College Certificate

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
JSS Technical Institutions, Mysuru - 570 006

Title of Course: Software Engineering
Subject Code: CS530
Semester: 5

Student Names and USN:

USN	Name
01JST17CS004	Adishi Jha
01JST17CS019	Anirudh Pai
01JST17CS051	Donapati VS Reddy
01JST17CS052	Eeshaan Achar
01JST17CS097	Navneet Hosmane

Internal Assessment Marks awarded: .....

....

Head Of Department Sign of Teacher University Seal

Declaration

We, Adishi, Anirudh, Sreekar, Eeshaan and Navneet students of 5th Semester B.E, Computer Science and Engineering at JSS Science and Technology University, Mysuru hereby declare that this group activity was carried out by our group and this report was prepared by us as a part of the course work CS530-Software Engineering under the guidance of Prof. Dr. Trisila Devi Nagavi.

Project Supervisor:

Name: DrTrisiladevi C. Nagavi

Signature : .....

Date: .....

Acknowledgement

We would like to express our special thanks of gratitude to our teacher Dr. Trisiladevi C Nagavi, who gave us the golden opportunity to do this wonderful project on the topic "JCE Foodie", which also helped us in doing a lot of research and we came to know about so many new things, got to meet people from various fields and their interests and opinions with regard to the project. At last we would like to thank all the other people who were involved directly or indirectly in collecting the requirements. We also enjoyed working as a team and are looking forward for implementing it successfully.

- Adishi Jha
- Anirudh Pai
- Eeshaan Achar
- Navneet Hosmane
- Donapati VS Reddy

Abstract

Deciding a cafe for having food is always a menial task because until now there was no way to compare between two cafes. If there was a way to show the popular choice to the public, then they can benefit from it. One of the major problems is the waste of time and money in the process of finding what one wants to eat. To provide a solution for this, we came up with an application that acts as a platform where people can pick one cafe for a food item over the other and contribute to it's overall ranking. Additionally, the preparation time and cost price is also added in case that is a factor of decision. Teachers and students can use this app to learn the best rated cafe for a particular dish. The entire process of creating this application - from requirements collection to the testing phase has been documented in this report.

Team Details

Figure 1: The Team

Name: Adishi Jha
Usn: 01JST17CS004

Name: Anirudh Pai
Usn: 01JST17CS019

Name: Donapati VS Reddy
Usn: 01JST17CS051

Name: Eeshaan Achar
Usn: 01JST17CS052

Name: Navneet Hosmane
Usn: 01JST17CS097

Contents

Part I
Introduction

1 Introduction to the problem domain

Being a student in a professional college with 5-6 hours of classes everyday on average, hunger or just the craving to eat is something that bothers the otherwise energetic young minds.

However, making a decision of where to eat from has always been a pain to students. Be it selecting the cafe to eat at or the dish to have. In our college SJCE-JSSSTU, we have so many cafes, canteens and bakeries that we face the dilemma of picking a place. We have Yampa, LKB, Cafe Frappe, Cafe Khushi, Mylari and Nescafe and the wide variety of food that they present. There are dosas, meals, burgers, sandwiches, milkshakes, machurians, noodles, etc.

The deciding factors in this process can be the type of dish(breakfast item, milkshake, snacks) , quality, amount of serving and price. The new students like to know what to savour before a long walk to the cafe. Customers that have tried different places, must be able to put forth their opinion about the better option. Also the absence of an online menu for all the cafes keeps the majority of new students in the dark. So, to provide them the light, an online platform that compares the common dishes between two different cafes or canteens is needed.

2 Aim of the problem

i. To come up with an app to help students with decision making in food selection.

ii. To help the students find the cafe or restaurant of their choice on Google maps.

iii. To give the students preparation time and cost details.

iv. To give existing customers a platform to present their comparison to the others.

v. To help cafes and restaurants to improve their products and services.

3 Objectives

The app basically has three users. The customers, i.e. students and teachers, and the cafe managers. The app helps the customers in making food selection. This in turn leads to saving time and money which is usually wasted in researching. The customers just have to login to the app through Google or their email. Using the app, customers can look at the top rated cafe for a particular type of food. Then they can locate the place on Google Maps which will navigate them to their destination. After having the same food in two or more places, any customer can select one cafe for a product over the other. This is a process simpler than having to rate a place out of 5 stars which leads to clashes most of the time.

The cafe managers or admins have the rights to enter the preparation time and cost of a food item. Only trusted people will be given these rights as the data must be accurate. This will allow the users to go for their option based on either popular choice or the cost and preparation time.

Part II
Requirements Engineering

Requirements Engineering is the process of defining, documenting and maintaining the requirements. It is a process of gathering and defining service provided by the system.

This is critically important for creating accurate results in software engineering. Requirements Engineering is also known as Requirements Analysis.

Objectives :

i. To describe the principal requirements engineering activities and their relationships.

ii. To introduce techniques for requirements elicitation and analysis.

iii. To describe requirements validation and the role of requirements reviews.

iv. To discuss the role of requirements management in support of other requirements engineering processes.

Requirements Engineering is divided into seven phases :

i. Inception

ii. Elicitation

iii. Elaboration

iv. Negotiation

v. Specification

vi. Validation

vii. Requirements Management

4 Inception

As a fresher, it can be overwhelming to find so many dining options in the college, whether it tastes good, if it's delicious, if it's expensive etc. Therefore, in this digital era we need a software solution to this problem. This solution must make it easier for students, and people in general to find different dishes and cuisines with appropriate ranking and other information such as the quality, quantity, cost etc.

4.1 Understanding the problem

Our college has students coming in from various parts of the country who have a variety of tastes, right from the saffron of Kashmir to the banana chips of Kanyakumari. As a fresher who is just getting used to the hostile environment of the hostel, not being able to enjoy the tastes of their hometown can seriously suppress their spirits. Therefore, it would be very useful to them if there was an easy way for them to know what foods are being served, where and for how much so that they can try them out and feel at home, at least in terms of food. We also have many other students, faculties and the general public who could use some help with the dining options in the form of a food guide.

4.2 People that want a solution

The target users for this software will be freshers who have little to no idea on the food options available across the campus and also around the campus. However, considering the large number of students studying in the college, there are many who could use a food guide to know where to find what food, the quality and quantity available, and also know about the rush/peak hours when it takes longer than normal to get the food. Moreover, we have many faculties who have their lunch/snacks in one of the college restaurants, and we also have people from outside living near the college who would be interested in exploring the food options available in the region.

4.3 Nature of the solution

The solution to the mentioned problem can be realized in the form of an android based application (which can then be extended to the iOS Platform) that the users would have to install on their phones, which will provide them with all the details regarding the food options in and around JSS Science and Technology University. The rankings of the food will be based on the collective opinions by the user group which includes students, faculty members (teaching and non teaching) and also other outsiders who want to try out the delicacies of JSS S&TU. At any given time, the user can open the app, look at the recommendations or the rankings for any particular dish, and then decide if it's a good time to visit based on the rush hour information that will be provided.

