SAVITRIBAI PHULE PUNE UNIVERSITY

A PROJECT REPORT

ON

INTELLIGENT REFRIGERATOR TO RECOMMEND RECIPES USING HYBRID RECOMMENDATION

Submitted toward the partial fulfillment of the Requirement of

BACHELOR OF ENGINEERING
(COMPUTER ENGINEERING)

BY

Rahul Golhar	B120284317
Manmohan Shinde	B120284326
Shivam Shriwas	B120284328

UNDER THE GUIDANCE OF

Dr. Sunita S. Barve

DEPARTMENT OF COMPUTER ENGINEERING

MIT Academy of Engineering

Alandi(D), PUNE - 412015

2017-2018

SAVITRIBAI PHULE PUNE UNIVERSITY

A PROJECT REPORT

ON

INTELLIGENT REFRIGERATOR TO RECOMMEND RECIPES USING HYBRID RECOMMENDATION

Submitted toward the partial fulfillment of the Requirement of

BACHELOR OF ENGINEERING
(COMPUTER ENGINEERING)

BY

Rahul Golhar	B120284317
Manmohan Shinde	B120284326
Shivam Shriwas	B120284328

UNDER THE GUIDANCE OF

Dr. Sunita S. Barve

DEPARTMENT OF COMPUTER ENGINEERING

MIT Academy of Engineering

Alandi(D), PUNE - 412015

2017-2018

MIT Academy of Engineering
(An Autonomous Institute affiliated to Savitribai Phule Pune University)

MIT ACADEMY OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING
ALANDI(D), PIN CODE-412 015

CERTIFICATE

This is certify that the project entitled

Intelligent Refrigerator to recommend recipes using Hybrid Recommendation

submitted by

Rahul Golhar	B120284317
Manmohan Shinde	B120284326
Shivam Shriwas	B120284328

is a bonafide work carried out by students under the Supervision of Dr. Sunita S. Barve and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering).

Date: / /

(Dr. Sunita S. Barve)
Project Guide

(Dr. Shitalkumar Jain)
HOD, CE Dept

(Dr. Yogesh J. Bhalerao)
Principal

Name of Internal Examiner

External Examiner

MIT Academy of Engineering

(An Autonomous Institute affiliated to Savitribai Phule Pune University)

PROJECT APPROVAL SHEET

Is successfully completed by

Intelligent Refrigerator to recommend recipes using Hybrid Recommendation

submitted by

Rahul Golhar	B120284317
Manmohan Shinde	B120284326
Shivam Shriwas	B120284328

at

DEPARTMENT OF COMPUTER ENGINEERING

MIT Academy of Engineering, Alandi(D)

SAVITRIBAI PHULE PUNE UNIVERSITY,PUNE

ACADEMIC YEAR 2017-2018

Dr. Sunita S. Barve	Dr. Shitalkumar Jain
Internal Guide	H.O.D
Dept. of Computer Engg.	Dept. of Computer Engg.

Acknowledgements

We are profoundly grateful to **Dr. Sunita S. Barve** for her expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion.

We would like to express deepest appreciation towards **Dr. Yogesh J. Bhalerao**, Principal, MIT Academy Of Engineering, **Dr. Shitalkumar Jain**, Head of Department of Computer Engineering and **Dr. Sunita S. Barve**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

Rahul Golhar
Manmohan Shinde
Shivam Shriwas

ABSTRACT

A lot of times we are confused about which dish to cook and end up wasting our time on thinking about it. Thus, we need a Smart System to provide us with a solution to this problem i.e. a system to recommend us which dish to prepare. Here, we make a Recipe recommending fridge to reduce the human efforts and time.

Our main goal here is to develop an Intelligent Refrigerator which will maintain a database containing the list of items. This data is then used to recommend a recipe to the user according to their preferences. It consists of a Smart panel on the door which is nothing but a touch Enabled LCD. It provides a complete list of all the things inside the refrigerator as well as the recipes that can be prepared using those items. And various other option are made accessible to user from here. Based on the vegetables and other products, the Refrigerator is smart enough to give the recommendation on what all possible dishes can be made. The user may then select the recipe to prepare from the list.

Keywords: Machine learning, Visual Recognition, Internet of Things, Data Analytic, Regression

Contents

List of Figures

List of Tables

Synopsis

Chapter 1

Synopsis

1.1 Project Title

Intelligent Refrigerator to recommend recipes using Hybrid Filtering

1.2 Project Option

Sponsored by Persistent Systems Pvt. Ltd.

1.3 Internal Guide

Dr. Sunita S. Barve

1.4 Sponsorship and External Guide

Under the guidance of Chandrashekhar Deshmukh at Persistent Systems Pvt. Ltd. for year 2017-18

1.5 Technical Keywords (As per ACM Keywords)

1. Internet of Things (IoT)
2. Machine learning
3. Tensorflow
4. Visual Recognition
5. Data Analytic

6. Django
7. Flask
8. Image Processing

1.6 Problem Statement

To develop an Intelligent Refrigerator to recommend recipes that can be prepared from the items present inside the fridge.

1.7 Abstract

A lot of times we are confused about which dish to cook and end up wasting our time on thinking about it. Thus, we need a Smart System to provide us with a solution to this problem i.e. a system to recommend us which dish to prepare. Here, we make a Recipe recommending fridge to reduce the human efforts and time.

Our main goal here is to develop an Intelligent Refrigerator which will maintain a database containing the list of items. This data is then used to recommend a recipe to the user according to their preferences. It consists of a Smart panel on the door which is nothing but a touch Enabled LCD. It provides a complete list of all the things inside the refrigerator as well as the recipes that can be prepared using those items. And various other option are made accessible to user from here. Based on the vegetables and other products, the Refrigerator is smart enough to give the recommendation on what all possible dishes can be made. The user may then select the recipe to prepare from the list.

1.8 Goals and Objectives

- Making human life simpler by providing above features.
- Inventory management would be simpler.
- Making better human to machine interaction.
- Save time
- The data generated may be used for analytics by government.

1.9 Relevant mathematics associated with the Project

System Description:

- Input: The Image of each vegetable inside the fridge.
- Output: Recipes that can be prepared from these items.
- Functions : Identify Objects, Morphisms, Overloading in functions, Functional relations
- Mathematical formulation :

Content Based Algorithm:

The recipe is recommended to the user on the basis of ingredients present using the following formula.

We find the score for each recipe as follows:

$$\text{score} = (2 * (Ri \cap I) \big/ (Ri + I))$$

where,
Ri= Recipe ingredients,
I= Set of available items,
$\cap$ = denotes Intersection of sets

Collaborative Filtering Algorithm:

Here, r is the recommended list and is given by:

$$\text{r=num/dem}$$
$$\text{num= n * ps - (s1 * s2)}$$
$$\text{dem= } \sqrt{(n * ss1) - (s1)^2 * (n * ss2) - (s2)^2 2)}$$

n= number of common items
s1= sum of scores of uname
s2= Sum of i

ss1= sum of squared scores of uname
ss2= sum of squared scores of Ui
ps= sum of products of common items

- Success Conditions: The items are detected successfully and the recipes are recommended correctly.
- Failure Conditions: Inability to detect items correctly or inability to give correct recommendation results.

1.10 Names of Conferences / Journals where papers can be published

- IEEE PuneCon 2018 International Conference On Data Science And Analytics

1.11 Review of Conference/Journal Papers supporting Project idea

The concept of smart Homes has now become reality with the help of Internet of Things, Machine Learning, Visual recognition, Image processing and many more technological advancements. Smart refrigeration is also a part of such smart homes, which use IoT to make human life simpler by providing various control options and advanced features.

1.11.1 An IoT-based Appliance Control System for Smart Homes

This paper describes development of home networking and its architecture which is totally wireless based in an attempt to eliminate the use of wires to control the devices completely. An IoT system[1] based on the use of 433 MHz wireless sensor along with actuators and controllers to control various things like air-condition, room lighting and their monitoring.433MHz frequency band is able to transfer data at low rate up to long distances at a very low power and it is available globally and licence-free. To control all the appliances a Smart Home Information System (SHIS)[1] is proposed which can be on the management server or the user computer. SHIS[1] is nothing but a web based application that lets user to control the various appliances through a user interface.

1.11.2 Vegetables detection from the glossary shop for the blind

They have presented the Automatic recognition system (Vegetable vision)[2] for blind people to help them to identify vegetables at the grocery store/supermarket. An image is captured and based on the captured image various parameters like colour, shape, size, density, texture are analyzed as a single parameter is not capable of identification of correct vegetable. Various methods like RGB colour modelling, histogram analysis, image segmentation, textured segments are to be used but increase in the number of parameters increases the recognition time. The future work suggests the use of neural networks in recognition system to reduce latency. Vegetable vision[2] system can be used to quantify quality aspects of vegetables which can be then used as an alternative of manual sorting and grading of vegetables.

1.11.3 Review Paper on Vegetable Identification and Detection using Image Processing

This paper describes project to develop a vegetable detection system[3] using computer vision to identify the vegetables. It is based on image processing, which can control the classification, qualification and segmentation of images and hence recognize the vegetable. The identification of vegetables has to be done by multiple recognition clues such as colour, shape, size, and texture are extracted and analyzed to classify and recognize the vegetables. They have Canny Edge Detection[3] algorithm as it is most optimal. Classification of vegetables is carried out after image detection and segmentation based on the colour classification and texture classification. Image processing does not guarantee the correct identification but it can generate optimal result using above methodology as output depends upon image quality and angle also.

1.11.4 A Content-aware Fridge Based on RFID in Smart Home for Home-Health care

In this paper we studied about Content-aware Fridge[4] based on RFID to provide several content-aware services like creating a grocery list intelligently when the stock in the fridge is less or providing the details about what food to eat to a particular user based on their health and eating habits. The system also recommends recipes in a way such that it ensures overall nutrition of the family. The Content aware fridge uses RFID[4] to integrate item details and user information. The use of RFID[4] is also a drawback of such implementation as most of the items inside the

fridge are kept after unwrapping them in other words removing the RFID[4].

1.11.5 Smart Fridges with Multimedia Capability for Better Nutrition and Health

The project aims at development of Smart Fridge[5] which maintains the user information weight, height, age, medical record, allergic food etc in a database, the database also contains specific nutrition information and based upon the above data it suggests nutritional recipes. It provides all the details via a display mounted on the door. It may notify the user by audio and visual medium. It also provides multimedia cooking demonstration (requires internet connectivity)[5]. Generating store list, generating shopping list, warning on food which is going to expire, displaying calories for various foods are some of its features. But the problem with the system is that uses RFID[5] to scan the items.

1.11.6 IoT Based Smart Refrigerator System

They have designed a smart refrigerator[6] and an android application Which can notify the user about the contents of the fridge. Various notifications are send to user based on the condition whether the fridge is loaded or not. They have also proposed various multimedia capability[6] of the fridge and dietary control and eating routine analysis.

1.11.7 A scalable recipe recommendation system for mobile application

In this paper, they have proposed a hybrid recommendation system[7] for personalized recipe mobile application. The system is a combination of content-based and collaborative filtering. The hybrid recommendation algorithm[7] is designed to improve the effectiveness of recipe recommendation. They have used spark clusters to manage and process high amount of data generated from mobile application. Spark[7] clusters are the open source implementation of map-reduce Framework[7]. It maintains two type of data log data and historical data for the hybrid recommendation system.

1.11.8 Recipe Recommendation Considering Flavours of Regional cuisines[8]

There are variety of recipe recommendation system[8] based on user preference, content based, nutrition based or user health based, in this paper they have presented and recommendation system based on the regional cuisines preferences[8]. It

is a score based system so the flavour preferred by a region is calculated from all regional recipes and hence broken into their ingredients and scored by the TF-IDF[8] (Term frequency and inverse document frequency). Finally, the recommended recipe score is calculated by local score and the score of similar recipes.

1.11.9 Intelligent Food Planning - Personalized Recipe Recommendation

This paper describes use of content based recommendation system[9] for recommending recipes.It has high coverage[9] and reasonable accuracy[9].It uses content based strategies with break down and construction to establish relationship between food items and recipes .There was only a marginal improvement in the accuracy when collaborative filtering was deployed. It does not use collaborative filtering[9] based on user preferences and other and eating habits.

1.11.10 Implementation of a Goal-Oriented Recipe Recommendation Providing Nutrition Information

In this paper they have proposed a recommendation System Providing Nutrition Information[10] this paper they have proposed a recommendation system that suggests recipes to user based on nutritional information[10] on the Internet but the end user gets the recommendation based on simple natural language input like want to cure acne[10]. They have maintained a co-occurrence database that maintains nutrients and their nouns associated with along with this they had ingredients database with the amount of nutrition they have and database which contained all recipes. Using above databases[10] nutrition value of each dish was calculated and then recommended dishes were then compared to manually recommended dishes and overall system performance is evaluated.

1.12 Plan of Project Execution

Table 1.1: Project Plan

Sr. No.	Task Performed	From Date	To Date	Duration in Days
1	Identification Of Problem Statement	15-07-2017	30-07-2017	15
2	Abstract And Research On Topic	01-08-2017	15-08-2017	14
3	Project Overview	15-08-2017	05-09-2017	21
4	Literature Survey	06-09-2017	30-09-2017	24
5	Identifying Objectives, Scope and Goals	15-09-2017	05-10-2017	20
6	Coding of Modules	15-10-2017	10-12-2017	56
7	Documentation	20-11-2017	10-01-2018	51
8	Integration of Different Modules	11-01-2018	05-02-2018	25
9	Implementation of Work Done	06-02-2018	28-02-2018	22
10	Presentation of Project	15-02-2018	10-03-2018	23
11	Testing and Correction	11-03-2018	31-03-2018	20
12	Result Analysis	27-03-2018	10-04-2018	14
13	Documentation	25-03-2018	20-04-2018	26

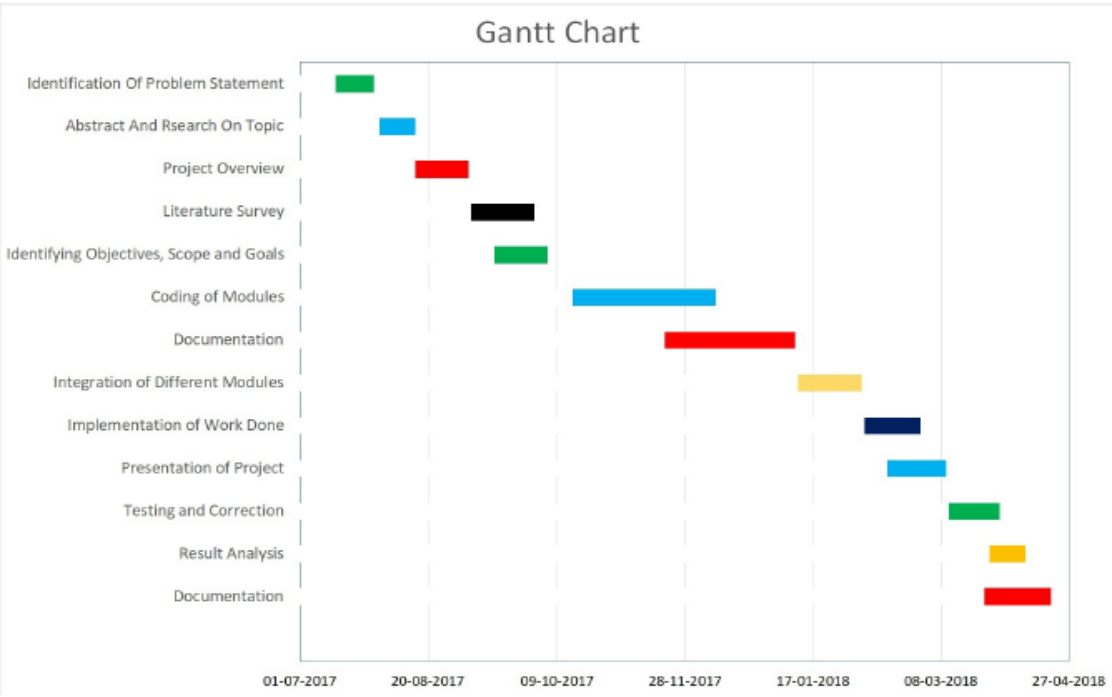


Figure 1.1: Prepared Grant chart for Project Plan

Chapter 2

Technical Keywords

2.1 Area of Project

Data Analytics, Internet Of Things, Smart Home Systems

2.2 Technical Keywords

1. Internet of Things (IoT)
2. Machine learning
3. Tensorflow
4. Visual Recognition
5. Data Analytic
6. Django
7. Flask
8. Image Processing

Chapter 3

Introduction

3.1 Project Idea

The main idea of the project is to make a system that recommends the recipe according to the vegetables present inside the refrigerator. The system maintains a table for keeping the records of items. The system has two options, one is for adding an item in the list and another is for removing it.
There are 2 options for profile related functions:

- Sign-up for new user
- Sign-in for existing user

As the name suggests these two options are used by user to sign in and sign up. Once the user signs up there is a unique ID allocated to the user and the further operations of user are monitored like the rating given and other few stuffs.

Once the user signs in there is a dashboard provided for every user. There are mainly 4 options available on the Dashboard:

- Capture the Image of a vegetable
- Reset the system
- Give Content Based Recommendation
- Give Collaborative Recommendation

Firstly, the Capture option is used to capture the image of the vegetables. Once the image is captured it is saved on the server and is then used for recognition later on.

Secondly, the user has options for resetting the list of items in the fridge. If the user resets the system all the data is lost i.e. all the images and list of vegetables are

lost. Once the user resets the system, user needs to scan all the vegetables again.

Thirdly, the user has an option of content based recommendation. Here, the system first analyses the list of items and then uses it to recommend the possible number of recipes.

Finally, we have an option which gives recommendations based on Collaborative filtering results. Recommending Recipe is based on the reviews given by other similar users.

3.2 Motivation of the Project

A lot of times we are confused about which dish to cook and end up wasting a lot of time on thinking about it. Thus, we need a Smart System to provide us with a solution to this problem I.e. a system to recommend us which item to prepare.

