IJAIT (International Journal of Applied Information Technology)

http://journals.telkomuniversity.ac.id/ijait/

MotivEat: A Web-Based Meal Planning Application with Ingredient Recognition for Personalized Nutritional Guidance

Britanny Baldovino [a, *], Beverly Estephany Ferrer [a]

[a] IT/CS Department, School of Accountancy, Management, Computing, and Information Studies, Saint Louis University, Philippines

bmbaldovino@slu.edu.ph, bpferrer@slu.edu.ph

ARTICLE INFO

Received July 26th, 2024
Revised July 7th, 2025
Accepted July 7th, 2025
Available online January 27th, 2026

Keywords
Machine learning, object detection, TensorFlow, meal planner, web application

ABSTRACT

Noncommunicable diseases are prevalent in the Philippines. Filipinos increasingly suffer from undernutrition or overnutrition. This can be caused by a lack of budget and nutritional awareness. Tolerating improper macronutrient intake is related to a higher risk of chronic diseases. Although varying solutions have been attempted, there is potential for discoveries. Specifically, with food choice technologies, motivation is an under-researched topic. Thus, this study explores the topic through the development of a meal planner web application that enables users to plan meals based on recipe selections tailored to their recommended macronutrient intake and available ingredients. Considering price, convenience, and experience, a proposed motivational concept based on nudging was to enable users to find recipes by scanning available ingredients. For this, a machine learning object detection model was trained to detect ingredients and find recipes. The model had a mean Average Precision (mAP) of 0.526 and an Average Recall (AR) of 0.539. Still, it could detect ingredients even when set at an accuracy of 90% and above. It was then integrated into a web application. Once accessible, the deployed web application was evaluated by nutritionist dietitians and other respondents. An average System Usability Scale (SUS) score of 79/100 was obtained. Moreover, the computed scores for the Intrinsic Motivation Inventory (IMI) subscales were all higher than 5 out of 7, with a Cronbach's Alpha value of 0.7. Responses to open-ended questions also showed that users are motivated to eat healthier if they have increased nutritional awareness and pre-existing ingredients in suggested recipes.

Acknowledgment

The authors acknowledge financial support from the Science and Technology Regional Alliance of Universities for National Development (STRAND) program by the Department of Science and Technology-Science Education Institute (DOST-SEI) in the Philippines.

* Corresponding author at:
School of Accountancy, Management, Computing, and Information Studies, Saint Louis University
Baguio City, Philippines
E-mail address: bmbaldovino@slu.edu.ph

ORCID ID:
First Author: 0000-0002-4576-0607

http://dx.doi.org/10.25124/ijait.v8i2.8134
Paper_reg_number IJAIT000080204

1. Introduction

"You are what you eat" is a proverbial expression often used to indicate the importance of one's food choices. Evidently, adequate nutrition from food is essential for achieving proper physical growth, productivity, survival, and overall health [1]. These are achieved by having a healthy diet in which nutrients are adequately consumed to sustain the human body's physiologic needs. Healthy diets often consist of fruits, vegetables, and whole grains. These foods contain macronutrients and micronutrients, which supply energy and provide health benefits to reduce the risk of developing diseases [2].

Nevertheless, the benefits of eating healthy food have not entirely made it preferable. The prevalence of unhealthy food, such as fast food, continues to rise. At one point in the pandemic, unhealthy food predominated in the volume of deliveries due to being promoted by food delivery services [3]. Unfortunately, having an unhealthy diet can cause non-communicable diseases (NCDs), which include cancer, diabetes, and cardiovascular diseases [4][5][6]. Thus, they are one of the principal risk factors for global deaths [7]. The World Health Organization (WHO) determined that NCDs accounted for 63% of all deaths worldwide [6]. While in the Philippines alone, NCDs were discovered as the cause of about 67% of the total deaths recorded [4].

The consumption of unhealthy food in the Philippines is comparably excessive in contrast to nearby countries. One reason may be that ultra-processed food and drinks are highly affordable and accessible to Filipino citizens [7]. There is also a perceived high cost of nutritious food and a lack of awareness regarding nutrition [8]. Although there are Filipinos who consume conventionally healthy food, their intake is less than favored. Abilgos-Ramos & Ballesteros [9] found that some children and adults were below the Acceptable Macronutrient Distribution Range (AMDR) for protein, fat, and carbohydrate intake. As they discussed, macronutrient consumption higher or lower than the AMDR implies nutrient inadequacy or amplified risk of chronic diseases.

The severity of associated NCDs bestows the utmost importance on the awareness of healthy eating. Thus, authorities apply different strategies to lessen the consumption of unhealthy food. Around the world, governments implement food regulatory policies. These include nutrition labeling, restrictions on food marketing, and taxation, which aim to support healthy food choices and decrease unhealthy food consumption [10]. Governments also reinforce dietary guidelines, which span from detailed advice to short and straightforward messages, including information on the quantity and frequency of how much food should be eaten [11]. In the Philippines, food-based dietary guidelines have also been created. However, the awareness and conformance of meal planners in the Philippines to them are still low [12]. It can, therefore, be seen that nutrition problems aggravate the country.

To solve the problem, technology is now being applied. The accelerated evolution of networks and technologies allows advanced data collection. This increases the possibility of food-related studies [13]. The stated tools are also used to develop food choice technologies with algorithms to promote healthy food to users [14]. These can take the form of food recommendation systems, meal planners, or diet and nutrition applications. Such systems can balance user preferences and nutritional needs. The information used to recommend food and recipes could be the user's personal information, eating habits, food preferences, physical activities, and health status. Therefore, these can help users improve their food consumption [15].

However, an essential requirement is that the recommendations are followed. de Ridder [11] discussed different precursors people need to follow a healthy diet. They determined that a person should know the details of a healthy diet, feel capable of abiding by it, and be motivated to follow it. Incidentally, related articles express the notion that systems recommending food and recipes lack the capability of igniting the needed motivation to encourage users to eat healthier food. Pecune et al. [16] found that the user's food choice depends on their eating goals. Thus, those not fully committed to eating healthy food may opt not to cook the suggested recipes. Trang Tran et al. [15] also discussed that although there is an abundance of papers proposing food recommendation systems, processes to change the eating behaviors of users are lacking. Their sentiment shows that there is an opportunity to explore the area further.

With food and recipe recommendation applications, recent techniques that have been researched for increasing user motivation regarding healthy food choices include rapport-building and nudging. Pecune et al. [17] applied a persuasive nudging conversational system to encourage people to consume healthier food. They utilized rapport-building to convince users to cook nutritious recipes. Rapport is implemented in their system by giving it social conversational abilities. To improve persuasion, they applied nudging by comparing healthy recipes with unhealthier ones. They found that the technique could convince users to follow the recipes. Similarly, Khan et al. [18] implemented smart nudges that display information to encourage users to eat healthier food. They proposed nutrition-aware strategies that can nudge user choices by placing nutritional guidelines with the recipes. They discovered that nudging could increase the user's cumulative first click rate to the healthier recipes.

Nudging involves presenting choices in a way that can change a person's behavior without decreasing their options [19]. It does not force a choice on users but instead lets them choose of their own volition. Therefore, nudges should be informative and motivational for users to select the suggested action or item [20]. Considering that price, convenience, and experience affect dietary habits, a motivational concept based on nudging that was envisioned is to allow users to find recipes that fit their nutritional needs by scanning the ingredients they have. It applies the concept of nudging concerning placement, where an item or choice can be more convenient to be selected. For instance, similar examples are positioning fruit at eye level [20] or placing healthy products near the cash register [17]. Only for the study, users were aimed to be motivated to eat food close to their recommended intake by suggesting recipes with ingredients they may own and also presenting nutritional information with it.

