IRS-PM-ISY5001-2020-01-18-GROUP5-A.I. Food Recommender

A.I. Food Recommender

NUS-ISS Master of Technology (Intelligent Systems)

Project Report

Full Name	Student ID	Email
Mok Kay Yong	A0214617J	e0529481@u.nus.edu
Chong Keng Han	A0213547H	e0508648@u.nus.edu
Ly Duy Khang (Ken)	A0032571N	k.ly@u.nus.edu
Harry Chan	A0213530X	e0508631@u.nus.edu

1. Executive Summary **4**

2. Business Problem Background **5**

3. Project Objective **5**

4. Data Pre-Processing **6**
4.1. Food Data Scraping from HPB website 6
4.2. Food Data Scraping from eatthismuch.com website 7
4.3. Additional Food Data Scraping for Keto Vegan food 8
4.4. Food Classification with Data Mining 9

5. Knowledge Representation **11**
5.1. Daily Calorie Requirement 11
5.2. Diet Style 12
5.2.1. General Diet 12
5.2.2. Ketogenic Diet 13
5.2.3. Preference and Restriction 13
5.2.3.1 Vegetarian 14
5.2.3.1 Vegan 14
5.2.3.1 Halal 14
5.3. Meal Plan Structure 14

6. System Architecture and Design **14**
6.1. Overall System Architecture 14
6.2. System Components 18
6.2.1. Web User Interface Design 18
6.2.2. Django REST Framework 19
6.2.3. Business Rules Implementation In PyKE 19
6.2.4. Constraint Satisfaction and Optimization with Google OR-Tools 20

7. Prototype Testing **22**

8. Limitations **24**

9. Future Improvements **25**

10. Conclusion **26**

11. References **27**

Appendix A - Project Proposal **30**

Appendix B - Mapped System Functionalities Against Courses **39**

Appendix C - Installation and User Guide **40**

Appendix D - Project Team Members Individual Report **41**
Appendix D1: Mok Kay Yong A0214617J 41
D1.1. Personal contribution 41
D1.2. What is learnt? 42
D1.3. How can you apply the knowledge and the skills gained? 42
Appendix D2: Chong Keng Han - A0213547H 43
D2.1. Personal Contribution Summary 43
D2.2. Lessons Learned In This Project 43
D2.3. Potential Application of Knowledge and Skill In My Workplace 43
Appendix D3: Ly Duy Khang (Ken) - A0032571N 44
D3.1. Personal Contribution Summary 44
D3.2. Lessons Learned In This Project 44
D3.3. Potential Application of Knowledge and Skill In My Workplace 44
Appendix D4: Harry Chan - A0213530X 45
D4.1. Personal Contribution Summary 45
D4.2. Lessons Learned In This Project 45
D4.3. Potential Application of Knowledge and Skill In My Workplace 46

1. Executive Summary

In developed countries, the obesity level has been increasing rapidly and people look for ways to effectively lose weight. People are highly aware that Diet plays the major part of people's health and especially in their effort to maintain healthy body weight. Several publications estimate that the impact of diet is about 75% to 80% as compared to exercise which only accounts for 20% to 25% when it comes to weight loss *[1] [2] [3]*.

When it comes to diet style and plan we need to acknowledge that each individual is unique. People want to have options in choosing the right diet style that is most suitable for them (lifestyle, type of food, belief, etc). Each diet style has its own rule which can be confusing. Therefore people need easy guidance and idea on the selection of food based on their diet style and their food preference based on their belief/values (i.e. religion, vegetarian, vegan, etc)

The above is the background why our team chose this area as our project. Simply saying, food is very dear to our daily life and we believe that food plays a major part in our overall health.

At the same time we saw that we could apply what we have learned in this semester:

- Machine Reasoning: applying rule-based for each variety of diet style

- Optimisation technique to ensure our search results in the most optimum food choice based on the selected diet style

- Knowledge Discovery with Data Mining: find food words that can help to classify food based on diet requirements, using word lemmatization and word counts to get the significant food names and rare food names. In addition, manually to more precisely classify food based on keywords to ensure food selection for various types of diets is proper.

2. Business Problem Background

To comply with a certain type of diet, people need to know the nutrients content of their food. In canned or mass produced food, this information can be easily obtained by reading the ingredients and nutrition table.

However it is not so easy for home cooked food. The best way to get approximate information is to get it from local dieticians or official websites provided by trusted institutions. In our project, we source our samples for food database from 2 sites:

a. Singapore Health Promotion Board (HPB) *[4]*.
b. Eatthismuch.com, a reputable website which contains a database of international food *[5]*.

By getting the right information on the nutrients content of our daily food facilitated by trusted information, dieters can apply the diet rules in a manual way (manual calculation). The manual way will still be tedious, error prone and not easy to optimize certain objectives (i.e. to determine the list of food that they can consume on their daily basis to adhere to the total number of calorie or amount of certain nutrients based on their diet style). This is what we aim to solve or at least makes it a lot easier for dieters to make their daily food choice.

3. Project Objective

This project's objective is mainly 2 prongs:

a. To help dieters to easily choose their daily food based on the diet style and restrictions which could be complicated to plan on daily basis
b. To facilitate food vendors to list down their food offering (their restaurant or food establishment menu) into our database

However for this project scope, we mainly concentrate on the first objective since the second objective may need further business/commercialisation discussions. The second objective can still be achieved because putting the food database of restaurants or any

other food establishments can be done as a back-end process without any user interface required at this stage.