3.3 Literature Survey

The concept of smart Homes has now become reality with the help of Internet of Things, Machine Learning, Visual recognition, Image processing and many more technological advancements. Smart refrigeration is also a part of such smart homes, which use IoT to make human life simpler by providing various control options and advanced features.

3.3.1 An IoT-based Appliance Control System for Smart Homes

This paper describes development of home networking and its architecture which is totally wireless based in an attempt to eliminate the use of wires to control the devices completely. An IoT system[1] based on the use of 433 MHz wireless sensor along with actuators and controllers to control various things like air-condition, room lighting and their monitoring.433MHz frequency band is able to transfer data at low rate up to long distances at a very low power and it is available globally and licence-free. To control all the appliances a Smart Home Information System (SHIS)[1] is proposed which can be on the management server or the user computer. SHIS[1] is nothing but a web based application that lets user to control the various appliances through a user interface.

3.3.2 Vegetables detection from the glossary shop for the blind

They have presented the Automatic recognition system (Vegetable vision)[2] for blind people to help them to identify vegetables at the grocery store/supermarket. An image is captured and based on the captured image various parameters like colour, shape, size, density, texture are analyzed as a single parameter is not capable of identification of correct vegetable. Various methods like RGB colour modelling, histogram analysis, image segmentation, textured segments are to be used but increase in the number of parameters increases the recognition time. The future work suggests the use of neural networks in recognition system to reduce latency. Vegetable vision[2] system can be used to quantify quality aspects of vegetables which can be then used as an alternative of manual sorting and grading of vegetables.

3.3.3 Review Paper on Vegetable Identification and Detection using Image Processing

This paper describes project to develop a vegetable detection system[3] using computer vision to identify the vegetables. It is based on image processing, which can control the classification, qualification and segmentation of images and hence recognize the vegetable. The identification of vegetables has to be done by multiple recognition clues such as colour, shape, size, and texture are extracted and analyzed to classify and recognize the vegetables. They have Canny Edge Detection[3] algorithm as it is most optimal. Classification of vegetables is carried out after image detection and segmentation based on the colour classification and texture classification. Image processing does not guarantee the correct identification but it can generate optimal result using above methodology as output depends upon image quality and angle also.

3.3.4 A Content-aware Fridge Based on RFID in Smart Home for Home-Health care

In this paper we studied about Content-aware Fridge[4] based on RFID to provide several content-aware services like creating a grocery list intelligently when the stock in the fridge is less or providing the details about what food to eat to a particular user based on their health and eating habits. The system also recommends recipes in a way such that it ensures overall nutrition of the family. The Content aware fridge uses RFID[4] to integrate item details and user information. The use of RFID[4] is also a drawback of such implementation as most of the items inside the

fridge are kept after unwrapping them in other words removing the RFID[4].

3.3.5 Smart Fridges with Multimedia Capability for Better Nutrition and Health

The project aims at development of Smart Fridge[5] which maintains the user information weight, height, age, medical record, allergic food etc in a database, the database also contains specific nutrition information and based upon the above data it suggests nutritional recipes. It provides all the details via a display mounted on the door. It may notify the user by audio and visual medium. It also provides multimedia cooking demonstration (requires internet connectivity)[5]. Generating store list, generating shopping list, warning on food which is going to expire, displaying calories for various foods are some of its features. But the problem with the system is that uses RFID[5] to scan the items.

3.3.6 IoT Based Smart Refrigerator System

They have designed a smart refrigerator[6] and an android application Which can notify the user about the contents of the fridge. Various notifications are send to user based on the condition whether the fridge is loaded or not. They have also proposed various multimedia capability[6] of the fridge and dietary control and eating routine analysis.

3.3.7 A scalable recipe recommendation system for mobile application

In this paper, they have proposed a hybrid recommendation system[7] for personalized recipe mobile application. The system is a combination of content-based and collaborative filtering. The hybrid recommendation algorithm[7] is designed to improve the effectiveness of recipe recommendation. They have used spark clusters to manage and process high amount of data generated from mobile application. Spark[7] clusters are the open source implementation of map-reduce Framework[7]. It maintains two type of data log data and historical data for the hybrid recommendation system.

3.3.8 Recipe Recommendation Considering Flavours of Regional cuisines[8]

There are variety of recipe recommendation system[8] based on user preference, content based, nutrition based or user health based, in this paper they have presented and recommendation system based on the regional cuisines preferences[8]. It

is a score based system so the flavour preferred by a region is calculated from all regional recipes and hence broken into their ingredients and scored by the TF-IDF[8] (Term frequency and inverse document frequency). Finally, the recommended recipe score is calculated by local score and the score of similar recipes.

3.3.9 Intelligent Food Planning - Personalized Recipe Recommendation

This paper describes use of content based recommendation system[9] for recommending recipes.It has high coverage[9] and reasonable accuracy[9].It uses content based strategies with break down and construction to establish relationship between food items and recipes .There was only a marginal improvement in the accuracy when collaborative filtering was deployed. It does not use collaborative filtering[9] based on user preferences and other and eating habits.

3.3.10 Implementation of a Goal-Oriented Recipe Recommendation Providing Nutrition Information

In this paper they have proposed a recommendation System Providing Nutrition Information[10] this paper they have proposed a recommendation system that suggests recipes to user based on nutritional information[10] on the Internet but the end user gets the recommendation based on simple natural language input like want to cure acne[10]. They have maintained a co-occurrence database that maintains nutrients and their nouns associated with along with this they had ingredients database with the amount of nutrition they have and database which contained all recipes. Using above databases[10] nutrition value of each dish was calculated and then recommended dishes were then compared to manually recommended dishes and overall system performance is evaluated.

Chapter 4

Problem Definition and scope

4.1 Problem Statement

To develop an Intelligent Refrigerator to recommend recipes that can be prepared from the items present inside the fridge.

4.1.1 Goals and Objectives

- Making human life simpler by providing above features.
- Inventory management would be simpler.
- Making better human to machine interaction.
- Save time
- The data generated may be used for analytics by government.

4.1.2 Statement of scope

The concept of smart Homes has now become reality with the help of Internet of Things, Machine Learning, Visual recognition, Image processing and many more technological advancements. Smart refrigeration is also a part of such smart homes,which use IoT to make human life simpler by providing various control options and advanced features.

- The system takes the images of vegetables as an input one by one and displays the results of recommendation based on the list generated. The user needs to provide the system only with images of vegetables otherwise the system might classify it as some vegetable and give some wrong outputs.

- The system can be used in our day to day homes and is very easy to use and user friendly as well.
- The system can also be used in canteens or food stalls but it might not be so much useful there as they already know what the user wants and they don't need kind of recommendation.

4.2 Major Constraints

Most important constraints are:

1. For image processing:
   (a) Proper functioning of pi camera
   (b) Clear images
   (c) Object scanned should be a vegetable only
   (d) Huge amount of training data for tensorflow
2. For content based recommendation:
   (a) Good Internet Connection
   (b) The database must be strong
   (c) The list of items should not be empty
   (d) The list of items should contain only vegetables
3. For recommendation using collaborative filtering:
   (a) Good Internet Connection
   (b) The database of user ratings must be strong
   (c) The user should provide time to time ratings to keep the data functioning smooth

4.3 Methodologies of Problem solving and efficiency issues

The single problem can be solved by different solutions. This considers the performance parameters for each approach. Thus considers the efficiency issues.

- We implement our system using servers so as to reduce the load on our Raspberry pi which is used for image capturing and classification.

- This provides faster execution and quick response thus it reduces the latency as well
- Storing and processing of data on servers is a secure and faster way of processing the day. It also increases the system's data reliability due to its fault tolerant properties.
- Use of Pi camera lets the system to save a huge amount of processing.
- Use of Firebase is done to store the user data by giving a unique id to each individual which increases the data security of the system.

4.4 Outcome

The main outcome of the project is to save user's time by recommending recipes:

- **Using Content Based Recommendation method:** The user gets the list of recipes that can be made from the current list of vegetables.
- **Using Collaborative Filtering method:** The user gets the list of some new recipes that could be tried based on ratings from other users.

4.5 Applications

- The system can be widely used in homes and due to its low cost the system will be affordable for a huge number of population.
- The system could be used in restaurants as well but it might not be of much use there as we already know what the user wants and there is no need of any recommendation there.

4.6 Hardware Resources Required

Basically there are 2 hardware components required:

- Raspberry Pi
- Pi Camera

The requirements of these components are given below:

1. Rapberry Pi:

Sr. No.	Parameter.	Minimum Requirement.
1	CPU	4 ARM Cortex-A53, 1.2GHz
2	GPU	Broadcom VideoCore IV
3	RAM	1GB LPDDR2 (900 MHz)
4	Networking	10/100 Ethernet, 802.11n wireless
5	Bluetooth	Bluetooth 4.1 Classic
6	Storage	microSD
7	GPIO	40-pin header, populated
8	USB Ports	4
9	Camera port	Camera Serial Interface (CSI)
10	SoC	Broadcom BCM2837

Table 4.1: Raspberry Pi Specifications

2. Pi Camera:

Sr. No.	Parameter.	Minimum Requirement.
1	Size	Around 25 24 9 mm
2	Weight	3g
3	Still resolution	5 Megapixels
4	Video modes	1080p30, 720p60 and 640 480p60/90
5	Linux integration	V4L2 driver available
6	C programming API	OpenMAX IL and others available
7	Sensor	OmniVision OV5647
8	Sensor resolution	2592 1944 pixels
9	Sensor image area	3.76 2.74 mm
10	Pixel size	1.4 m 1.4 m
11	Optical size	1/4”

Table 4.2: Pi Camera Specifications

4.7 Software Resources Required

Platform :

1. **Operating System:** Raspian OS
2. **IDE:** PyCharm
3. **Programming Language:** Python 3.1
4. **Database:** Firebase(JSON format)
5. **Web Frameworks:** Django and Flask

Chapter 5

Project Plan

5.1 Project Estimates

5.1.1 Reconciled Estimates

Cost Estimate

The actual system consists of only Raspberry Pi and Pi camera. The fridge could be of any type, any company and any size. Cost could be estimated as follows:

- Raspberry Pi: 3000-3500 Rs
- Pi camera: 550-650 Rs

The total amount could be anywhere between 3600-4200. But one thing to consider is that this cost is excluding the cost of the fridge so it may wary for different type and size of refrigerators. Another important factor to be considered here is, the cost of setting up servers and storage. The recipe data will be mostly static but the personal data of users needs to be updated each time so the server needs to function accordingly. This cost maybe estimated around 2L to 3.5L Rupees.

Time Estimates

It took us around 8 months to complete the project and one month for testing purpose.

5.1.2 Project Resources

1. Software Resources Required:
   (a) **Operating System:** Raspian OS
   (b) **IDE:** PyCharm

(c) **Programming Language:** Python 3.1

(d) **Database:** Firebase(XML format)

(e) **Web Frameworks:** Django and Flask

2. Hardware Resources Required:

(a) **Hardware device:** Raspberry pi

(b) **Camera:** Pi Camera

5.2 Risk Management w.r.t. NP Hard analysis

Before embarking on the project it is necessary to review all the risks that might be involve in it. These risks have been documented as seen before the implementation of the project.

5.2.1 Risk Identification

We have identified a few risks and these are mentioned below:

- Availability of internet is necessary.
- Cold start problem for collaborative filtering.
- Possibility of having no recipes based on the vegetables present
- System might recommend recipes that user doesn't like
- Failure of server due to some technical errors
- Failure of Hardware components
- Images captured might not be clear enough to correctly classify the vegetable

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Table 5.1: Risk Analysis Table

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Availability of internet is necessary	High	Low	Low	Medium
2	Cold start problem for collaborative filtering	High	High	High	High
3	No recipes based on the vegetables present	Low	Low	Low	Low
4	Recommend recipes that user doesnt like	Medium	Low	Low	Low
5	Failure of server	Low	High	High	High
6	Failure of Hardware components	Low	High	High	High
7	Images not suitable for classification	Low	High	High	High

Probability	Value	Description
High	Probability of occurrence is	$> 75\%$
Medium	Probability of occurrence is	$26 - 75\%$
Low	Probability of occurrence is	$< 25\%$

Table 5.2: Risk Probability definitions

Impact	Value	Description
Very high	$> 10\%$	Schedule impact or Unacceptable quality
High	$5 - 10\%$	Schedule impact or Some parts of the project have low quality
Medium	$< 5\%$	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk Impact definitions

Risk ID	1
Risk Description	Availability of internet is necessary
Category	Development Environment
Source	Software requirement Specification document
Probability	High
Impact	Low
Response	Mitigate
Strategy	Having good internet connection
Risk Status	Occurred

Risk ID	2
Risk Description	Cold start problem for collaborative filtering
Category	Technology
Source	This was identified during early development and testing.
Probability	High
Impact	High
Response	Accept
Strategy	Letting the system get mature
Risk Status	Occurred

Risk ID	3
Risk Description	No recipes based on the vegetables present
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Low
Response	Accept
Strategy	To have a good database of recipes
Risk Status	Identified

Risk ID	4
Risk Description	Recommend recipes that user doesn't like
Category	Technology
Source	This was identified during early development and testing.
Probability	Medium
Impact	Low
Response	Accept
Strategy	To provide good collaborative filtering result to overcome the result of content based results.
Risk Status	Occurred

Risk ID	5
Risk Description	Failure of server
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	High
Response	Mitigate
Strategy	To provide good fault tolerent server structure.
Risk Status	Identified

Risk ID	6
Risk Description	Failure of Hardware components
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	High
Response	Mitigate
Strategy	To provide good hardware structure.
Risk Status	Occurred

Risk ID	7
Risk Description	Images not suitable for classification
Category	Technology
Source	This was identified during early development and testing.
Probability	Medium
Impact	High
Response	Mitigate
Strategy	To provide good error free images.
Risk Status	Occurred

5.3 Project Schedule

5.3.1 Project task set

- Task 1: Identification Of Problem Statement
- Task 2: Abstract And Research On Topic
- Task 3: Project Overview
- Task 4: Literature Survey
- Task 5: Identifying Objectives, Scope and Goals
- Task 6: Coding of Modules
- Task 7: Documentation
- Task 8: Integration of Different Modules
- Task 9: Implementation of Work Done
- Task 10: Presentation of Project
- Task 11: Testing and Correction
- Task 12: Result Analysis
- 13: Documentation

5.3.2 Timeline Chart

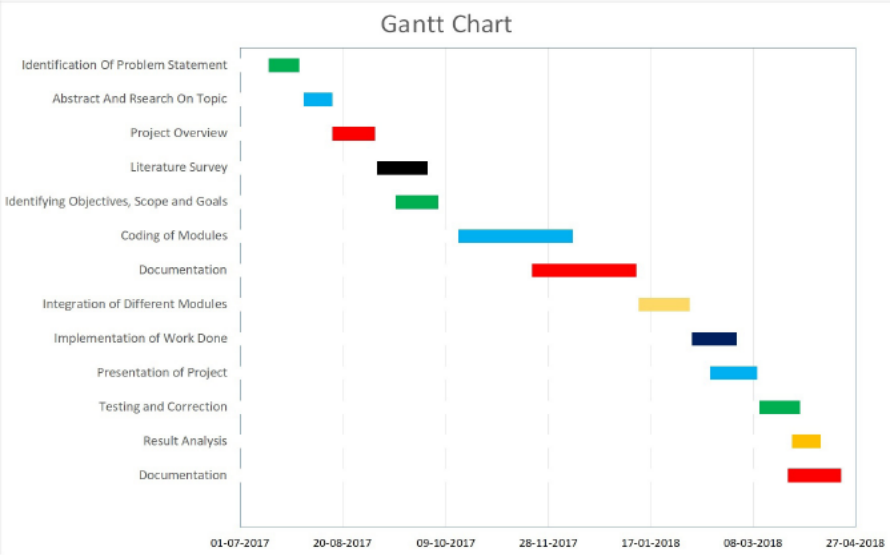


Figure 5.1: Timeline Chart

5.4 Team Organization

5.4.1 Team structure

The team consists of 3 people. Their contribution has been given in the table below.

Sr. No.	Name	Tasks
3	Rahul Golhar	Documentation and Development
2	Shivam Shriwas	Development and Server Setup
3	Manmohan Shinde	Tensorflow training and Database Management

Table 5.4: Roles of members

5.4.2 Management reporting and communication

Mechanisms for progress reporting and inter/intra team communication are identified as per assessment sheet and lab time table.

Sr. no.	**Audience**	**Information**	**Frequency**
1	Team+Guide	Accomplishments and challenges	Monthly
2	Team+Guide	Detailed Status	Every 2-3weeks
3	Team	High level project status	Weekly
4	Team	Issues and Challenges	Every 2-3 days
5	Team+External Guide	High level status	Every 2-3 months

Table 5.5: Roles of members

Software requirement specification

Chapter 6

Software requirement specification

6.1 Introduction

6.1.1 Purpose and Scope of Document

The Purpose of this document is to give detailed description of the requirement for the **"Intelligent Refrigerator to Recommend Recipes Using Hybrid Recommendation"** . It will illustrate the purpose and complete declaration for the development of system.It will also explain system constraint, interface and interaction with other external applications.

6.1.2 Overview of responsibilities of Developer

- Develop the quality product
- Develop the models and documentation which is required for the purpose of supporting the solution in live use
- Participating in any quality assurance work required to ensure the delivered products are truly fit for purpose
- Testing the output of their own work prior to independent testing

6.2 Usage Scenario

This section provides various usage scenarios for the system to be developed.

6.2.1 User profiles

There are basically 2 actors:

1. **Customer:** The customer is the one who uses the whole system. In other words, we can say, a customer is the one who needs recommendation regarding recipes.

2. **Server:** The database server contains all the recipes and their ingredients. In addition to this, it also maintains a database of all users.

6.2.2 Use-cases

Use case diagram at its simplest is a representation of users interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and often be accompanied by other types of diagram as well.

The use case diagram is shown in which there are 2 actors, Customer and Server.

1. **Customer:** The user needs to Signup to the system and a profile is created. If the user is an existing user he/she need to sign in to the system. Once the user has signed in, the user needs to capture images of the vegetables one by one by clicking on the Capture button. Once, this is done a list is created by the system that contains all the vegetables. Then to get recommendations, the user needs to click on the recommendation buttons provided.

