JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Food Recommendation System

[1]Ainsley Cardozo, [2]Mahesh Poojari, [3]Jobin Joseph, [4]Omprakash Yadav

[1]Student, [2]Student, [3]Student, [4]Supervisor
[1]Computer Engineering,
[1]Affiliation (Computer Engineering, Xavier Institute of engineering, Mumbai University)

Abstract: Premature heart disease, diabetes, and cancer are examples of non-communicable diseases that are becoming a severe problem. Unhealthy eating habits have been linked to a variety of ailments. In this paper, we offer a recommender system that makes nutritionally appropriate meal recommendations based on nutritional knowledge. Physical traits, physiological data, and other personal information in recent years, a number of studies have explored computer models for personalised meal recommendations based on nutritional information and user data. This article puts forth a general framework for daily food plan options. To produce recommendations for the essential food items, this system applied machine learning methods and methodologies. The K-means clustering and Random Forest classification methods are employed in this system. This system focuses on providing meal suggestions that assist the user in preserving and improving his or her health.

Index Terms **- Unhealthy diets, meal, nutritional, data, users, Recommendation System, K-means clustering, Random Forest.**

1. INTRODUCTION

Nutrition is the process of obtaining the food and nutrients needed for good health and growth. We become weak, ill, and, in the worst-case scenario, die if we do not consume food. We miss developmental milestones and are unable to challenge our bodies on a daily basis with mental and physical challenges.

A well-balanced diet is essential for good health and nutrition. It protects against chronic noncommunicable diseases such as heart disease, diabetes, and cancer. They are necessary for the survival of humans, plants, animals, and all other living things. Nutrition recommendation systems (NRS) are one of the most common technologies used in the field of nutrition information. They are being studied as a potential tool for assisting users in improving their eating habits and achieving their goal of making healthier food choices. A recommender system can predict users' preferences for unrated goods and then recommend new items to them.

2. RESEARCH METHODOLOGY

2.1. PROCESS

Food recommender systems are gaining popularity due to their importance in maintaining a healthy lifestyle. Most existing studies in the food domain focus on recommendations that suggest appropriate food items for individual users based on their preferences or health problems. In addition, recommendation features are extremely useful in the food industry. Such scenarios present numerous challenges for food recommender systems because the system must recommend food based on the height and weight of the users, which must be considered appropriately. These systems also include features for tracking nutritional consumption and persuading users to change their eating habits for the better. We present an overview of recommendation techniques for individuals and groups in the healthy food domain in this paper. Furthermore, we examine the current state-of-the-art in food recommender systems and discuss research challenges associated with the development of future food recommendation technologies.

2.2. COLLECTION OF DATA

The data has specific food related information which is related to the changes in the nutrients for the human being. The data used is observed from the lifestyle and daily records. The data collection process is responsible for the framework to do an inspection of the data collected and the data that can be compared to previously obtained data. This task further leads to the next task which is pre-processing which is applied to normalize the dataset using normalizing techniques. This task helps in dealing with data collecting and balancing to make a dataset that will be based on Machine learning.

2.3. DATA PREPARATION

In this study, the data preparation stage is data pre-processing that prepares raw data before the following process (clustering and memory-based processes) to obtain clean data. One way to data pre-processing is to reduce irrelevant attributes.

2.4. CLUSTERING PROCESS

This stage consists of two steps. The first step is to determine the number of clusters to get the optimal number of clusters using the Silhouette Coefficient. The second step is to group the user data using the k-means algorithm, where k is the number of clusters with the maximum Silhouette Coefficient value.

2.5. PRE-PROCESSING

In this task, the raw data obtained is being transformed into a type of data that can be given to algorithms of Machine learning. Training of data is done directly on the raw material which can lead to poor results. This task is very much necessary to fasten the training process. Pre-processing of the data helps in solving all the issues of the dataset.

3. IMPLEMENTATION

3.1. DATASET

Datasets were read using Pandas. Numpy was used to convert features into NumPy, which was then used to perform the subsequent operations.

We began by reading the CSV file with pandas and then converted it to a numpy array with NumPy.

Using the K-Means Algorithm, we divided the dataset into three clusters based on calories.

Then, for each row, we created weightlosscat, weightgaincat, and healthycat by selecting some columns from the nutritiondistribution.csv file and then adding respective bmicl and agecl as a separate column.

For the Weight Loss Function, we create an X test np array with zeros of length weightlosscat, including all weightlosscat columns, and append bmicl & agecl by multiplying it with ti, where ti=(clbmi+agecl)/2. We now build a Random Forest classifier with estimator=100.Then we use a classifier to train our model fit () which is used for Dumping the weightlosscat and Trained model for Weight Loss Function for recommendation.

Then, for the Weight Gain Function, we create an X test np array with zeros of length weightgaincat that includes all columns of weightgaincat and append bmicl & agecl by multiplying it with ti where ti=(clbmi+agecl)/2. We now build a Random Forest classifier with estimator=100. Then we use a classifier to train our model.fit () Dumping the weightgaincat and Trained model for Weight Gain Function Recommendation.

For Healthy Function, then we make an X test np array with zeros of length healthycat, including all healthycat columns, and append bmicl & agecl by multiplying it with ti, where ti=(clbmi+agecl)/2.