This study, therefore, involved the creation of a daily meal planner that considers the user's macronutrient intake and available ingredients. As implied, the AMDR conveys macronutrient consumption to avoid undernutrition or chronic disease risk. Hence, using the AMDR for analyzing nutrition and machine learning for ingredient scanning, appropriate recipe selections could be shown based on the user's recommended macronutrient intake and existing ingredients. Users could then create a meal plan using the recipe selections. These would be presented with helpful information, such as nutrition facts and preparation time, for convenience. The functionalities were implemented into a web application where people can find recipes to create a meal plan. The main aim of this study was the development of a daily meal planner web application named "Motiveat," which is based on the user's recommended macronutrient intake and available ingredients. For this, the following were the specific objectives.

1. Generate common Filipino recipes based on the user's attributes.
2. Develop a machine learning model that determines recipes based on ingredients.
3. Create and evaluate a daily meal planner web application by integrating the output from the previous objectives.

2. Methods

The main steps to perform the study are based on the stated objectives. First, it was planned how Filipino recipes would be generated based on the user's attributes. Second, a machine learning model was developed to determine recipes based on given ingredients. Finally, a meal planner web application that integrates these components was created.

2.1. Generate Filipino Recipes Based on User Attributes

2.1.1. Data Collection & Preparation

The websites panlasangpinoy.com and allrecipes.com were web scraped using Python to collect Filipino recipes. After this, the official website of the Food and Nutrition Research Institute-Department of Science and Technology (FNRI-DOST) in the Philippines was used to gather the values of the AMDR table. The CSV files containing the gathered data were considered "raw," meaning that the data may not adhere to a proper structure [21]. Therefore, a Python program was also developed to organize and merge the recipe information.

2.1.2. Generating Recipes Based on User Information

After web scraping the recipes and analyzing their nutritional content, it was discerned how these would be filtered based on the user's nutritional information. For this, the research study of Garcia [22] was followed, wherein the Total Daily Energy Expenditure (TDEE) will be computed using the Basal Metabolic Rate (BMR). The Mifflin-St Jeor Equation will be used to compute the BMR since it is identified to be more accurate [23]. Equation 1 shows the formula used for males, and Equation 2 shows the formula for females. The resulting BMR value would then be multiplied by the Katch-McArdle physical activity multipliers to be chosen by the user.

$$(10\ x\ weight[kg]) + (6.25\ x\ height[cm]) - (5\ x\ age[years]) + 5 \qquad \text{Equation 1}$$

$$(10\ x\ weight[kg]) + (6.25\ x\ height[cm]) - (5\ x\ age[years]) - 161 \qquad \text{Equation 2}$$

The obtained AMDR table by the FNRI-DOST will be used for the values of the macronutrient distribution. The nutritional content of the recipes per meal type should fit the computed values. For example, if the user is an adult, the AMDR values for each macronutrient can be 10-15% for proteins, 15-30% for fats, and 55-75% for carbohydrates. As long as the total is 100%, any percentage can be used for each macronutrient. There will be default values but users can edit them. Depending on how many meals the user selects, the important consideration is that the total calories and values under each macronutrient per day are appropriate to the computed values. This procedure of calculating and suggesting recipes was then placed in the web application code, as user input was required. The user attributes that users would input include their information, such as weight, height, age, and physical activity. Since there are other factors affecting nutrition, disclaimers were placed upon creating the application.

2.2. Developing a Machine Learning Model for Identifying Ingredients to Determine Recipes

2.2.1. *Data Collection*

From the collected recipes, ingredients were identified by creating a Python code that used the RegEx or Regular Expressions and Collections libraries. Words that are ingredients were extracted per recipe and separated by a comma. After which, condiments and non-food words, such as sizes or verbs, were removed. The result was exported as a CSV file. Web research was then performed to identify the ingredients' prices and commonalities.

Different pictures of each ingredient were gathered after determining the ingredients. First, at least 90 individual images of each ingredient were collected. Second, images that contained different ingredients were also gathered. Since there are uncommon ingredients local to the Philippines, ingredient items were also physically bought in markets and then photographed. Therefore, the images were obtained from Pexels, Unsplash, Google Images, Bing Images, and by taking photographs. Some ingredients that did not have a substantial amount of images or could not be obtained were removed from the ingredients list.

2.2.2. *Data Preprocessing*

Data preprocessing involved resizing, editing, and annotating the images for the dataset. Before training, the images must be adjusted because the graphics processing unit (GPU) and memory of the computer used for training could limit performance [24]. Thus, the collected images for each ingredient were resized to 800x800 pixels and then converted to JPG. As seen in Figure 1, images that only needed resizing were modified with Paint 3D, while images with a complex background were edited with Photoshop to extend the background of the images. This ensured that there would be varying backgrounds. In addition, to have images with different ingredients, Photoshop was also used to place a mix of various ingredients in one photo. Once all the needed images were resized and edited, LabelImg was used to annotate or manually add bounding boxes to the images of the ingredient being trained to be detected. Annotating involved dragging the cursor to create a box that contained an ingredient. The name of the ingredient was then tagged to the specific portion of the image. The annotations were saved as XML files in the Pascal VOC format.

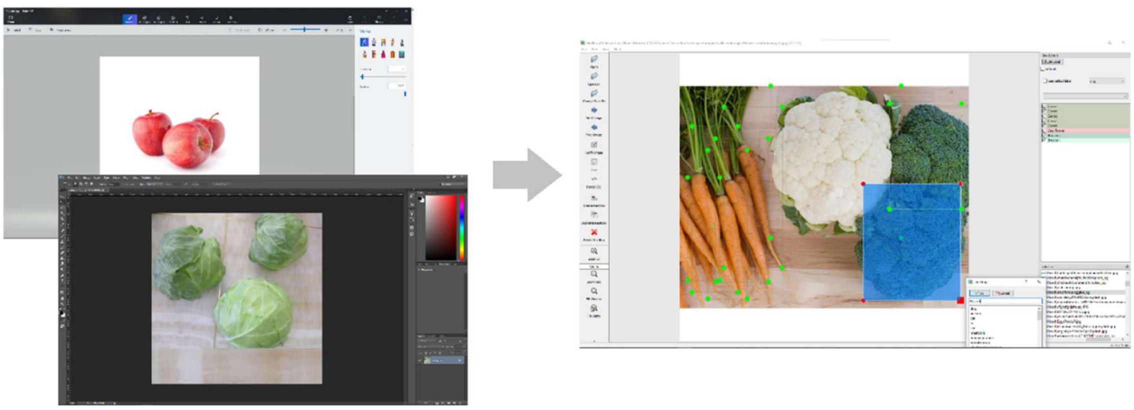

Figure 1 Editing and Annotating Images

2.2.3. Developing the Machine Learning Model

Deep learning is an advanced type of machine learning that can use neural networks to classify and predict data [25]. For object detection, MobileNet is a deep learning model that is quick, accurate, and small. Single Shot Multibox Detector (SSD) models that utilize MobileNet are lightweight and can be run in real-time [26]. MobileNet SSD was used since the meal planner is a web application that is intended to be lightweight. Development predominantly consisted of training and testing using the gathered images. For convenience, two different laptops were used. An attempt was also made to determine if the laptop with higher specifications could produce a better model. Table 1 shows the specifications of the laptops used.