2. **Server:** The server has 2 tasks, profile validation and recommendation. When a new user signs up the server stores its data onto the database and creates a unique ID to maintain his/her profile. Once this is done, when user signs in the server is again used for validation of the user. After logging in the user performs his/her tasks as required. There are 2 more option for user and those are logout from the system and reset the system.
   Once the user clicks on the recommendation buttons the control is transferred to the server. Here, the list is analyzed and the recipes are recommended using content based filtering. Another type of recommendation is given by Collaborative Filtering method. The ratings given by other users are analyzed and then the recipe is recommended by considering score from similar users.

Sr No.	Use Case	Description	Actors
1	Use Case 1	The interaction of user and the system in defined	Customer and Server

Table 6.1: Use Cases

6.2.3 Use Case View

Use Case Diagram. Example is given below

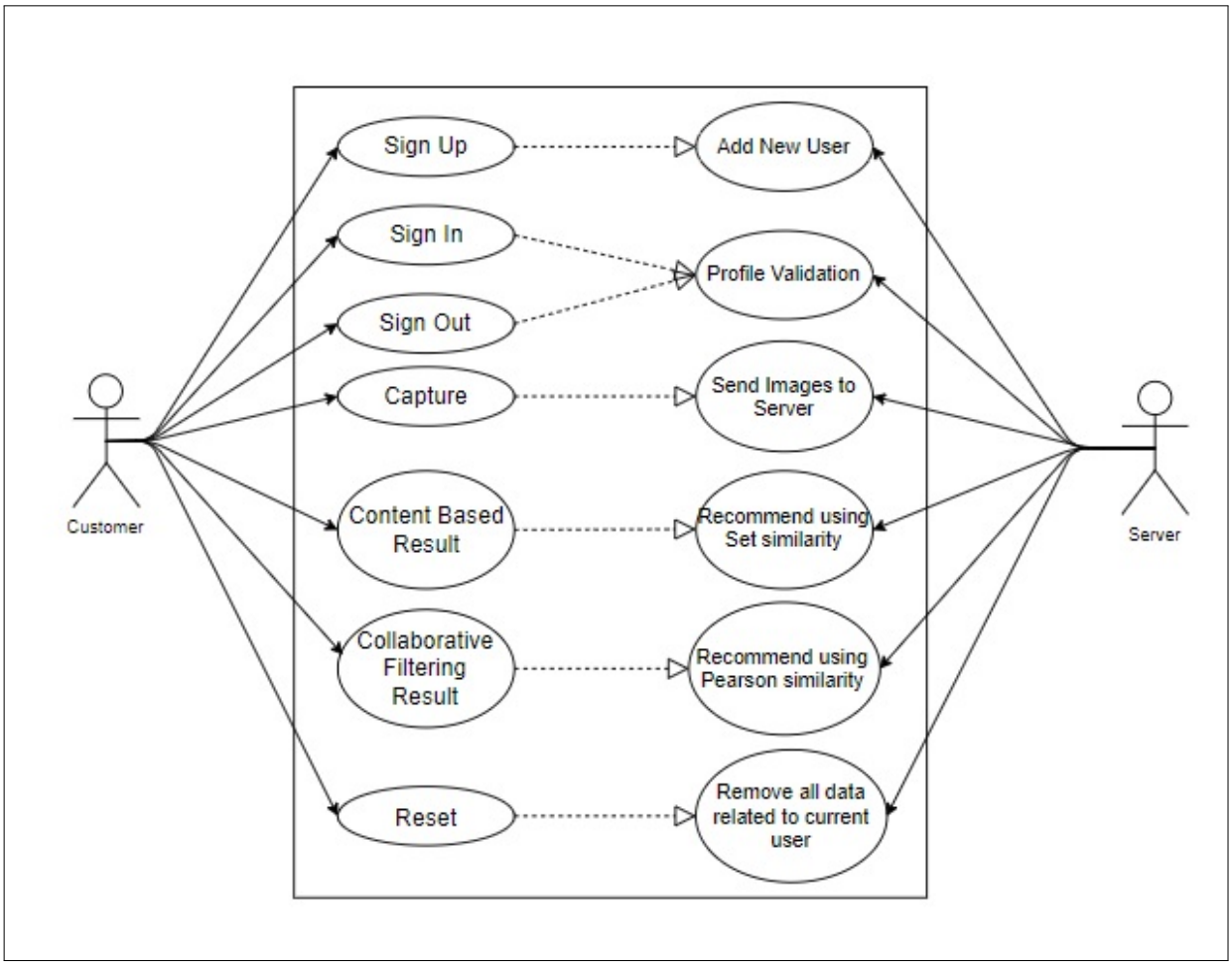


Figure 6.1: Use case diagram

6.3 Functional Model and Description

6.3.1 Data Flow Diagram

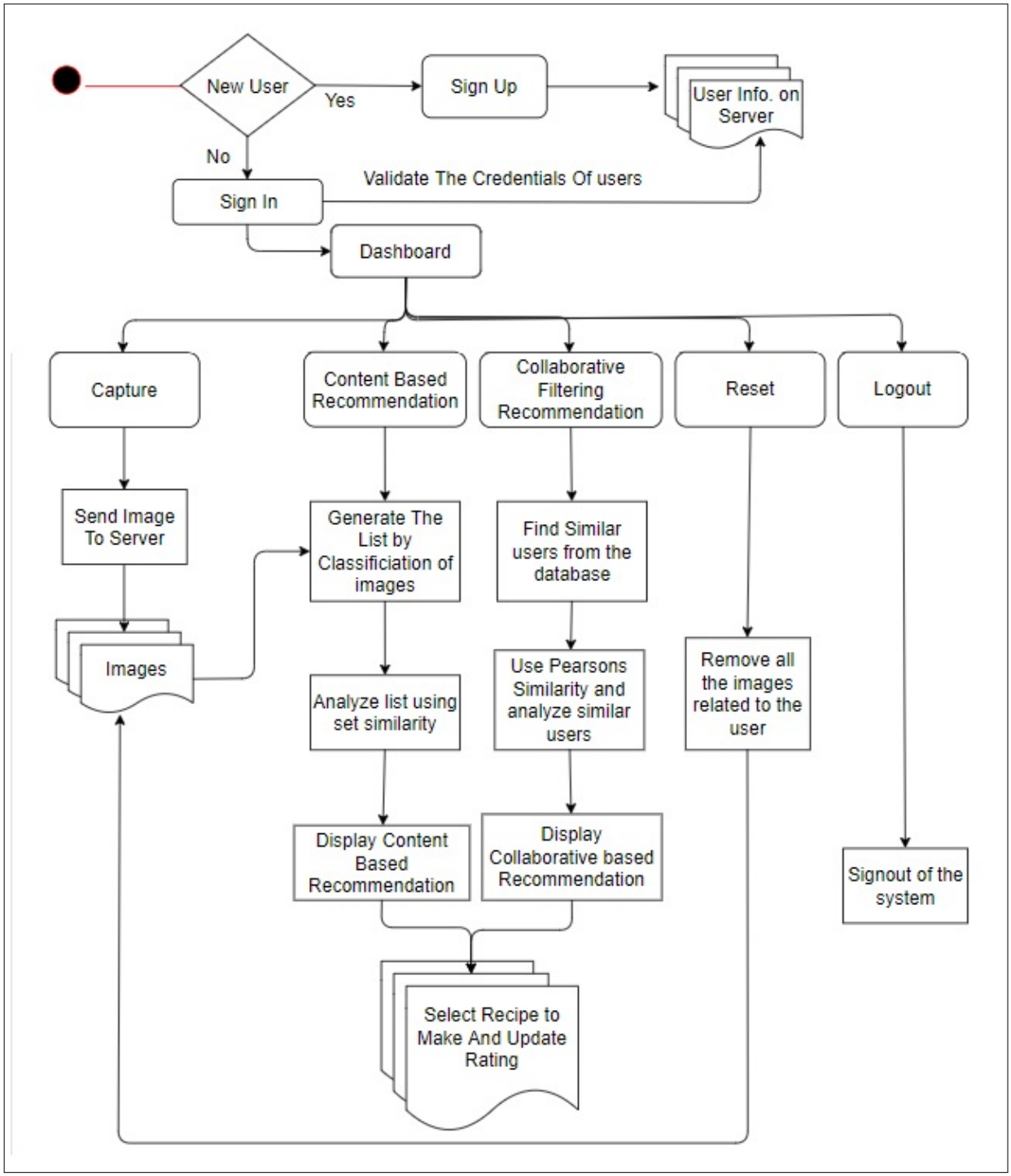


Figure 6.2: Data Flow diagram

6.3.2 Activity Diagram:

- The Activity diagram represents the steps taken.

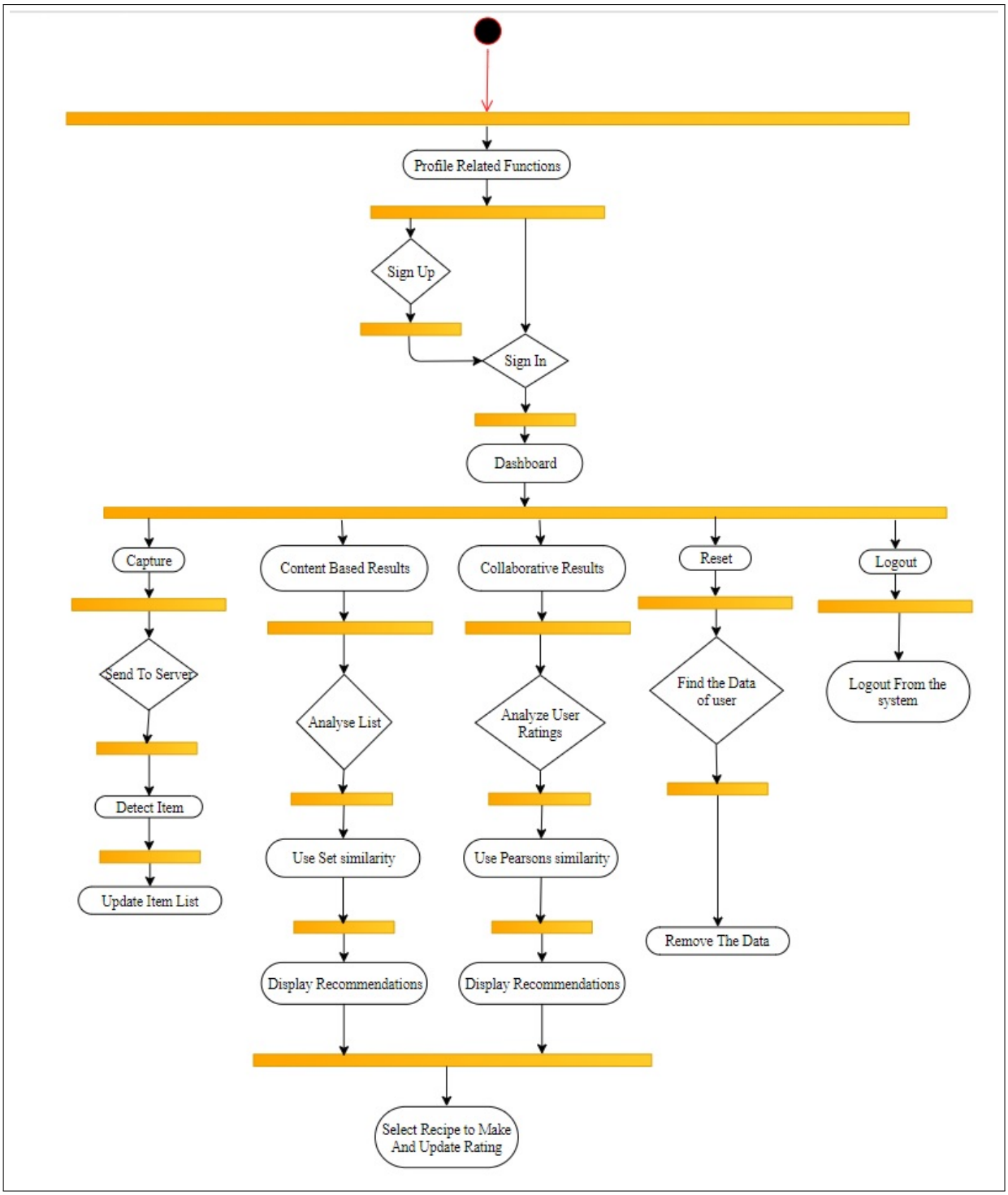


Figure 6.3: Activity diagram

6.3.3 State Diagram:

State Transition Diagram

Fig.6.4 example shows the state transition diagram of our system. The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating our project.

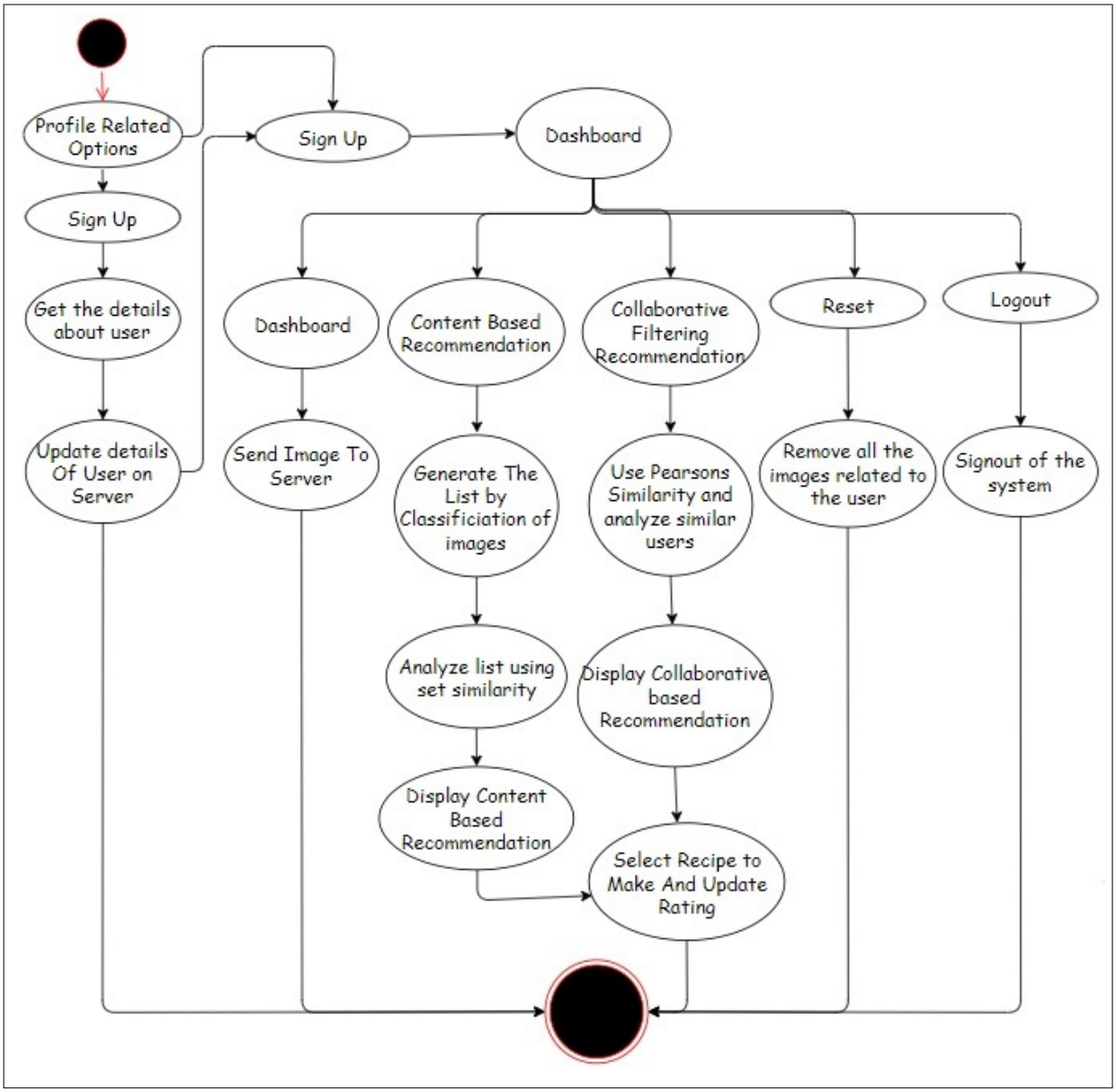


Figure 6.4: State transition diagram

Detailed Design Document using Appendix A and B

Chapter 7

Detailed Design Document

7.1 Introduction

This document specifies the design that is used to solve the problem of Product.

7.2 Architectural Design

A description of the program architecture is presented. Subsystem design or Block diagram,Package Diagram,Deployment diagram with description is to be presented.