We now build a Random Forest classifier with estimator=100. Then we use a classifier to train our model fit () Dumping the healthycat and Trained model for Healthy Function recommendation

3.2. WORKING PRINCIPLE

When the "Get Recommendation" button is clicked, the frontend sends the contents of the form to "/bmi" and eventually calls predict () in app.py.

Flask is being used as the backend technology. The first step is to import all the dumped files, which are weightloss.joblib,weightlossprediction.joblib,weightgain.joblib,weightgainprediction.joblib, healthycat.joblib, and healthyprediction.joblib. Flask.request.form will fill in all the required data in the form for us. [‘NameOfHTMLElement’] The variable "function" will contain the words "weightloss," "weightgain," or "healthy." And the task is carried out based on the value of the function variable.

Considering function for "weightloss", We must again create an X test np array with zeros of length weightlosscat, including all weightlosscat columns, and appending bmicl & agecl by multiplying each element with ti, where ti=(clbmi+agecl)/2. Then, using "model loss.prdict(X test)," we will predict using the dumped trained model. As a result of the above predict method returning a list, our flask will send the generated list to the frontend, where we can easily traverse the list in an html page and display the recommended food items.

Considering function for "weightgain", We must again create an X test np array with zeros of length weightgaincat, including all weightgaincat columns, and appending bmicl & agecl by multiplying each element with ti, where ti=(clbmi+agecl)/2. Then, using "model gain.prdict(X test)," we will predict using the dumped trained model. As a result of the above predict method returning a list, our flask will send the generated list to the frontend, where we can easily traverse the list in an html page and display the recommended food items.

Taking function for "healthy" into account We must again create an X test np array with zeros of length healthycat, including all healthycat columns, and appending bmicl & agecl by multiplying each element with ti, where ti=(clbmi+agecl)/2. Then, using "model healthy.prdict(X test)," we will predict using the dumped trained model. As a result of the above predict method returning a list, our flask will send the generated list to the frontend, where we can easily traverse the list in an html page and display the recommended food items.

3.3. DETAILS OF HARDWARE AND SOFTWARE SPECIFICATIONS

HARDWARE DETAILS

To be used effectively, all computer software needs certain hardware components or other software resources to be present on a computer. These prerequisites are known as (computer) system requirements and are often used as a guideline as opposed to an absolute rule.

Processor: - Pentium IV 2.6GHZ or above
Ram: - 2GB or above
Hard disk Space: - 256GB

SOFTWARE DETAILS

Along with the hardware used in the system, it requires some software to make a system as to run a system with the computer hardware. For efficient and proper working of any system, a software must be installed.

Operating System: - Windows 7 or above
Languages used are: - Html, CSS, JavaScript, Bootstrap, Python.
Software used: Jupyter Notebook and Visual Studio Code.

3.4. MODULES

Flask: Flask is a micro web framework written in Python. It is classified as a microframework because it does not require tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

Joblib: Joblib is a set of tools to provide lightweight pipelining in Python. In particular: transparent disk-caching of functions and lazy re-evaluation (memorize pattern) easy simple parallel computing.

Pandas: Pandas is an open-source Python package that is most widely used for data science/data analysis 21 and machine learning tasks. It is built on top of another package named NumPy, which provides support for multi-dimensional arrays.

NumPy: NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

Random Forest: A random forest is a meta estimator that fits several decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

K-means: k-means clustering is a method of vector quantization, originally from signal processing, that aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster

4. IMPLEMENTATION METHODOLOGIES

4.1. IMPLEMENTED ALGORITHMS

4.1.1.1. K-MEANS

K-means clustering algorithm computes the centroids and iterates until we it finds optimal centroid. It assumes that the number of clusters are already known. It is also called flat clustering algorithm. The number of clusters identified from data by algorithm is represented by ‘K’ in K-means.

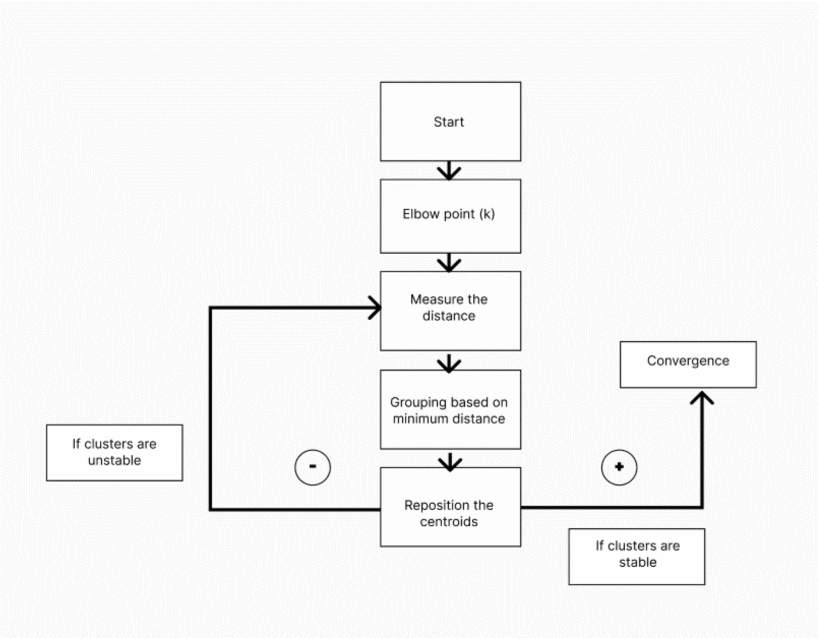
In this algorithm, the data points are assigned to a cluster in such a manner that the sum of the squared distance between the data points and centroid would be minimum. It is to be understood that less variation within the clusters will lead to more similar data points within same cluster.