4.4 Communication and collaboration

Due to the extensive nature of the problem that this system overcomes, anyone with a plan to dine in and around the campus is a potential client or stakeholder. To obtain a more specic and realistic approach towards obtaining requirements we reduced our target audience to three specific groups:

i. Students: The main inspiration to start with the project was the problems faced by freshers, especially the ones who reside in the hostel where good food is scarce. Students in their second, third and fourth years, are the people that do have a good idea about the various food options available in and around the campus. They form the backbone of the rating system as they can provide valuable insights to the feedback system.

ii. Faculties: They don't get time to try all food items avaliable in the restaurants. They can try various food once the ratings are available.

iii. Others: With the college canteens open to everyone, and also with the inclusion of restaurants outside the campus, the general public thats interested in trying out any of these food chains make for stakeholders.

Our team was in communication with all three clients and prepared a set of general questions for each of these groups along with some specic questions for each set. These questions were structured in order to obtain the maximum possible information from the users. The elicitation (section 2) contains more detailed interviews with clients.

5 Elicitation

The following questions were structured to obtain all the required information from the target audience. It is to be noted that this set of questions are standard and were answered and veried by each customer. The following section contains the questions along with the paraphrased responses of the user.

5.1 Questionnaires

5.1.1 Students

This set of questions was answered by Vaibhav (EEE, 1st semester), a student of JSS S&TU. The interview was conducted in the lawn infront of Admin block. He lives in Shivarathri hostel because seniors told him food isn't good in the college hostel. He travels 4km just for the sake of food. His requirements can be taken as the standard for all the students who are new to college.

Did you face any issue while choosing what to eat?

I was clueless about what to eat when I joined the college. I ate samosa in Yampa and burger in Cafe Frappe. I don't have an idea about the best food items in the college. It takes me around 15 minutes to ask people around.

Are you satisfied with the recommendations?

No, I am not satisfied with the recommendations. My friends are new to college and they don't know much about food served in various canteens. I don't go out often but whenever I go, I try new food item every time. If I don't make a correct choice, time and money gets wasted.

If our application was available to you right now, how would you use its features? Does the application help in simplifying tasks?

If it is possible to get honest reviews on your application, I would use the system to decide which food item to choose. Reviews about quantity and quality of food will help all freshers in investing their money in the right food item. Mentioning waiting time will help us plan our day easily. Currently, I don't go anywhere in 10:30- 11:00 break because I may be late for my class.

Can you think of any reason why our product would fail to deliver on performance or otherwise?

I have no specific idea about this. One issue can be that people may not have tried enough food items from all the canteens. Another issue would be people may not be honest with their reviews.

We also met another student, Preetham (MECH, 1st semester) and asked him the same set of questions. He gave similar answers. He lives in the campus hostel. He also told us that dinner isn't served on Sundays and he would like recommendations regarding the same. This interview was conducted in the second floor of Admin Block.

5.1.2 Faculties

This set of questions were answered by our teachers in JSS S&TU. Interview was conducted in the Staff room of Golden Jubilee Block. Their requirements can be taken as the standard for all the teachers in our college who have time constraints.

Did you face any food related issue?

We have a hectic day because of meetings and classes. We can't try new food items because waiting time for food isn't fixed. We really don't have extra time to decide among various food items. Therefore, we eat the same food every time.

Do you ask other people for recommendations?

We have a busy schedule, therefore none of us gets time to try different food items. Hence, we can't get suggestions.

If our application was available to you right now, how would you use its features? Does the application help in simplifying tasks?

If it is possible to get honest reviews on your application, I would use the system to decide which food item to choose. I can try new food items. Since, waiting time is specified I can make wise decisions.

Is there any other feature that we can add to our product?

In this establishment there is nothing. If you are considering going to all restaurants in Mysore, then you might have to add more features that include mode of payment, distance, ambience and other details about the food.

5.1.3 Restaurant Owners

This set of questions were answered by the owner and manager of Cafe Frappe. This Cafe specializes in fast food and North Indian Cuisine. The interview was conducted in the same restaurant. Her requirements can be taken as the standard for all the restaurants.

What are the peak hours in your restaurant?

This place is usually crowded in the afternoon, 1pm- 3pm. We have additional staff during that time so that everyone gets food on time. We cater to everyone's need.

If our application was available to you right now, how would you use its features? Does the application help in simplifying tasks?

It will help me improve the food items served in my canteen. I can prepare the bestseller food items in advance to save time. In addition to that,I can change the ingredients of the food items not liked by the students. I can get feedback from both students and teachers. This product will help me provide good food items to everyone.

How many people can you accommodate at one time and at what time food is available?

I can accommodate around 50 people. Food is served from 10am- 8pm. We are trying to expand our place to accommodate more people.

We asked the same set of questions to the owner of Yampa, and we got a similar response. He also told us that lately he has been seeing decline in the sales of shakes, and would like to improve upon it if the application would help him. This interview was conducted in his canteen.

5.2 Association of requirements

From the above questionnaire we have collected all the necessary requirements and created a list to attribute all the features to their source.

i. Students: Honest reviews about food items, quality, quantity, waiting time, price, regular updates, food delivering rate

ii. Faculties: Honest reviews about food items, quality, quantity, waiting time, regular updates, food delivering rate

iii. Restaurant owners: Feedback from customers, regular updates

6 Elaboration

From the information collected via the questions in the previous sections, we were able to create a refined model containing the basic functions that the system will implement, the interactions between the users and the task management system as well as the overall association between each module. We also came up with the list of features to implement.

6.1 Developing Use Cases

The first step is to identify the set of actors that are involved. This will include different people that use the system or product within the context of the project. Actors represent the roles that people play as system operators.

Actors identified

i. **Customers** (who will rate and will check out the results)

ii. **Admins**(who will update the data related to ordering food)

6.2 Customer

For the actor, customers (who will rate and see the results):

i. **Rating**

   a. **Use Case:** Picking one food item over the other

   b. **Primary Actor:** Student or Teacher that is a customer to the cafes

   c. **Goal in context:** To be able to pick one food item between two cafes that is better in the actor's point of view

   d. **Preconditions:** The primary actor has been provided with credentials to login to the app. The credentials are set during registration (email and password)

   e. **Scenario:**

      . The user opens the application and selects the login button to go to the login screen

      . The user then enters their unique credentials, i.e. email and password, to login to the app.

      . The user picks one of the eight different food items (burger, sandwich, oreo shake, meal, etc) and reaches the rankings page.

      . The user then clicks on the 'Rate Now' button to enter the screen to select between two displayed options where they select one of the two given options or skip the choice in case they haven't tried both the places.

      . The choice made by the user is saved into the database.

Exceptions:

. Password is incorrect, user re-renters correct password.

. Password not recognised, technical team is contacted to reprogram the password.