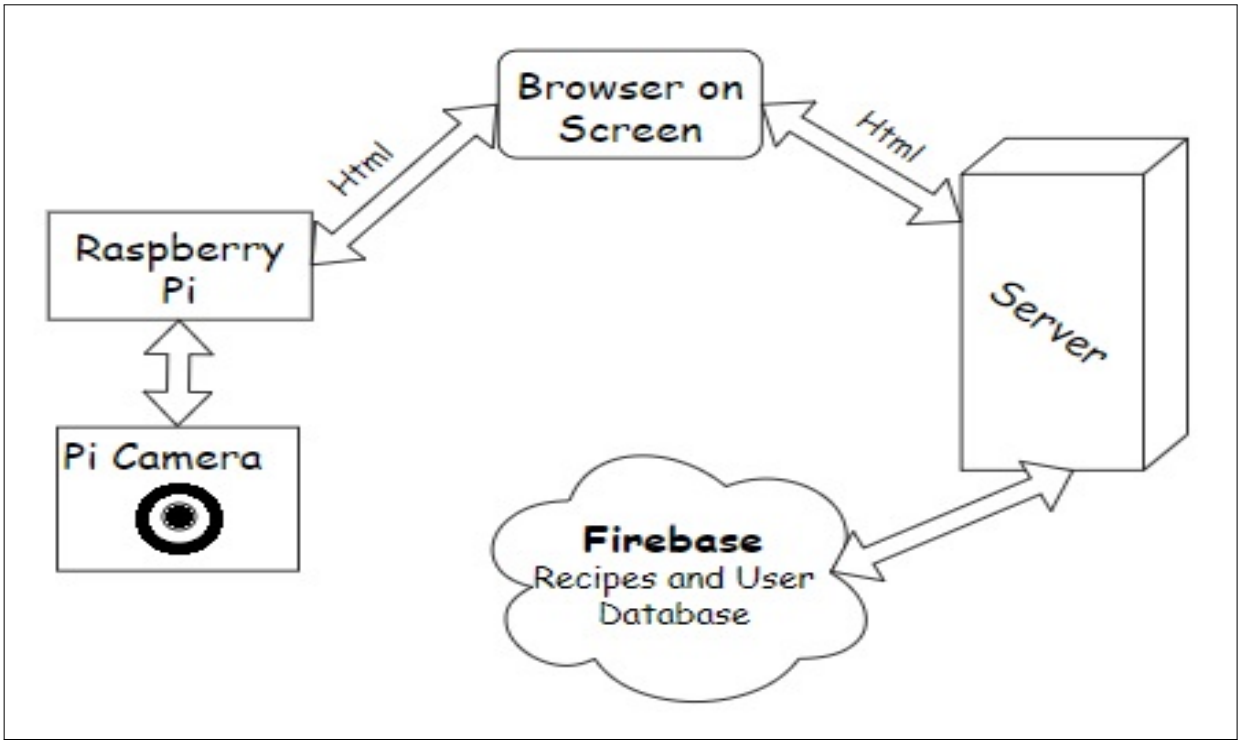


Figure 7.1: Architecture diagram

7.3 Data design (using Appendices A and B)

A description of all data structures including internal, global, and temporary data structures, database design (tables), file formats.

7.3.1 Internal software data structure

Data structures used are:

- List : For storing list of vegetables after classification
- Dictionary: Store user ratings, Store Key Value Pair of content based results where key is recipe name and value is the rating of the recipe provded
- Images: jpeg format for saving images
- Database: JSON format

7.4 Component Design

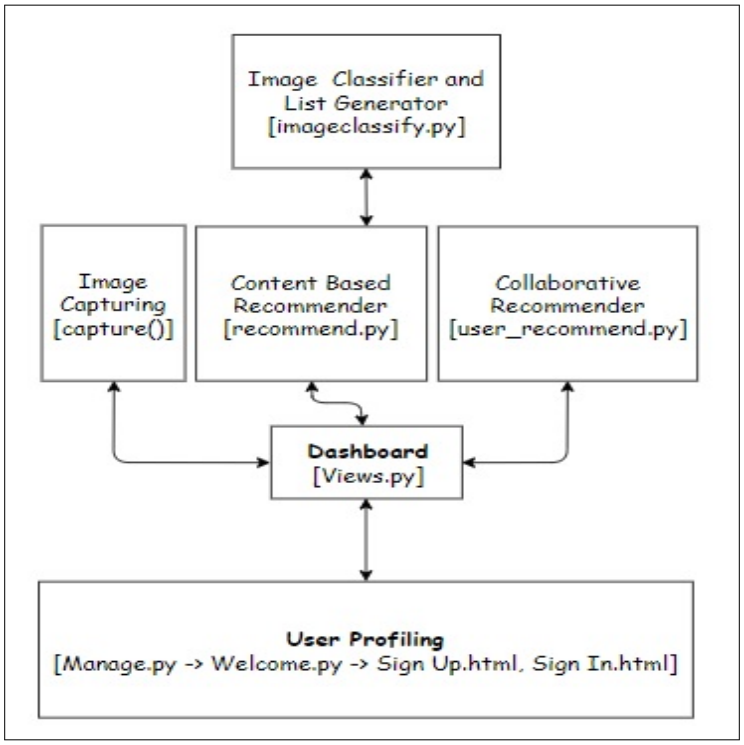


Figure 7.2: Component diagram

Project Implementation

Chapter 8

Project Implementation

8.1 Introduction

Due to the advancements of computer technologies and the tremendous use of the Internet, intelligent systems with multimedia and smart operation capabilities have been emerging into our daily life. Kitchen is one of the places where such intelligent systems have been used a lot. Since modern life style is giving people less time because of their busy schedules, they dont get a lot of time to thinking of which recipe to cook, what are the items they have in their fridge, what else do they need to buy, etc. So, an enjoyable and healthy life style can be achieved by using an intelligent system such as a smart fridge. In this paper we introduce a novel application for a smart fridge with multiple intelligent capability. It is designed for managing items stored in it and advising its users with cooking recipes depending on what kind of food is stored. It can also perform other functions such as adding new recipe discovered, reading out the recipe for user, finding related videos online, eating habit analysis, etc. We are confident that an Intelligent Fridge will be an important part in future smart homes.

The Internet of Things (IoT) a technology which is a global network of machines and devices capable of interacting with each other as well as the Internet has been predicted to be a new wave of technological advancements by many academics and manufactures and will also improve, or at least change the lives from conventional to modern life. There are a lot of applications of this technology. And is used to make daily routines quicker, smarter and more efficient what were once tedious and hectic (Lee and Lee, 2015).

The idea of an Intelligent Refrigerator is the application that we will explore in this project. Already there have been some attempts of making a Smart Fridges specifically for household use. However, many thought of it as a seemingly obvious n great opportunity to introduce such technology into homes, they have failed to make an impact and have been branded as commercial flops (Kuniavsky, 2010).

Here we will build our idea upon these models and try to come up with a solution to save time and money as well. We commonly have features that help track inventory, set the user preferences of recipes, get recommendation about recipes - to create a smart fridge which has the additionally functionality of helping the person to save time by not wasting their time on thinking about which recipe to make today. This additional feature is expected to increase the commercial appeal of an Intelligent Fridge, as time has become more important in the busy life of a modern person on a global scale.

Some of the initial ideas for our Intelligent Fridge system include:

- LCD screen display - showing all the options available for the user to choose from.
- User Profile - In the beginning, customer needs to create his/her profile. Once the profile is created the user needs to sign in to access the dashboard.
- Content Based Recommendation - This option is used to recommend the recipes based on the contents inside the refrigerator.
- Collaborative Filtering Based Recommendation - This option is used to recommend the recipes based on the rating given by the similar users.
- Once all the recipes are recommended the user chooses one and then rates the recipe.

There are also certain features which could be added to make it a better. This Fridge can also hold information on household food that doesn't need to be kept in the refrigerator (i.e. cupboard food) by scanning an item or receipt to provide better information on recipes. It can also show and vary temperature according to the food stored inside the fridge. Warning when there is a shortage of food is also given before the items get finished. Online ordering feature based on the users average

shopping of items, connected mobile app - so you can track your diet and items inside the fridge anywhere, to receive notifications about what you need when you are at the store.

In the next section we will provide an evaluation of current systems which will draw upon task analysis, so that a better understanding of user requirements as well as flaws can be made.

8.2 Tools and Technologies Used

8.2.1 Recommendation systems:

Recommendation systems[18] changed the way the websites communicated with their users. Rather than providing a static experience in which users searched for and potentially bought products, recommendation systems increased interaction to provide a richer experience. Recommendation systems[18] identify recommendations autonomously for each individual user based on users' behavior.

Most recommendation systems take either of two basic approaches collaborative filtering or content-based filtering. Other approaches (such as hybrid approaches) also exist.

- **Collaborative filtering :**
  Collaborative filtering[17] gives us a recommendation which is based on a model of prior user behavior. The model can be constructed either from a single user's behavior or more effectivelyalso from the behavior of other users who have similar interests. When it takes other users' behavior into account then collaborative filtering[17] uses group knowledge to form a recommendation based on similar users. Recommendations are based on an automatic collaboration of multiple users and filtered on those who exhibit similar preferences or behaviors.[17]

- **Content-based filtering :**
  Content-based filtering[17] constructs a recommendation on the basis of a user's current behavior. For example, this approach might use historical browsing information, such as which blogs the user reads and the characteristics of those blogs. If a user commonly reads articles about Linux or is likely to leave comments on blogs about software engineering, content-based filtering can use this history to identify and recommend similar content (articles on Linux or other

blogs about software engineering). This content can be manually defined or automatically extracted based on other similarity methods.

- **Hybrid approach :**
  Hybrid approaches that combine collaborative and content-based filtering are also increasing the efficiency (and complexity) of all recommendation systems. Incorporating the results of collaborative and content-based filtering creates a more accurate recommendation. The hybrid approach is also used to address collaborative filtering that starts with sparse data also known as cold start by enabling the results to be weighted towards content-based filtering initially, then shifting the weight toward collaborative filtering as available user data set matures with time.

In our system, the user is suggested recipes which can be made from items in his fridge. The list of items is analyzed and then suitable recipe is recommended. In addition to this, the recipes chosen by people in a certain area can be used to recommend the recipes for a person in that area. By using the information from many users in different areas, you can group those users based on their preferences. So, we can group together users who prefer same type of dishes. From this information, you identify the most popularly prepared recipes by the people in the area. Then 68 for a particular user in the group you recommend some of the most popular recipe that he or she can prepare with items present inside the fridge.

If a person is living in India then he must be recommended Indian Dishes as all people around him prefer those dishes but in case if he is an American person living in India his first preference should be American recipes and then Indian recipes.

Content-Based Filtering Strategy

Content-based filtering[16] is used in a lot of applications, like information retrieval (search engines) and also recommender systems.[13]

WHY CONTENT-BASED FILTERING?

Collaborative filtering[16] is the go to option when it comes to systems, but content-based filtering does have a lot of advantages, especially in certain situations.[13]

- `Results seem highly relevant.` As content-based recommendations[16] depends on characteristics of objects themselves, they are expected to be highly

relevant to interests of user[16]. This makes them especially valuable for organizations with massive libraries of a single type of content.

- `There is transparency in recommendations.` Another benefit is that the process by which recommendation is generated can be easily made transparent[16], which also increases users trust in the recommendations. With collaborative-filtering, the process is a black box type the algorithm and users do not really know why theyre getting the recommendations they are watching.
- `Users can start quickly.` Content-based filtering[16] removes the cold-start problem that collaborative-filtering techniques has.Still the system needs some initial inputs from user to start making recommendations the quality of those recommendations not much better. When a system becomes robust after data items have been added and correlated.[13]
- `Immediately we can recommend new items.` Another problem with collaborative-filtering[16] is that new objects added to the database will have few (if any) interactions, i.e. they wont be recommended too often. Unlike collaborative-filtering systems, content-based systems dont require other users to interact with an object before it starts recommending items.
- `Easier to implement.` Compared to the difficult math involved in building collaborative-filtering system, the data science behind a content-based system is much straightforward.[16] The main task, we've seen is in assigning the attributes.

Collaborative Filtering Strategy

When designing a recommendation system, there are two major ways to go about it. Weve already talked about content-based filtering, but what do you do when your content is simply too massive or diverse to manually apply attributes? For that, theres collaborative filtering, a technique thats widely used across social media, retail, and streaming services. In this article, well explore how collaborative filtering works, where its used, and what skills you might need to get started.

For all the sophisticated math and machine learning techniques involved, the concept behind collaborative filtering is pretty straightforward: Its based on the idea that people who share an interest in certain things will probably have similar tastes in other things as well. You experience collaborative filtering first-hand every time you go online and see Customers Who Bought This Item Also Bought, or Users like you also liked

WHY COLLABORATIVE FILTERING?

The main difference between collaborative filtering and content-based filtering is conceptual. Where content-based filtering is built around the attributes of a given object, collaborative filtering relies on the behavior of users. This approach has some distinct advantages over content-based filtering:

- `Results seem highly relevant.` As content-based recommendations depends on characteristics of objects themselves, they are expected to be highly relevant to interests of user. This makes them especially valuable for organizations with massive libraries of a single type of content.[13] It benefits from large user bases. Simply put, the more people are using the service, the better your recommendations will become, without doing additional development work or relying on subject area expertise.
- `Flexible across different domains.` Collaborative filtering approaches are well suited to highly diverse sets of items. Where content-based filters rely on metadata, collaborative filtering is based on real-life activity, allowing it to make connections between seemingly disparate items (like say, an outboard motor and a fishing rod) that nonetheless might be relevant to some set of users (in this case, people who like to fish).
- `It produces more serendipitous recommendations.` When it comes to recommendations, accuracy isnt always the highest priority. Content-based filtering approaches tend to show users items that are very similar to items theyve already liked, which can lead to filter bubble problems. By contrast, most users have interests that span different subsets, which in theory can result in more diverse (and interesting) recommendations.
- `It can capture more nuance around items.` Even a highly detailed content-based filtering system will only capture some of the features of a given item. By relying on actual human experience, collaborative filtering can sometimes recommend items that have a greater affinity with one another than a strict comparison of their attributes would suggest.

8.2.2 Hardware Components:

Raspberry Pi:

The Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries. The original model became far more popular than anticipated, selling outside of its target market for uses such as robotics. Peripherals (including keyboards, mice and cases) are not included with the Raspberry Pi. Some accessories however have been included in several official and unofficial bundles.[12]

Sr. No.	Parameter.	Minimum Requirement.
1	CPU	4 ARM Cortex-A53, 1.2GHz
2	GPU	Broadcom VideoCore IV
3	RAM	1GB LPDDR2 (900 MHz)
4	Networking	10/100 Ethernet, 802.11n wireless
5	Bluetooth	Bluetooth 4.1 Classic
6	Storage	microSD
7	GPIO	40-pin header, populated
8	USB Ports	4
9	Camera port	Camera Serial Interface (CSI)
10	SoC	Broadcom BCM2837

Table 8.1: Raspberry Pi Specifications

PiCamera:

The Raspberry Pi camera module can be used to take high-definition video, as well as still photos. Its easy to use, but has plenty to offer advanced users if youre looking to achieve more knowledge. There are lots of examples online of people using it for time-lapse, slow-motion and many more videos. You can also use the libraries we bundle with the camera to create effects.

In our system, we capture the image and then send it to our server to find the content by using tensorflow for Visual Recognition.

Sr. No.	Parameter.	Minimum Requirement.
1	Size	Around 25 24 9 mm
2	Weight	3g
3	Still resolution	5 Megapixels
4	Video modes	1080p30, 720p60 and 640 480p60/90
5	Linux integration	V4L2 driver available
6	C programming API	OpenMAX IL and others available
7	Sensor	OmniVision OV5647
8	Sensor resolution	2592 1944 pixels
9	Sensor image area	3.76 2.74 mm
10	Pixel size	1.4 m 1.4 m
11	Optical size	1/4”

Table 8.2: Pi Camera Specifications

8.2.3 Software Components:

Raspbian OS :

Raspbian is a widely used operating system for Raspberry Pi. It is a free OS which is based on Debian, which is optimised for the Raspberry Pi hardware.[19] It also comes with over 35,000 packages i.e. precompiled software bundled in an easy installation format on your Raspberry Pi. Raspbian is a community project, with a focus on improving the Debian package’s stability and performance.[19]

VNC Connect :

VNC Connect consists of two apps, VNC Server and VNC Viewer:

1. VNC Server: It enables user to connect to Raspberry Pi from a desktop or mobile, obe can watch its screen in real-time, and user can control system as if they are sitting in front of it.[20]
2. VNC Viewer: It enables user to connect to and control a desktop computer (or another Pi) from your own Raspberry Pi.[20]

PyCharm :

Pycharm is an IDE used in our system. It has **Intelligent Python Assistance**[21] as it provides smart code completion, code inspections, quick-fixes and error highlighting, along with automated code refactoring and great navigation capabilities. It also has a collection of **Web Development Frameworks**[21] as it offers great framework-specific support for latest web development frameworks such as Django, Flask, Google App Engine, Pyramid, and web2py. In addition to this it has great support of **Scientific Tools**[21] as it integrates with IPython Notebook which supports multiple scientific packages including matplotlib and NumPy. It also has **Cross-technology Development**[21] as it supports JavaScript, CoffeeScript, TypeScript, Cython, SQL, HTML/CSS, template languages, AngularJS, Node.js, and more. It has **Remote Development Capabilities**[21] as well which enable to Run, debug, test, and deploy applications on remote hosts or virtual machines, with remote interpreters, an integrated ssh terminal, and Docker and Vagrant integration. Along with this it has a great collection of **built-in Developer Tools**: [21]an integrated debugger and test runner; Python profiler; a built-in terminal; and integration with major VCS and built-in Database Tools.