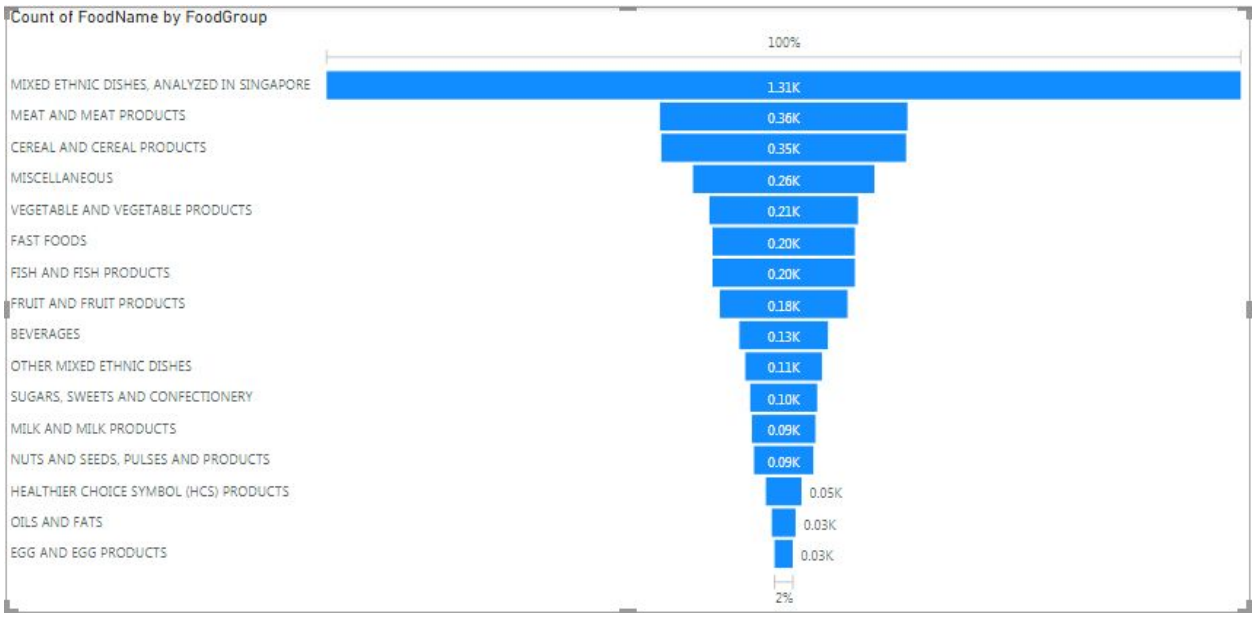
4. Data Pre-Processing

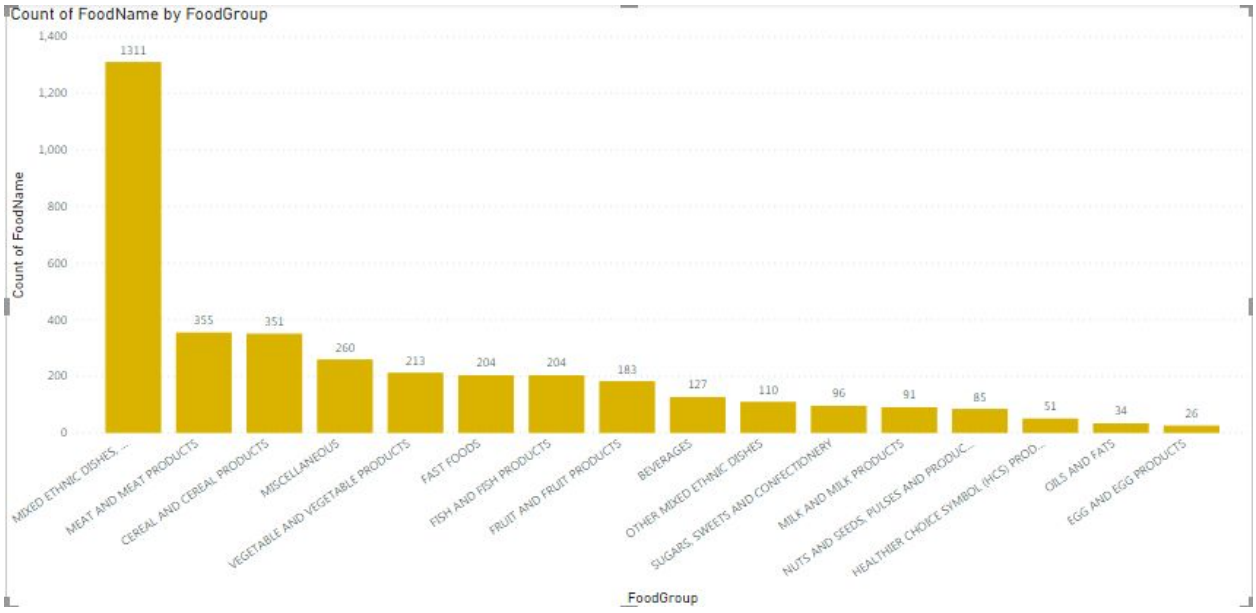
4.1. Food Data Scraping from HPB website

The URL used for food database scraping: https://focos.hpb.gov.sg/eservices/ENCF/

Below are the summary of data scraped from the above URL:

- Total food records scraped is 3701 records
- The biggest food group is “Mixed Ethnic Dishes Analyzed In Singapore”
- There are 16 food groups, represented below in both funnel chart and bar chart with indication on the number of records for each food group
- Important data points:
    - Food Name
    - Food Group
    - Food Sub Group
    - Energy or Calorie Amount in Kilo Calorie
    - Carbohydrate Amount in gram
    - Protein Amount in gram
    - Fat Amount in gram

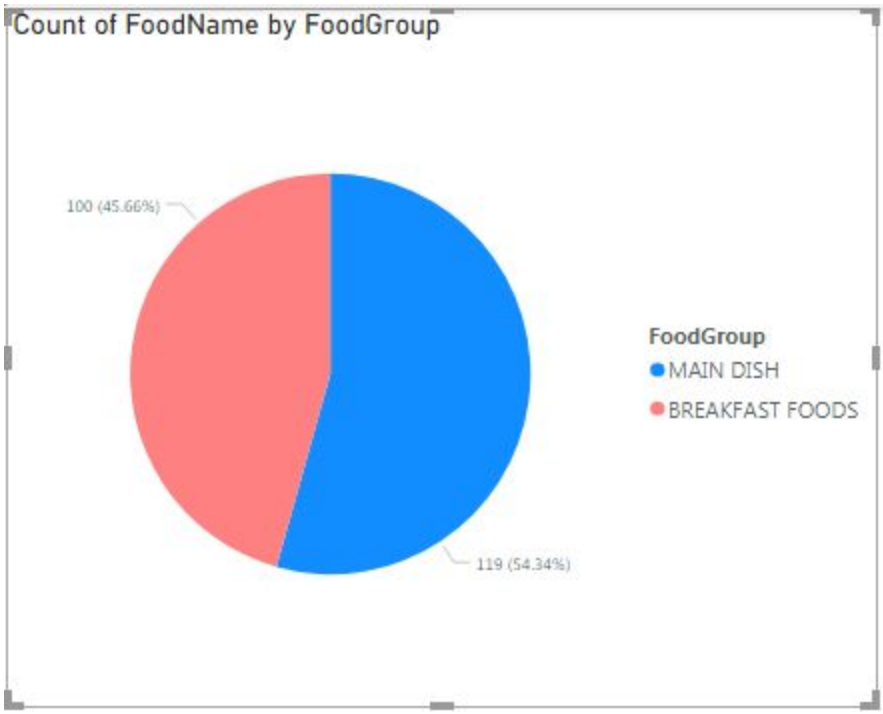




4.2. Food Data Scraping from eatthismuch.com website

To supplement the food data scraped from HPB website, we decided to scrap some food records from https://www.eatthismuch.com/:

- Breakfast Food: this is to ensure that we can get the right food for breakfast because not all types of food is suitable for breakfast. The breakfast type of food from Singapore HPB is only limited to cereal type of food.
- Main Dish: we selected main dish with meat to make sure ketogenic diet has more options since ketogenic diet is a type of diet that is very low in carb and high in fat and protein



FoodGroup	FoodSubGroup	Count of FoodName
MAIN DISH	Main dish with meat	119
BREAKFAST FOODS	Breakfast foods	100
Total		**219**

4.3. Additional Food Data Scraping for Keto Vegan food

In general vegan preference is the most restrictive among all other preferences or restrictions since it cannot include any animal source or derived food material. When it is combined with the ketogenic diet, that will make the food selection to be very limited because the ketogenic diet is supposed to be rich in fat and most food with rich fat content is mostly animal based.

To make sure that we get results with this combination (i.e. Keto Diet and Vegan preference), some food items are scraped especially from an additional website that provides recipes for Ketogenic Vegan food *[17]*.

4.4. Food Classification with Data Mining

To make it easier for the optimizer program to make the food selection, the team decided to add some boolean flags into the food database table:

- IsVegan
- IsVegetarian
- IsHalal

The indicators of whether the food is of the 3 categories is in their names. We use what we learnt in the ***Cognitive Systems*** to find rules for classification. We break down the list of food names into a distribution of the words vs the number of occurrences can provide a list of words for use in classification. Before that, preprocessing of the list of food names is required.

The preprocesses are:

1. Removal of punctuations
2. Lemmatize words to get an accurate word count
3. Removal of known stop words from words list

A dictionary of words vs count is generated using “Counter” from “collections” library, which has 2009 words. The number of words are too huge to sift through, and so the dictionary is sorted in descending order by counts. This can show the most common words, which is useful in getting keywords that can help to classify the food. This can also show the least common words, where some non-english words such as Malay words can indicate the food to be Halal.

These words are then analysed and used in rules for classification.

However, there are food names that contain ambiguity such as “fish ball, vegetarian”, which makes classification difficult. Therefore to gain accuracy, there is the subsequent process of manual filtering keywords in food names.