Table 1 The Laptop Specifications

Specifications	HP Omen Laptop 15	ASUS ROG Strix G15
Processor	Intel i7-7700HQ CPU	AMD Ryzen 7 4800H
RAM	8GB	16GB
Graphics Card	NVIDIA GeForce GTX 1050	RTX 3050 Ti
Tensorflow Version	Tensorflow 2.9	Tensorflow 2.10

To develop the model, 80% of the total images and annotations of each ingredient were used for training, while 20% were used for testing.

- ***Training*** - The Python code provided by Nicholas Renotte [27] was used with Jupyter Notebook, where the parameters only needed to be updated and run per cell. Following the program, a "labelmap" file was generated to define the classes or objects to be detected. Inside the code, a cell can be run to start training. The parameters that were experimented with were the batch size and the number of steps. Errors such as "Out of Memory" were encountered while using the model training code. For the errors, the parameters had to be adjusted. However, even if the two laptops had different specifications, their GPU memory was the same and served as a limitation. Once the training parameters were fixed, training started. Different iterations of training were performed. The default settings were kept except for the batch size and the number of steps. In case an error was encountered, the values were edited. Table 2 shows the tested training specifications and the errors encountered, if any.

Table 2 The Training Specifications

Device	Training #	Pre-trained Model	Batch Size	Steps	Error(s) Encountered
HP Omen	Training 1	SSD MobileNet V2 FPNLite 640x640	64 -> 32 -> 16	50,000 (Default)	GPU ran out of memory -> None
	Training 2		16	80,000	None
	Training 3	SSD MobileNet V2 FPNLite 320x320	24	40,000	• Resource Exhausted: failed to allocate memory • OOM (Out of Memory)
	Training 4	SSD MobileNet V2 FPNLite 640x640	16	52,000	None
Asus ROG Strix	Training 1	SSD MobileNet V2 FPNLite 320x320	64 -> 32 -> 16	60,000	GPU ran out of memory -> None
	Training 2		16	20,000	None

Using both laptops, errors were prevented by using a batch size of only 16. Moreover, the pre-trained model that was used had to be "SSD MobileNet V2 FPNLite 320x320."

- ***Testing*** - The TensorBoard tool was used to test the model with 20% of the prepared images. Based on the testing results, training parameters were modified before trying to train the model again.

2.2.4. Using the Machine Learning Model

The trained model that yielded the highest performance was converted into Tensorflow.js. The resulting files were then uploaded to an IBM cloud object storage instance which could be used for a web application. Concerning how recipes would be determined, the steps are as follows. Firstly, the label or name of detected ingredients will be searched from the ingredients of each recipe in a list of recipes. Next, the recipes containing the ingredients will be returned. If there is no recipe applicable to the ingredients, the application will display "No applicable recipes found."

2.3. Creating and Evaluating a Daily Meal Planner Web Application

2.3.1. Prototyping

Prototyping could be used to experiment with a draft UI and evaluate design ideas [28]. For the study, prototyping was performed to create a blueprint for the application. Figma was used to create a prototype that represents the pages where users will be redirected when using the application. Using Figma, the web application's design, user interactions and page redirections were created.

Before finalization, the prototype was initially evaluated through a face-to-face semi-structured interview with nutritionist dietitians and a short online open-ended survey with 25 respondents. The respondents were recruited through convenience sampling. They were asked about their opinions on the created prototype. Thus, their responses helped improve it in terms of its design and functionalities.

2.3.2. Development and Integration

Development involved programming the application and integrating the components. It started with programming the application's front end by following the prototype. This involved coding the web pages, functionalities, components, design, and responsiveness. HTML, CSS, JavaScript, and different libraries were the primary tools. The code was placed in a GitHub repository. Following this, the recipe dataset was stored in a database. After the application's front end and database had been set up, the previously discussed components were integrated. First, the code for computing user nutritional information to generate recipe selections was added. The values that the user places in the created form interface will be used to compute the TDEE. Next, the trained machine learning model was integrated. It was done by adding the "Public URL" of the model uploaded in an IBM cloud storage instance. Code that could get the name of the ingredients scanned and then search it from the ingredients of the recipes was then placed. Different web hosting services were checked, but a free web hosting service was found for the web application. The selected web hosting service was used to launch the application and make it accessible using a web browser.

2.3.3. Testing the Application's Performance

The developed web application was used on a mobile phone to see how accurate the predictions are and what constraints there could be. The ingredient scanning component was tested to scan ingredients under different lighting and distance settings. In addition, the PageSpeed Insights web testing tool was also utilized. The performance metrics include the application's performance, accessibility, and others. These helped identify if the application had any severe issues.

2.3.4. Evaluating the Application in Terms of Usability and Motivation

The application was also evaluated in terms of usability and motivation. First, an interview was conducted with registered nutritionist dietitians to verify and ask about the application. Next, two surveys were performed to initially see the impression of possible users of the application. They were asked to try the application and then answer the surveys. A mixed-method approach was followed by using quantitative and qualitative questions. To evaluate usability, the System Usability Scale (SUS) was adapted from the paper of Brooke [29]. Next, to evaluate motivation, the Intrinsic Motivation Inventory (IMI) – "Activity Perception Questionnaire" was adapted from the Self Determination Theory website [30]. Open-ended questions were then added to the surveys to ask about the features of the deployed web application and the experience of the respondents.

3. Results and Discussion

3.1. Generating Filipino Recipes Based on User Attributes

3.1.1. Collected & Prepared Data

From the Panlasang Pinoy website, 367 recipes were obtained. Next, 127 recipes were gathered from Allrecipes. The collected data for each recipe were the recipe's name, URL, image URL, author, course, cuisine, preparation time, cook time, ingredients, and nutritional information. The FNRI-DOST website [31] was then used to obtain the values of the AMDR table. It contains the acceptable macronutrient distribution ranges based on age and primary macronutrients. The content of the table was later added to the web application code, where the computations would be made. Utilizing the Python program for preparing data, the two CSV files were formatted and merged into one file. In total, 272 recipes were gathered after removing duplicates and recipes that lacked nutritional information.

3.1.2. Generating Recipes Based on User Nutritional Information

The discussed formulas in the methodology were eventually placed in the web application code. The computed value for the TDEE is how many total calories should be consumed by a person to maintain their weight [32]. On the other hand, the computed values using the AMDR are how many calories each macronutrient should be eaten per day [33]. A database query that selects recipes that have less than the computed amounts was written. The important consideration was that the total values of the selected meals for all types would not significantly be lesser or greater than the computed values.

3.2. The Developed Machine Learning Model for Identifying Ingredients to Determine Recipes

3.2.1. Collected Data

192 ingredients were initially extracted. Web research was performed to find common ingredients. Finally, 34 main ingredients were selected. Based on the selected ingredients, individual and mixed images of each ingredient were gathered. These were manually downloaded and ingredients were also bought and photographed. A total of 3,732 images were obtained. Table 3 contains the complete list of ingredients and the total images collected for each.