Image Classification using Tensorflow:

Image recognition and classification are the tasks of the system. Using the Inception-v3 model, we will be classifying images using our own classifier that contains trained images of vegetables.[22] It is the fastest and the simplest way to do image recognition on your laptop or computer without any GPU because it is just an API and your CPU is good enough for this.[22]

Django Web Framework:

Django is a high-level Python Web framework which encourages faster development and clean, pragmatic design. It has been built by experienced developers. It takes care of much of the issues that occur in Web development. It is an open source and free sorftware. It was designed to help us to take applications from concept to completion as rapidly as possible. In addition to this, it takes security seriously and helps us avoid many security mistakes. Another feature is it's ability to quickly and flexibly scale.[23]

Flask:

Flask is an easy-to-extend Python web framework built with a small core. It is considered more Pythonic than the Django web framework because equivalent Flask web application is more explicit in common situations. It is also easy to get started with as a beginner because there is little boilerplate code for getting a simple app up and running.[24]

Firebase:

Firebase is a mobile and web app development platform that provides developers with a plethora of tools and services to help them develop high-quality apps, grow their user base, and earn more profit.[25] We can build apps fast and that too without managing infrastructure. Firebase gives us functionalities like analytics, databases, messaging and crash reporting. And as it is backed by Google it can be trusted. As it is built on Google infrastructure and scales automatically it can be used for even the largest apps.[26]

8.3 Methodologies/Algorithm Details

8.3.1 Algorithm 1 - Capturing Of Images

1. get url with local id='LocalId'
2. imgcount++
3. Disable capture button
4. Call Ajax functions using type-'GET' and url-'url'
5. get LocalId
6. set camera resolution
7. set variable filename to 'timestamp.jpg'
8. capture image as 'filename.jpg' under directory/Pictures
9. store image in Firebase
10. get the url of captured image and store it in list of urls
11. store urls in dictionary data where {Key: Image, Value: url}

12. return (data)
13. If successful then display all images using url

8.3.2 Algorithm 2 - Content based algorithm

1. Set url=`http://ip:portnumber/recommend`
2. Call function using Ajax function where type=post, url=url
3. Get localId
4. Get urlList by replacing'b' by '' and '"' by '', and spliting by ','
5. Create directory localID and if it already exists then print "Already Exists"
6. Download each image in urlList using required url retrieves
7. Call imageClassifier function to get list of available items in I
8. Get list of recipes from Firebase
9. Then score for a recipe is calculated as follow:
   score = $(2 * (Ri \cap I) / (Ri + I))$
   where,
   Ri= Recipe ingredients,
   I= Set of available items,
   $\cap$ = denotes Intersection of sets
10. if score is greater than 0 add recipe name and score to "data" variable in dictionary format
11. Sort data according to score
12. return data to ajax function
13. return data

8.3.3 Algorithm 3 - Collaborative Filtering Algorithm

1. if user clicks for Collaborative results
2. Set variable
   url=;http://ip:portnumber/userrecommend/;

3. Call Ajax functions using type-'GET' and url-'URI'
4. get Local Id
5. get username using local id from Firebase
6. results=userRecommend(uname, 1, data)
   Where, uname- current username
   1- bound value
   data- user Rating
   In userRecommend function
7. Declare list scores=[]
8. For users(i) in data loop:
   n= number of common items
   s1= sum of scores of uname
   s2= Sum of i
   uname= current username
   i= users other than uname

   ss1= sum of squared scores of uname
   ss2= sum of squared scores of
   ps= sum of products of common items

   num= n * ps - (s1 * s2)
   dem= $\sqrt{(n*ss1)-(s1)^2*(n*ss2)-(s2)^2 2)}$

   If den!= 0:
   scores.append(num)/(den))
   Else: scores.append(0)
   end for
9. Sort list scores in reverse order

10. Declare recommend an empty dictionary
    item resultList=[]
    Return list from userRecommend function

11. for i in len(scores):
    sim,others = scores
    ranked = data[others]
    End for

12. for item in ranked:
    if item not in data-of-person do:
    weight = sim*ranked[item] if item in recommend:
    s,weight=recommend[item]
    recommend[item] = (s+sim,weight + weight[])
    else
    recommend[item]=(sim,weight[])
    End if
    End if
    End For

13. For r in recommend:
    sim,item = recommend[r]
    recommend[r] = sim[item]/sim
    End For

14. return(recommend)

15. Append key of result in resultList
    If no item in result goto step 9, Else repeat step 8

16. Return responce in string format

17. On success ajax function will print data on the webpage.

Software Testing

Chapter 9

Software Testing

9.1 Type of Testing Used

1. **White box testing :** It is a testing technique, that examines the program structure and derives test data from the program logic/code. The other names of glass box testing are clear box testing, open box testing, logic driven testing or path driven testing or structural testing.

2. **Black-box testing :** It is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance.

3. **Unit Testing :** Unit testing focuses on the smallest unit of software design i.e. the smallest component or module. Important control paths are tested to uncover errors within the boundary of the module. It focuses on the internal processing logic and data structures within the boundaries of a component. This type of testing can be conducted in parallel for multiple components.

4. **Integration Testing :** Integration testing is a systematic technique for constructing the software architecture while at the same time conducting tests to uncover errors associated with interfacing. The different modules in our project were interfaced and tested in small increments, thus making the errors easy to isolate and correct. This is known as incremental integration.

5. **Validation Testing :** Validation testing begins at the culmination of integration testing, when individual components have been exercised, software is completely assembled as a package, and interfacing errors have been uncovered and corrected. Here, testing focuses on user visible actions and user recognizable output from the system. Validation succeeds when the software functions in

a manner that can be reasonably expected by the customer. In our project, all functions and performance characteristics are tested and they conform to the required specifications and are accepted.

6. **System Testing :** This is the final step in testing. In this phase, we tested the entire system as a whole with all forms, code, modules and class modules. This form of testing is known as Black Box testing or System testing. Black Box testing enables us to derive sets of input conditions that will fully exercise all functional requirements for a program. Black box testing helps to discover incorrect or missing functions, interface errors, errors in data structure, performance errors and initialization and termination errors.

9.2 Test Cases and Test Results

Sr. No.	Test Case	Expected Results	Actual Results	status
1	Login using invalid User-ID or Password	Rejected	Rejected	Pass
2	Login using valid User-ID or Password	Accepted	Accepted	Pass
3	Scan non vegetables	Reject	Reject	Pass
4	Scan untrained vegetable	Reject	Reject	Pass
5	Scan trained vegetable	Accept	Accept	Pass
6	Collaborative Recommendation for new user	Reject	Reject	Pass
7	Collaborative Recommendation for old user	Accept	Accept	Pass

