuclideanDistance(food-product, nutritional-requirements)

5: distance 🡨 distance + dist

6: **done**

7: ranked-food-products 🡨 RankFoodProducts(distance, food-products)

8: return ranked-food-products

9: **end**

**Get recipes for ranked food products**

The last functional component in the solution flow is to get recipes for each of the ranked food products back to the user to enhance user experience. This module receives the ranked food products as input. An upstream request is made to YouTube’s consumer API. YouTube’s Consumer APIs are freely available for an application to consume at will. The recipe for each of the food products is queried in this process and the most relevant video from the list the API returns is picked. A final set is created with recipes and nutritional information for each of the food products in order of their rankings and returned to the user.

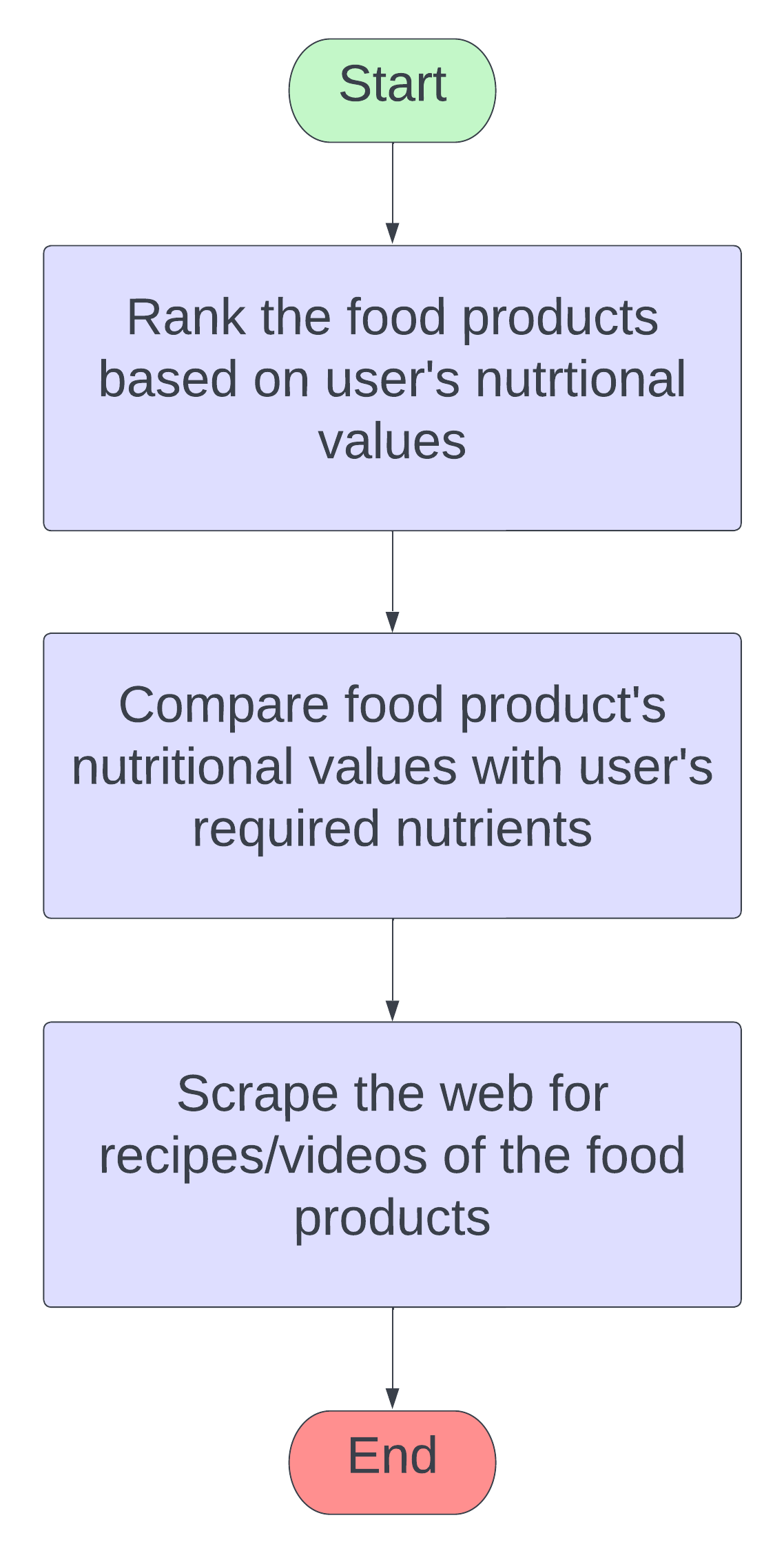


Fig. 13. Obtaining best recommendations for user

*Get Recipes procedure:*

1: **procedure** get-recipe-procedure(ranked-food-products)

2: recipeURLs 🡨 []

3: **while** (food-product in ranked-food-product) **do**

4: videoURL 🡨 YoutubeAPI(food-product, sort = relevance)

5: recipeURLs 🡨 recipeURLs + videoURL

6: **done**

7: return recipeURLs

8: **end**

**7.6 Conclusion**

This chapter highlights the necessary tools and technology that were required to implement the solution that is capable of functioning in a deployment scenario. The entire team discussed the problem statement, existing gap, and solution base with the mentor when the topic was first chosen. An initial design representing the flow of the solution was showcased to the mentor and was approved. Each of the functional components was individually developed on Jupyter Notebook simultaneously by using mock inputs for each of the components to reduce the time taken for general implementation. Each of the components was then assembled together to simulate the solution flow and tested. The sanity of the solution provided was then verified by testing against multiple inputs. Once the base solution was tested and approved, the team arbitrated on the design for the application. Once the specific UI was fixed, the team developed a mobile-cum-web application using Flutter and Android Studio IDE that a user could consume. The scripts used to verify the solution were optimized and converted into scripts that could be used by a flask server. This would enable the users to utilize the solution as the application would query the backed solution for each user request and provide users with a list of recipes that would cater to their nutritional requirements.

**Chapter 8**

**TESTING**

**8.1 Introduction**

Testing for the project took place in four phases as follows:-