K-means follows Expectation-Maximization approach to solve the problem. The Expectation-step is used for assigning the data points to the closest cluster and the Maximization-step is used for computing the centroid of each cluster.

While working with K-means algorithm we need to take care of the following things –

- While working with clustering algorithms including K-Means, it is recommended to standardize the data because such algorithms use distance-based measurement to determine the similarity between data points.
- Due to the iterative nature of K-Means and random initialization of centroids, K-Means may stick in a local optimum and may not converge to global optimum. That is why it is recommended to use different initializations of centroids.

Figure 1: Working of K-means

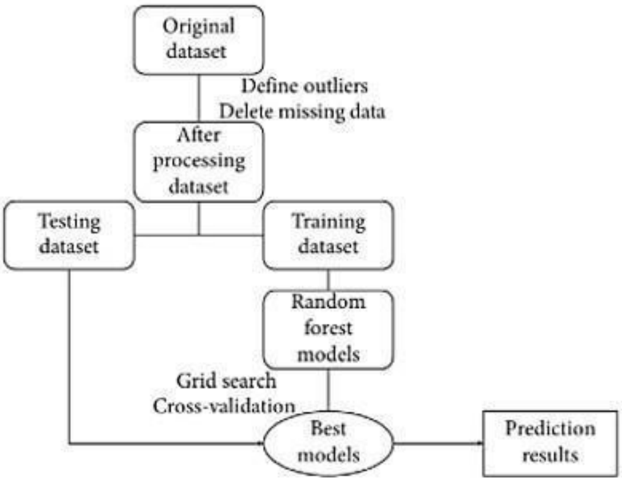


4.1.1.2. RANDOM FOREST

The concept of Random Forests was first developed by and is an Ensemble learning method. This classification technique of Machine Learning was constructed using Bootstrap aggregation (bagging technique) that made use of randomized features using subset sampling. This randomization resulted in the creation of multiple trees that is further capable to take accurate and precise decisions that could be combined in a single model. Each tree in a random forest learns from its superior random set of features. This subset of the trained data is randomly sampled with replacement. The classification of this test sample is developed using most votes by combining the outputs from all the models so observed. Some of the hyperparameters for constructing an RF are, the number of trees, and the max number of features for each split. The combined estimator is used in later stages to prevent the occurrence of overfitting henceforth enhancing the predictive accuracy of the model. Based on the implementation so executed, all the parameters are further optimized. After a set of instructions were carried out the following two hyperparameters were selected to be optimized:

Total number of trees in the forest and the number of forests needs to be considered when the test split needs to be performed. The graph below depicts the workflow of a conventional random forest:

Figure 2: Working of Random Forest



5. OUTPUT

Figure 3: Home Page

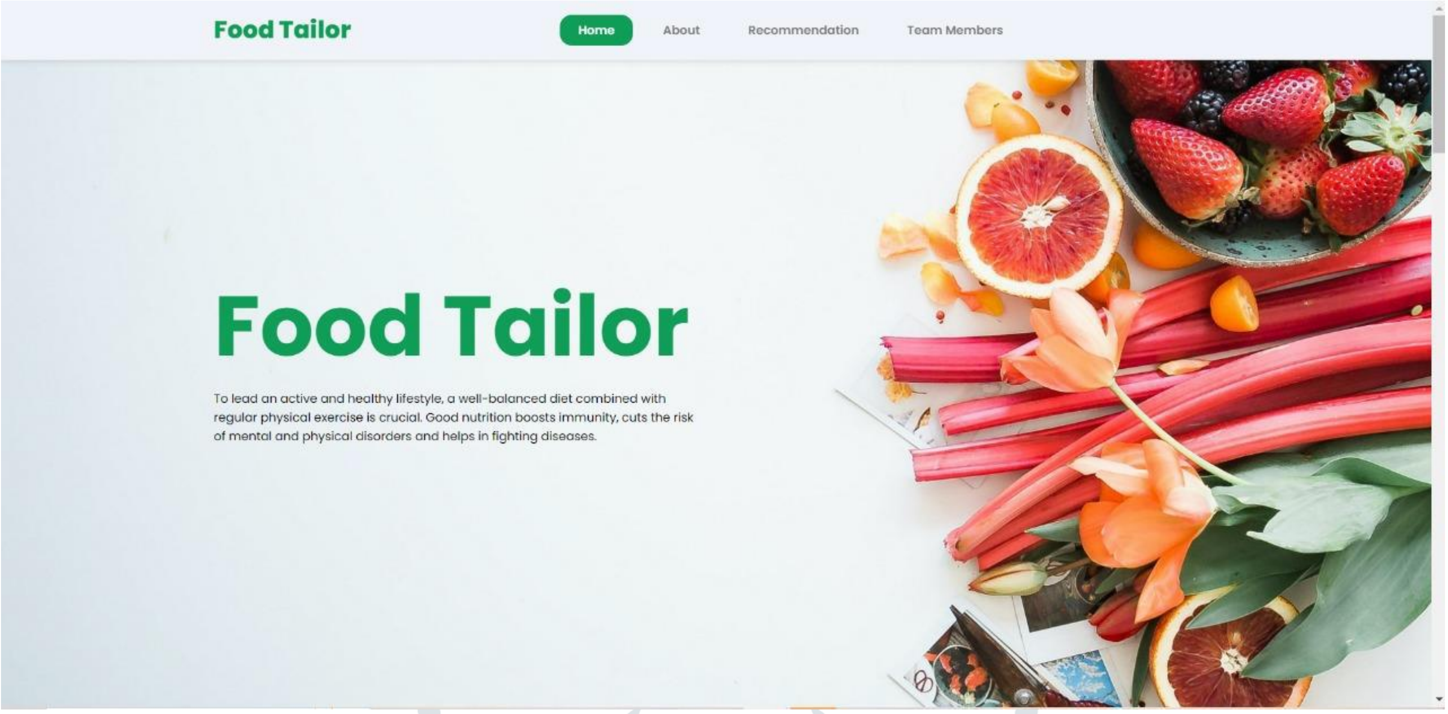


Figure 4: Bmi Calculation

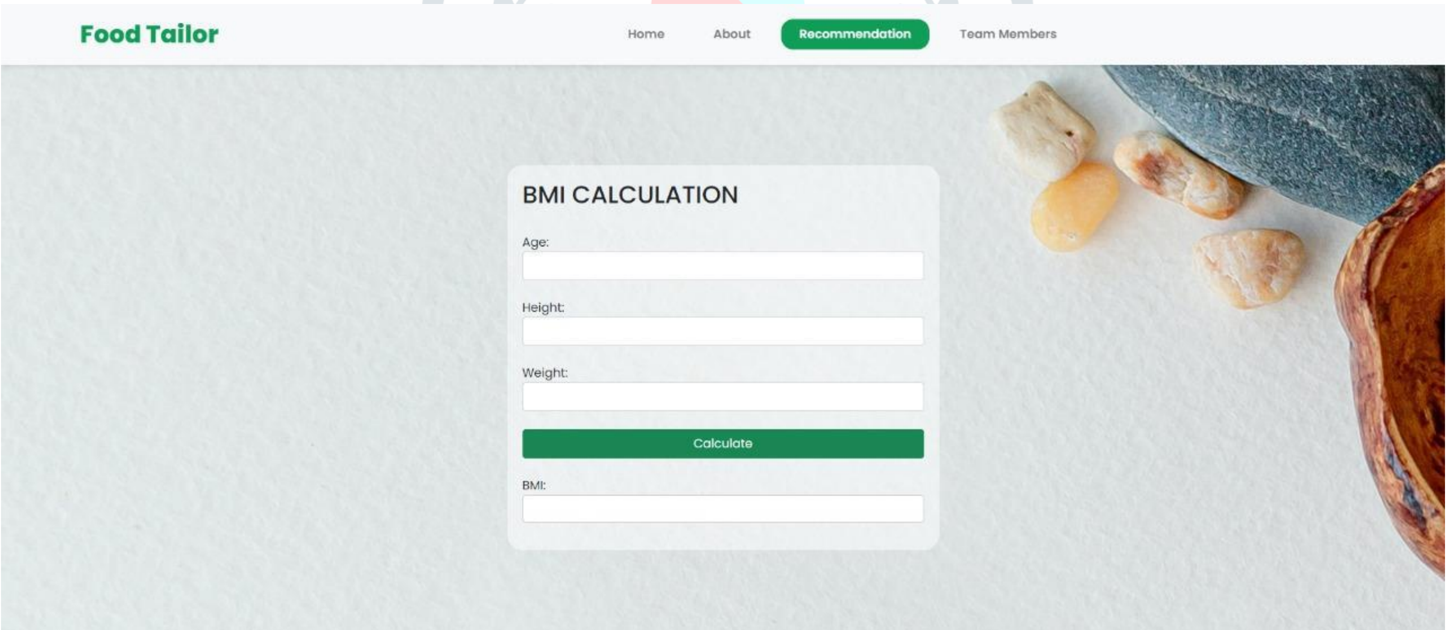


Figure 5: Bmi Calculation Healthy Result