Table 9.1: Test Cases

Result

Chapter 10

Results

10.1 Screen shots

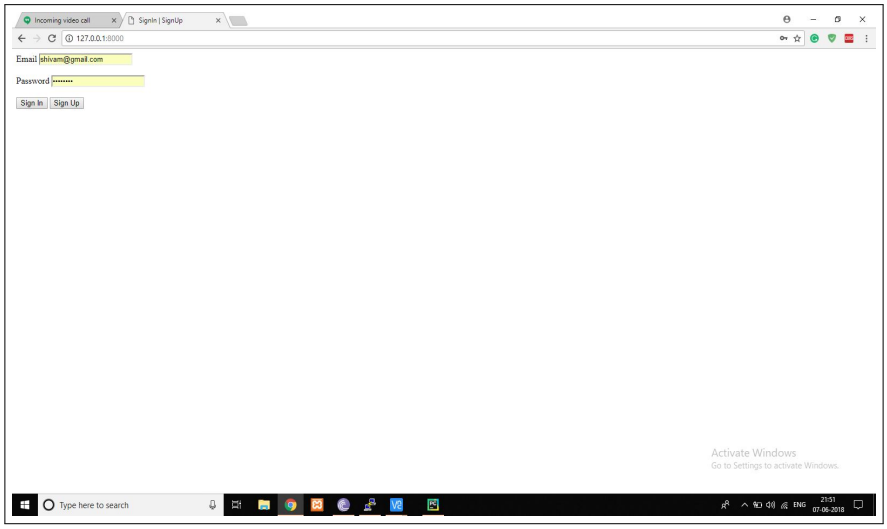


Figure 10.1: SignIn Page

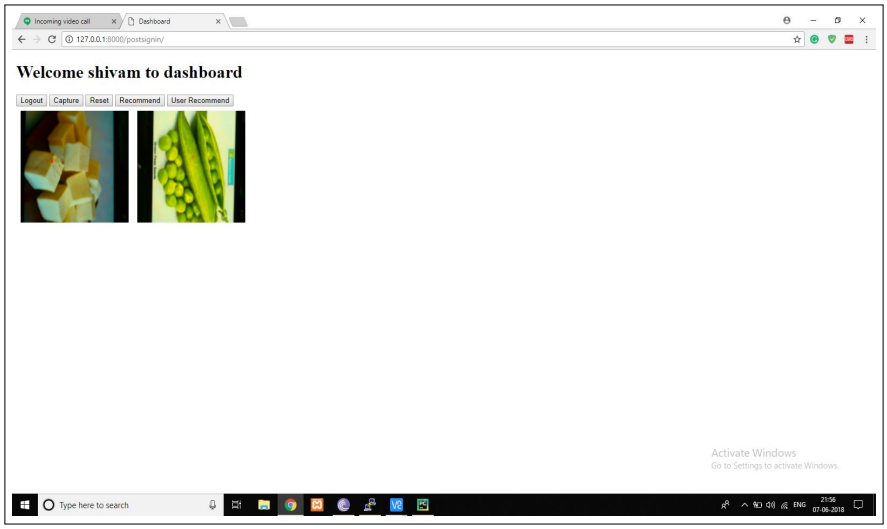


Figure 10.2: Image Capturing 1

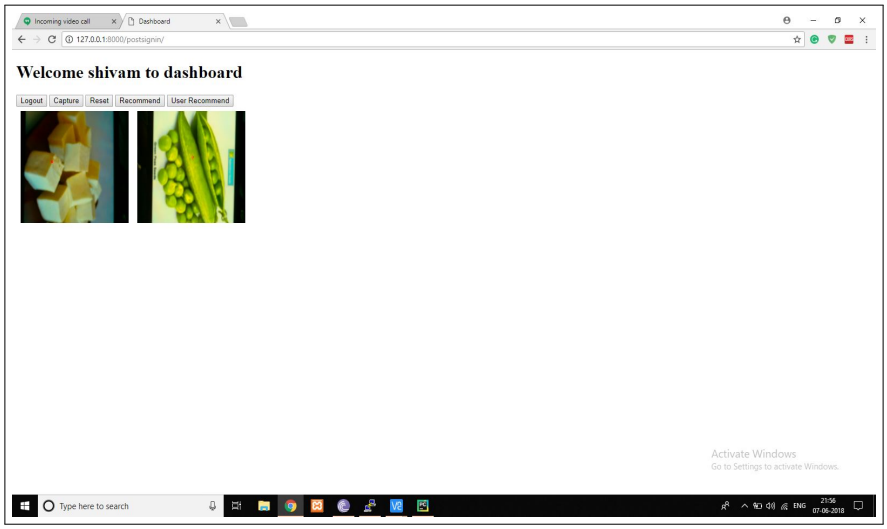


Figure 10.3: Image Capturing 2

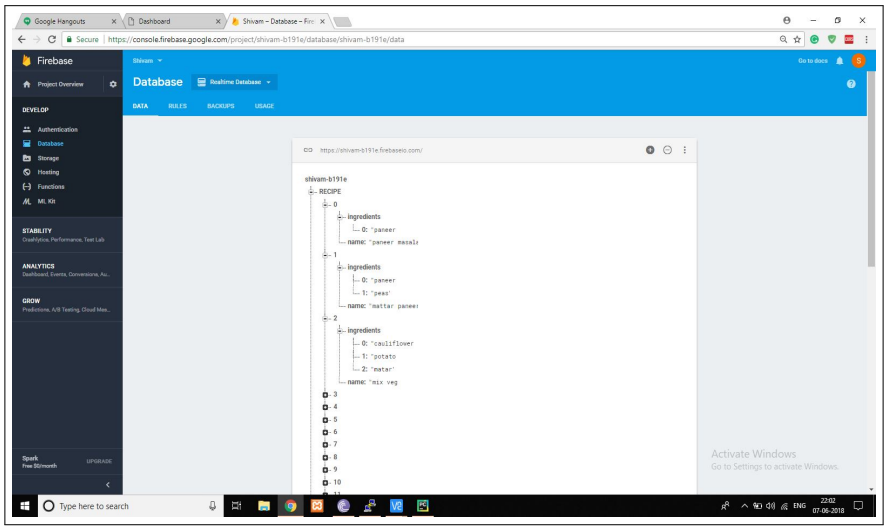


Figure 10.4: Recipe Database Structure

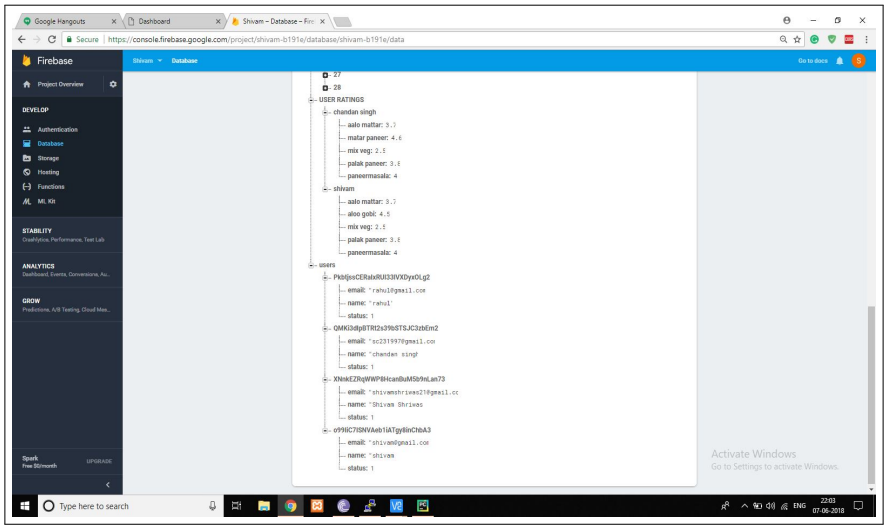


Figure 10.5: Users Database Structure

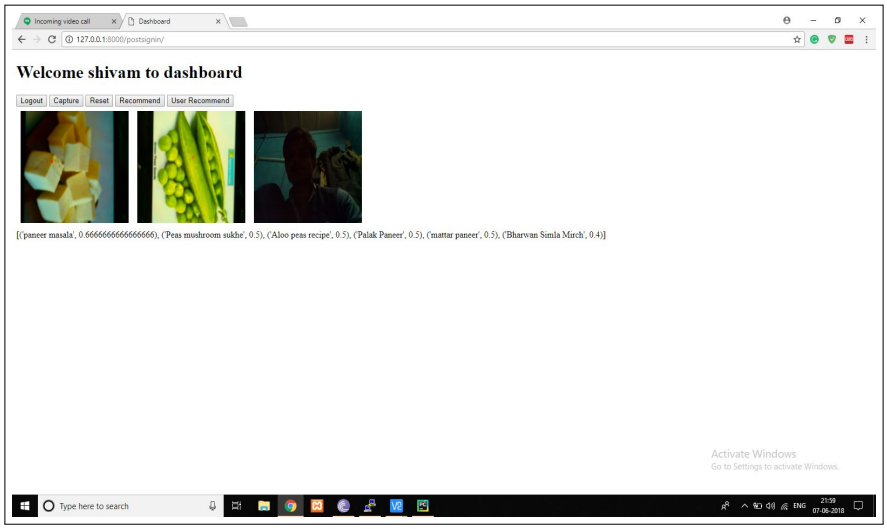


Figure 10.6: Content Based Output

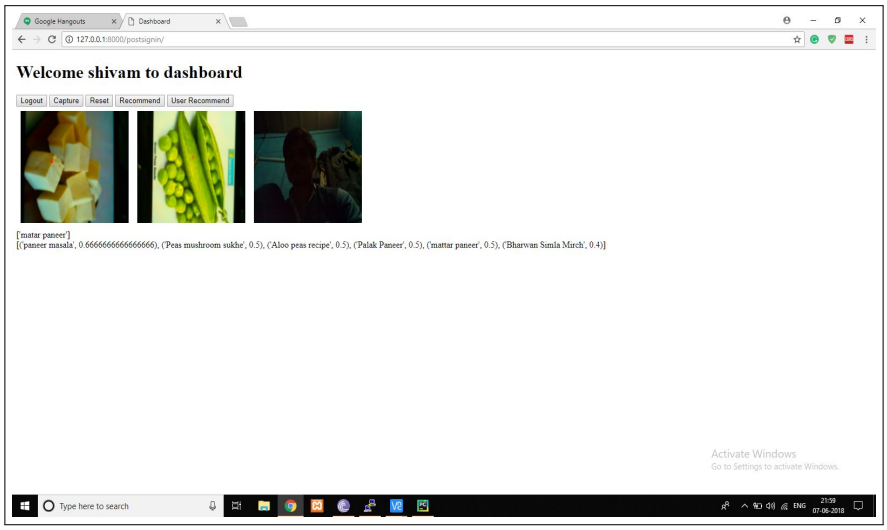


Figure 10.7: Collaborative Filtering Based Output

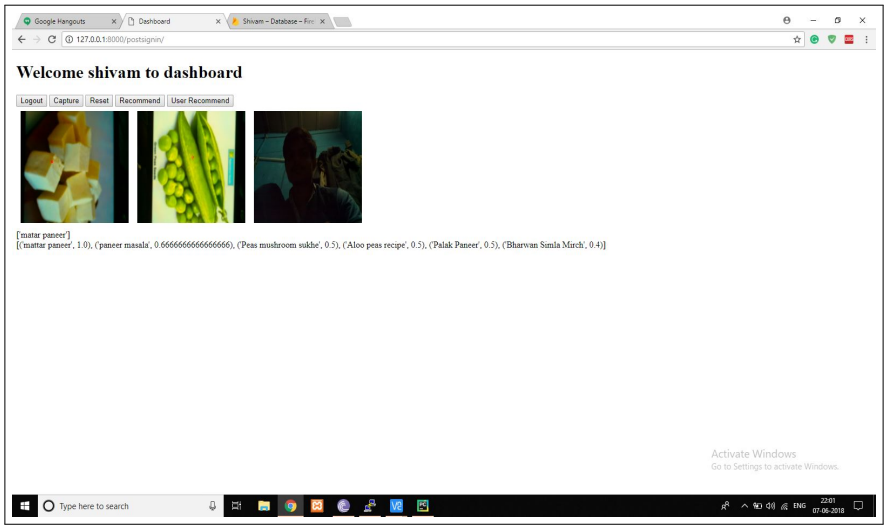


Figure 10.8: Combined Output

Deployment and Maintenance

Chapter 11

Deployment and Maintenance

11.1 Installation and un-installation

11.1.1 Enabling Camera On Raspberry Pi

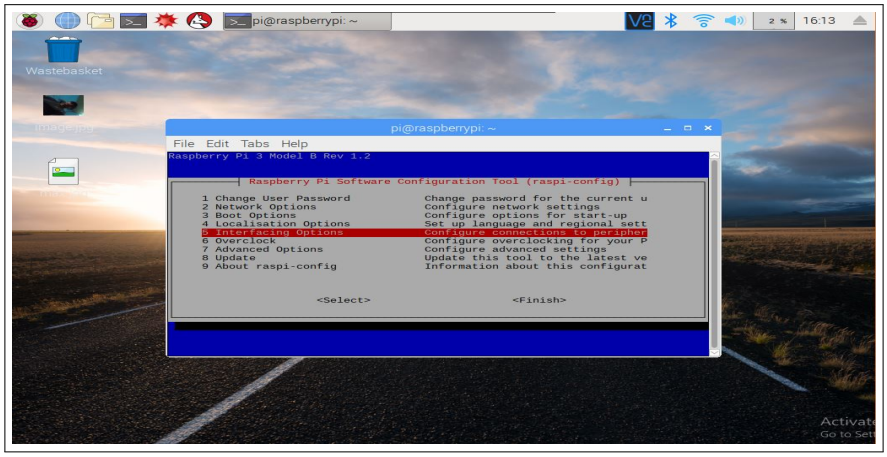


Figure 11.1: Raspberry Pi Interface Options

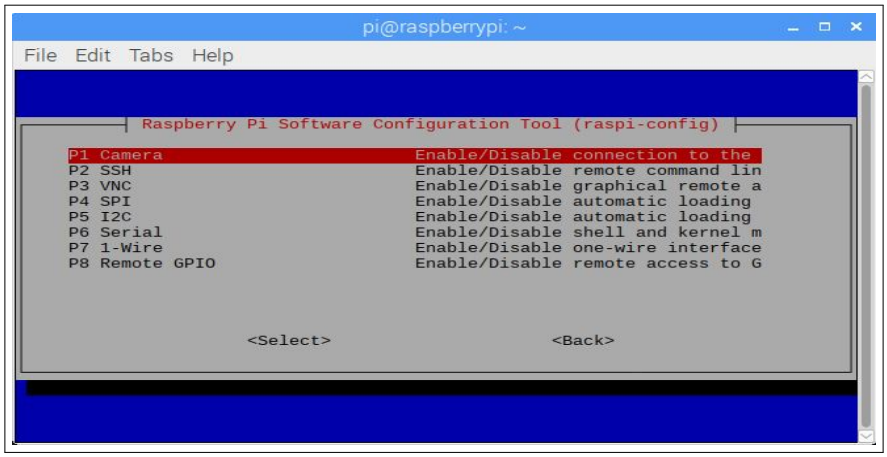


Figure 11.2: Camera Interface Selection

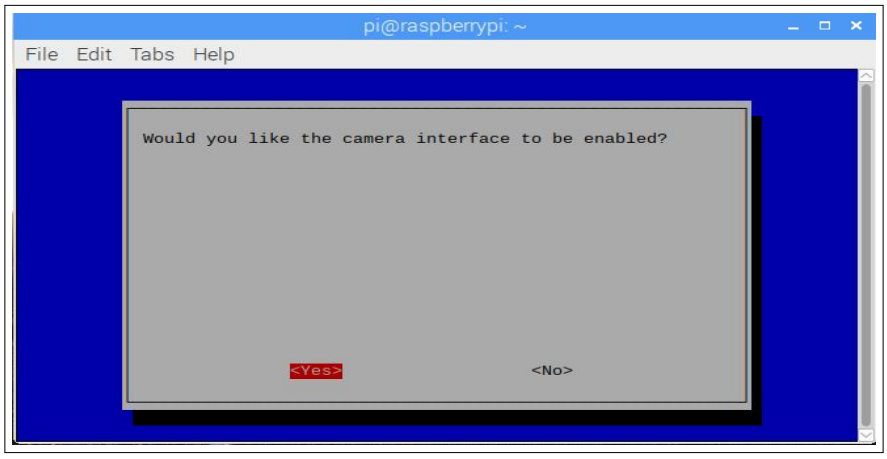


Figure 11.3: Camera Enable Confirmation

11.1.2 Flask Installation On Raspberry Pi

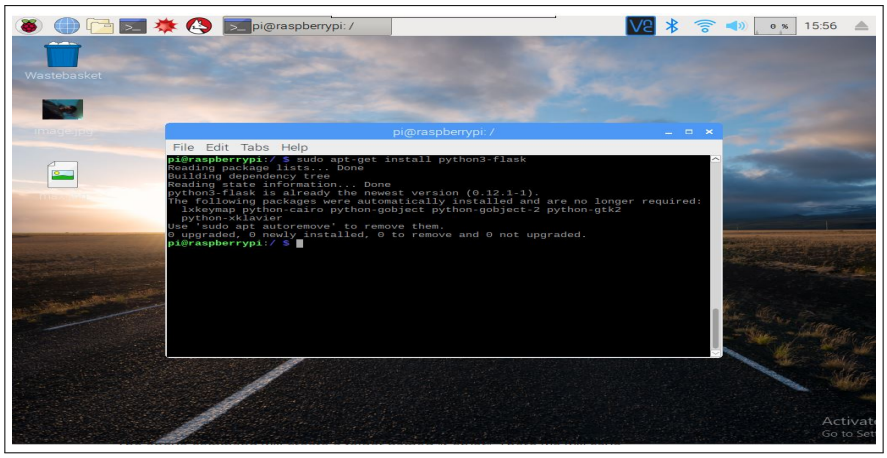


Figure 11.4: Flask Installation on Raspberry Pi

11.1.3 Pyrebase Installation On Raspberry Pi

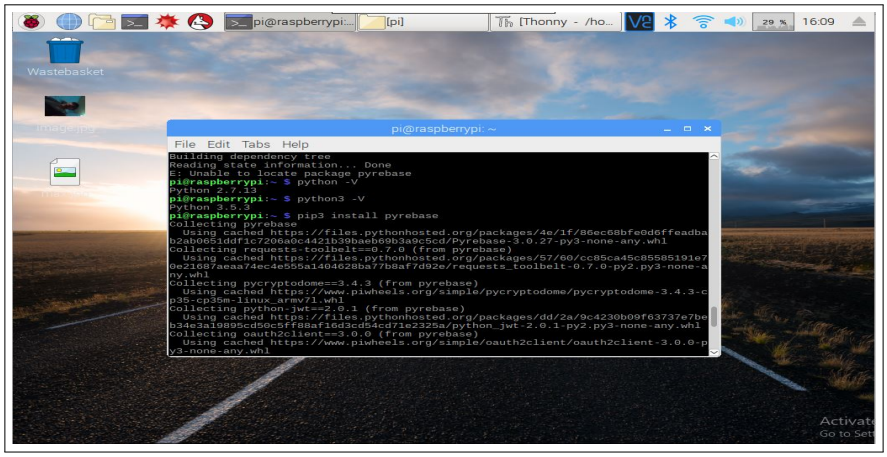


Figure 11.5: Pyrebase Installation On Raspberry Pi

11.1.4 PyCharm Packages Installation

Interpreter settings

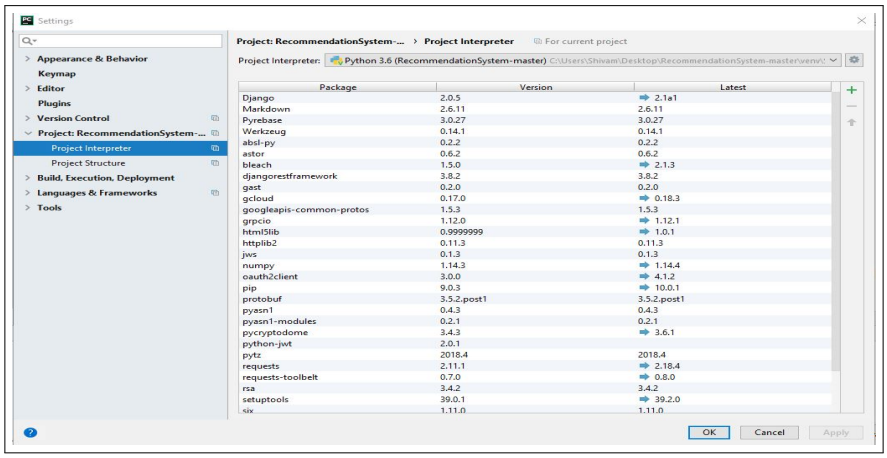


Figure 11.6: Interpreter settings

Django Installation

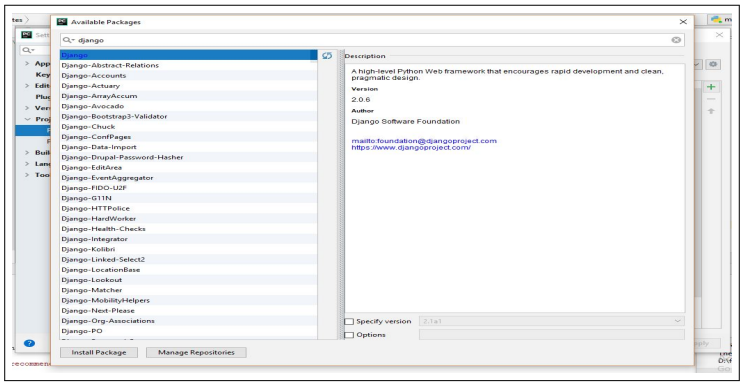


Figure 11.7: Django Installation

Django Rest Framework Installation

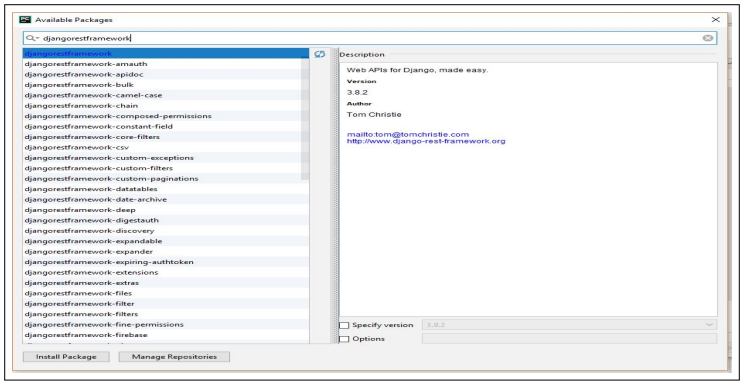


Figure 11.8: Django Rest Framework Installation

Pyrebase Installation

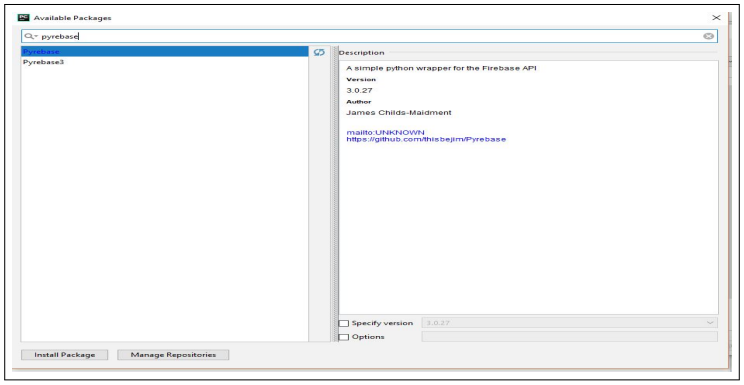


Figure 11.9: Pyrebase Installation

Tensorflow Installation

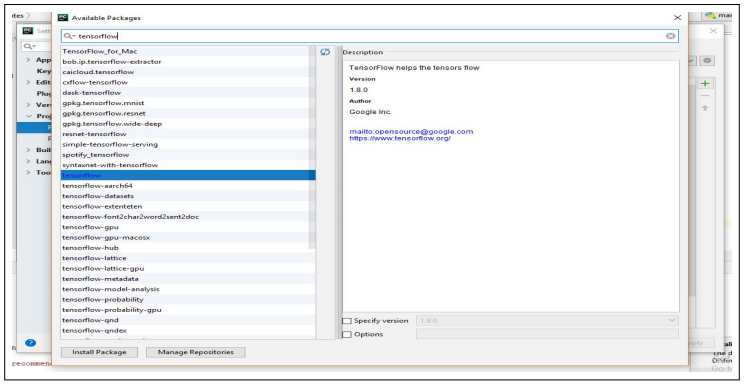


Figure 11.10: Tensorflow Installation

Conclusion and future scope

Chapter 12

Conclusion and future scope

12.1 Conclusion

There have been a lot of attempts to make a Smart Refrigerator but none of them had an option for recommending recipes based on the contents present inside the refrigerator. By implementing this project we have successfully achieved this task. We can conclude that, the system which is proposed here tries to overcome that disadvantage as well as also some of its own disadvantages such as accuracy of Visual Recognition, connection to the internet and many more. But, the system proposed here very well solves the problem statement.

12.2 Future Scope

The scope of the project and its functionality can be defined in following ways:-

- Use of multiple camera for identification of the vegetable.
- In single frame of image system can identify multiple vegetable present,which in current system identifies single vegetable from a single image.
- More collaborative results may be generated , by using user profiling and user history.
- various sensors may be deployed inside the refrigerator which may be used to regulate the temperature inside the fridge and change it accordingly.

Appendices

References

References

[1] An IoT-based Appliance Control System for Smart Homes Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang, Chengdong Li 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP) June 9 11, 2013, Beijing, China

[2] Vegetables detection from the glossary shop for the blind

[3] Review Paper on Vegetable Identification and Detection using Image Processing [Aashna Ahluwalia* and Ruhina Karani Computer Department, DJSCOE, Vile-Parle (W), Mumbai 400056, India Accepted 20 Nov 2014, Available online 30 Dec 2014, Vol.4, No.6 (Dec 2014)]

[4] A Content-aware Fridge Based on RFID in Smart Home for Home-Healthcare Hanshen Gu, Dong Wang School of Software, Shanghai Jiaotong University

[5] Smart Fridges with Multimedia Capability for Better Nutrition and Health [Suhuai Luo, Hongfeng Xia, Yuan Gao, Jesse S. Jin, and Rukshan Athauda School of Design, Communication Information Technology, The University of Newcastle, Callaghan NSW 2308]

[6] IoT Based Smart Refrigerator System [Deepti Singh 1, Preet Jain2 Electronics and Communication Department Shri Vaishnav Institute of Technology and ScienceIndore, India1, 2]

[7] A scalable recipe recommendation system for mobile application, ZhengXian Li, Jinlong Hu*, Jiazhao Shen, Yong Xu, School of Computer Science and Engineering, South China University of Technology, Guangzhou, China [2016 3rd International Conference on Information Science and Control Engineering]

[8] Recipe Recommendation Considering Flavours of Regional cuisines [Xuchui Mao1, Shizhong Yuan1, Weimin Xu1, Daming Wei2 1. School of Computer Engineering abd Science, Shanghai University, Shanghai 200444, China 2. Graduate School of Medicine, Tohoku University, Seiryo-machi, Aoba-ku, Sendai, Miyagi 980-8575, Japan]

[9] Intelligent Food Planning

[10] Implementation of a Goal-Oriented Recipe Recommendation Providing Nutrition Information [Tsuguya UETA, School of Techno-Business Administration Computer Science and Engineering, Nagoya Institute of Technology, Aichi, Japan Masashi IWAKAMI, Nagoya Institute of Technology Aichi, Japan Email: ueta@itolab.nitech.ac.jp Email: iwakami-almuni2011@itolab.jp Takayuki ITO Department of Computer Science, School of Techno-Business Administration Nagoya Institute of Technology Aichi, Japan Email: ito.takayuki@nitech.ac.jp]

[11] https://console.bluemix.net/docs/

[12] https://www.ibm.com/watson/

[13] https://www.ibm.com/watson/services/visual-recognition/

[14] https://www.ibm.com/watson/services/conversation/

[15] https://en.wikipedia.org/wiki/Raspberry_Pi

[16] https://www.upwork.com/hiring/data/what-is-content-based-filtering/

[17] https://www.upwork.com/hiring/data/how-collaborative-filtering-works/

[18] https://www.analyticsvidhya.com/blog/2015/10/recommendation-engines/

[19] https://www.raspberrypi.org/documentation/raspbian/

[20] https://www.realvnc.com/en/connect/docs/raspberry-pi.html#raspberry-pi-connect-direct/

[21] https://www.jetbrains.com/pycharm/

[22] https://towardsdatascience.com/

[23] https://www.djangoproject.com/

[24] https://www.fullstackpython.com/flask.html

[25] https://hackernoon.com/introduction-to-firebase-218a23186cd7

[26] https://firebase.google.com/

Laboratory assignments on Project Quality and Reliability Testing of Project Design

Appendix A

Laboratory assignments on Project Quality and Reliability Testing of Project Design

A.1 Type of Testing Used

1. **White box testing :** It is a testing technique, that examines the program structure and derives test data from the program logic/code. The other names of glass box testing are clear box testing, open box testing, logic driven testing or path driven testing or structural testing.

2. **Black-box testing :** It is a method of software testing that examines the functionality of an application without peering into its internal structures or workings. This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance.

3. **Unit Testing :** Unit testing focuses on the smallest unit of software design i.e. the smallest component or module. Important control paths are tested to uncover errors within the boundary of the module. It focuses on the internal processing logic and data structures within the boundaries of a component. This type of testing can be conducted in parallel for multiple components.

4. **Integration Testing :** Integration testing is a systematic technique for constructing the software architecture while at the same time conducting tests to uncover errors associated with interfacing. The different modules in our project were interfaced and tested in small increments, thus making the errors easy to isolate and correct. This is known as incremental integration.

5. **Validation Testing :** Validation testing begins at the culmination of integration testing, when individual components have been exercised, software is completely assembled as a package, and interfacing errors have been uncovered and

corrected. Here, testing focuses on user visible actions and user recognizable output from the system. Validation succeeds when the software functions in a manner that can be reasonably expected by the customer. In our project, all functions and performance characteristics are tested and they conform to the required specifications and are accepted.

6. **System Testing :** This is the final step in testing. In this phase, we tested the entire system as a whole with all forms, code, modules and class modules. This form of testing is known as Black Box testing or System testing. Black Box testing enables us to derive sets of input conditions that will fully exercise all functional requirements for a program. Black box testing helps to discover incorrect or missing functions, interface errors, errors in data structure, performance errors and initialization and termination errors.