* The first phase was to check the sanity and correctness of the recipe recommendation engine
* The second phase was to test the flask backend
* The third phase was to test the flutter application
* The fourth and final phase was to test the cohesion of the backend and frontend together.

**8.2 Testing tools and Environment**

Different testing tools and environments were chosen for different phases of the project.

* We used Jupyter Notebook to check the recipe recommendation engine
* To test the flask backend Postman was used to make REST API calls
* The flutter application UI was tested by writing unit and widget(component) tests
* The complete application was tested with a custom integration test

**8.3 Test cases**

A recommendation engine's performance can be tested, controlled, and measured in a variety of ways. The following are a few metrics which can be considered for testing and evaluation.

* **Coverage of users** with recommendations is an important factor, depending on the techniques implemented, the system might be able to generate recommendations to 1, 5, 42 or 100% of the users.
* **Coverage and diversity of items** the engine is capable of recommending. It is the measure of whether the system is recommending only 5% of all the available items. For a given user, are the recommendations diverse enough, e.g. items from different categories, price ranges, colours, etc.
* **System performances**. e.g. Can the system provide recommendations under 50ms at the 99.99th percentile? Depending on how the recommendations are used, one can set tight or loose constraints here

For each of the above metrics a threshold must be determined to accept or reject a recommendation engine.

A unit test verifies that every individual unit of software (often a function) performs its intended task correctly.

All the test files in a Flutter app (except for integration tests) are placed in the test directory.

For each module present in the application a unit test was written and the correctness of the module was checked.

Widget testing uses the testWidget() function instead of the test() function. It also takes two parameters: the description, and the callback. But here, the callback takes a WidgetTester as an argument.

Widget tests use TestFlutterWidgetsBinding, a class that provides the same resources to your widgets that they would have in a running app (information about screen size, the ability to schedule animations, and so on), but without the actual app. Instead, a virtual environment is used to run the widget, measure it, and so on, then tests the results.

The widget testing framework provides finders to find widgets (for example, text(), byType(), byIcon()) and also matchers to verify the results.