Table 3 The Number of Images for Each Identified Ingredient

Ingredient	Images	Ingredient	Images	Ingredient	Images
Ampalaya	97	Eggplant	99	Potato	95
Apple	96	Fish	105	QuailEgg	98
Banana	159	GreenPapaya	93	Sayote	98
BananaBlossom	94	Kalabasa	101	ShrimpOrPrawn	107
Beef	125	Kamote	98	Sigarilyas	95
Broccoli	72	Kangkong	96	Singkamas	82
Cabbage	96	Labanos	99	Sitaw	103
Carrot	97	Lemon	106	Squid	92
Cauliflower	101	Malunggay	92	Tahong	107
Chicken	154	Okra	86	Tomato	107
Crab	96	Pechay	119	Mixed	195
Egg	122	Pork	150	**Total**	3,732

3.2.2. Preprocessed Data

Preprocessing the images involved editing and annotating them. These annotations were saved as XML files. In total, 7464 files resulted from this process. These were then split into training and testing folders.

3.2.3. Developed Machine Learning Model

The process of testing the models was performed after each training iteration. For this, the TensorBoard tool was mainly used. Using the HP Omen 15 laptop, only the loss and learning rate graphs were obtained for the first two training iterations. An error occurred when testing the trained models because some images were not in the proper format. After noticing the error, the images were fixed. For the third training iteration, no model was tested because training did not complete due to other errors. At the fourth training iteration, the issues were fixed. Therefore, the mean Average Precision (mAP) and Average Recall (AR) were also acquired. Figure 2 shows the different stages containing the loss and learning rate graphs when testing the produced models using the HP Omen 15 laptop. The x-axis in each graph is the number of steps, while the y-axis represents the loss/learning rate.

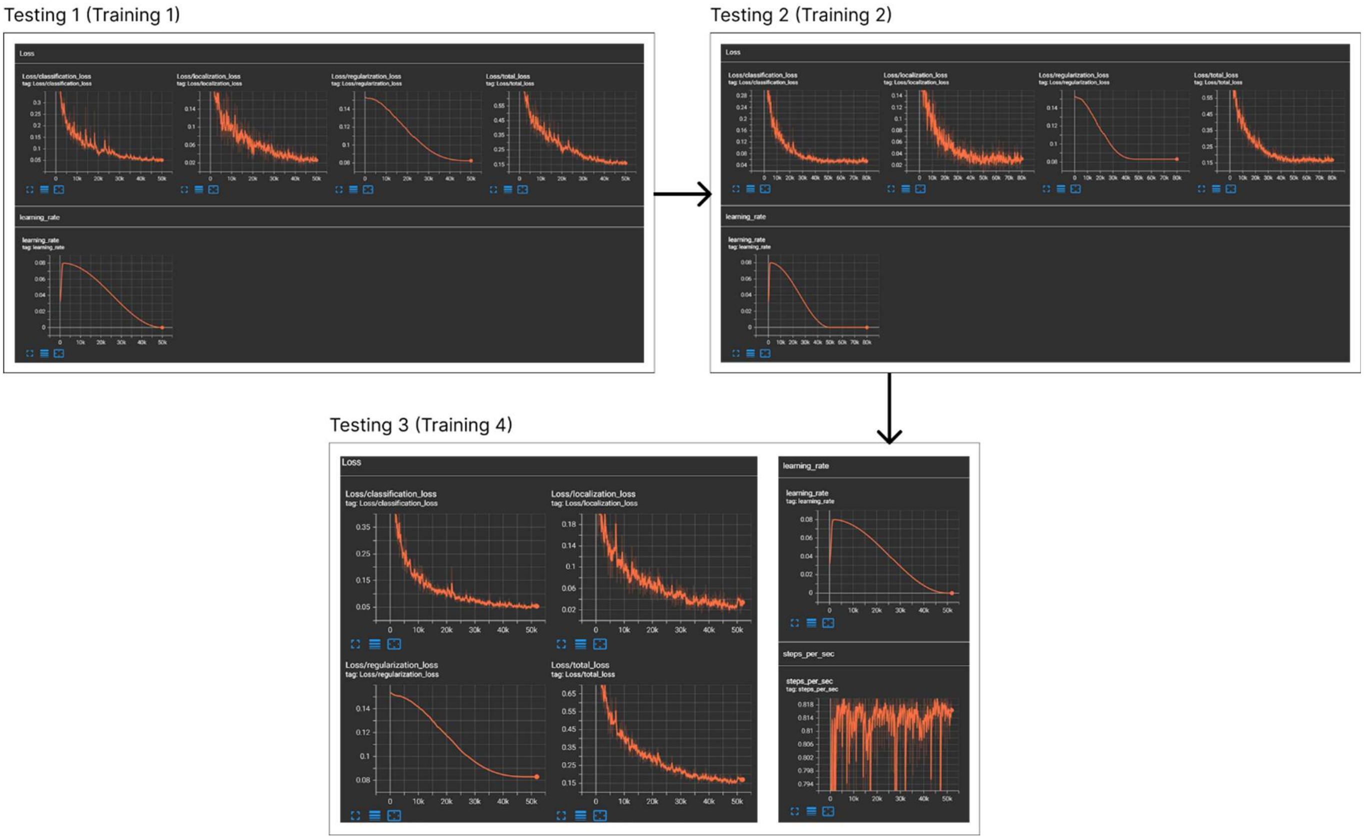

Figure 2 The Results of Testing The Models Trained

As depicted by most of the graphs in Figure 2, the different types of loss decreased significantly around step 45,000. On the other hand, the learning rate reached 0 around step 50,000. Thus, the optimal value for the number of steps is around 45,000 to 50,000. From these numbers, using a lower value would be inefficient, and using a higher value could overtrain the model.

Next, Figure 3 shows the loss and learning rate graphs when testing the models developed using the ASUS ROG Strix G15 laptop. The graphs from "Testing 1 (Training 1)" show that the optimal value for the number of steps is also around 45,000 to 50,000. On the other hand, from the graphs in "Testing 2 (Training 2)", it can be noticed that more training steps could still be added since the values were not yet at their lowest.

Figure 3 The Results of Testing The Models Trained Using The ROG Strix G15 Laptop

Table 4 shows the testing results of all the trained models. As stated, when training the fourth trained model, the type of images was fixed. Thus, the mAP and AR values were generated.

Table 4 The Testing Results

Device	Training #	Testing #	mean Average Precision	Average Recall	Error(s) Encountered
HP Omen	Training 1	Testing 1	N/A	N/A	Image Type
	Training 2	Testing 2	N/A	N/A	
	Training 3	N/A	N/A	N/A	OOM, No Model trained
	Training 4	Testing 3	0.526	0.539	None
Asus ROG Strix	Training 1	Testing 1	0.438	0.603	None
	Training 2	Testing 2	0.123	0.147	None

For the fourth trained model, the mAP is 0.526 and the AR is 0.539. The results indicate that the trained model is relatively balanced and may correctly detect ingredients. Next, with the first model trained using the ASUS ROG Strix G15 laptop, the mAP is 0.438 and the AR is 0.603. This indicates that the model may detect more ingredients but with more false positives. Therefore, the model would have fewer correct predictions overall. Although the training was stopped for the second model trained using the second laptop, it was still tested. Its mAP was 0.123 and AR was 0.147. Since these values are significantly low, the model would not be able to predict ingredients properly, if at all.