A.2 Test Cases and Test Results

Sr. No.	Test Case	Expected Results	Actual Results	status
1	Login using invalid User-ID or Password	Rejected	Rejected	Pass
2	Login using valid User-ID or Password	Accepted	Accepted	Pass
3	Scan non vegetables	Reject	Reject	Pass
4	Scan untrained vegetable	Reject	Reject	Pass
5	Scan trained vegetable	Accept	Accept	Pass
6	Collaborative Recommendation for new user	Reject	Reject	Pass
7	Collaborative Recommendation for old user	Accept	Accept	Pass

Table A.1: Tests Performed

Project Planner

Appendix B

Project Planner

Table B.1: Project Plan

Sr. No.	Task Performed	From Date	To Date	Duration in Days
1	Identification Of Problem Statement	15-07-2017	30-07-2017	15
2	Abstract And Research On Topic	01-08-2017	15-08-2017	14
3	Project Overview	15-08-2017	05-09-2017	21
4	Literature Survey	06-09-2017	30-09-2017	24
5	Identifying Objectives, Scope and Goals	15-09-2017	05-10-2017	20
6	Coding of Modules	15-10-2017	10-12-2017	56
7	Documentation	20-11-2017	10-01-2018	51
8	Integration of Different Modules	11-01-2018	05-02-2018	25
9	Implementation of Work Done	06-02-2018	28-02-2018	22
10	Presentation of Project	15-02-2018	10-03-2018	23
11	Testing and Correction	11-03-2018	31-03-2018	20
12	Result Analysis	27-03-2018	10-04-2018	14
13	Documentation	25-03-2018	20-04-2018	26

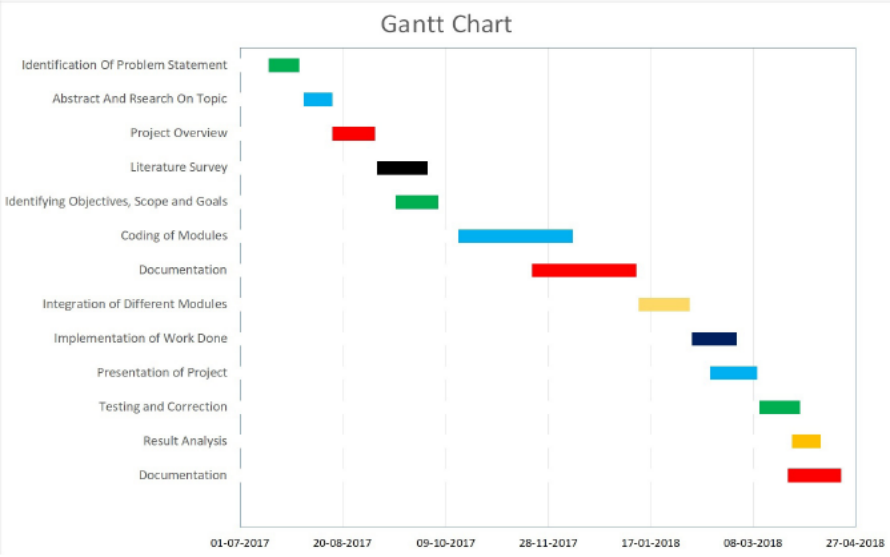


Figure B.1: Prepared Grant chart for Project Plan

Reviewers Comments of Paper Submitted

Appendix C

Reviewers Comments of Paper Submitted

1. **Paper Title:** Intelligent Refrigerator to Recommend Recipes Using Hybrid Recommendation
2. **Name of the Conference/Journal where paper submitted :** NCETCET18
3. **Paper accepted/rejected :** Accepted
4. **Review comments by reviewer :** Use a good Algorithm for recommendation and use server so as to reduce processing load on Raspberry Pi.
5. **Corrective actions if any :** Used new algorithms which were faster and efficient and added a server for processing everything.

Plagiarism Report

Appendix D

Plagiarism Report

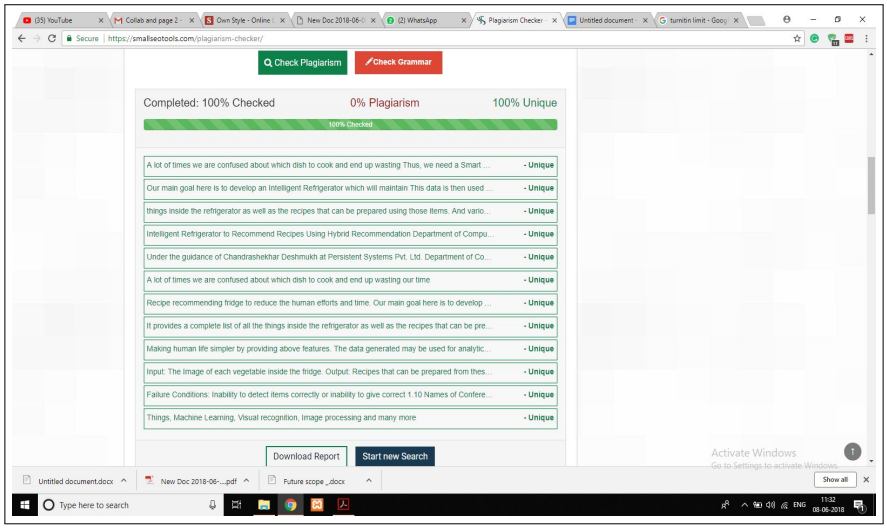


Figure D.1: Plagarism Report 1

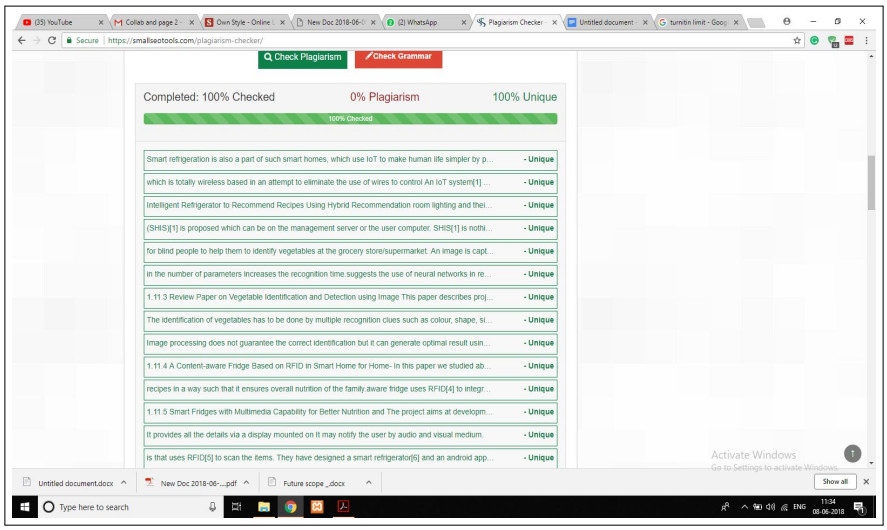


Figure D.2: Plagarism Report 2

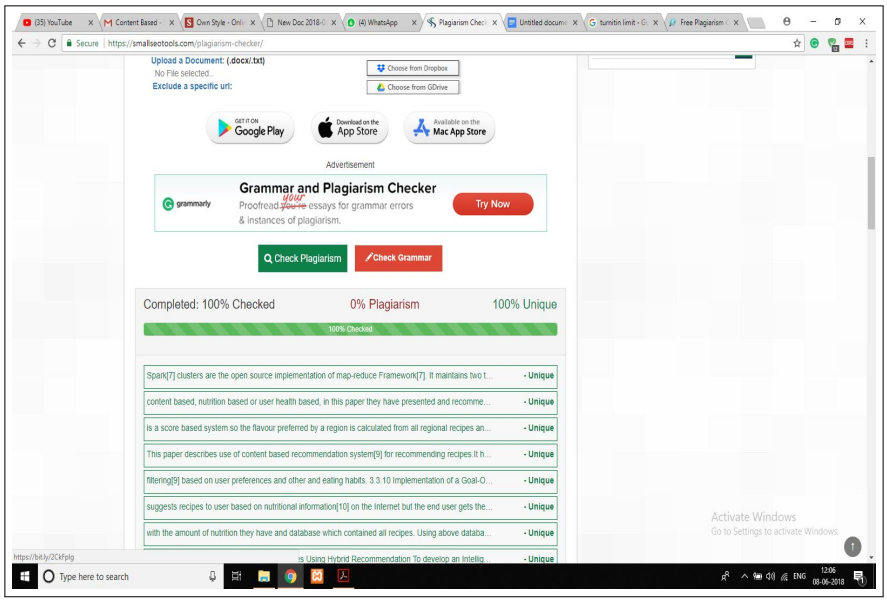


Figure D.3: Plagarism Report 3

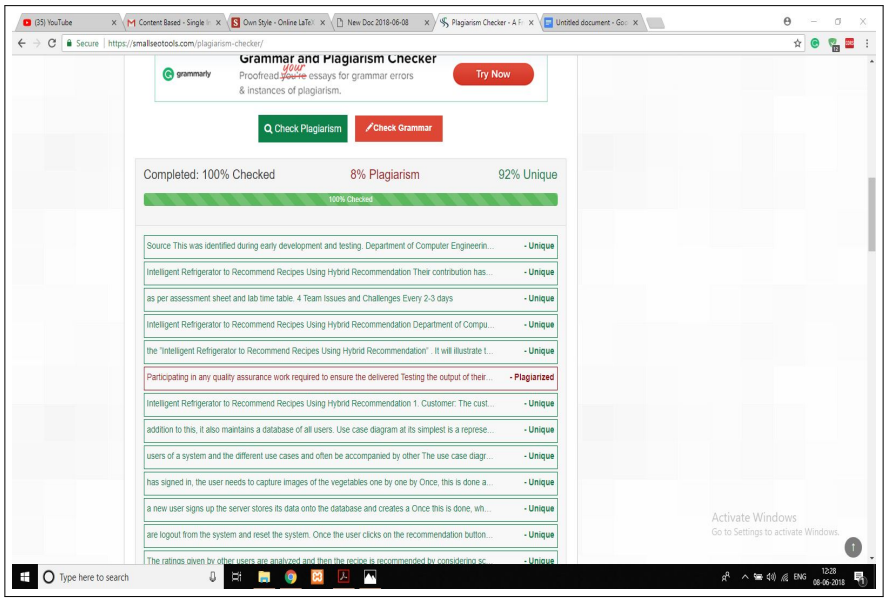


Figure D.4: Plagarism Report 4

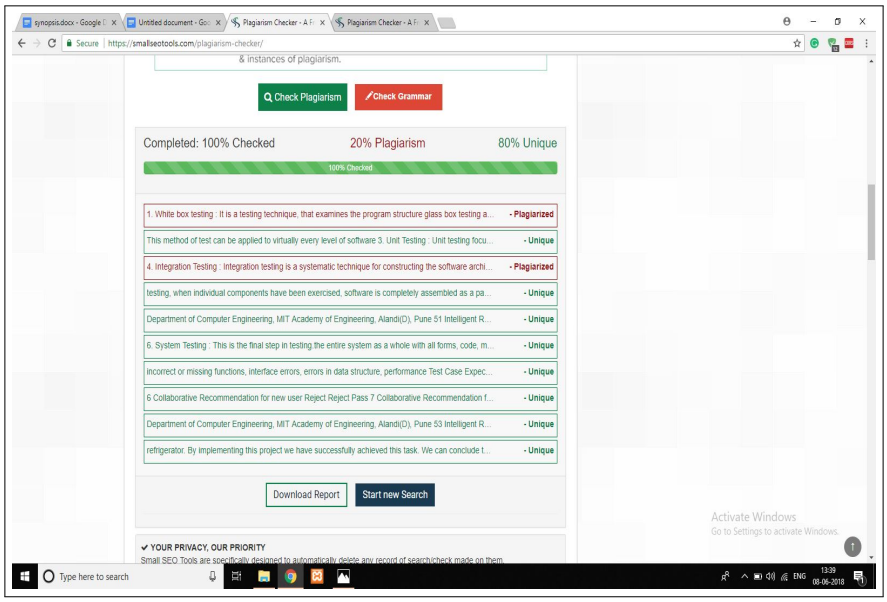


Figure D.5: Plagarism Report 5

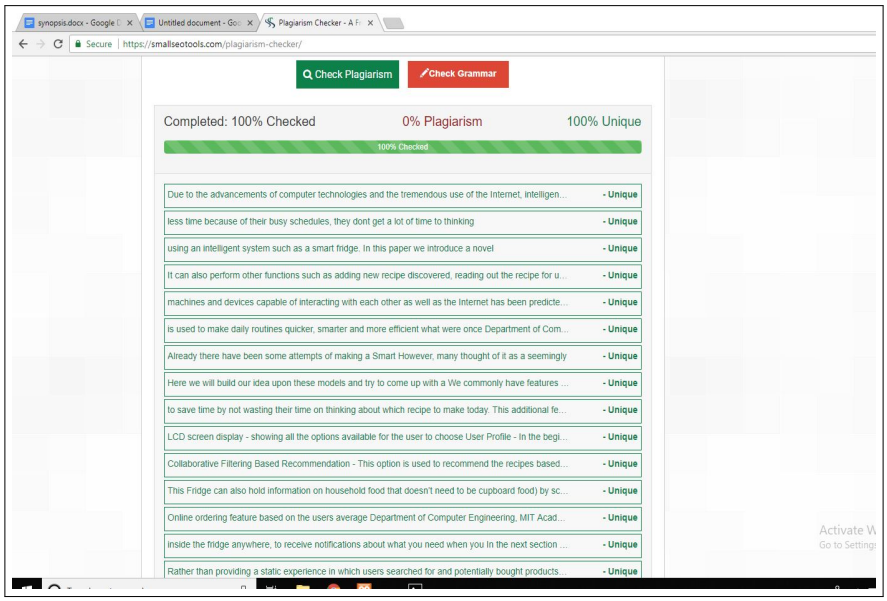


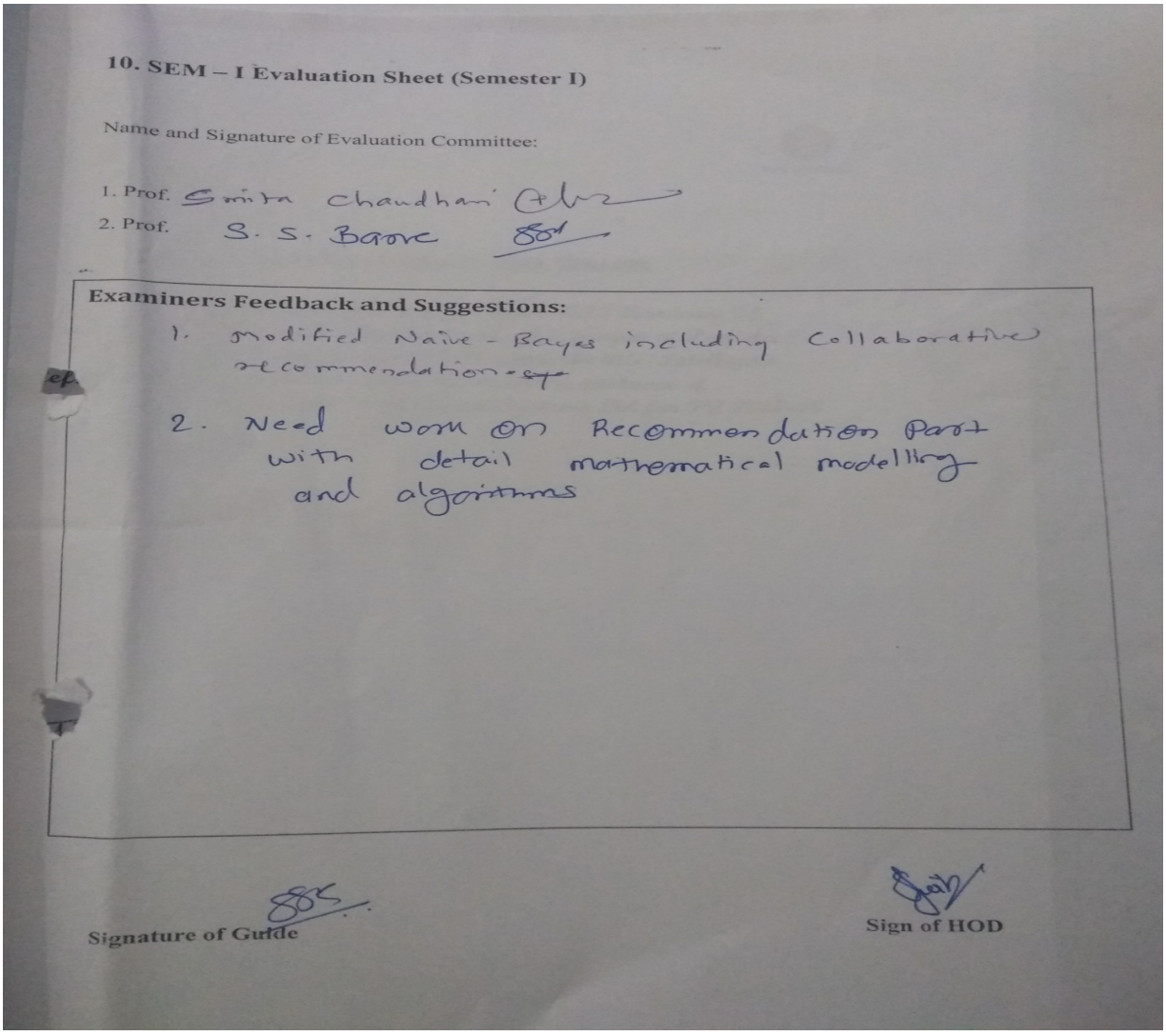
Figure D.6: Plagarism Report 6

Term-II Project Laboratory Assignments

Appendix E

Term-II Project Laboratory Assignments

1. Review of proposed design and necessary corrective actions taking into consideration the feedback report of Term I assessment and other competitions/conferences participated.



10. SEM – I Evaluation Sheet (Semester I)

Name and Signature of Evaluation Committee:

1. Prof. Smita Chaudhari
2. Prof. S. S. Barve

Examiners Feedback and Suggestions:

1. Modified Naive - Bayes including Collaborative recommendation - ef
2. Need work on Recommendation Part with detail mathematical modelling and algorithms

Signature of Guide

Sign of HOD

Figure E.1: Certificate of Merit

Feedback Received:

(a) Modification in the Naive-Bayes algorithm or use another algorithm for collaborative recommendation.

(b) More work on recommendation part.