The data goes through a series of python scripts in Jupyter notebook. The summary of steps are below:

1. Initialize the flags for IsVegan, IsVegetarian, IsHalal to true

2. Update Beverages:
    a. Set all food name containing key-words associated with milk so that IsVegan = false
    b. Set all beverages not containing milk so that IsVegan = true
    c. Set all alcoholic beverages so that IsHalal = false
3. For food group 'CEREAL AND CEREAL PRODUCTS':
    a. For food sub-group 'Biscuits(eg crackers)': set all food names containing key words associated with milk i.e. 'chocolate','cream','ovaltine','cheese','butter' so that IsVegan = false
    b. Overwrite step 3a such that food containing plant based compound keywords such as 'cream cracker','peanut butter' will be set = true for the IsVegan flag
    c. For food sub-group 'Breads and rolls' (Note: subsequently these 3 sub-steps below are called as Vegetarian-Vegan-Halal filter and it will be applied to other food sub-groups in subsequent steps)
        i. Set all food names containing meat products so that IsVegetarian and IsVegan flags are false
        ii. Set all food names containing animal-sourced or animal-derivatives keywords so that IsVegan = false
        iii. Set food names containing non-halal keywords so that IsHalal flag = false. (Note: halal doesn't mean only without pork and lard, some dishes with chicken and beef by default are non-halal unless certified by the Moslem Authority body in the country since the way the animal is slaughtered needs to adhere to a standard. Generally all fish based products are considered halal. All food containing alcohol is considered as Non-Halal)
    d. For food sub-group 'Breakfast cereals': all food names containing animal derivatives word such as 'honey' should be set false for their IsVegan flag
    e. For food sub-group 'Buns (baked and steamed,plain and with sweet fillings)', apply the Vegetarian-Vegan-Halal filter
    f. For food sub-group 'Cakes and pastries(eg cakes mixes,croissants)', apply the Vegetarian-Vegan-Halal filter
4. Classification at overall food-group level: apply the Vegetarian-Vegan-Halal filter

5. Classification at overall food sub-group level: apply Vegetarian-Vegan-Halal filter
6. Apply the finer level detail filter which can only be done at food name level based on certain key-words. This needs fine-tuning by checking the result at records level after applying the filter flag. For example we can check all those food names flagged as Halal and if we see some food names that are not supposed to be there i.e. 'kway chap' because we know kway chap is pork related food then the specific filter based on food name (specific to 'kway chap') must be applied as the last stage filter.

5. Knowledge Representation

In this section, we would like to explain the basic knowledge representation which are then translated as rules and objectives inside system

5.1. Daily Calorie Requirement

Below are the logical steps on how Daily Calorie requirement is derived:

1. Calculate BMR

   Basal Metabolic Rate (BMR) is the total number of calories that your body needs to perform basic, life-sustaining functions. These basal functions include circulation, breathing, cell production, nutrient processing, protein synthesis, and ion transport.

   The Harris-Benedict Equation is often used to estimate/calculate basal metabolic rate *[6] [7]*

   Input user gender, height, weight and age

- 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

2. Calculate TDEE (Total Daily Energy Expenditure)

   From BMR, we can derive the Total Daily Energy Expenditure (TDEE) or Total Calorie Budget based on the estimation of physical activity level below *[8]* :

- Sedentary. You regularly have to tell Netflix you are still watching. You don't intentionally exercise at all. Sedentary = BMR x 1.2
- Lightly Active. You casually stroll through your neighborhood a few times a week. On average, you walk for exercise about 30 minutes a day. Another way to think about this would be 15 minutes per day of vigorous exercise like running or lifting weights. Light Active = BMR x 1.375
- Moderately Active. If we called the gym on a weeknight looking for you, they'd find you. This averages out to about one hour and 45 minutes of walking (for exercise, not going around your house) a day, or 50 minutes of vigorous exercise a day. Moderately Active = BMR x 1.55
- Very Active. You work in construction during the day and you're on the company softball team. This averages out to about four hours and 15 minutes of walking (again, for intentional exercise) a day, or two hours of vigorous exercise. Very Active = BMR x 1.725

5.2. Diet Style

5.2.1. General Diet

We define general diet as a diet style with no particular restriction of food. The only constraint that is applicable is the number of calories per day and the amount of protein that a person needs to consume to preserve muscle mass. These 2 constraints are applicable for any other type of diet program.

Muscle mass is specifically important because muscle cells burn more calories as compared to fat cells. Any type of sustainable diet needs to maintain muscle mass to be effective for the long term period. Muscle mass significantly determines the level of metabolism in our body.

Constraints:

1. Amount of calorie per day cannot exceed the budgeted calorie as per TDEE (Total Daily Energy Expenditure) in section 5.1.

2. Minimum amount of protein required to preserve a person's muscle mass is: 0.8 gram per kg of body weight *[9]*. For example if the person's body weight is 70kg, the minimum amount of protein to be consumed on daily basis would be 70 x 0.8 = 56 grams
3. To make the diet balanced, a balance proportion of 40%:30%:30% is added for the proportion of calorie derived from carbohydrate, fat and protein. This is to make sure that the optimizer will not pick up foods which are concentrated on a certain type of nutrient.

5.2.2. Ketogenic Diet

Ketogenic diet is a type of low carbohydrates high fat diet with the objective to change body metabolism from burning glucose to burning fat (in the form of ketones) *[10]*.

This diet has been used effectively to combat obesity, diabetic, epilepsy, and other chronic illness conditions (i.e. high blood pressure, . We choose this diet in this project due to its proven impact and benefit on many people. There is even active research on the usage of this diet to prevent long term neurodegenerative diseases like alzheimer and dementia *[11] [12]*.

The simplified rule of ketogenic diet:

1. A healthy ketogenic diet should consist of about 75% fat, 20% protein and only 5% or less than 50 grams of carbs per day. The percentage represents the amount of calorie from each nutrient (not the weight of the nutrients). General assumption: 1 gram of fat = 9 calories, 1 gram of carb = 4 calories and 1 gram of protein = 4 calories.
2. Carb limit of 20 gram per day (strict) or 50 gram per day (medium). Ideally this can be set in the parameter because some people can be in ketosis state only when they are in super low carb (< 20 gram per day) and some people can still be in ketosis state even when their carbs consumption is more than 50 gram per day. In this project we simplify it. We just have one criteria that carbohydrates must be less than 50 gram per day.

5.2.3. Preference and Restriction

Preference and Restrictions that we selected in this project are limited to three: Vegetarian, Vegan and Halal because these food restrictions is something that our data mining program in Python can classify

5.2.3.1 Vegetarian

Vegetarian diet is a diet free of animal meat/flesh or any organ, but it allows the consumption of milk, egg, cheese, butter, honey etc which are derived from animal's products.

5.2.3.1 Vegan

Vegan diet is similar with Vegetarian diet but it is more strict because it excludes all animal products, so milk, egg, cheese, butter, honey and all other animal derivative products are excluded from this diet.

5.2.3.1 Halal

Halal food restriction is required for Moslem community. In general all vegetables and seafood are considered Halal. For meat, it is well known that halal food excludes pork, lard and any other source from pigs. All products which contain alcohol also need to be excluded.

General meat such as chicken, beef and lamb are considered halal only if they are certified by the national/official Moslem body because it has something to do with the way the animals are slaughtered.

5.3. Meal Plan Structure

In this project, the meal plan structure is simplified into 3 main meals: Breakfast, Lunch and Dinner.