3.2.4. Using the Machine Learning Model

The trained models were compared based on the values resulting from being tested. The model that yielded the better performance was the fourth trained model using the HP Omen 15 laptop. Therefore, it was selected as the model to be used for the web application. The HP Omen 15 - Training 4 model was then converted to Tensorflow.js. After conversion, the files were uploaded to an IBM cloud storage. It was then configured to be accessible using a "Public URL." The IBM cloud storage "Public URL" was used in the application code, where the steps for identifying ingredients and recipes were added. It was also set that the minimum accuracy for ingredients to be detected was 90% but was eventually changed to 80% for more ingredient detections.

3.3. The Created and Evaluated Daily Meal Planner Web Application

3.3.1. The Designed Prototype

A clickable prototype of the meal planner web application was created using Figma. Figure 4 shows the created prototype for the mobile version of the application. The main pages designed are the "Home" page, "Generate" page, "Recipes" page, and "About" page. For each page, subpages were created. A desktop version of the prototype, as seen in Figure 5, was then created. The desktop version shows how the pages and subpages would look on a bigger screen.

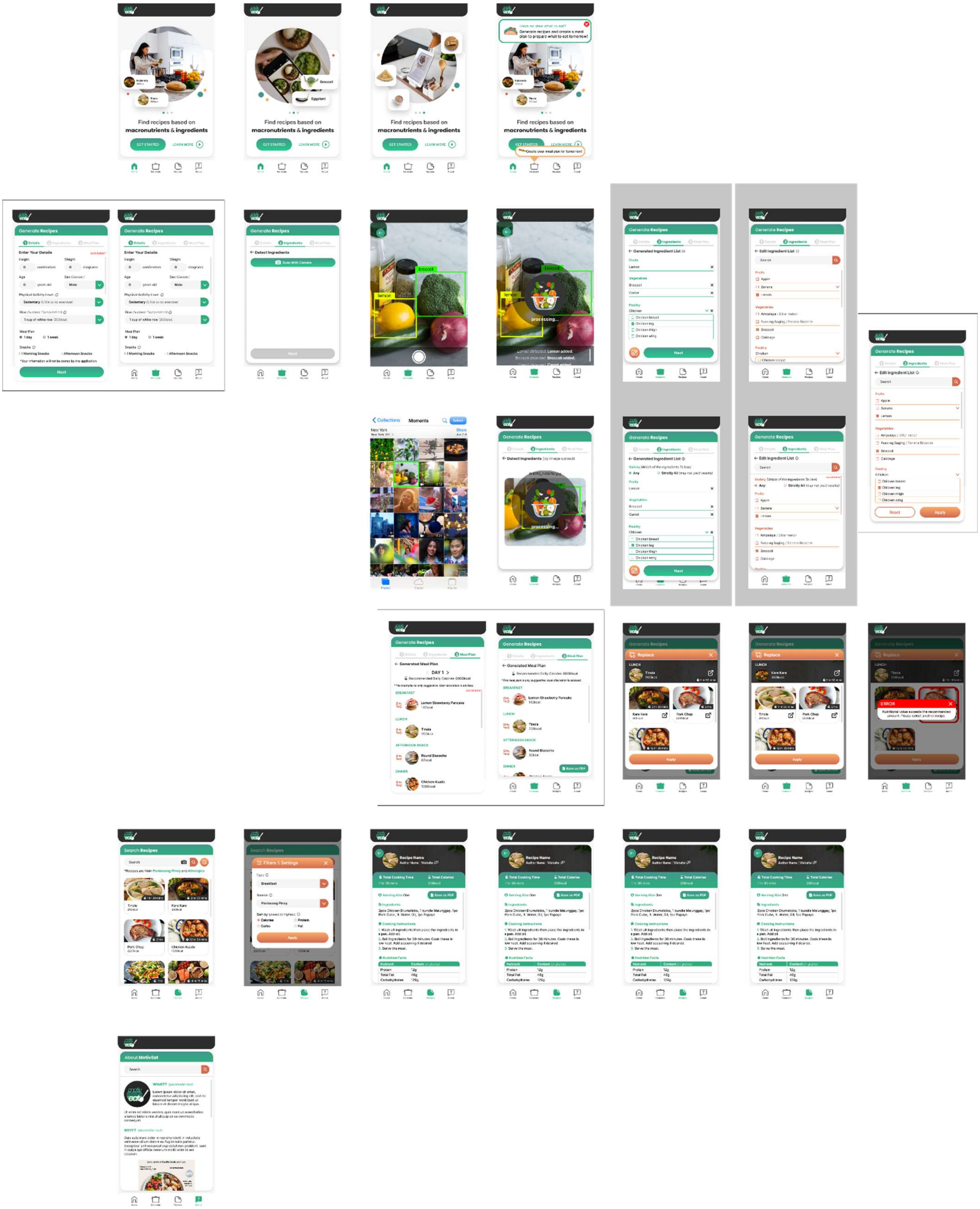

Figure 4 The Mobile Prototype of The Meal Planner

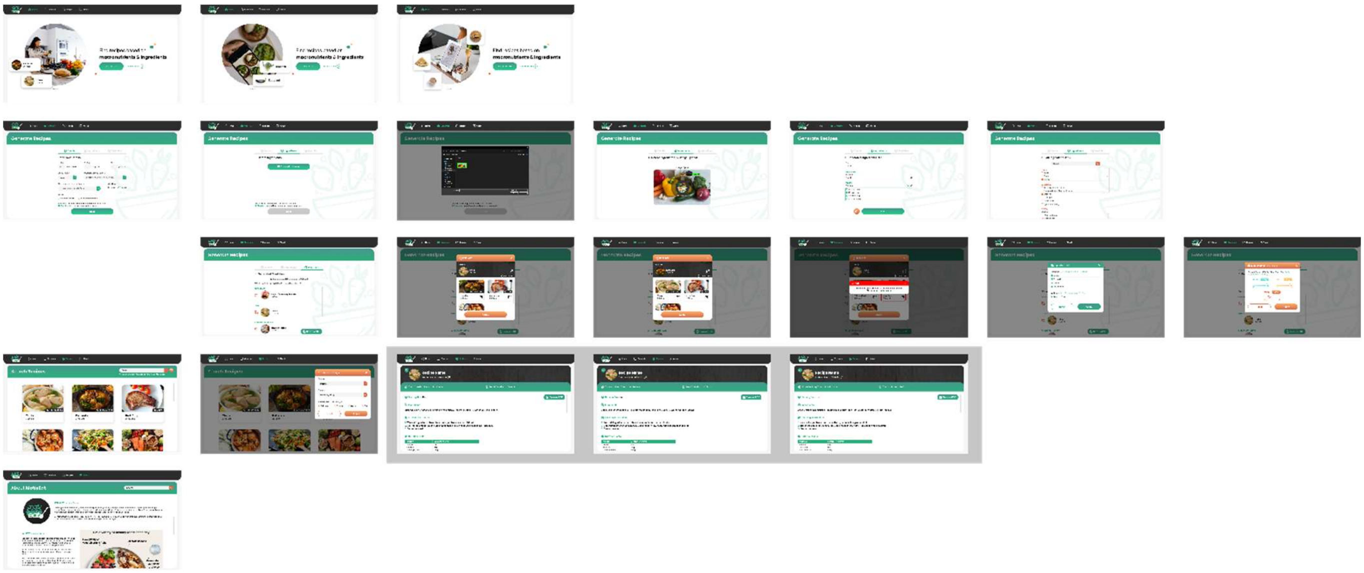

Figure 5 The Desktop Prototype of The Meal Planner

The responses to the interview and survey were used to improve the prototype. Not all the suggestions were followed since these could be inapplicable or beyond the scope of the research. Also, as stated by Vu et al. [34], end users should not be enabled to manage the design process since their requests could cause a less optimal design. Therefore, only some suggestions were followed. Based on the interview with nutritionist dietitians, the main changes were to update the formulas, to change the rice values using a local source, and to let the users change the macronutrient distribution. From the survey responses, the applied changes include allowing the changing of the color theme, avoiding wasting the top navigation bar space, adding the option to specify ingredients through a list, increasing the font size, lessening rounded borders, and adding more color.