Corrective Actions Taken:

Used Pearson Correlation Algorithm which is used to measure the strength between users and relationship between multiple users.This algorithm provides more accurate results than Naive-Bayes. Naive-Bayes works on the principle that the presence of any particular feature in a class is unrelated to the presence of any other feature. But Pearson Correlation algorithm finds all the similar features in a class. That means Pearson algorithm finds all the similarities between the users and generate results based on similarities. Pearson Correlation Formula gives value from -1 to +1.The negative value suggest that there are no similarities between classes or users.On the other hand the positive value suggests that there are similarities between users. So based on the matched features it generate correlation value from -1 to +1.

2. Project workstation selection, installations along with setup and installation report preparations.

E.0.1 Enabling Camera On Raspberry Pi

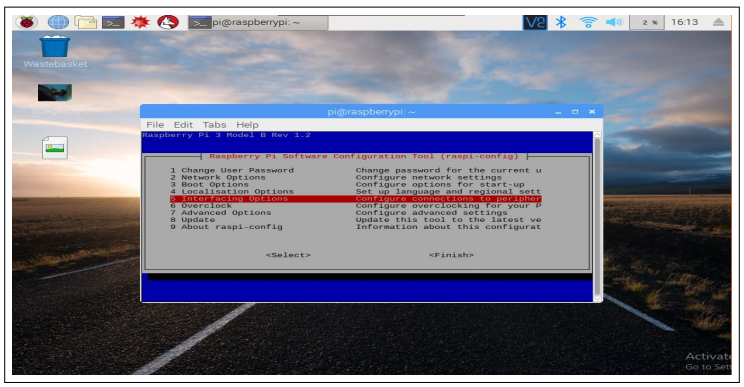


Figure E.2: Raspberry Pi Interface Options

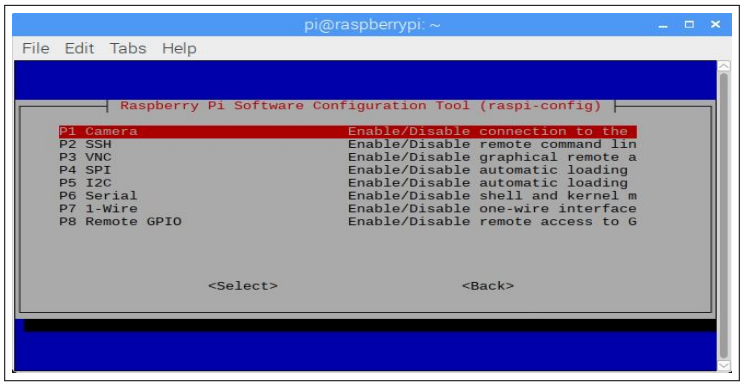


Figure E.3: Camera Interface Selection

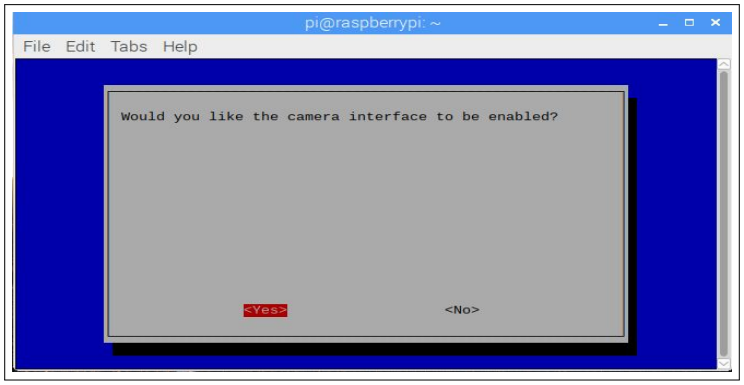


Figure E.4: Camera Enable Confirmation

E.0.2 Flask Installation On Raspberry Pi

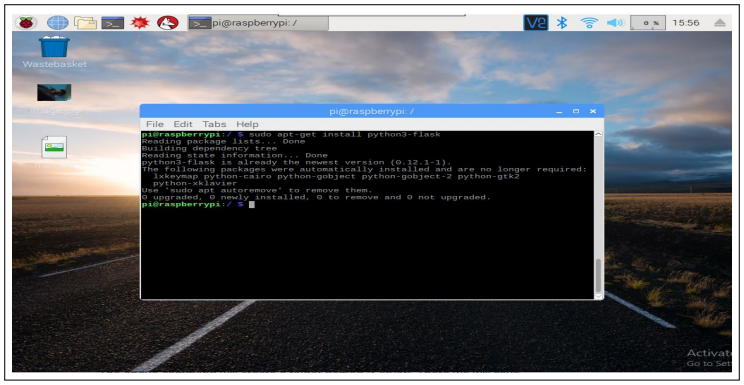


Figure E.5: Flask Installation on Raspberry Pi

E.0.3 Pyrebase Installation On Raspberry Pi

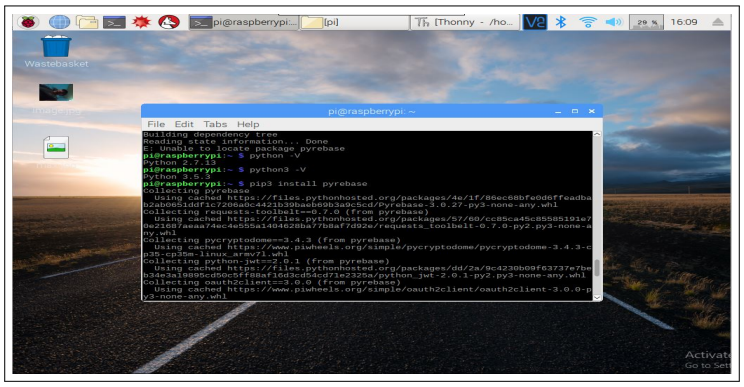


Figure E.6: Pyrebase Installation On Raspberry Pi

E.0.4 PyCharm Packages Installation

Interpreter settings

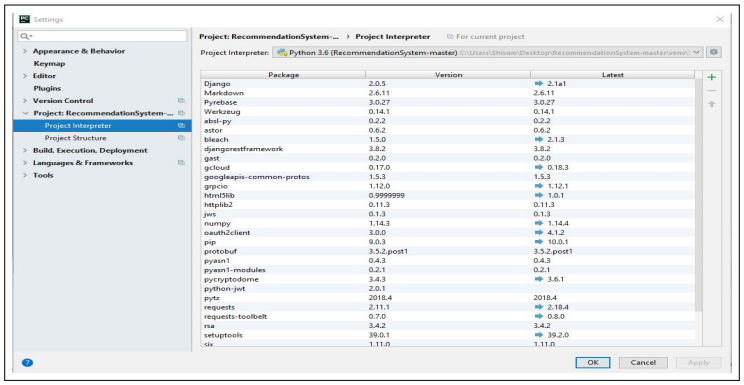


Figure E.7: Interpreter settings

Django Installation

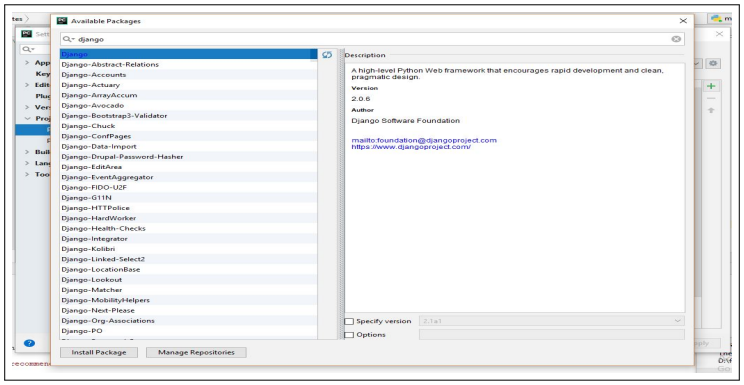


Figure E.8: Django Installation

Django Rest Framework Installation

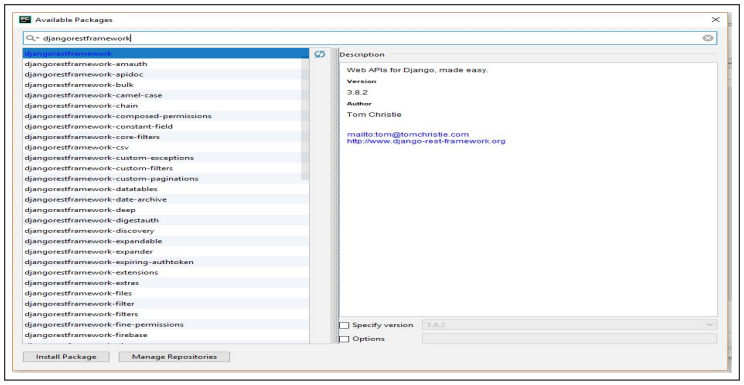


Figure E.9: Django Rest Framework Installation

Pyrebase Installation

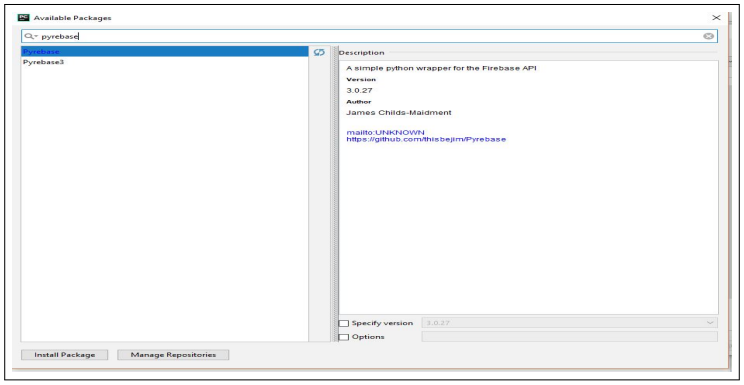


Figure E.10: Pyrebase Installation

Tensorflow Installation

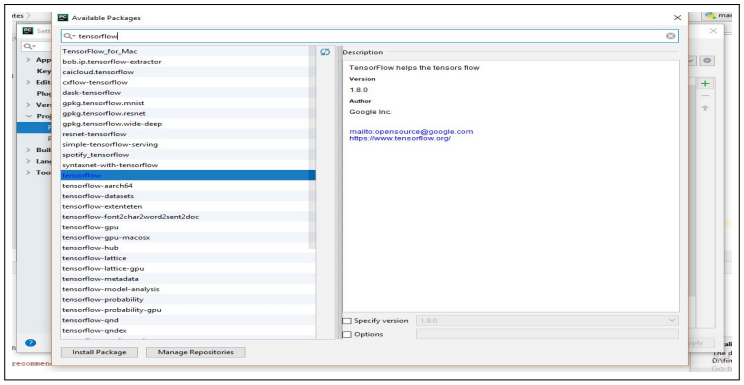


Figure E.11: Tensorflow Installation

3. Programming of the project functions, interfaces and GUI (if any) as per 1 st Term term-work submission using corrective actions recommended in Term-I assessment of Term-work.

   - We implemented the Collaborative filtering algorithm of Pearson Similarity as suggested in the first semester. The results are given below.

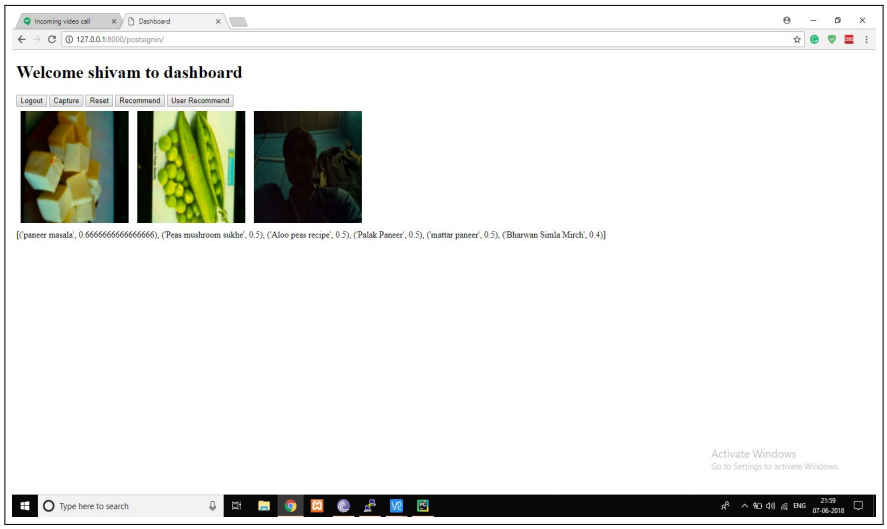


Figure E.12: Content Based Output

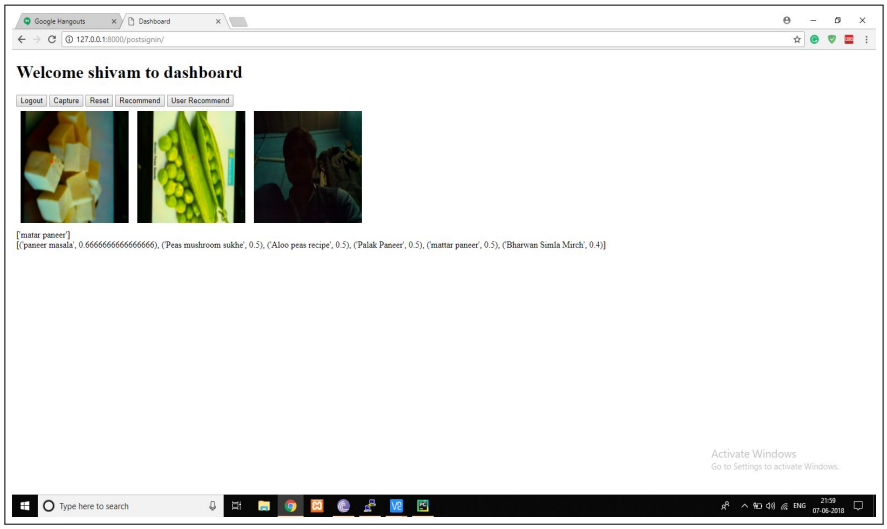


Figure E.13: Collaborative Filtering Based Output

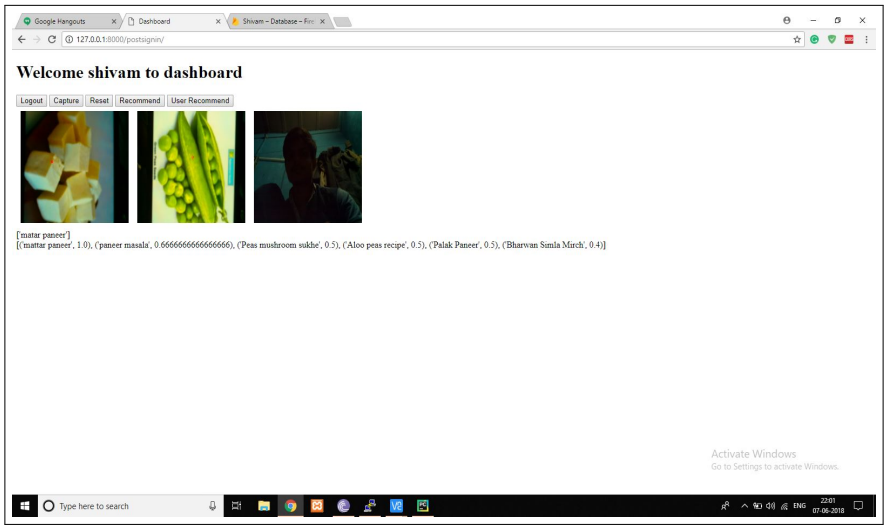


Figure E.14: Combined Output

4. Reliability Testing Reports at the client end.

Test Cases and Test Results

Sr. No.	Test Case	Expected Results	Actual Results	status
1	Login using invalid User-ID or Password	Rejected	Rejected	Pass
2	Login using valid User-ID or Password	Accepted	Accepted	Pass
3	Scan non vegetables	Reject	Reject	Pass
4	Scan untrained vegetable	Reject	Reject	Pass
5	Scan trained vegetable	Accept	Accept	Pass
6	Collaborative Recommendation for new user	Reject	Reject	Pass
7	Collaborative Recommendation for old user	Accept	Accept	Pass

Table E.1: Testing Results

Appendix F

Information of Project Group Members

1. **Name :** Rahul Golhar
   Date of Birth : 20th Sept 1996
   Gender : Male
   Permanent Address : A 701, Savannah, BAIF Road, Wagholi, Pune-412207
   E-Mail : golhar.rahul@gmail.com
   Mobile/Contact No. : 8805425520
   Paper Published : Intelligent Refrigerator to Recommend Recipes Using Hybrid Recommendation

2. **Name :** Shivam Shriwas
   Date of Birth : 21st June 1996
   Gender : Male
   Permanent Address : E-35,Shastri Nagar near, Indira Gandhi school, opp. Sudarshan stationery, Pimpri, Pune-411017
   E-Mail : shivamshriwas21@gmail.com
   Mobile/Contact No. : 7040108815
   Paper Published : Intelligent Refrigerator to Recommend Recipes Using Hybrid Recommendation

3. **Name :** Manmohan Shinde
   Date of Birth : 07th Oct 1996
   Gender : Male
   Permanent Address : Ashirwad Niwas, Shrikrishna Nagar, Near Balaji temple, Bhokar,Nanded-431801
   E-Mail : shindemanmohan77@gmail.com
   Mobile/Contact No. : 8446494441
   Paper Published : Intelligent Refrigerator to Recommend Recipes Using Hybrid Recommendation

Appendix G

Sponsorship Details

PERSISTENT

August 17, 2017

To Whomsoever it May Concern

This is to certify that following students from MIT Academy Of Engineering are undergoing their final year B.E. project at Persistent Systems Ltd. for academic year 2017-18 under the title 'Intelligent Refrigerator using Machine Learning' under the guidance of Chandrashekhar Deshmukh at Persistent Systems Ltd for FY 2017-18

Name of Students:

i. Rahul Golhar
ii. Manmohan Shinde
iii. Shivam Shriwas

For Persistent Systems Ltd.

Mahesh Gosavi
Senior Manager - Human Resource

Figure G.1: Sponsorship Letter

Appendix H

Publication Details

Figure H.1: Certificate of Merit