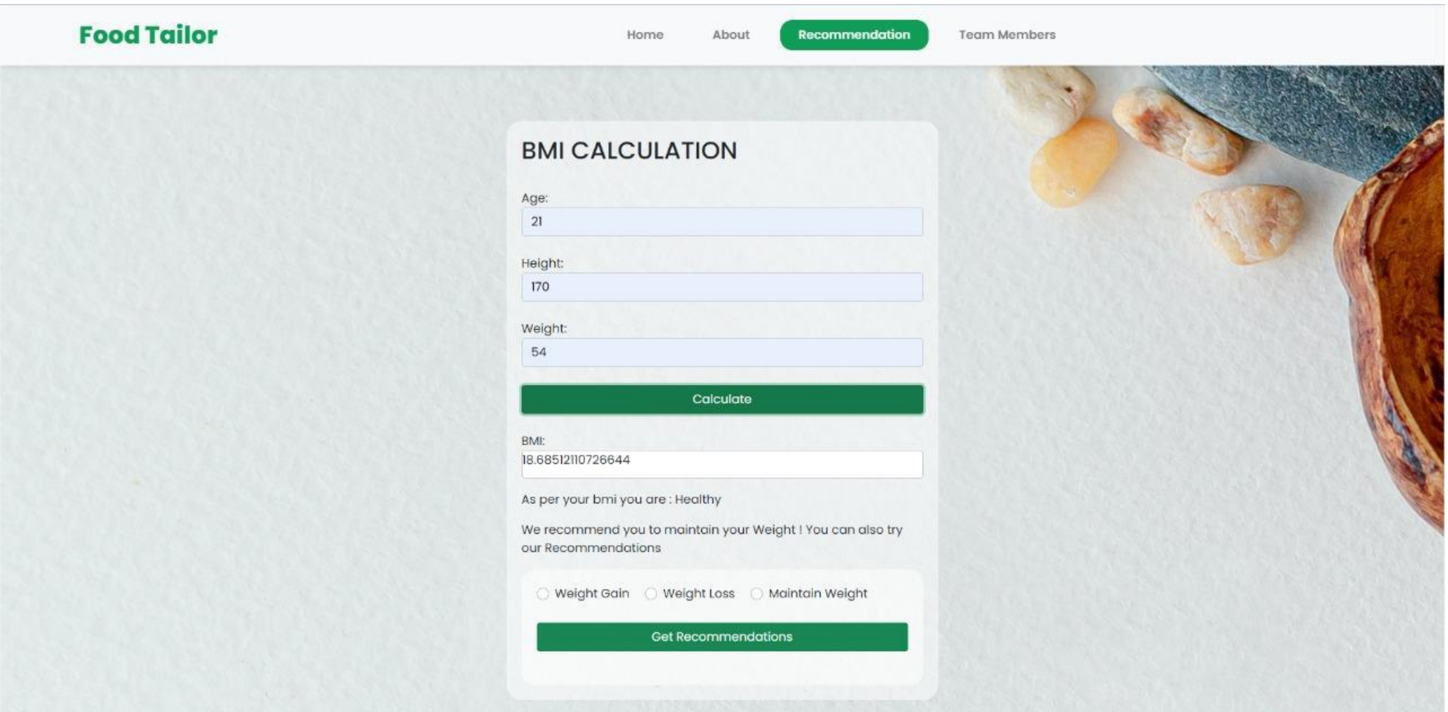


Figure 6: Bmi calculation Underweight Result

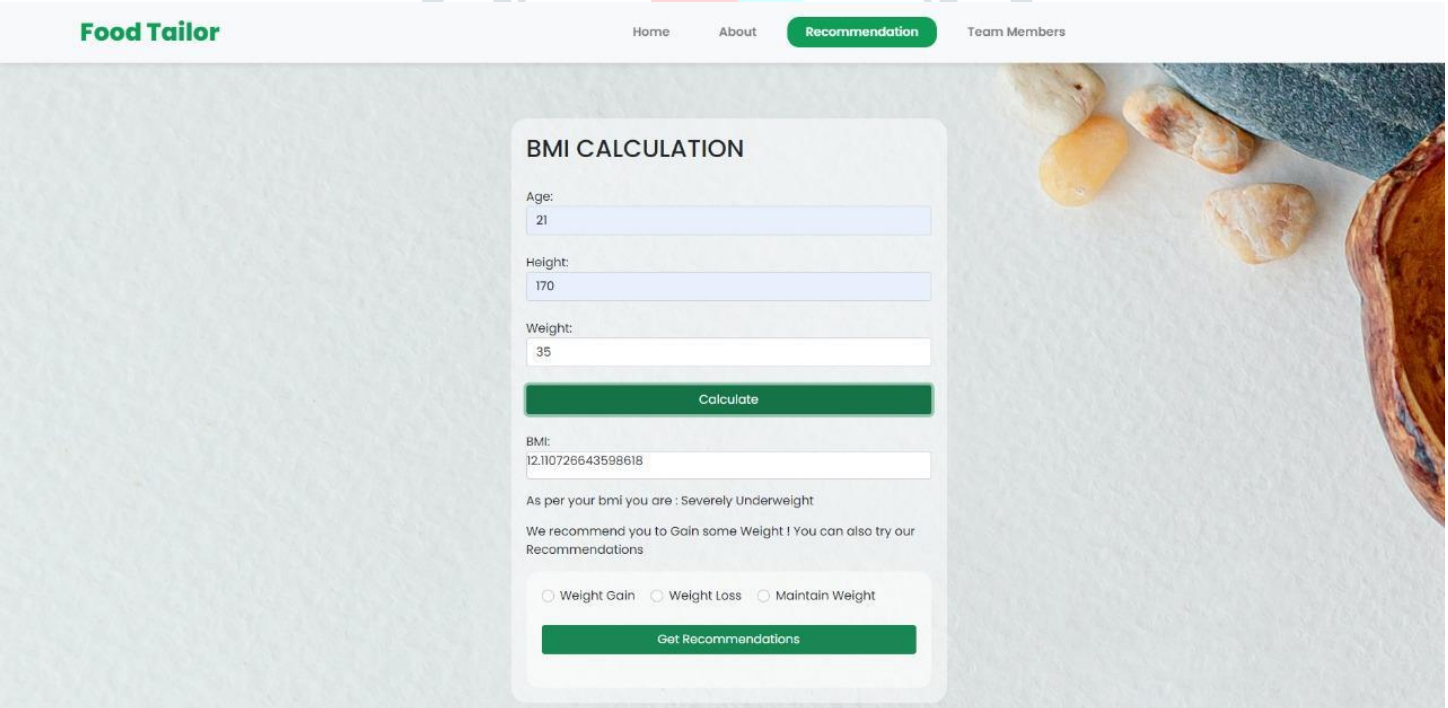


Figure 7: Bmi Calculation Overweight Result

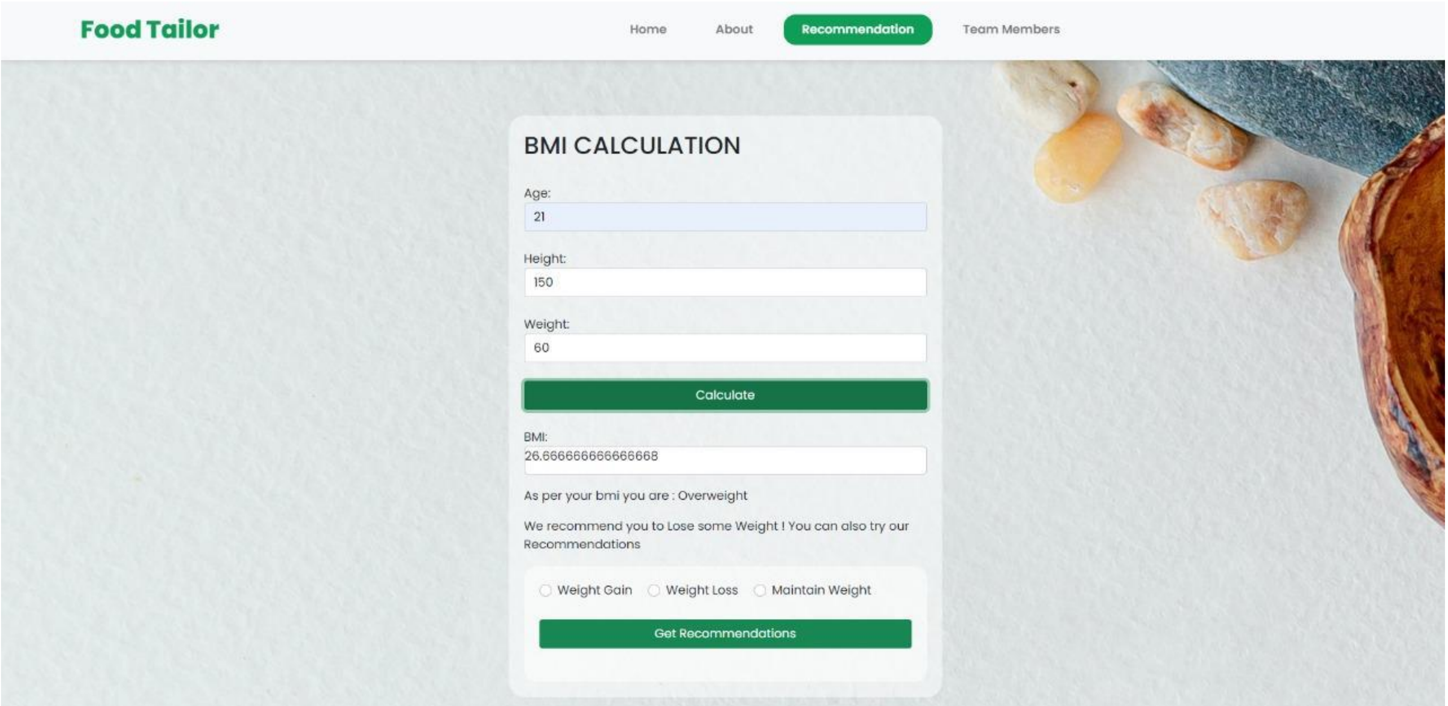