6. System Architecture and Design

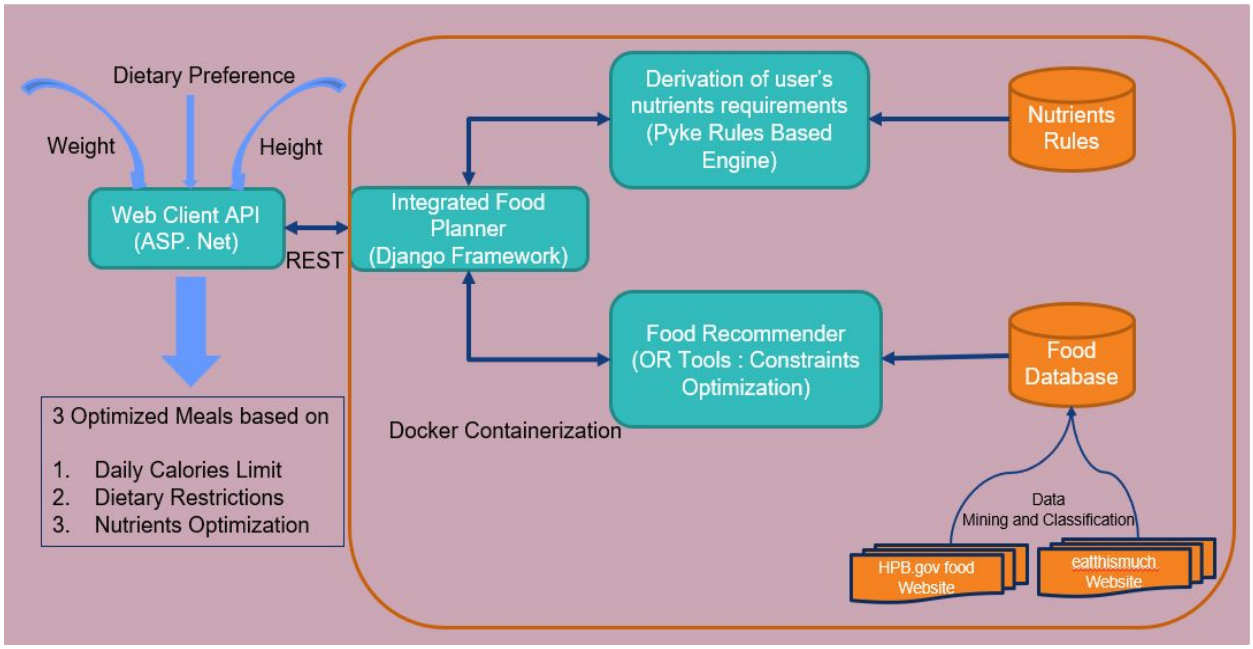
6.1. Overall System Architecture

In this system, the backend which is done in the Django Framework has 2 functions.

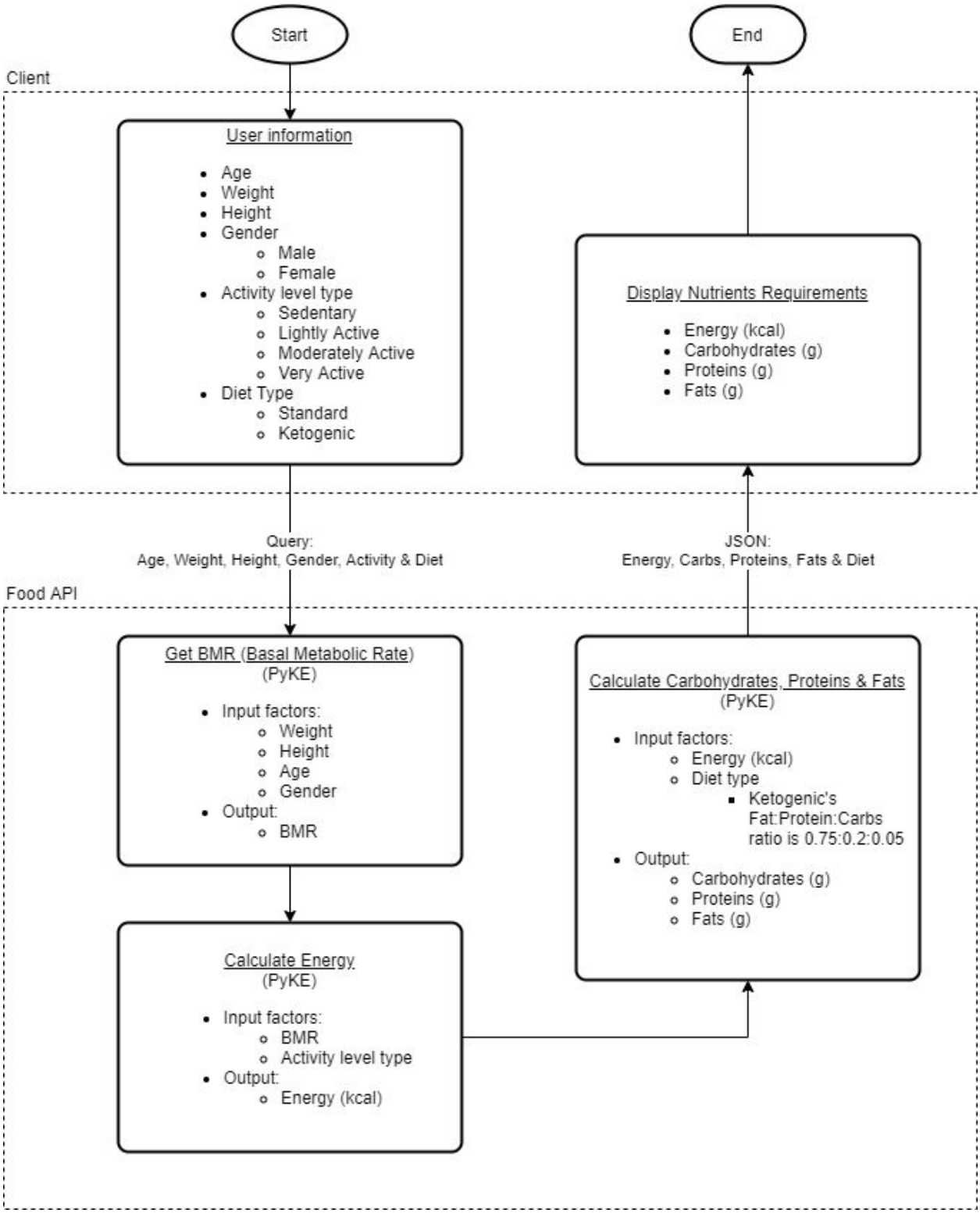
1. Compute recommended nutrients from the user's information.
2. Search and optimise food list from the food database, given the constraints from the user's required nutrients.

At the front end, a simple web client is used as a user interface to the backend. It passes the user inputs as queries to each function in the backend, and displays the result to the user.

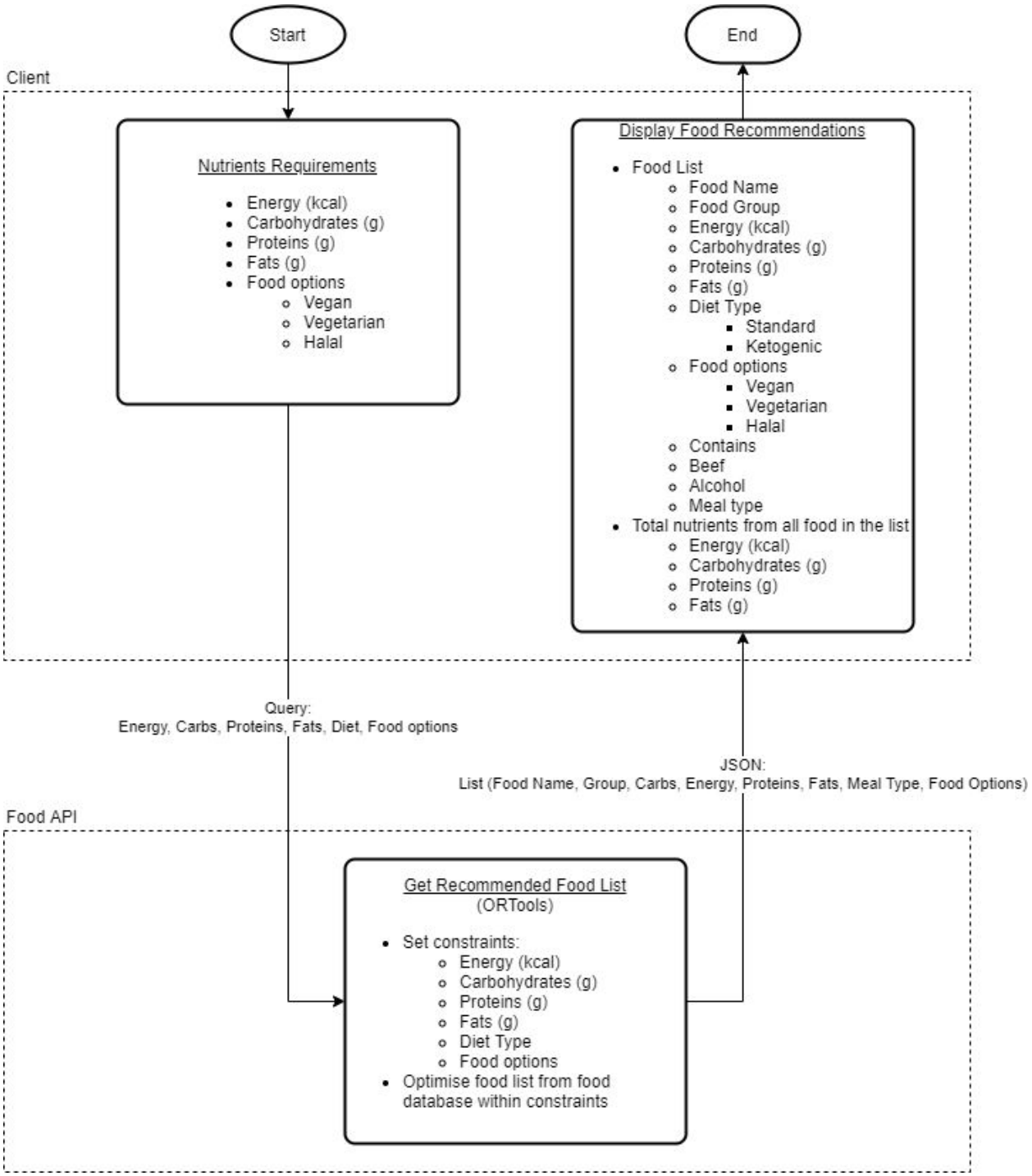
Below is the overall architecture of the food recommendation application:



Below is the flow chart for calculating the required nutrients from user information:



Below is the flow chart for getting the food recommendations from the required nutrients:



6.2. System Components

6.2.1. Web User Interface Design

The web client user interface is written using ASP.NET's MVC (model-view-controller).

The models are the data objects used in the views, and the data objects deserialize from the JSON return from the Food API. The data objects in the client are

1. User profile
2. Nutrients
3. Food options
4. Food list
5. View's data

The views are razor views (.cshtml) which can handle both HTML and C# codes. Data object is passed to the view from the controller, either as inputs from textboxes or to populate the display fields.

The controller processes the user inputs and updates the data objects to the view. Since this is a simple UI which interacts with the Food API, the data objects for the view are static objects in the controller.

There are 2 queries to the Food API.

1. Get recommended nutrients from user information
    a. Input: user profile data from UI
    b. Return: nutrients data
2. Get a recommended food list from nutrients and food options.
    a. Input: nutrients and food options from UI
    b. Return: food list

A class is added to interface with the Food API. It converts the data objects from the controller into queries to the Food API, and then deserialize the returned JSON into data objects back to the controller. It has 2 public methods, which corresponds to the 2 queries to the Food API.

Errors generated by the controller or the food API interface will be shown on the UI, so that the user is able to respond to fix the error. For example, when the client is not connected to the Food API, the System.IO.Stream will throw the error “No connection could be made because the target machine actively refused it.”.

6.2.2. Django REST Framework

Django REST Framework (DRF) [14] is a powerful and flexible toolkit for building HTTP APIs, that is extensible from the popular Django open-source web framework.

In particular, some of the reasons that we consider DRF are:

- Playing into the strength and experience of the majority of our project team members, Python is chosen as the main programming language for the core backend system.
- Django itself is a high-level Python Web framework that bundles in a lot of clean, pragmatic designs influenced by experienced open-source communities, and therefore facilitates rapid development without the need to reinvent the wheel.
- DRF’s Web browsable API is a huge usability win for everyone in the team. We are able to bootstrap the development of the backend quickly, and visualize the APIs inputs and outputs dynamically - out of the box.

6.2.3. Business Rules Implementation In PyKE

In this project, PyKE [15] powers the knowledge-based inference engine (i.e. expert system) for calculating nutrients based on different user health profile parameter inputs, as well as diet and restriction choices. Here, we apply what we learnt from ***Machine Reasoning***.

One important advantage of PyKE is that it integrates seamlessly with Python allowing the invocation of PyKE from normal Python programs, as well as intermingle Python statements and expressions within PyKE’s expert system. Furthermore, the knowledge base in PyKE is constructed with familiar facts and rules - a form of Logic Programming, that makes it very intuitive for translating the knowledge representation described in section 5. The Backward chaining rules are then processed when our program asks PyKE a question (i.e., asks PyKE to prove a specific goal) about the value that we need calculation for.

As an example, we have implemented the following BMR calculation rule in our system:

```
rule_calculate_bmr_male

    use calculate_bmr($bmr, 'male', $weight, $height, $age)

    when

        python bmr = calculate_bmr_male($weight, $height, $age)

        $bmr = bmr

def calculate_bmr_male(weight, height, age):

        return 10 * weight + 6.25 * height - 5 * age + 5
```

Given the input of a person's weight (`$weight`), height (`$height`), age (`$age`), and gender (`'male'`), this specific rule will be matched and activated by PyKE to calculate **`$bmr`** - the BMR value.

6.2.4. Constraint Satisfaction and Optimization with Google OR-Tools

Google OR-Tools [16] is an open-source optimization software suite that is very lightweight but equipped with powerful algorithms to solve some of the world's toughest problems in vehicle routing, flows, integer and linear programming, and constraint programming. This is where we applied what we learnt in the ***Reasoning Systems***.

For our Food Recommendation application, we model a Linear Programming problem where all variables are integers, and use the built-in solver to obtain the results for objective function optimization as well as constraints requirements. In details:

- Data for the problem: a 2-dimensional array of nutritional data *`food_data[i][j]`* that we have prepared in section [], where each row represents a unique food item, and the columns are its corresponding attributes including Name/Food group, Nutrient Values (Calories, Protein, Fat, etc.), and Diet Restriction labels (isVegan, isVegetarian, etc.)
- Variables: a 1-dimensional array of food items *`food[i]`*, mapped to each row in the data input. Each element of this variable array can either have the value of 0 or 1.

`food[i] = 1` if the food item i^{th} is selected in the recommendation output result, and vice versa.

The job of the optimization solver is thus to find the values of this variable array that satisfy the constraints and minimize the objective function defined next.

- Objective function: We will minimize the amount of total calories from the selected output food items. The objective function can be defined as

$$\sum_{i=1}^{n} food[i] * food\_data[i]['Calories']$$

- Constraints: The nutrients amount (Carb, Fat, Protein) as instructed for each Diet type (Standard or Ketogenic) - the results of PyKE calculation previously, together with the Restrictions for food items (Halal, Vegan or Vegetarian) are the constraints that we need to satisfy in optimizing the objective function.
    - Example of Diet Types constraint for Carb:

$$\sum_{i=1}^{n} food[i] * food\_data[i]['carb'] \geq carb\_amount_{PyKE}$$

    - Example of Restriction constraint for Vegan (i.e. all food must be Vegan):

food[i] == 0 if food_data[i]['is_vegan'] = False

    - In addition, to aid the solver to find a solution that conforms to our meal plan structure mentioned in section 5.3, additional constraint for the total number of meals are also defined:

$$\sum_{i=1}^{n} food[i] == 3$$

Randomness is introduced throughout the objective function, and constraints calculation to simulate the effect that the users now have the possibility of getting back different output results in multiple runs of the program, even with the same set of inputs.

However, even with randomness, due to the optimization nature of the solver, it is not sufficient to guarantee unique items in the output results, especially when our program

also allows users to explicitly **refresh** the results - either by changing the entire result set, or just changing a subset of the result set while retaining the rest (e.g. just change the breakfast, for the same lunch/dinner). Therefore, session variables are implemented in the API to keep track of the history result of the solver and the choice of the user. These session variables are then used to influence the solver in the next refresh run by marking the variable as 1 or 0 upfront.