. Rating is showing an error, technical team is contacted for feedback and repairs are made.

f. **Frequency of use:** any number of times per day

g. **Channel to actor:** via the main application user interface

h. **Secondary actors:** technical team

i. **Channels to secondary actors:** phone line, text messages, emails and other online messages.

j. **Open issues:**

. Should there be additional factors to select one cafe for a food item?

. How many tries are allowed to enter the right password?

ii. **Ranking**

a. **Use Case:** Viewing the ranking for a food item

b. **Primary Actor:** Student or Teacher that is a new customer to the cafes

a. **Goal in context:** To be able to find a better choice as per a majority of customers.

c. **Preconditions:** The primary actor has been provided with credentials to login to the app. The credentials are set during registration (email and password). The users must have rated for the existing food items.

d. **Scenario:**

. The user opens the application and selects the login button to go to the login screen

. The user then enters their unique credentials, i.e. email and password, to login to the app.

. The user picks one of the eight different food items (burger, sandwich, oreo shake, meal, etc) and reaches the rankings page.

. The user will then view the ranking as per all the users' ratings.

Exceptions:

. Password is incorrect, user re-renters correct password.

. Password not recognised, technical team is contacted to reprogram the password.

e. **Frequency of use:** any number of times per day

f. **Channel to actor:** via the main application user interface

g. **Secondary actors:** technical team Channels to secondary actors: phone line, text messages, emails and other online messages.

h. **Open issues:**

. Should there be a way to access the ranking list without logging in?

. How many tries are allowed to enter the right password?

6.3 Admin

For the actor, admins (who will update preparation time and cost of food):

Data updating:

a. **Use Case:** Entering the preparation time and food price

b. **Primary actor:** Admin of the application or the managers of the cafes

c. **Goal in context:** To be able to update the database with the preparation time of food and its cost price

d. **Preconditions:** The primary actor has been provided with credentials to login to the insider app. The credentials are given to the actor by the admin of the application

e. **Scenario:**

. The admin will open the insider app and enter the unique login credentials. **.**The admin will select the cafe from the drop down list.

. The admin will select the food item from the drop down list.

. The admin will enter the preparation time and cost price of the food

. item of that cafe and press the submit button to enter details.

f. **Exceptions:**

. Password is incorrect, user re-renters correct password.

. Password not recognised, technical team is contacted to reprogram the password.

. Data can't be entered by the admin, technical team is contacted to help with the database errors.

g. **Frequency of use:** once for registering a food item at each cafe

h. **Channel to actor:** via insider app user interface

i. **Secondary actors:** technical team

j. **Channels to secondary actors:** phone line, text messages, emails and other online messages.

k. **Open issues:** How many tries are allowed to enter the right password?

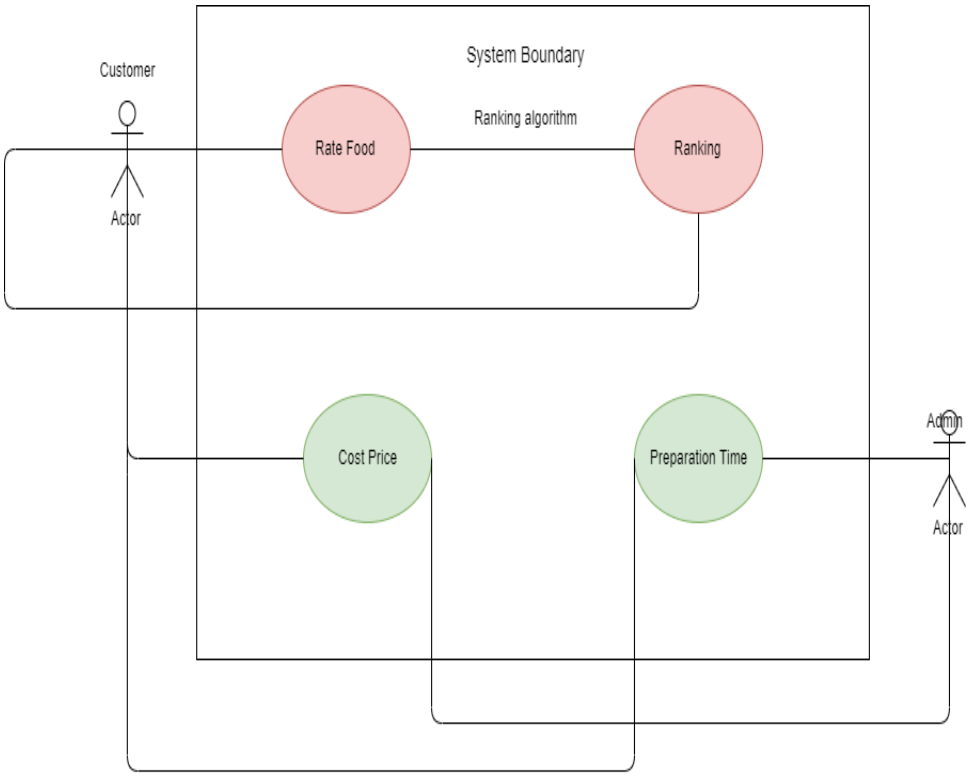

Figure 2: USE CASE Diagram

7 Negotiation

Our team had in mind the deadline of the requirement sheet submission and thus had to cut down a few requirements that were out of scope of the project and that were out of scope of the available skill set that our team possessed. Our development team had to keep in mind that we were developing a Food ranking system that was not specic to a single type of client but to general users who wanted to enjoy the delicacies of JSS S&TU.

7.1 Scalability Issues

We will be making the app for cafes and restaurants in the campus, including Frappe, Nescafe, etc, but not places like Aroma the Bakers, Kannan Bakery, Lassi Shop. This is because of a couple of reasons:

i. The main stakeholders of the app are students of JSS S&TU. The collge gives breaks of about 30 minutes to 1 hour. Hence to be on time to class, students have to eat inside the campus rather than outside.

ii. The other issue is distance to the outside places. Aroma the Bakers is 1km, while Lassi Shop is 2.5km from college, which is a major problem because most of the 1st year students do not have motor vehicles. So, they won't be visiting them most of the time.

7.2 Food delivery applications integration issues

Residents who live in hostels or Paying Guest establishments, where cooking is either not available or gets tiring, often order food using online apps like Zomato, Uber Eats and Swiggy. We could implement our idea into these platforms. If one was to decide where to order a particular dish available in two or more places, then the platform will tell them which one would be better. However, due to time limit for the submission, we have to delay the said integration.