3.3.2. Development and Integration

The prototype was followed to start developing the application. For the specific functionalities of the application, the Mifflin-St Jeor equation was placed in the interface with forms to get user information. The equation was used to get the needed calories per day to maintain weight. The computed calorie value was then used with the set AMDR values to compute the ideal values of protein, fat, and carbs. Therefore, the query to the database indicated that the recipes should be less than the computed maximum nutritional values. Next, for the machine learning model, the Public URL was placed in the “tf.loadGraphModel” code. The detected ingredients were then used to filter the recipes obtained from querying the database.

Once the web application named “MotivEat” was developed, it was deployed using Vercel. After this, the web application became accessible using a web browser. Figure 6 shows previews of the important steps when using the daily meal planner web application on a mobile device. It shows the initial user information form, the ingredient scanning interface, and a sample meal plan being created. During meal plan creation, warnings will be displayed if the user exceeds the computed nutritional values. Once the user is satisfied with their created meal plan, it can then be saved as a PDF.

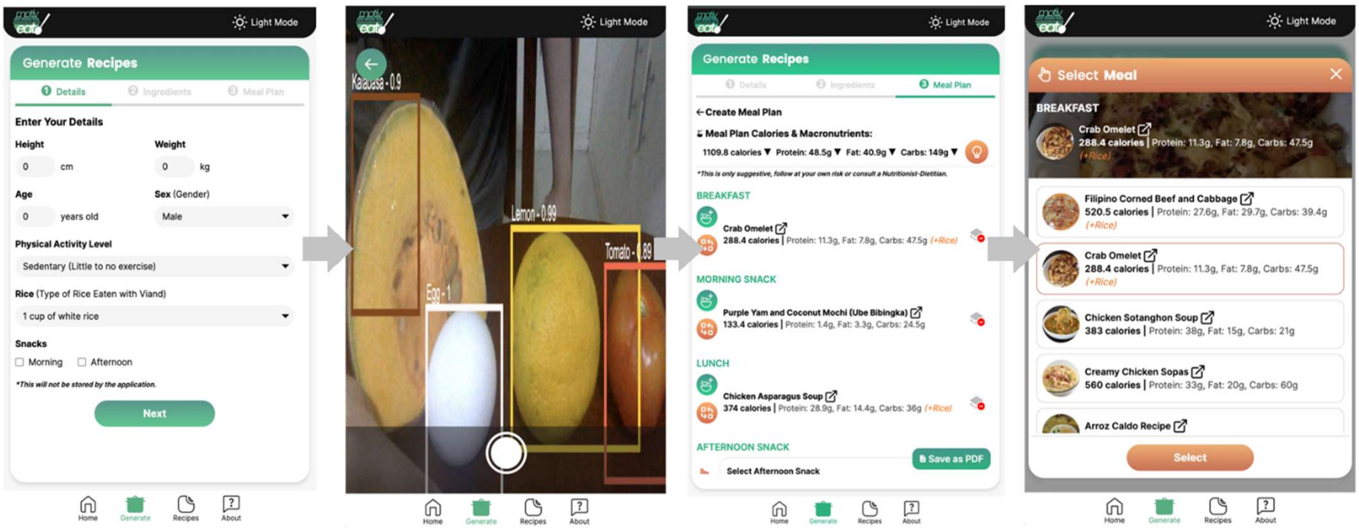

Figure 6 The Web Application's Interface

3.3.3. *Testing the Application's Performance*

The web application was initially tested using the Chrome web browser on a Samsung Galaxy S8+ Edge phone. As shown in Figure 7, the better the lighting in the room, the better the results. For the distance, it was preferable that the camera of the device being used be closer to the ingredients, wherein a 0.5-foot distance would be optimal.

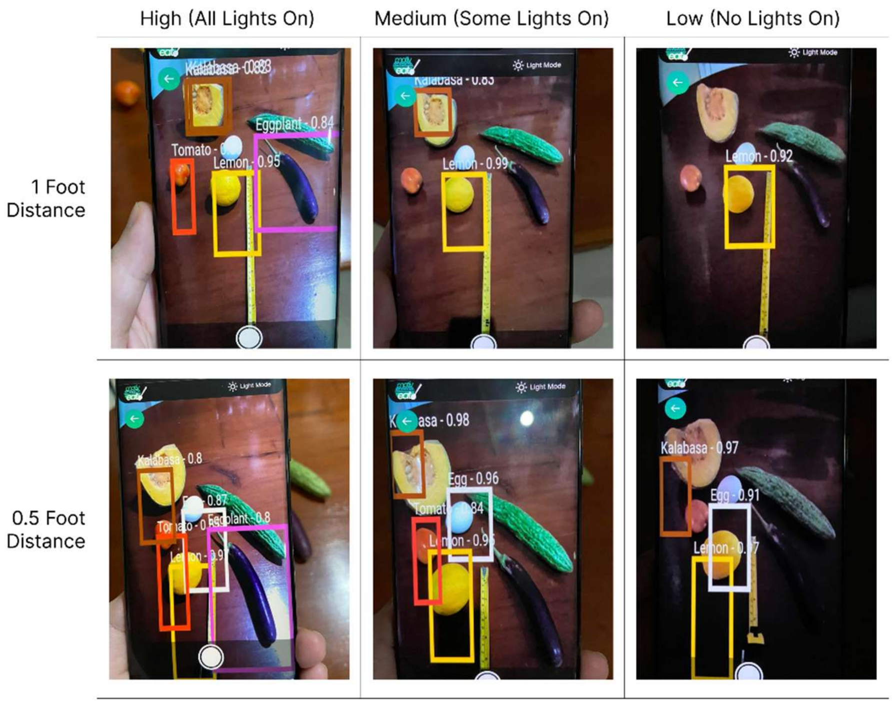

Figure 7 Manual validation of the machine learning object detection model

The application was then tested using the PageSpeed Insights web testing tool, where the performance, accessibility, best practices, and search engine optimization (SEO) for both mobile and desktop were evaluated. As seen on Table 5, most of the scores were acceptable except for performance. It was found that the reason for the low performance and other issues was the JavaScript code. Specifically, reducing unused JavaScript code could improve the performance of the web application. However, upon rechecking the code, none could be removed. The cause for the bad performance could be the machine learning model or other components that require an action to be rendered. It was thus discovered that the visual aspects are acceptable, but the performance could still be improved.