7. Prototype Testing

Profile Inputs	**Nutrients Outputs**	**Food Recommendation Result**
Age: *40* Height: *180* Weight: *90* Gender: *Male* Activity: *Lightly Active* Diet: *Standard*	Total calories: *2516.25* Carbohydrates: *251.62* Protein: *188.72* Fats: *83.88* Restriction: *None*	Breakfast: - Name: *Tropical Skin Cleanser Green Smoothie* - Energy: *1174 KCal* - Carbohydrates: *93 gr* - Proteins: *134 gr* - Fats: *23 gr* Lunch: - Name: *Maize, raw* - Energy: *743.7 KCal* - Carbohydrates: *147.85 gr* - Proteins: *19.31 gr* - Fats: *8.44 gr* Dinner: - Name: *Chicken kurma* - Energy: *620.83 KCal* - Carbohydrates: *2.78 gr* - Proteins: *53.73 gr* - Fats: *43.85 gr* ========== Nutrients Proportion: - Carbohydrates: *38.39%* - Proteins: *32.62%* - Fats: *26.69%*
Age: *53* Height: *175* Weight: *63* Gender: *Male* Activity: *Very Active* Diet: *Ketogenic*	Total calories: *2524.97* Carbohydrates: *31.56* Protein: *126.25* Fats: *210.41* Restriction: *Vegan*	Breakfast: - Name: *Gouda & Green Onion Omelet x2* - Energy: *1459 KCal* - Carbohydrates: *10 gr* - Proteins: *81 gr* - Fats: *122 gr* Lunch: - Name: *Seeds, melon seed, flesh only*

		- Energy: *601.56 KCal* - Carbohydrates: *16.52 gr* - Proteins: *30.56 gr* - Fats: *51.19 gr* Dinner: - *Name: Coconut, grated and desiccated* - Energy: *497.37 KCal* - Carbohydrates: *5.48 gr* - Proteins: *4.73 gr* - Fats: *48.83 gr* ========== Nutrients Proportion: - Carbohydrates: 5% - Proteins: 18.19% - Fats: 78.11%
Age: *28* Height: *173* Weight: *63* Gender: *Female* Activity: *Moderately Active* Diet: *Ketogenic*	Total calories: *1939.09* Carbohydrates: *24.24* Protein: *96.95* Fats: *161.59* Restriction: *Vegetarian*	Breakfast: - Name: *Gouda & Green Onion Omelet x2* - Energy: *1459 KCal* - Carbohydrates: *10 gr* - Proteins: *81 gr* - Fats: *122 gr* Lunch: - Name: *Seeds, melon seed, flesh only* - *Energy: 601.56 KCal* - Carbohydrates: *16.52 gr* - Proteins: *30.56 gr* - Fats: *51.19 gr* Dinner: - Name: *Braised taupok with vegetable* - *Energy: 137.98 KCal* - Carbohydrates: *1.2 gr* - Proteins: *6.6 gr* - Fats: *11.86 gr* ========== Nutrients Proportion: - Carbohydrates: *5.04%* - Proteins: *21.5%* - Fats: *75.75%*
Age: *76* Height: *161* Weight: *55* Gender: *Female* Activity: *Sedentary* Diet: *Standard*	Total calories: *1218.3* Carbohydrates: *121.83* Protein: *91.37* Fats: *40.61* Restriction: *Halal*	Breakfast: - Name: *Cottage Cheese with Blueberries and Honey* - Energy: *262 KCal* - Carbohydrates: *31 gr* - Proteins: *24 gr* - Fats: *5 gr*

		Lunch: - Name: *Dry ban mian* - Energy: *749.39 KCal* - Carbohydrates: *86.97 gr* - Proteins: *35.01 gr* - Fats: *28.99 gr* Dinner: - Name: *Fish, soup, with eggplant, tamarind and pepper* - Energy: *227.83 KCal* - Carbohydrates: *6.25 gr* - Proteins: *26.95 gr* - Fats: *10.56 gr* ========== Nutrients Proportion: - Carbohydrates: *40.1%* - Proteins: *27.75%* - Fats: *32.36%*

8. Limitations

Below are some known limitation for the system delivered in our project:

1. Number of meals is something that we define as 3 meals (breakfast, lunch and dinner) to cater for the majority of the population. We understand that some dieters can prefer less (i.e. 2 meals or even 1 meal) and some dieters can prefer more (3 meals with snacks in between)
2. For the ketogenic diet, we put the assumption that fat is of the same quality. In practice, ketogenic dieters are strongly advised to consume healthy fat and avoid unhealthy fat *[13]*. This is not defined in our project because it will require much scrutiny at the food ingredients detail level
3. The data mining to classify food into IsVegetarian, IsVegan and IsHalal are based on classification (food category and food sub-category) as well as food name. There is no scrutiny up to food ingredients/recipe level
4. Meal serving is solely based on food database whilst we acknowledge that serving size can be very different from one person/dieter to another

9. Future Improvements

Below are the list of potential future improvements that can be done:

1. Food recommendation should be enriched to include number of meals to be defined by dieters and allow them to select category (main dish or snack)
2. Amount of food portion or number of servings can be recommended for each meal, which requires a more complicated optimiser. This is sometimes required because some food serving is too small or too big for individual dieters.
3. More food restrictions can be added, for example:
    a. some people with Hindu or Buddhist religion avoid beef related products
    b. some major allergies can be added so that dieters can avoid food which can cause them severe allergic reactions (i.e. seafood, lactose, nuts, etc).
4. Learning of user food preferences to create a more customised food recommendation. However, this may have implications on the legal issues or privacy.
5. More features for food providers (restaurants or other food establishments) to list down their menu can be added along with premium features if the product is commercialised
    a. This will increase our food database, and the accuracy of the food database to reflect the local food scene.
    b. The earnings can be through payment from food vendors to increase their food visibility in the application.
6. Recommend food based on location of the user. This is more feasible when another database of food providers are added into the application. But there is the legal issue of user privacy.

10. Conclusion

- The food recommendation system that we build for this project is able to recommend the correct food based on 2 diet plan (standard or ketogenic) combined with restrictions (no restrictions, vegetarian, vegan and halal) as a result of the right rules based being pumped into PyKE and with set of goals and restrictions defined properly inside Google OR-Tools
- Teamwork experience during this project is very valuable to us. Initially our project team planned to meet regularly in one physical location to plan the project and work on activities or certain parts of the project that are very difficult to be done via online meeting. However due to the recent COVID-19 Circuit Breaker situation, we didn't manage to do that and purely manage the project via online meetings that we do on a weekly basis (Tuesday 9pm via Microsoft Teams provided by NUS Office 365). We are quite surprised that despite all these limitations, we can still finish the project on time. It is an encouraging feat and very valuable experience to all of us as a team. Below is a summary of each project team contribution:
    - Ly Duy Khang (Ken): our project and technical lead. Ken is very fast in mastering new technology. He willingly and patiently transfers the knowledge to all other team members. He setups the github and plan the technology components used in this project (i.e. PyKE, Google OR-Tools, DJANGO REST framework, Docker containerization, etc). Back-end components are done by Ken and he makes it easily connected to the front end and modularised such that other team members can contribute in the coding segments
    - Mok Kay Yong: our second technical lead. Mok is the one in charge for developing the front end user interface as well as initiating and developing the data mining and food classifications.
    - Dennis Chong: Dennis acquires the food database data through scraping the HPB website and puts it into the csv file which is used as the database for this project. He is also the creator of our project video animation
    - Harry Chan: Harry researches the articles on this topic because he has personal interest in diet (especially ketogenic where he has got the benefit of practising it although not in a strict manner). He works with Mok on the food

classification/categorization and added food database by scraping from eatthismuch.com. He is also the main writer for project documentation

- The technology used in this project: PyKE, Google OR-Tools, DJANGO REST framework along with tools below will provide each team member with valuable skill, knowledge and experience in near future
    - Github
    - Working together using Google Docs (note: in my workplace I use Office 365 which is commercial tool, so nice to know that Google docs allow free use of their document collaboration)
    - Trello, for the project management, breaking down into manageable task and assign it to team members
    - Visual Studio Code which provides single UI to edit codes and test them

11. References

[1] Wexler, Sarah. 6th Dec 2017. “Exercise Vs. Diet: The Truth About Weight Loss”. Huffpost.com, [Online]. Available: https://www.huffpost.com/entry/exercise-vs-diet-for-weight-loss_n_5207271

[2] Bellus and Haubursin. 2nd Jan 2019. “The science is in: exercise won’t help you lose much weight”. Vox.com, [Online]. Available: https://www.vox.com/2018/1/3/16845438/exercise-weight-loss-myth-burn-calories

[3] Bellus and Zarracina. 31st Oct 2017. “Why you shouldn't exercise to lose weight, explained with 60+ studies”. Vox.com, [Online]. Available: https://www.vox.com/2016/4/28/11518804/weight-loss-exercise-myth-burn-calories

[4] Health Promotion Board, Singapore. "Energy and Nutrient Composition of Food," Health Promotion Board, Singapore, [Online]. Available: https://focos.hpb.gov.sg/eservices/ENCF . [Accessed 1 Mar 2020].