7.3 Machine Learning implementation issues

i. Lack of quality data: Comparing two or more food items has a lot more than just the calories, quantity, price and distance to the cafe. There are factors like the ambiance of the cafe and taste, which both depend on the customer. Converting these factors into data will be a time consuming process and also very hard to implement.

ii. Inadequate infrastructure: Machine learning requires a vast amounts of data processing power. Our system wont be able to handle the workload and will crash under pressure. Considering the computers in our college labs, only the newly installed PARAM supercomputer will be able to handle it(but the large number of machine learning and data science projects taken up by students will make it time consuming, waiting for our chance).

iii. Implementation issues: To implement, we will need Analytic engines. Implementing newer Machine Learning methodologies into existing methodologies is a complicated task.

iv. Lack of skilled resources: Deep analytics and machine learning in their current forms are still new technologies. Thus, there is a shortage of skilled developers available to manage such analytical content for Machine Learning. Data scientists often need a combination of domain experience as well as in-depth knowledge. We teammates being 3rd students, don't have enough knowledge.

7.4 Result

Our team decided to negotiate such requirements with the stakeholders so that we could strive for a win-win result. In this win-win situation, what we intended for was that the stakeholders would win by getting all the functional requirements which were absolutely necessary and those which would satisfy their needs, and also we would win by working to realistic and achievable deadlines. Thus, during the negotiation phase, our team successfully identified the key stakeholders and determined the win conditions of the stakeholders. Then we approached the stakeholders and negotiated the stakeholders win conditions in order to have them satisfied with those requirements and also to make us complete the task within the specified deadline. This was a key step for our team to carry out the further plans which include the actual design and implementation of the project.

8 Specifications

The following subsections of the Software Requirements Specications (SRS) document provide an overview of the entire SRS. This document provides details on how the developers will design the system, however this document is not for designing.

8.1 Introduction

8.1.1 Purpose

i. To come up with efficient real time application that will provide users with a better option among a couple of restaurants.

ii. To give feedbacks to the restaurant managers, to better their service and hence to improve sales.

8.1.2 Document Conventions

LaTex is used to document the Requirements gathering phase.

8.1.3 Intended Audience and Reading Suggestions

i. Front end developers - They will have to know about the customer and Restaurant manager side of the app, i.e. XML and Web Development.

ii. Back end developers - They will have to know Java or Kotlin development.

iii. Testers and Debuggers - They will have to know about the assumptions and dependencies made, software quality attributes and overall description of the application.

iv. Customers and Restaurant Owners - They will have to know the features of the apps to be using it to the full potiential.

8.1.4 Product Scope

i. The app will be working for the restaurants in the campus of JSS S&TU, however, we will be increasing the scale expanding to restaurants and cafes outside the campus.

8.1.5 References

i. https://ieeexplore.ieee.org/document/7991732

ii. https://ieeexplore.ieee.org/document/6785539

8.2 Overall Description

8.2.1 Product Perspective

i. This is a new self contained product, which is independent of any other applications.

ii. The idea was born when we experienced dilemma in which restaurant to visit, so we wanted to make it easy for the freshers, importantly not exclusively, in the college.

iii. The application will play an important role in aiding the restaurant managers to introduce changes in their menu or service to improve sales.

8.2.2 Product Functions

i. The product should identify the best restaurant for a particular dish.

ii. It should then identify the reason why one restaurant is rated over another, so the lower rated restaurants can improve.

iii. It should give the customers opening and closing timings for each restaurant.

iv. Details about each food item is given, so that food choices can be made easily.

8.2.3 User Classes and Characteristics

Product functions used by :

i. Product maintenance people : for modification and updation of the product

ii. End Users : for finding the best rated food items of the college

8.2.4 Operating Environment

i. The product is Android Based Application. Any mobile phone hosting versions of Android 4.3 or higher can be used.

ii. There should also be reliable internet connection.

8.2.5 Design and Implementation Constraints

i. Languages used: Java

ii. Front end development: XML

8.2.6 User Documentation

i. User manual will be provided for the product.

ii. Tutorials given for beginners.

iii. Email id's of the product creators and designers provided for any user support or issue.

iv. Additional reading references also provided for efficient usage.

8.2.7 Assumptions and Dependencies

Assumptions:

i. It is assumed that the the user ratings are honest.

ii. It is also assumed users have tried enough food items.

Dependencies:

i. Interfaces should work properly and values should be updated regularly.

8.3 External Interface Requirements

8.3.1 User Interfaces

i. Login page for authentication of the user.

ii. If no reliable internet connection, then display error message: Internet not found/ network connection is slow, depending on the state.

iii. Interactive buttons to get current status of food ranking.

8.3.2 Hardware Interfaces

Minimum requirements of the hardware components for the system :

i. Mobile phone connected to WiFi or mobile data connection or any other kind of internet connection.

ii. 5 MBPS connection speed, which is ideal.

8.3.3 Software Interfaces

i. Android 4.3 Jelly Bean or higher.

8.3.4 Communication Interfaces

i. Any Android device to use Internet facilities.

8.4 System Features

i. Ranking for food.

ii. Menu of the restaurants, along with other information like quality, quantity, price etc.

iii. Timings, ambience,hygiene etc of the restaurant.

8.5 Other Nonfunctional Requirements

8.5.1 Safety Requirements

i. Only secure connections to be used while browsing or Internet usage.

8.5.2 Security Requirements

i. Only users authorised to modify and change the characteristics of the application can use the tool.

8.5.3 Software Quality Attributes

Portability Application must work on different machines, irrespective of their hardware components, provided the hardware requirements mentioned in section 3.4 are satisfied. It also must be platform independent.

Reliability The software must be reliable and function desirably on all Android platfrom, and at all times.

Usability Software must be user friendly, that is, it should be easy to understand and use by all of its users. Instructions for use must be provided on the form, to help the user understand the procedure to use the network monitoring system. The different buttons must also be named appropriately, to avoid confusion.

Modifiability Software must be easy to understand and modify, in order to cater to future needs. It should be possible for the system to accept inputs of other forms like static dataset, if the necessary changes are made to the respective modules. Modification of one module or subcomponent, should have minimum effect on the other modules or subcomponents.

8.5.4 Business Rules

i. The software requires someone with a good knowledge of network.

ii. The software provides information which the administrator must understand in order to commit his/her changes.

8.6 Other Requirements

No other requirements identified as of now.

9 Validation

Validation of the collected requirements plays an important role in the overall process of requirements gathering. The process of validation of requirements involves activities like checking for any ambiguities in the requirements collected, omissions of any requirements and finding out if any inconsistent requirements are present or not. Once the requirement model is shown to the stakeholders, the stakeholders prioritize the requirements based on their needs and we, as requirement engineers, group the requirements into packages so that they can implemented easily as software increments. During this step, the requirement engineer must ask questions that are shown in the following subsection.