Integration tests are used to test how individual pieces of an app work together as a whole. The integration\_test library is used to perform integration tests in Flutter. This is Flutter's version of Selenium WebDriver (generic web), Protractor (Angular), Espresso (Android), or Earl Gray (iOS). The package internally uses flutter\_driver to drive the test on a device.

**Chapter 9**

**RESULTS AND PERORMANCE ANALYSIS**

**9.1 Result Snapshots**

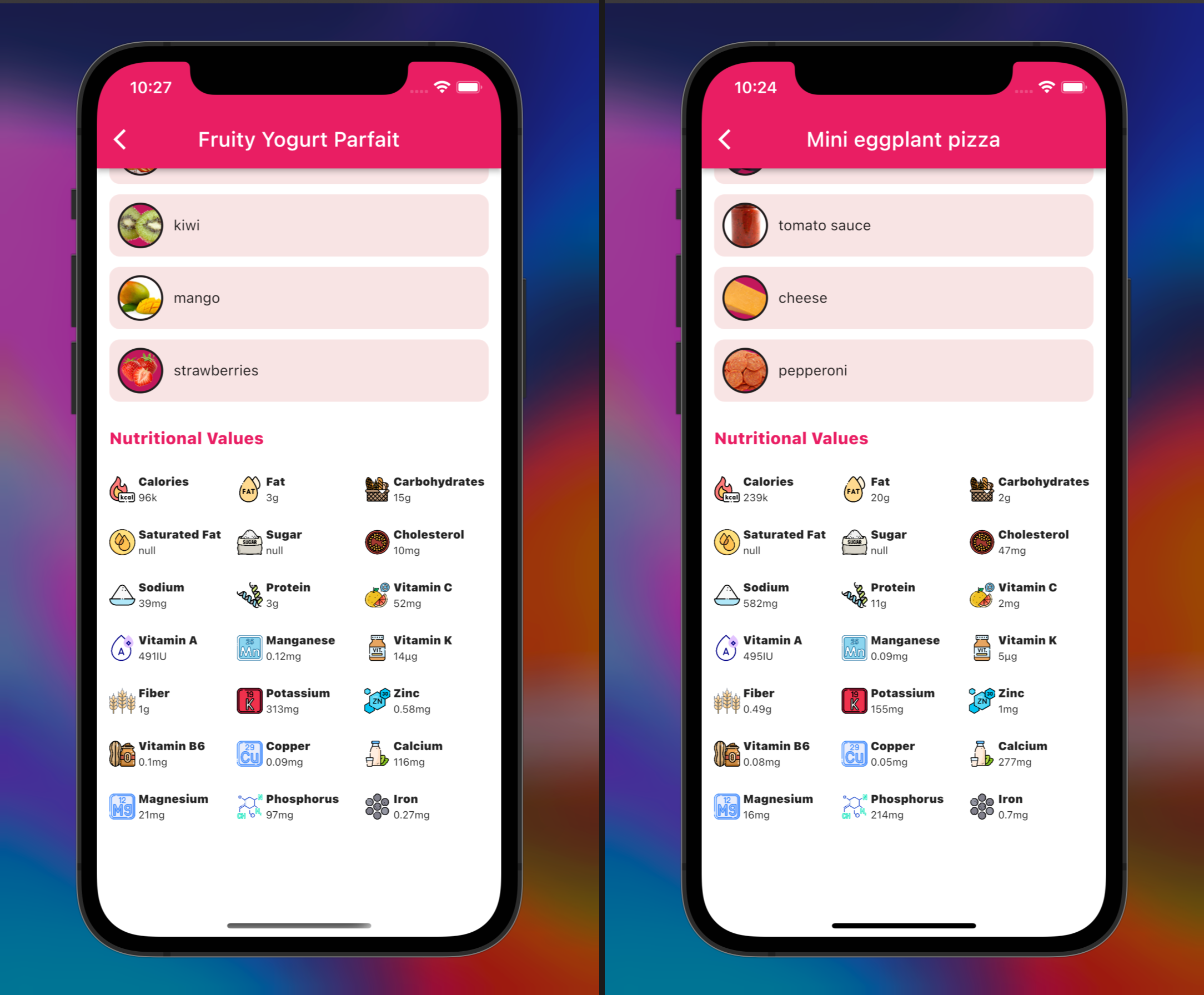
****

Fig. 14(a). High Calorie User Fig. 14(b). Low Protein User

Fig. 14(a) captures one of the food products suggested to a user with a high calorie value of 550kcal. The output is a food product with much lesser calories of just 96kcal, which is way below the standard 290kcal. By consuming a lower calorie food the user gradually improves their diet. As the user’s nutritional profile changes, the recipes suggested will also change accordingly.