[5] Eatthismuch.com, “Food and Recipe browser”, Eatthismuch.com [Online]. Available: https://www.eatthismuch.com/food/browse/?type=recipe

[6] Frey, Malia. 10th April 2020. “Calculate Your Basal Metabolic Rate to Lose Weight”. Verywellfit.com [Online]. Available: https://www.verywellfit.com/what-is-bmr-or-basal-metabolic-rate-3495380

[7] Wikipedia.com. “Harris-Benedict equation”. Wikipedia.com [Online]. Available: https://en.wikipedia.org/wiki/Harris%E2%80%93Benedict_equation

[8] Kamb, Steve. 1st Jan 2020. “How Many Calories Should I Eat Every Day? A TDEE Calculator for Total Daily Energy Expenditure”, [Online]. Available: https://www.nerdfitness.com/blog/how-many-calories-should-i-eat-every-day-a-look-at-total-daily-energy-expenditure-tdee/

[9] Gunnars, Chris. 5th July 2018. “Protein Intake – How Much Protein Should You Eat Per Day?”. Healthline.com [Online]. Available: https://www.healthline.com/nutrition/how-much-protein-per-day

[10] Wikipedia.com. “Ketogenic diet”. Wikipedia.com [Online]. Available: https://en.wikipedia.org/wiki/Ketogenic_diet

[11] Wlodarek, Dariusz. 15th Jan 2019. “Role of Ketogenic Diets in Neurodegenerative Diseases (Alzheimer’s Disease and Parkinson’s Disease)”. PubMedCentral (PMC) [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6356942/

[12] Rusek, Pluta, Ulamek-Koziol, and Czuczwar. 9th Aug 2019. “Ketogenic Diet in Alzheimer’s Disease”. PubMedCentral (PMC) [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6720297/

[13] Carb Manager. “The Best and Worst Fats to Consume on Keto”, [Online]. Available: https://www.carbmanager.com/article/the-best-and-worst-fats-to-consume-on-keto-f673a066-f455-99f4-62b6-c5c1fdcd0fbc

[14] Django REST framework. https://www.django-rest-framework.org/

[15] PyKE - Python Knowledge Engine. http://pyke.sourceforge.net/about_pyke/index.html

[16] Google’s Operations Research tools - https://developers.google.com/optimization

[17] Clarke, Craig. 18th Feb 2019. “A Comprehensive Guide To The Vegan Ketogenic Diet”, [Online]. Available: https://www.ruled.me/comprehensive-guide-vegan-ketogenic-diet/

Appendix A - Project Proposal

GRADUATE CERTIFICATE: Intelligent Reasoning Systems (IRS)

PRACTICE MODULE: Project Proposal

Date of proposal: April 2020
Project Title: IRS-PM-ISY5001-2020-01-18-Group 5-A.I. Food Recommender
Sponsor/Client: 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: The proposed intelligent food recommender will make use of various advanced machine reasoning techniques and web framework for a user one-stop solution.

Requirements Overview:

- Research ability
- Programming ability
- System integration ability

Resource Requirements (please list Hardware, Software and any other resources)

Hardware proposed for consideration:

- Standard PC with browser.

Software proposed for consideration:

- Reasoning systems: **PyKe Tools** , **Google OR-tools**
- Pertained machine learning models: **NLP (nltk) for Data Classification for Rules**
- Machine learning use cases: **Jupyter Notebook** , **Python3**
- Application Framework: **Django REST Framework**
- Application container: **Docker**

Number of Learner Interns required: (Please specify their tasks if possible)

4 pax.

- Ken Ly: our project and technical lead. Ken is very fast in mastering new technology. He willingly and patiently transfers the knowledge to all other team members. He setups the github and plan the technology components used in this project (i.e. PyKE, Google OR-Tools, DJANGO REST framework, Docker containerization, etc). Back-end components are done by Ken and he makes it easily connected to the front end and modularised such that other team members can contribute in the coding segments
- Mok Kay Yong: our second technical lead. Mok is the one in charge for developing the front end user interface as well as initiating and developing the data mining and food classifications.
- Dennis Chong: Dennis acquires the food database data through scraping the HPB website and puts it into the csv file which is used as the database for this project. He is also the creator of our project video animation
- Harry Chan: Harry researches the articles on this topic because he has personal interest in diet (especially ketogenic where he has got the benefit of practising it although not in a strict manner). He works with Mok on the food classification/categorization and added food database by scraping from eatthismuch.com. He is also the main writer for project documentation

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 to 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	To ensure interface compatibility and confirm that the integrated system	1. Prepare System Test Specifications

acceptance testing	hardware and system software meets requirements and is ready for acceptance testing.	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
Delivery	To deploy the system into 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)

Team Formation & Registration

Team Name: Group 5
Project Title (repeated): A.I. Food Recommender
System Name (if decided): n.a.
Team Member 1 Name: Ly Duy Khang (Ken)

Team Member 1 Matriculation Number: A0032571N
Team Member 1 Contact (Mobile/Email): k.ly@u.nus.edu
Team Member 2 Name: Mok Kay Yong
Team Member 2 Matriculation Number: A0214617J
Team Member 2 Contact (Mobile/Email): e0529481@u.nus.edu
Team Member 3 Name: Harry Chan
Team Member 3 Matriculation Number: A0213530X

Team Member 3 Contact (Mobile/Email): e0508631@u.nus.edu
Team Member 4 Name: Chong Keng Han
Team Member 4 Matriculation Number: A0213547H
Team Member 4 Contact (Mobile/Email): e0508648@u.nus.edu

For ISS Use Only		
Programme Name:	**Project No:**	**Learner Batch:**
Accepted/Rejected/KIV:		
Learners Assigned:		

Advisor Assigned:

Contact: Mr. GU ZHAN / Lecturer & Consultant

Telephone No.: 65-6516 8021

Email: zhan.gu@nus.edu.sg

Appendix B - Mapped System Functionalities Against Courses

- **(MR**) Machine Reasoning: Applying rule-based for each variety of diet style using PyKE rules engine.

- **(RS)** Optimisation technique to ensure our search results in the most optimum food choice based on the selected diet style using Google OR-tools

- **(CGS)** Knowledge Discovery with Data Mining: find food words that can help to classify food based on diet requirements, using word lemmatization and word counts to get the significant food names and rare food names. In addition, manually to more precisely classify food based on keywords to ensure food selection for various types of diets is proper.

Appendix C - Installation and User Guide

The “Installation & User Guide” can be found at the following link

Appendix D - Project Team Members Individual Report

Appendix D1: Mok Kay Yong A0214617J

D1.1. Personal contribution

Web client user interface

In this project, I volunteered to build the user interface for the application. I used Microsoft ASP.NET MVC to create a web client to the food recommender API. In the web client, I created classes for models and data objects. The models are used in the view page. The data objects are the data to be passed to the food API as queries, the data serialised from the JSON returned from the food API and data in the models for the view page.

I created another class to interface with the food API, to handle the data from the view-controller to the food API as queries and to serialise the returned JSON into data objects for the view-controller. Errors, such as invalid arguments and food API connection error, are thrown to the controller to be added to the model state errors, so that the view page can display the error beside the button that triggers the query.