9.1 Validation Checklist

i. ***Is each requirement bounded and unambiguous?***

We went through the requirements which were mentioned previously and found out that there were a few ambiguous ones, like the stakeholder category of Students gave us a lot of ambiguous requirements eg. some of them were willing to pay a lot while others weren't. We eliminated such requirements by introducing price as a feature which helps in making things clear. Our team provided abstract details to the stakeholders which hid the implementation part of the project. This made easy for the team to gather requirements without confusing the stakeholders of the implementation part. A few requirements which were out of scope of the project were realized in this stage, and were eliminated. The students category of stakeholders requested to get food ratings for all the restaurants in Mysore. This requirement was out of scope of our project and was not feasible in terms of implementation, so our team eliminated this specific requirement. Apart from this, the lecturers category of stakeholders wanted recommendations based on their order history. This requirement was out of scope of our project, because it is about ranking food and not using machine learning to learn from the order history.

ii. ***Is the requirement really necessary or does it represent an add-on feature that may not be essential to the objective of the system?***

While answering to this question, we thought that the requirement of having a feature to show the billing details was not necessary and hence we eliminated it. Our main objective with this application is to find the best food items by collecting several reviews. This feature is general to all people.

iii. ***Is each requirement achievable in the technical environment that will house the system or product?***

The team decided to remove the feature of recommendations based on order history during this phase of requirement collection as it is difficult to achieve in the technical environment that will house the application. The feature would definitely be useful in our application however due to our strict deadline and lack of required skills we discarded this feature.

iv. ***Do any requirements conflict with other requirements?***

We surely did come across a few ambiguous and a few out of scope requirements, but our team found out that there were no requirements that conflicted the other. Validation of requirements through this question was done easily.

v. ***Does each requirement have attribution? That is, is a source (generally, a specific individual) noted for each requirement?***

During the process of requirement collection, our team carefully noticed and made a note of the source of each requirement. This has been successfully reflected in the elicitation phase of the requirement collection. Each client has been attributed to a specific set of requirements.

vi. ***Is the specification structured in a way that leads to easy understanding, easy reference, and easy translation into more technical work products?***

A Software Requirements Specification(SRS) document has been created to allow easy understanding of the models and functions of the project, along with formal diagrams, descriptions and other graphical models. The SRS is a part of this report and has been included in previous sections.

These were the questions asked and answered to ensure that the requirements model was an accurate reflection of stakeholder needs and that it provided a solid foundation for design.

10 Requirement Management

Requirement Management can be defined as a process of eliciting, documenting, organizing, and controlling changes to the requirements.

We started this process of management of requirements as soon as the elicitation phase began. Our team had a closer look at the requirements collected and had a clear understanding of what the requirements were, and also got an abstract idea of how these requirements can be implemented into actual software, which is just a mobile application in this case.

During this phase, we had a better control on our project. We followed a particular pattern for managing the requirements. What we did first was, identified each of the requirements clearly. This was possible because our team maintained a note of all requirements collected. Then we went through the list and reassembled the requirements based on a priority given by the stakeholder. During this process of reassembling, we also made sure that the requirements were reassembled taking into consideration, the various problems we might face during the implementation of the project.

11 Product

After collecting the requirements from various stakeholders and analysing their feasibility, we associated the requirements needed from the software. Requirements engineering helped us to have a clear mindset about the features of our product. Keeping all that in mind, we started developing our software.

12 Roles and responsibilities

. Adishi Jha: Tester, Front end developer

. Anirudh D Pai: Business Analyst, Tester

. Eeshaan Achar: Project leader, Back end developer

. Navneet Hosmane: Lead developer, Front end developer

. Donapati VS Reddy: Business Analyst, Back end developer

Front end developer: A front-end android developer is responsible for implementing visual elements that users see and interact with.

Business Analyst: The analyst interacts with the business stakeholders and subject matter experts in order to understand their problems and needs.

Back end developer: They are responsible for application logic and integration of the work front-end developers did.

Tester: They are involved in planning and implementing strategies for quality management and testing.

Part III
Gallery

13 Discussion with the Students

Figure 3: Inteview with a Fresher

Figure 4: Inteview with another Fresher

14 Discussion with the Faculties

Figure 5: Inteview with a Faculty

Figure 6: Inteview with more Faculties

15 Discussion with the Restaurant Owners

Figure 7: Inteview with Cafe Frappe's Owner

Figure 8: Inteview with Yampa's owner

16 The Team

Figure 9: Group Disscussions

Figure 10: Team Members

Part IV
System Design and Implementation

17 Design Concepts

System design is the process or art of defining the hardware and software architecture, components, modules, interfaces, and data for a computer system to satisfy specified requirements. The design of the system is essentially a blueprint, or a plan for a solution for the system. A system is considered to be a set of components, with a clear and defined behavior, which interact with each other in a fixed, defined manner, to produce some behavior or services to it's environment.

17.1 Database System

The system is itself vertexed on students of the college (who will act as the major repecients) consequently making them an important entity. As the service can be extended to people beyond the college, we consider these people in the student category as well.

As we will be profiling the different customers in categories like teachers, students and shops we have an attribute signifying their categories. We also have an entity called 'FOOD' which will have all the details of the different food items along with a relation which will connect it to the entity called 'SHOP' defining the places that particular item may be found.

The ***Entities*** are:

Customer: This entity includes all personal information of the different customers including their category (freshers, veterens, PeopleNearby). Freshers would have a more diverse recommendation encouraging them to try different places and try out new things. Veterens will have a more stronger profiling with recommendations rendering only to their taste. PeopleNearby will include people opting for the sevice situated outside the college in the nearby areas.

Food: This entity would have all information about the food items including it's ratings, vicinity (calculated dynamically with the customer's location), price, discount (calculated dynamically from saleOn relationship), picture(blob), and the number of friends who like that item.

Shop: This entity will contain all the information regarding the shop, it's contact, delivery options, it's location and the general popularity. This entity is related to the entity food with two relations 'foundIn' and 'SaleOn' where the latter lets the shop provide different discounts on food items.