Fig. 14(b) captures one of the food products suggested to a user with low protein value of just 3g. The output is a food product with a higher amount of protein, 11g for a meal. This increase in protein improves the user’s well-being and diet. As the user’s nutritional profile changes, the recipes suggested will also change accordingly.

Chart, bubble chart

Description automatically generated

Fig. 15. Results from K-Means Clustering

Fig. 15. is the visualization that comprises 30 clusters with 10  data points(ingredients)  in each  cluster. There is no visible hyperplane in this 2D visualization that segregates the clusters because there exists numerous features (nutritional components) based on which its being clustered. We choose the cluster that contains the vector we appended ( deficiency vector) since it’ll have various ingredients that can cover the deficits and in turn be used to make some food product.

**9.2 Comparison Results**

Given below are some visualizations that can better help understand the nuances of the developed recommendation engine.

Chart, bar chart

Description automatically generated

Fig. 16. Standard vs Recommended values for protein and carbohydrates (in mg)

Fig. 16. shows the levels of protein that the engine recommends for  a user without a deficiency  vs. a user with a deficiency in protein and carbohydrates. As it is clearly visible, user’s with a deficiency are provided diets and recipes with ingredients which help in improving their intake for a given deficiency.

Chart, bar chart

Description automatically generated

Fig. 17. Standard vs Recommended values for saturated fat (in mg)

Fig. 17. shows the levels of saturated fat the engine recommends for a user that consumes a healthy diet vs. a user that consumes a fatty diet. As it is clearly visible, users who have an excess intake of a particular nutrient are provided diets and recipes with ingredients which help in reducing their consumption to have a good and healthy overall diet.

**9.3 Result Analysis**

The final recipe video recommendation is made based on multiple stages of intermediate results. Initially, two main datasets are used. All the intermediate results and computations are made based on these datasets. One consists of the standard nutritional values of a given age and gender. The other dataset consists of 300 ingredients along with the nutritional component values of each ingredient. On receiving the user inputs through the UI, it is compared with the standard values of the user.

Assume a user of a certain age and gender has a standard vector:

Sx = [s1,s2,s3,.....,sn],  where s1,s2,s3,.....,sn are all standard nutritional component values such as carbohydrates, proteins,  saturated fats and so on.

User input is fed in the form of the user vector:

Ux = [u1,u2,u3,.....,un],  where u1, u2, u3,....., un  are the user’s nutritional component values such as carbohydrates, proteins, saturated fats and so on.

A deficiency vector Dx is created by subtracting Ux from Sx, and its absolute value is considered.

Dx = Abs( Sx - Ux )

This deficiency vector Dx is further used as an input to the  clustering algorithm along with all the ingredients.

The second aforementioned dataset comes into play for the clustering algorithm. Assume each ingredient vector to be Ix=[ I1, I2, I3,....,I300] each of which is a vector with values [i1,i2,i3,.....,in], where i1, i2, i3,....., in are the ingredient’s nutritional component values such as carbohydrates, proteins,  saturated fats and so on.

Dx is then appended to Ix to  create the final dataset for clustering (Fx).

Fx = Dx + Ix

The Euclidean norm is computed in order to create a data point for each vector fi in Fx. On computing the clusters which comprise ingredients similar to Dx, we essentially identify and obtain the ingredients that could potentially cover up the deficiencies.

We further move to choose all the ingredients in the cluster that Dx exists. These ingredients are then fed into the Spoonacular API which proceeds to suggest various recipes. Upon obtaining multiple recipe recommendations, we further personalize the recommendations by choosing the top choices among the API recommendations by comparing the Euclidean norm of each recommendation with the user.