Figure 8: Bmi Calculation Weight Gain Recommendation

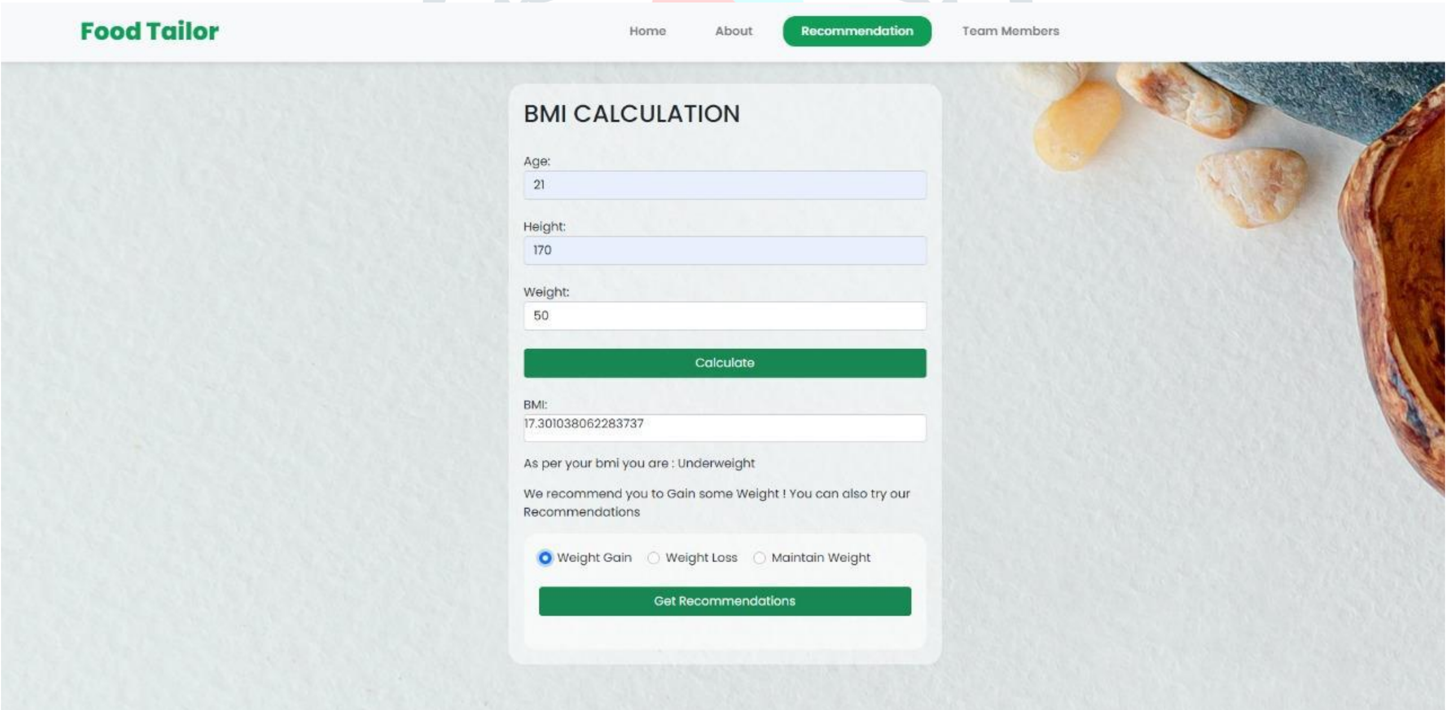
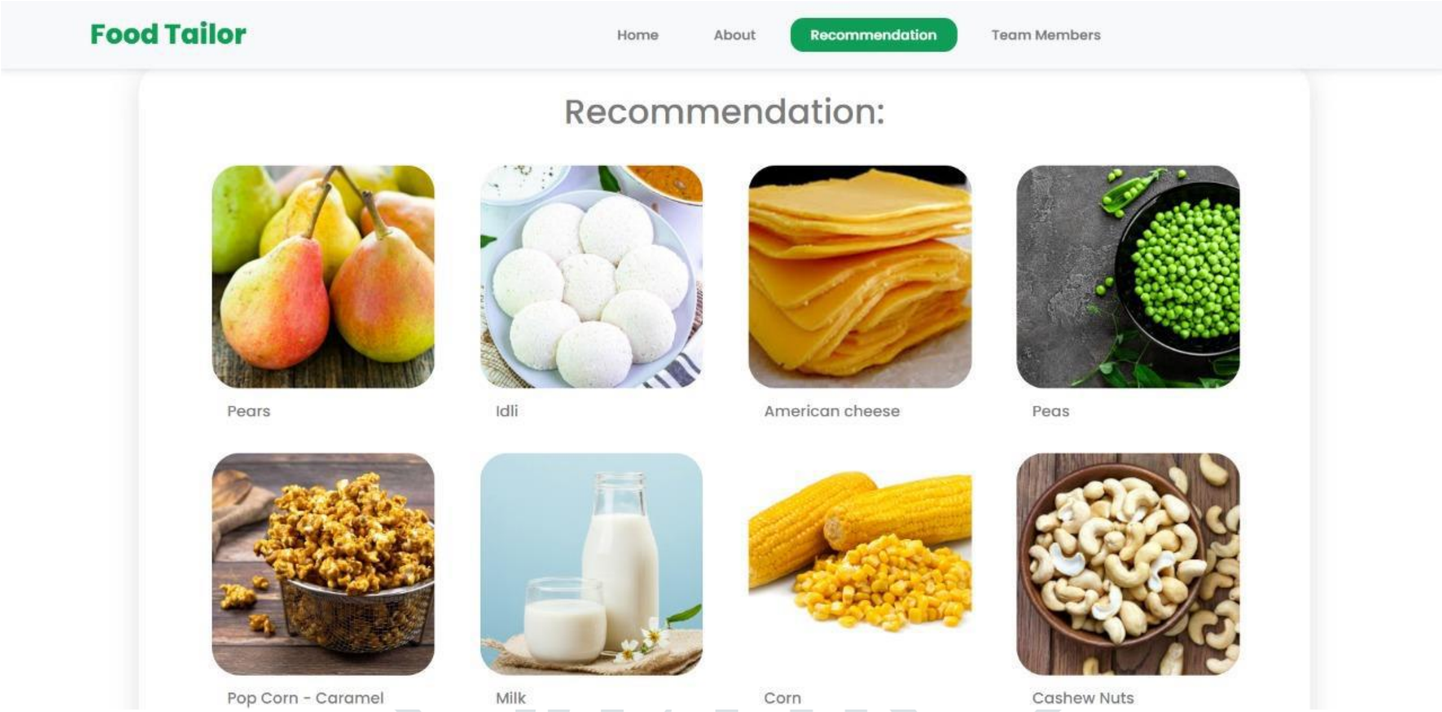


Figure 9: Food Recommendation Based on Bmi Result



6. FUTURE SCOPE AND CONCLUSION

In the future, we plan to expand our project to include an Android application that will recommend foods for specific diseases as well as a basic exercise plan and its maintenance.