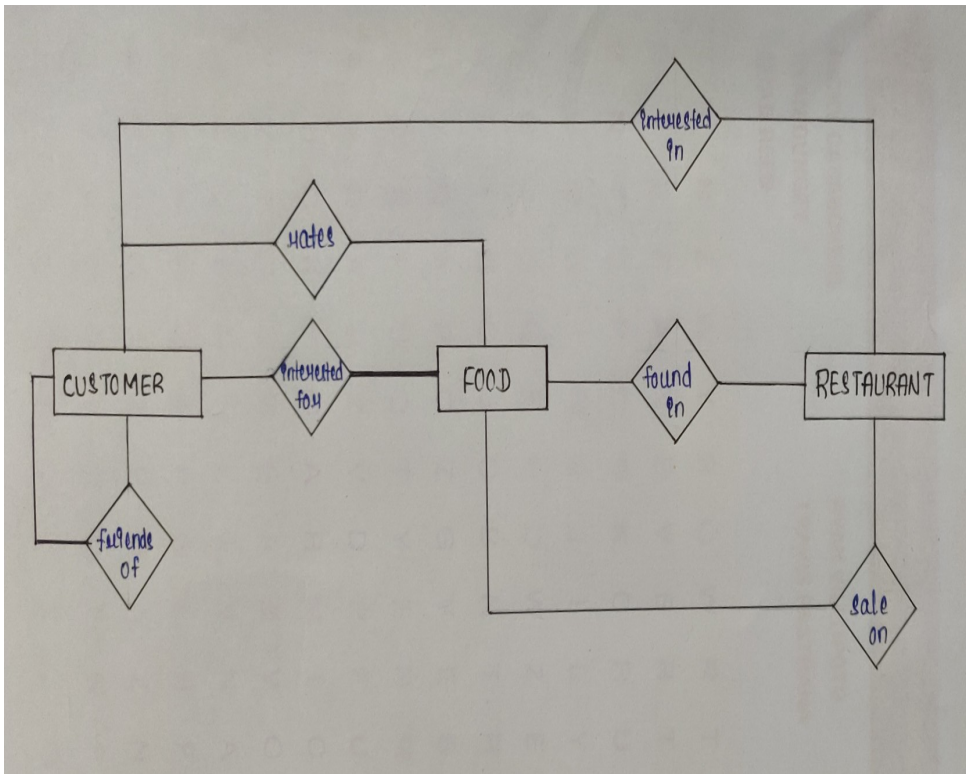

Figure 11: Entity-Relationship diagram

The ***Relationships*** are:

InterestedIn: This is a many to many relationship existing between Customers and Shops.

InterestedFor: This is another many to many relationship connecting the customers and the food items.

FoundIn: This relation links the different food items to the shops they are found in.

SaleOn: This relation lets shops to provide different kinds of discounts on food items.

FriendsOf: This is a self relationship on customers keeping a track of their friend circles.

Rate: This relation allows customers to rate.

17.2 Architectural Design

Architectural design elements give us an overall view of the software. The architectural design element is usually depicted as a set of interconnected sub-systems, often derived from analysis packages within the requirements model.

17.3 Architectural Genre

Our Architectural genre is 'Commercial and nonprofit'. Systems that are fundamental to the operation of a business enterprise. We collect feedback in form of votes which helps in ranking food items served by various restaurants.

17.4 Architectural Style

Our model is based on the Data Centered Architecture.

This architecture is applied when a data store (e.g database) resides at the center of this architecture and is accessed frequently by other components that update, add, delete, or otherwise modify data within the store. Client software accesses a central repository. Client software accesses the data independent of any changes to the data or the actions of other client software. A variation on this approach transforms the repository into a blackboard that sends notifications to client software when data of interest to the client changes.

We are also working to rank a particular food item served in various restaurants. We need to store ranks in the database and update them from time to time. Hence, Data Centered architecture style suits the best for this project.

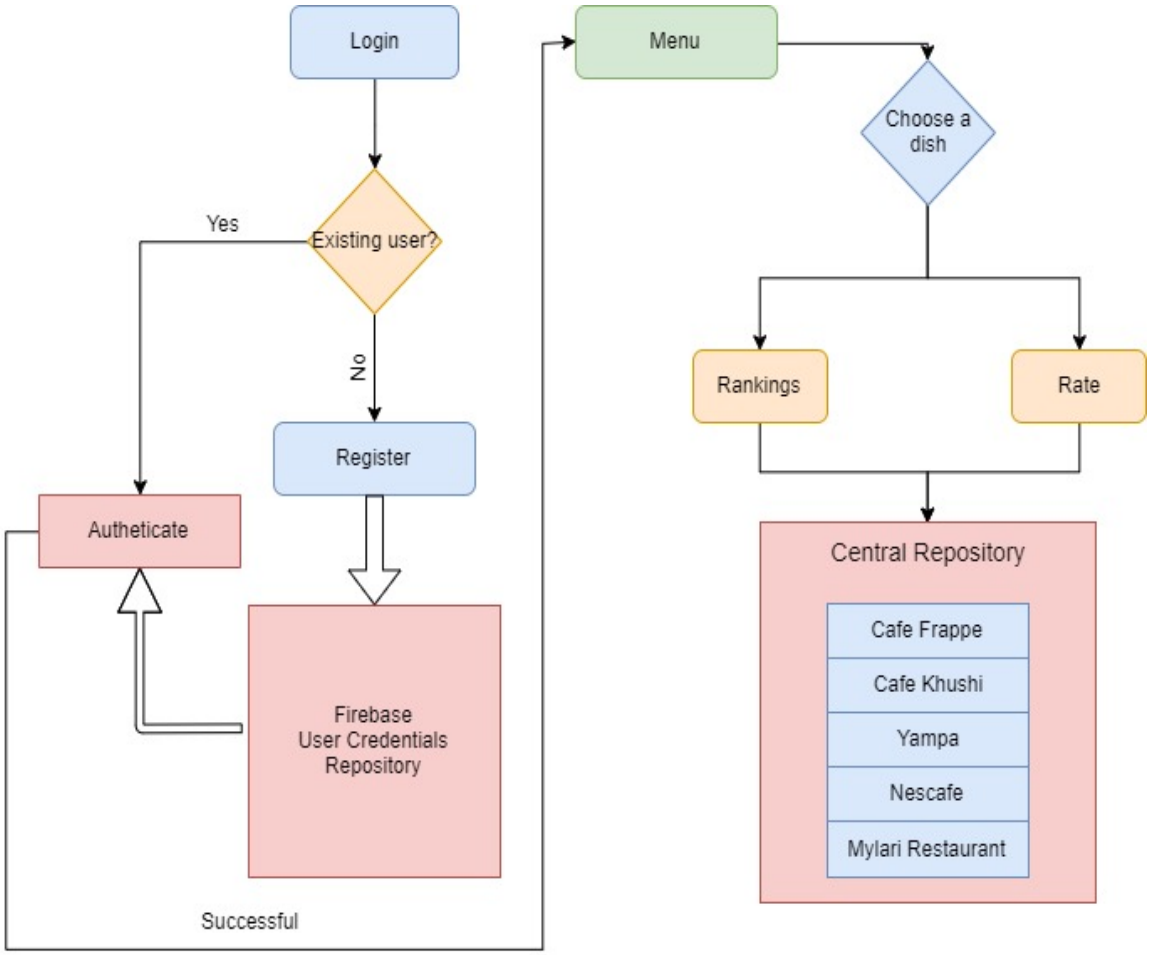

Figure 12: Architectural Style

18 Data Flow Diagram

i. **Users**: Users are the external entities that use the software.

ii. **Authentication**: Users will enter their login credentials and that has to be matched with the data available in the database. Users acn click on forgot password to enter new password through a link sent to their email.

iii. **Display Menu**: The food items available in the various restaurants are displayed.

iv. **Ranking Database**: All restaurants have rank assigned to them for a particular food item in the database.

v. **Retrieve rankings**: Whenever a dish is selected, restaurants are displayed according to their ranks.

vi. **Rate**: For a particular dish, users can choose and vote in between two restaurants. This option can be skipped if the user has not tried that food item yet.