Thus, the recommendations as well as the ingredients each pass through a layer of personalization which ultimately aids in obtaining robust and accurate results.

**Chapter 10**

**CONCLUSION & SCOPE FOR FUTURE WORK**

In conclusion, the result of this research is an algorithm which ingests a user’s age, gender and nutritional profile to obtain a list of suitable ingredients for the user. The user’s personal preferences are also considered for recipe recommendation and depending on the user’s preferences certain ingredients are removed. The final list of ingredients are then fed into the Spoonacular API that gives various choices of final food products that can be made using the ingredients obtained from K-Means clustering. Another layer of personalization is added by ranking the choices suggested by the API on considering the user’s nutritional profile. The YouTube API is used to scrape the web to find videos of recipes of the final recommended food products. These videos are then displayed to the user along with other relevant information through the flutter application with an elegant UI and user-friendly design that encompasses a smart and personalized recipe recommendation engine. This research might manufacture a system that can greatly contribute to the cause by saving a considerable amount of time in choosing an appropriate recipe to fulfil all needs and help individuals evade life-threatening health conditions and malnourishment. In the future, the app may be improved so that users can take pictures of the ingredients they have at home using their phone's camera to get recipes which include only those ingredients and are nutritious. Users can also search for generic recipes based on a specific deficiency to enhance their nutritional profile.