Table 5 The Web Application's PageSpeed Insights Scores

Device	Performance Metrics *(Total Score Out of 100)*			
	Performance	Accessibility	Best Practices	SEO
Mobile	42	94	75	75
Desktop	72	94	75	80

3.3.4. Evaluating the Application in Terms of Usability and Motivation

Based on the interview with nutritionist dietitians, the application was easy to follow. They also verified that by using the web application, users can at least check how much they go over or under each calorie or macronutrient. In relation to this, the nutritionist dietitians also stated that the nutrition information table and the warnings were relevant. Concerning the ingredient scanning component, the nutritionist dietitians noted that it worked and could detect ingredients, but other types of ingredients and their characteristics should be taken into account.

Next, Google Forms surveys containing the SUS, IMI, and open-ended questions were created. There were 21 respondents, including one nutritionist dietitian who had previously been interviewed. The computed average SUS score rounded off to no decimal places is 79. Therefore, the SUS score of the web application is 79 out of 100 which is acceptable since the average SUS score is around 68 [35]. The IMI subscale scores for each respondent were then computed. The scores were eventually averaged to compute the total score for each subscale. The result is a score out of 7. For reliability, Cronbach's Alpha (α) was computed. It was computed that $\alpha = 0.7$.

Table 6 shows the computed average scores for each IMI motivation subscale, rounded off to the nearest tenth. First, the score for interest/enjoyment is 6.2 out of 7. Next, the score for value/usefulness is 6.4 out of 7. Lastly, the score for perceived choice is 5.6 out of 7. All the scores are considerably high. This means that the respondents felt a high interest/enjoyment, value/usefulness, and perceived freedom of choice towards using the web application. The IMI statements were updated to relate to eating healthy. Therefore, the three subscales, overall, indicate that the web application was motivational for the respondents concerning healthy eating or the consideration of their nutritional intake.

Table 6 The Average Scores for Each IMI Motivation Subscale

	Motivation Subscales		
	Interest/Enjoyment	Value/Usefulness	Perceived Choice
Score	6.2	6.4	5.6

From the open-ended questions, most of the respondents answered that the web application could motivate them to find, explore, or follow recipes based on their recommended intake. As indicated by the responses, the reason is that the web application easily allows them to gain knowledge of their nutrition and also the food they can cook with their available ingredients. When respondents were asked about the helpful web application features that they liked, the main features they stated were ingredient scanning, displaying additional recipe information such as nutritional guidelines, meal planning and recipe generation, and others. Notably, they found ingredient scanning helpful because the ingredients can simply be scanned which is useful if they forget the name of an ingredient. For features they disliked, most of the respondents answered “None.” However, those who did place an answer stated that their disliked features were the inconsistency of the camera feature, the forms with inputs which could reset in some instances, the possible complication of adding ingredients manually if not scanned by their camera, and the “Macronutrient Distribution” selection which could be complex for some.

Respondents were also asked about their experience and for any additional comments. The responses were a summary of how the respondents felt about the application. Most of the respondents found the application interesting in a positive way. Some of the respondents also stated that they had fun and were motivated to start having healthier eating habits. However, they encountered technical issues, especially concerning the ingredient scanning feature. They also suggested adding other filters and types for the recipes and including more non-Filipino food in the recipe selections.

4. Conclusion and Recommendations

In this study, a daily meal planner web application, named “MotivEat”, that generates recipe selections based on the user’s recommended macronutrient intake and available ingredients was developed. For this, three objectives were achieved.

First, common Filipino recipes were generated based on the user’s attributes by analyzing the nutritional content of a recipe dataset and relating it to the user’s ideal calorie and macronutrient intake. The dataset was constructed from web scraping Filipino recipes. Related user nutritional information was calculated using identified formulas.

Second, a machine learning model that can determine recipes based on available ingredients was developed. TensorFlow was used to train a machine learning model that can detect ingredients and find recipes that can be cooked using them. As a result, the trained model with the highest performance had a mAP of 0.526 and an AR of 0.539. The performance could be improved if a computer with higher specifications is used for training. Nevertheless, the model was able to detect ingredients even when set at 90% accuracy and above in the application.

Third and lastly, the daily meal planner web application was created by designing a prototype and following it to develop the web application. The web application was then hosted online before being finally evaluated. Through evaluation, it was discovered that the model would perform better if the lighting is good and the ingredients are closer to the camera. It was also found, through web testing, that the performance of the web application concerning its JavaScript components was unsatisfactory, but the other aspects were adequate. From surveys, respondents had positive impressions of the application. An SUS score of 79 and IMI scores above 5 were obtained. However, although these scores are acceptable, the features of the application should be improved to be further accepted by more users.

The findings of the study show promising results. Still, there are improvements that could be made. For the machine learning model, a computer with better specifications can be used to train a model with better performance. On the other hand, for the web application, the code to use the machine learning model could be improved to make it faster. The code and other JavaScript components of the application can also be refactored for improvement. Once these enhancements are implemented, it could then be evaluated if the application could have a long-term effect to consistently keep motivating users to eat healthier food for an extended period of time.

Bibliography

[1] S. Abhari *et al.*, “A Systematic Review of Nutrition Recommendation Systems: With Focus on Technical Aspects,” *J Biomed Phys Eng*, vol. 9, no. 6, Art. no. 6, 2019, doi: 10.31661/JBPE.V0I0.1248.

[2] H. Cena and P. C. Calder, “Defining a Healthy Diet: Evidence for the Role of Contemporary Dietary Patterns in Health and Disease,” *Nutrients*, vol. 12, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/nu12020334.

[3] J. A. Coleman and N. Carrillo, “Pandemic Food Delivery: Plummet or Prosper?,” *UFJUR*, vol. 23, Oct. 2021, doi: 10.32473/ufjur.v23i.128732.

[4] J. P. Guevarra, Y. M. H. Zuñiga, D. R. Uezono, J. J. L. Go, C. N. Granada, and D. T. Manese, “Developing an interpersonal communication material for primary health-care workers for the prevention and control of noncommunicable diseases: experience from a case in Manila, Philippines,” *JHR*, vol. 36, no. 4, Art. no. 4, Jul. 2022, doi: 10.1108/JHR-07-2020-0283.

[5] I. B. Idris, N. A. Azit, S. R. A. Ghani, S. F. S. Nor, and A. M. Nawi, “A systematic review on noncommunicable diseases among working women,” *Ind Health*, vol. 59, no. 3, Art. no. 3, Feb. 2021, doi: 10.2486/indhealth.2020-0204.

[6] R. Y. Toledo, A. A. Alzahrani, and L. Martinez, “A Food Recommender System Considering Nutritional Information and User Preferences,” *IEEE Access*, vol. 7, pp. 96695–96711, 2019, doi: 10.1109/ACCESS.2019.2929413.

[7] E. Reeve *et al.*, “Implementation lessons for school food policies and marketing restrictions in the Philippines: a qualitative policy analysis,” *Global Health*, vol. 14, no. 1, Art. no. 1, Dec. 2018, doi: 10.1186/s12992-017-0320-y.

[8] J. R. Bagos and M. D. Oruga, “Awareness of the Food-Based Diet Guidelines, Eating Practices, and Preferred Sources of Nutrition Information of Employees in an Open University in the Philippines,” *J Kep Padjadjaran*, vol. 9, no. 2, Art. no. 2, Aug. 2021, doi: 10.24198/jkp.v9i2.1654.

[9] R. G. Abilgos-Ramos and J. F. Ballesteros, “Food consumption, diet quality, and diversity of rice-based farm households in Central Luzon, Philippines,” *Rice-Based Biosyst J*, no. 4, Art. no. 4, Jun. 2018.