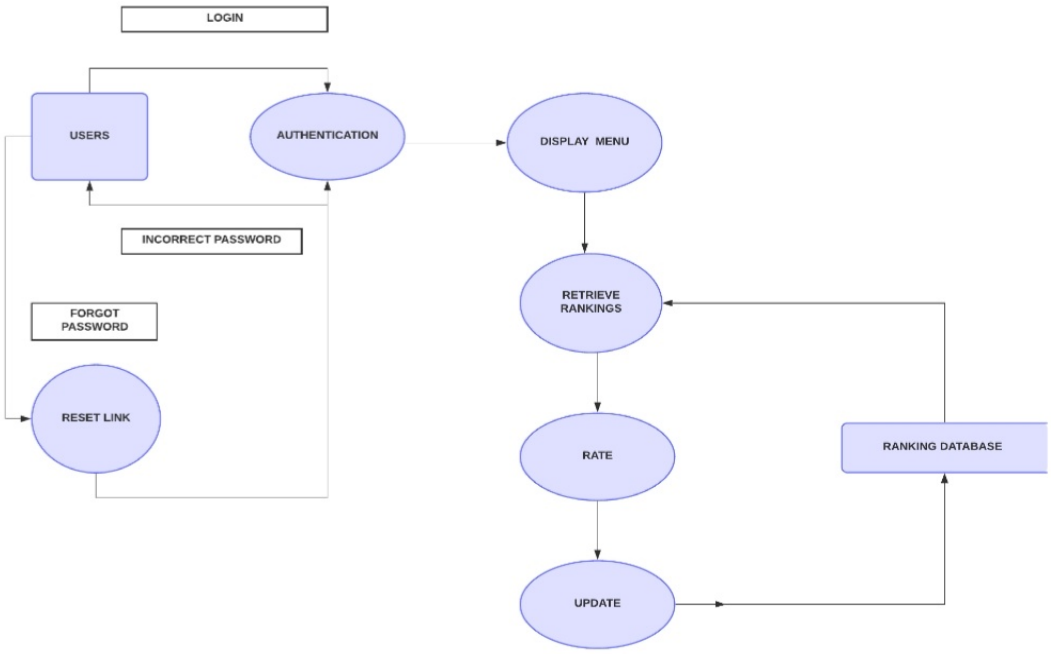

Figure 13: Data Flow diagram

vii. **Update**: After a User rates the food item, the corresponding change is reflected in the database.

19 System overview

An application is to be designed which gives suggestion about food items based on the rank assigned to them. Rank depends on the number of votes given to a food item of a restaurant. For a particular dish, users can choose between two restaurants and vote for it.

20 System Implementation

We used android studio for our software development. Firebase was used in the backend. A series of steps were followed to make our software work.

20.1 Algorithm For Login

Step 1: User registers himself with his email id and password. Registered users can login directly.

Step 2: The password and email are matched with the data stored in the database.

Step 3: If the credentials are correct, the users can access the software for viewing ranking and rating the food items.

Step 4: If the user forgot the password, he can recover it through the link sent to his email.

20.2 Algorithm For Ranking

Step 1: After login, menu is displayed.

Step 2: One dish can be choosed at a time.

Step 3: For that particular dish, ranking of various restaurants are retrived and displayed.

20.3 Algorithm For Rating

Step 1: After login, rate button is displayed. When a user clicks on that, two restaurants are displayed for one food item.

Step 2: User can vote for the restaurant which serves that food item in a better way. Vote for that food item in the database is updated.

20.4 Algorithm For JCE Insider

Step 1: The admins can enter price and prepartion time for each foof item.

Step 2: Data gets stored in firebase.

Part V
System testing and Result Analysis

Different kinds of difficulties came our way as we advanced with the project.

i. Inadequate knowledge about Android Studio.

i. Inadequate knowledge about Android Studio.

ii. Exposure to a new programming language called Java, hence we had to refer for each and every small errors encountered.

iii. We had to learn how to use realtime database of Firebase.

WHITE BOX TESTING

a. **Unit Testing:** Unit Testing is a level of the software testing process where individual units/components of a software /system are tested. The purpose is to validate that each unit of the softwareperforms are designed. The goal of unit testing is to isolate each part of the program and show that individual parts are correct in terms of requirements and functionality.

b. **Integration Testing:** Integration Testing is a level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. The testing of combined parts of an application to determine if they function correctly is Integration testing.

BLACK BOX TESTING

a. **Integration Testing:** Integration Testing is a level of the software testing process where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. The testing of combined parts of an application to determine if they function correctly is Integration testing.

b. **Acceptance Testing:** Acceptance Testing is a level of software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and asses whether it is acceptable for delivery.

21 Plan of Testing and Data collection

The plan of testing for our android application is as follows:

i. The content model is reviewed to uncover errors.

ii. The interface model is reviewed to ensure that all use cases can be accomodated.

iii. Design model is reviewed to uncover navigation errors.

iv. The user interface is tested to uncover errors in presentation.

v. Each functional component is unit tested.

v. Performance tests are conducted.

vi. Application is implemented in a variety of different environmental configurations.

We are dividing our project into four modules.

i. Login module: It take user information and updates in the database.

ii. Ranking module: It retrieves previously stored rank of restaurants for a particular food item.

iii. Rating module: It gives user choice to vote for one restaurant which serves a particular dish better.

iv. JCE Insider: It takes information about price and preparation time for each food item by the admin.

We are collecting data by taking inputs from 100+ users who have registered to our application. They can enter their choice and value will be updated to the database.