Classification of food

There was a need for the food data to be classified into food suitable for vegan, vegetarian and halal, but there was no classification of food in the food data. We decided to use the food names, food group name and food sub group name to determine the classification. Harry did a research for vegan came out with the rules, non-dairy, no eggs and no meat. For vegetarian, we used no meat rule. And for halal, it should be the halal certification from the food vendor, but since the food data has no food vendor data, we use the malay food names to determine the food as halal.

I created the data processing pipeline in Jupyter Notebook, so that it is easier to do EDA. First, the words from the food names are tokenized, to find out which keywords can be used as rules for classification for vegan, vegetarian and halal. Then, we fine-tuned the rules further by scanning the processed data for wrong food classification.

D1.2. What is learnt?

ASP.NET Web Client

This is the first web client that I build, which my team-mates requested me to do rather than build windows forms that I am familiar with. I chose the Model-View-Controller template because it has samples and tutorials which makes it easier to pick up. In this project, I learnt about handling the data objects and error handling in the web client, which is not as straightforward compared to windows forms.

EDA and NLP

The requirement to classify food into vegan, vegetarian and halal is not simple, because the food database does not have information of the breakdown of each food, just the nutrients. So we decided to classify food based on their names and group names. I have to learn about tokenizing words list of food names to use them as rules for the classification, and found that to build an accurate classification, it requires a huge set of rules. The lemmatization of words learnt in Dr Fan's class is very useful to get an accurate count of words, which is to get the significant words. However, the food data that we have are very local, and has Malay and Indian names, thus it is difficult to create useful rules with the nltk libraries.

D1.3. How can you apply the knowledge and the skills gained?

I would like to expand my knowledge on what was learnt while doing this project. Even though I learnt a lot from building a web client and food data classification, there are a lot more to learn, especially in the classification of data. Using keywords as rules are not good enough, and may need more combinations of words in the list of food names to make it more accurate, such as dairy versus non-dairy. The web client that I build is very simple and functional. There are more to learn of user interface or UX and client security, especially when user data is collected.

Appendix D2: Chong Keng Han - A0213547H

D2.1. Personal Contribution Summary

Below are the list of my contribution to the group:

A. Data scraping from https://focos.hpb.gov.sg/ for the first cut compilation of the working database.
B. Initial classification of various categories within the database (IsMain,IsBreakfast IsVegetarian, IsVegan and IsHalal).
C. Experiment and verification with various constraints with Google OR-tools.
D. Created Project Presentation Video.

D2.2. Lessons Learned In This Project

Below are lessons learned during this project:

A. First and foremost, teamwork. Different strengths within each team member is played out perfectly for the successful execution of this project.
B. DJANGO REST web framework are especially useful along with the Docker-based container concept for future projects
C. Rules-based implementation using PyKE and constraints optimization with OR-tools further complements the knowledge learned during and throughout the course.

D2.3. Potential Application of Knowledge and Skill In My Workplace

A. Man-power planning (Constraints Optimization)
B. Job scheduling for Plant Maintenance work (Constraints Optimization)
C. Plant process efficiency optimization (Constraints Optimization & Cognitive)
D. Prototyping with containers and web REST framework
E. Group collaboration tools (Trello , GitHub , Teams)
F. Video editing for project/proposal presentation

Appendix D3: Ly Duy Khang (Ken) - A0032571N

D3.1. Personal Contribution Summary

A. Setup the initial github repo + organization + tutorial for team members
B. Research on the overall framework and compatible technologies for the backend.
C. Boilerplate implementation to glue all the backend components together so that other members can contribute easily
D. Implementation of Django Rest Framework (DRF) bootstrapping + Documentation
E. Implementation of PyKE bootstrapping + Documentation
F. Implementation of OR-Tools bootstrapping + Documentation
G. Implementation of Production deployment strategy with Docker + Documentation
H. Weekly meeting discussion + Demo
I. Co-author the section “System Architecture and Design” in the Project Report
J. Co-author the section related to Backend deployment in the User Guide

D3.2. Lessons Learned In This Project

A. Harness the ability to overcome steep learning curve when interfaced with any new technology - to learn how to learn
B. Planning is very important for the success of the project. We started early and were able to adapt to the situation where all of us have to work remotely and independently due to COVID-19
C. Teamwork is crucial - to trust and motivate each other
D. Consistency is key - our team has never missed a weekly meeting

D3.3. Potential Application of Knowledge and Skill In My Workplace

A. Web framework for quick API prototyping + Minimum Viable Product (MVP)
B. Collaboration tools - github, Google docs, Microsoft Team
C. Knowledge about Rule Engines, Optimization and Cognitive as foundations for A.I.
D. The experience gained in our journey from idea inception to technology selection to implementation is most valuable as it helps stimulate a real life project.

Appendix D4: Harry Chan - A0213530X

D4.1. Personal Contribution Summary

Below are the list of my contribution to the group:

A. Research on how to calculate BMR (Basal Metabolic Rate), TDEE (Total Daily Energy Expenditure), etc.
B. Research on type of diets (especially on standard calorie based and ketogenic). I personally have been through a few types of diet and have a good interest in this topic. On the ketogenic diet, even though some people still view it as controversial, I am convinced that it will bring much benefit because I personally have been through this diet even though not strictly applying it on a day to day basis but it has been very helpful for me personally to maintain my weight in an acceptable range.
   The rules from the research are translated into PyKE
C. Research on food restrictions implemented in this project, i.e. Vegetarian, Vegan and Halal. This is translated into the data mining exercise to indicate each food into the restrictions category (IsVegetarian, IsVegan and IsHalal)
D. Food data scraping from eatthismuch.com to add the variety in breakfast food and main dishes
E. Checking/Verifying the food database and the data mining process to classify the food into various categories (to facilitate food restrictions such as IsVegetarian, IsVegan and IsHalal).
F. Testing the Google Ortools and verify whether the rule based work correctly with all team members
G. Writing the Project Report, cite the relevant references from internet articles and research papers available on the internet.
H. Setting the weekly project meeting in Microsoft Teams and record the meeting so all team members can refer to it if needed (especially if any of the team member cannot attend the session)

D4.2. Lessons Learned In This Project

Below are lessons learned during this project:

A. Teamwork is very very important. Each person in the team has been encouraging each other to contribute positively. Each of the team member's strengths makes up for other team member's weaknesses and at the end everybody is contributing towards the success of the project.
B. During the initial team forming, team members feel the need to meet physically at one place to discuss and work on certain parts of the project which is very difficult to do via online meeting. However at the end due to COVID-19 situation (Circuit Breaker) we ended up just meeting online via Microsoft Teams on a weekly basis. We are quite surprised that despite all these limitations, we can still finish the project. It is an encouraging feat and very valuable experience to me personally and all of us as a team
C. Technology components used in this project are very valuable to learn: PyKE, Google OR-Tools, DJANGO REST framework and also Python scripts implemented in Jupyter notebook for the Food Database classification and data mining
D. Accompanying tools that we have used are also very valuable learning experience:
    a. Github
    b. Working together using Google Docs (note: in my workplace I use Office 365 which is commercial tool, so nice to know that Google docs allow free use of their document collaboration)
    c. Trello, for the project management, breaking down into manageable task and assign it to team members
    d. Visual Studio Code to edit codes and test it

D4.3. Potential Application of Knowledge and Skill In My Workplace

Below are some potential applications in my workplace based on the knowledge and skill I have learned during this project:

A. The skills I learn from the python scripts applied in Jupyter notebook will be very handy in my workplace since I am in charge of Data Warehouse in my company and work mostly with data to realize the potential benefit, gain actionable insights that we can derive from our data asset. The data mining techniques and how the food classification is done will be crucial skill for me and my team in my company to enrich and enhance the data to be used for future projects

B. The other tool that will be very handy in my work context will be the Google OR-Tools which I can use to apply complex logical constraints and get the system to solve the problem and show the most optimal solution based on the list of constraints
C. The other technology that is worth learning is definitely DJANGO REST framework which enables back-end system to interact with web-based front end or user interface