[10] Y. Ngqangashe, S. Goldman, A. Schram, and S. Friel, “A narrative review of regulatory governance factors that shape food and nutrition policies,” *Nutrition Reviews*, vol. 80, no. 2, Art. no. 2, Jan. 2022, doi: 10.1093/nutrit/nuab023.

[11] D. de Ridder, F. Kroese, C. Evers, M. Adriaanse, and M. Gillebaart, “Healthy diet: Health impact, prevalence, correlates, and interventions,” *Psychology & Health*, vol. 32, no. 8, Art. no. 8, Aug. 2017, doi: 10.1080/08870446.2017.1316849.

[12] M. M. Lopez-Madrid, C. C. S. Acuin, C. L. Orense, C. A. Duante, R. C. A. Tan, and M. V. Capanzana, “Awareness of and Adherence to the Food Based Dietary Guidelines Among Household Meal Planners in the Philippines,” vol. 147, no. 3, Art. no. 3, 2018.

[13] W. Min, S. Jiang, L. Liu, Y. Rui, and R. Jain, “A Survey on Food Computing,” Jul. 16, 2019, *arXiv*: arXiv:1808.07202. Accessed: Jun. 11, 2023. [Online]. Available: http://arxiv.org/abs/1808.07202

[14] J. Marshall, P. Jimenez-Pazmino, R. Metoyer, and N. V. Chawla, “A Survey on Healthy Food Decision Influences Through Technological Innovations,” *ACM Trans. Comput. Healthcare*, vol. 3, no. 2, Art. no. 2, Apr. 2022, doi: 10.1145/3494580.

[15] T. N. Trang Tran, M. Atas, A. Felfernig, and M. Stettinger, “An overview of recommender systems in the healthy food domain,” *J Intell Inf Syst*, vol. 50, no. 3, Art. no. 3, Jun. 2018, doi: 10.1007/s10844-017-0469-0.

[16] F. Pecune, L. Callebert, and S. Marsella, “A Recommender System for Healthy and Personalized Recipe Recommendations,” in *HealthRecSys@ RecSy*, Sep. 2020, pp. 15–20.

[17] F. Pecune, L. Callebert, and S. Marsella, “Designing Persuasive Food Conversational Recommender Systems With Nudging and Socially-Aware Conversational Strategies,” *Front. Robot. AI*, vol. 8, p. 733835, Jan. 2022, doi: 10.3389/frobt.2021.733835.

[18] M. A. Khan, K. Muhammad, B. Smyth, and D. Coyle, “Investigating Health-Aware Smart-Nudging with Machine Learning to Help People Pursue Healthier Eating-Habits,” Oct. 05, 2021, *arXiv*: arXiv:2110.07045. Accessed: Jun. 11, 2023. [Online]. Available: http://arxiv.org/abs/2110.07045

[19] C. Henkel, A.-R. Seidler, J. Kranz, and M. Fiedler, “How to Nudge Pro-Environmental Behaviour: An Experimental Study,” in *Twenty-Seventh European Conference on Information Systems (ECIS2019)*, Jun. 2019.

[20] R. Karlsen and A. Andersen, “Recommendations with a Nudge,” *Technologies*, vol. 7, no. 2, Art. no. 2, Jun. 2019, doi: 10.3390/technologies7020045.

[21] M. Hameed and F. Naumann, “Data Preparation: A Survey of Commercial Tools,” *SIGMOD Rec.*, vol. 49, no. 3, Art. no. 3, Dec. 2020, doi: 10.1145/3444831.3444835.

[22] M. B. Garcia, “Plan-Cook-Eat: A Meal Planner App with Optimal Macronutrient Distribution of Calories Based on Personal Total Daily Energy Expenditure,” in *2019 IEEE 11th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, Laoag, Philippines: IEEE, Nov. 2019, pp. 1–5. doi: 10.1109/HNICEM48295.2019.9073490.

[23] A. Sampath Kumar, G. Arun Maiya, B. A. Shastry, K. Vaishali, S. Maiya, and S. Umakanth, “Correlation between basal metabolic rate, visceral fat and insulin resistance among type 2 diabetes mellitus with peripheral neuropathy,” *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, vol. 13, no. 1, pp. 344–348, Jan. 2019, doi: 10.1016/j.dsx.2018.10.005.

[24] I. Cesar, V. Solina, R. Kramberger, and T. Kramberger, “Enhancing the performance of image preprocessing for classification and object detection,” *Polytechnic and Design*, vol. 8, no. 1, Art. no. 1, 2020, doi: 10.19279/TVZ.PD.2020-8-1-12.

[25] J. Latif, C. Xiao, A. Imran, and S. Tu, “Medical Imaging using Machine Learning and Deep Learning Algorithms: A Review,” in *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, Sukkur, Pakistan: IEEE, Jan. 2019, pp. 1–5. doi: 10.1109/ICOMET.2019.8673502.

[26] R. L. Galvez, A. A. Bandala, E. P. Dadios, R. R. P. Vicerra, and J. M. Z. Maningo, “Object Detection Using Convolutional Neural Networks,” in *TENCON 2018 - 2018 IEEE Region 10 Conference*, Jeju, Korea (South): IEEE, Oct. 2018, pp. 2023–2027. doi: 10.1109/TENCON.2018.8650517.

[27] Nicholas Renotte, *Tensorflow Object Detection in 5 Hours with Python | Full Course with 3 Projects*, (Apr. 09, 2021). Accessed: Jun. 11, 2023. [Online Video]. Available: https://www.youtube.com/watch?v=yqkISICHH-U

[28] K. Moran, C. Bernal-Cardenas, M. Curcio, R. Bonett, and D. Poshyvanyk, “Machine Learning-Based Prototyping of Graphical User Interfaces for Mobile Apps,” *IIEEE Trans. Software Eng.*, vol. 46, no. 2, Art. no. 2, Jun. 2018, doi: 10.1109/TSE.2018.2844788.

[29] J. Brooke, “SUS - A quick and dirty usability scale,” *Usability evaluation in industry*, vol. 189, no. 3, 1996.

[30] “Intrinsic Motivation Inventory (IMI) – selfdeterminationtheory.org.” Accessed: Apr. 12, 2023. [Online]. Available: https://selfdeterminationtheory.org/intrinsic-motivation-inventory/

[31] FNRI-DOST, “Philippine Dietary Reference Intakes (PDRI),” Food and Nutrition Research Institute Department of Science and Technology. Accessed: May 06, 2022. [Online]. Available: https://www.fnri.dost.gov.ph/index.php/tools-and-standard/philippine-dietary-reference-intakes-pdri

[32] L. Kosmiski, “Energy expenditure in HIV infection,” *The American Journal of Clinical Nutrition*, vol. 94, no. 6, Art. no. 6, Dec. 2011, doi: 10.3945/ajcn.111.012625.

[33] J. Kubala, “How to Count Macros: A Step-By-Step Guide,” healthline. Accessed: May 08, 2022. [Online]. Available: https://www.healthline.com/nutrition/how-to-count-macros

[34] K.-P. L. Vu, R. W. Proctor, and Y.-H. Hung, “Website design and evaluation,” *Handbook of human factors and ergonomics*, pp. 1323–1352, Aug. 2021.

[35] J. Brooke, “SUS: A Retrospective,” *Journal of Usability Studies*, vol. 8, no. 2, pp. 29–40, Feb. 2013.