By the end of our project, we concluded that if NRS was properly designed, implemented, and evaluated, it could be used as an effective tool to improve nutrition and promote a healthy lifestyle. This study can help to inform nutrition informatics specialists, which was required to design and develop NRS.

7. REFERENCES:

[1] A Mobile Application for Managing Diabetic Patients' Nutrition: A Food Recommender System (2018) Somaye Norouzi 1, Azade Kamel Ghalibaf 2, Samane Sistani 1, Vahideh Banazadeh 3, Fateme Keykhaei 3, Parisa Zareishargh 3, Fateme Amiri 3, Mohsen Nematy 4, Kobra Etminani 1

[2] A Health-Awareness Nutrition Recommender System RACIEL YERA TOLEDO1, AHMAD A. ALZAHRANI2, AND LUIS MARTíNEZ 3, (Member, IEEE) (2019)

[3] Food Recommendation in a Worksite Canteen (2021) Vincenza Carchiolo1 a, Marco Grassia2 b, Alessandro Longheu2 c, Michele Malgeri2 d and Giuseppe Mangioni2e

[4] Neutralize a Personalized Nutrition Recommender System: an enable study (2018) Nadja Leipold, Mira Madenach, Hanna Schäfer, Martin Lurz, Nađa Terzimehić , Georg Groh, Markus Böhm, Kurt Gedrich, Helmut Krcmar

[5] A Disease-driven Nutrition Recommender System based on a Multi-agent Architecture (2018) Todor Ivaşcu, Adriana Diniş, Kristijan Cincar,

[6] A DASH Diet Recommendation System for Hypertensive Patients Using Machine Learning (2019) Romeshwar Sookrah, Jaysree Devee Dhowtal, Soulakshmee Devi Nagowah,

[7] K-means clustering https://www.simplilearn.com/tutorials/machine-learning-tutorial/k-means-clustering-algorithm

[8] A healthy food recommendation system by combining clustering technology with the Weighted slope one Predictor

[9] Building a Food Recommendation System https://towardsdatascience.com/building-a-food-recommendation-system-90788f78691a 33 An overview of recommender systems in the healthy food domain https://link.springer.com/article/10.1007/s10844-017-0469-0

[10] J.J. Otten, J.P. Hellwig, and L.D. Meyers et al. 2006. Dietary reference intakes: the essential guide to nutrient requirements. National Academies Press.

[11] J. Painter, J.-H. Rah, and L. Yeon-Kyung. 2002. Comparison of international food guide pictorial representations. Journal of the Academy of Nutrition and Dietetics (2002)

[12] R. Poinhos and M.D.V. de Almeida. 2015. Personalised nutrition: paving a way to better population health (A White Paper from the Food4Me project). Technical Report. [19] M. Rabbi, M.H. Aung, M. Zhang, and T. Choudhury. 2015. MyBehavior: automatic personalized health feedback from user behaviors and preferences using smartphones. In UbiComp.[20] H. Schäfer, S. Hors-Fraile, R.P. Karumur, A. Calero Valdez, A. Said, H. Torkamaan, T. Ulmer, and C. Trattner. 2017. Towards Health (Aware) Recommender Systems.

[13] In Proceedings of the 2017 International Conference on Digital Health - DH '17. [21] N. Terzimehić, N. Leipold, H. Schäfer, M. Madenach, M. Böhm, G. Groh, K. Gedrich, and H. Krcmar. 2016.

[14] Can an Automated Personalized Nutrition Assistance System Successfully Change Nutrition Behavior? - Study Design. In Thirty Seventh International Conference on Information Systems.[22] T.N. Trang Tran, M. Atas, A. Felfernig, and M. Stettinger. 2018.

[15] An overview of recommender systems in the healthy food domain. Journal of Intelligent Information Systems (2018).[23] D. Trattner and D. Elsweiler. 2019.

[16] Food Recommender Systems: Important Contributions, Challenges and Future Research 34 Directions. In Collaborative Recommendations: Algorithms, Practical Challenges and Applications, Shlomo Berkovsky, Iván Cantador, and Domonkos Tikk (Eds.). World Scientific Publishing. [24] WHO. 2014.

[17] Kaggle https://www.kaggle.com/datasets/izharalam150/diet-recommendation?select=food.csv

[18] Future scope https://www.quora.com/What-is-the-future-of-recommender-systems-research

[19] Global status report on noncommunicable diseases 2014. World Health (2014). [25] D. Zeevi, T. Korem, and N. Zmora et al. 2015. Personalized Nutrition by Prediction of Glycemic Responses. Cell (2015).

JETIR