**Chapter 11**

**REFERENCES**

1. Chuan-Jun Su, Yin-An Chen, and Chia-Wen Chih. (2013). Personalized Ubiquitous Diet Plan service based on ontology and web services. http://www.ijiet.org/papers/329-K012.pdf
2. Gao, Xiaooyan & Feng, Fuli & Huang, Heyan & Mao, Xian-Ling & Lan, Tian & Chi, Zewen. (2021). Food Recommendation with Graph Convolutional Network. Information Sciences. 584. 10.1016/j.ins.2021.10.040.
3. Wagner, Claudia & Singer, Philipp & Strohmaier, Markus. (2014). The nature and evolution of online food preferences. EPJ Data Science. 3. 10.1140/epjds/s13688-014-0036-7.
4. Kusmierczyk, Tomasz & Trattner, Christoph & Nørvåg, Kjetil. (2015). Temporal Patterns in Online Food Innovation. 10.1145/2740908.2741700.
5. Salloum, George & Tekli, Joe. (2021). Automated and Personalized Nutrition Health Assessment, Recommendation, and Progress Evaluation using Fuzzy Reasoning. International Journal of Human-Computer Studies. 151. 102610. 10.1016/j.ijhcs.2021.102610.
6. Gao, Xiaooyan & Feng, Fuli & He, Xiangnan & Huang, Heyan & Guan, Xinyu & Feng, Chong & Ming, Zhaoyan & Chua, Tat-Seng. (2019). Hierarchical Attention Network for Visually-Aware Food Recommendation. IEEE Transactions on Multimedia. PP. 1-1. 10.1109/TMM.2019.2945180.
7. Tanzil, Fidelson & Wulandhari, Lili & M. Isa, Sani. (2016). Artificial Bee Colony — Based for dietary recommendation in daily nutrition requirements. 1-6. 10.1109/KICSS.2016.7951406.
8. Harvey, Morgan & Ludwig, Bernd & Elsweiler, David. (2013). You Are What You Eat: Learning User Tastes for Rating Prediction. 8214. 153-164. 10.1007/978-3-319-02432-5\_19.
9. Rivolli, Adriano & Parker, Larissa & de Carvalho, Andre. (2017). Food Truck Recommendation Using Multi-label Classification. 585-596. 10.1007/978-3-319-65340-2\_48.
10. Mustafa, Annasari & Maulidiana, Annisa. (2019). The Effectiveness of Nutrition Education about Local Specific Food-based Balanced Nutrition Recommendation on Dietary Intake Level and Anemia Status in Female Adolescents at the Hidayatullah Arrohmah Islamic Boarding School Malang. KnE Life Sciences. 10.18502/kls.v4i15.5730.
11. Bundasak, Supaporn & Yoksuriyan, Prasopchok & Kuntawee, Patipan & Kotama, Rahat. (2021). Food recommendation system for the elderly. 18. 152-167.
12. Manoharan, Samuel & Ammayappan, Sathesh. (2020). Patient Diet Recommendation System Using K Clique and Deep learning Classifiers. 121-130.
13. Starke, Alain & Trattner, Christoph & Bakken, Hedda & Johannessen, Martin & Solberg, Vegard. (2021). The cholesterol factor : Balancing accuracy and health in recipe recommendation through a nutrient-specific metric.
14. Larson, Leila & Young, Melissa & Ramakrishnan, Usha & Webb-Girard, Aimee & Verma, Pankaj & Chaudhuri, Indrajit & Srikantiah, Sridhar & Martorell, Reynaldo. (2017). A Cross-Sectional Survey in Rural Bihar, India, Indicates That Nutritional Status, Diet, and Stimulation Are Associated with Motor and Mental Development in Young Children. The Journal of Nutrition. 147. jn251231. 10.3945/jn.117.251231.
15. Geleijnse, Gijs & Nachtigall, Peggy & Kaam, Pim & Wijgergangs, Luciënne. (2011). A personalized recipe advice system to promote healthful choices. 437-438. 10.1145/1943403.1943487.
16. Pinxteren, Youri & Geleijnse, Gijs & Kamsteeg, Paul. (2011). Deriving a recipe similarity measure for recommending healthful meals. 105-114. 10.1145/1943403.1943422.
17. Herrera, Juan. (2020). Sustainable Recipes. A Food Recipe Sourcing and Recommendation System to Minimize Food Miles.
18. Chun-Yuen Teng, Yu-Ru Lin, and Lada A. Adamic. 2012. Recipe recommendation using ingredient networks. In Proceedings of the 4th Annual ACM Web Science Conference (WebSci '12). Association for Computing Machinery, New York, NY, USA, 298–307. https://doi.org/10.1145/2380718.2380757
19. Chen, Yu & Subburathinam, Ananya & Chen, Ching-Hua & Zaki, Mohammed. (2021). Personalized Food Recommendation as Constrained Question Answering over a Large-scale Food Knowledge Graph.
20. Hao Cheng, Markus Rokicki, and Eelco Herder. 2017. The Influence of City Size on Dietary Choices. In Adjunct Publication of the 25th Conference on User Modeling, Adaptation and Personalization (UMAP '17). Association for Computing Machinery, New York, NY, USA, 231–236. <https://doi.org/10.1145/3099023.3099058>
21. Phanich, Maiyaporn & Pholkul, Phathrajarin & Phimoltares, Suphakant. (2010). Food Recommendation System Using Clustering Analysis for Diabetic Patients. 2010 International Conference on Information Science and Applications, ICISA 2010. 10.1109/ICISA.2010.5480416.
22. Mouzhi Ge, Mehdi Elahi, Ignacio Fernaández-Tobías, Francesco Ricci, and David Massimo. 2015. Using Tags and Latent Factors in a Food Recommender System. In Proceedings of the 5th International Conference on Digital Health 2015 (DH '15). Association for Computing Machinery, New York, NY, USA, 105–112. https://doi.org/10.1145/2750511.2750528
23. Trattner, Christoph & Elsweiler, David. (2017). Investigating the Healthiness of Internet-Sourced Recipes Implications for Meal Planning and Recommender Systems. 10.1145/3038912.3052573.
24. Min, Weiqing & Jiang, Shuqiang & Jain, Ramesh. (2019). Food Recommendation: Framework, Existing Solutions and Challenges. IEEE Transactions on Multimedia. PP. 1-1. 10.1109/TMM.2019.2958761.
25. Freyne, Jill & Berkovsky, Shlomo. (2010). Intelligent food planning: personalized recipe recommendation. 321-324. 10.1145/1719970.1720021.
26. National Institute of Nutrition. <https://www.nin.res.in/>
27. Spoonacular API, <https://spoonacular.com/food-api>

**Chapter 12**

**APPENDIX**

**12.1 Published paper certificate**