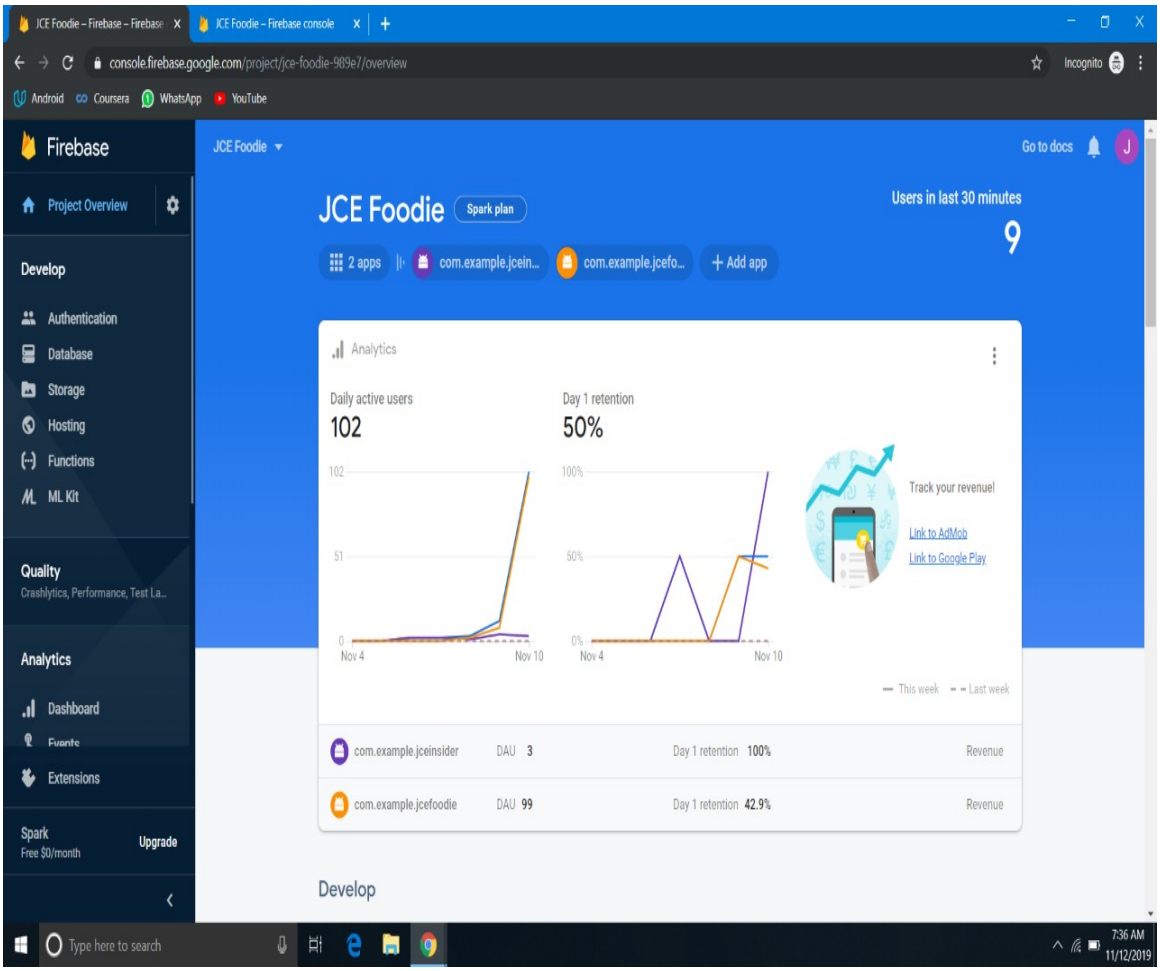

Figure 14: Data Collection by various users

22 Unit Testing and Result Analysis

TEST-ID	MODULE	FEATURE	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT
TC1	Login	Login and Registration of new user	Email id and password	Verified Email and password which leads to the menu page	If the Email and password are valid, it goes to the next page. However, if the Email is invalid or the password is too short, Login fails.
TC2	Login	Forgot Password	Email id	A link to be sent to the email address	A link was sent to reset the password
TC3	Rank	Rank of restaurants which serve a common dish	Any dish should be selected from the Menu.	Ranks of restaurants are displayed	This module works as expected and list of restaurants are displayed
TC4	Rating	User can rate a particular dish as per their experience.	One restaurant is selected between two restaurants which serve a common dish.	The values in the database should get updated	This module works as expected and number of votes get updated
TC5	JCE Insider	Stores price and preparation time of food items	Price and preparation time	The values in the database should get updated	Data gets stored

RESULT ANALYSIS: All our modules are working fine independently.

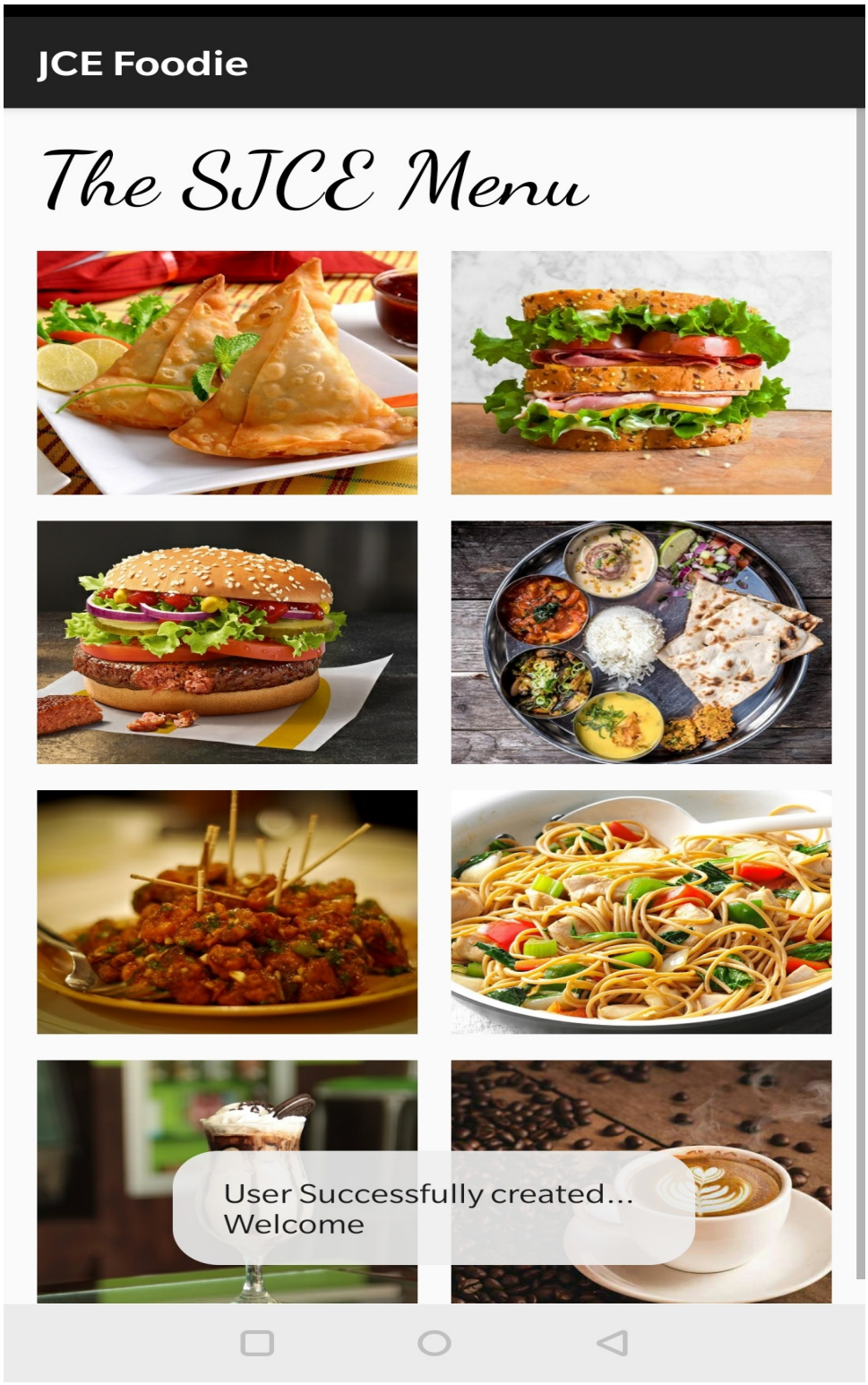

Figure 15: Registration of New user

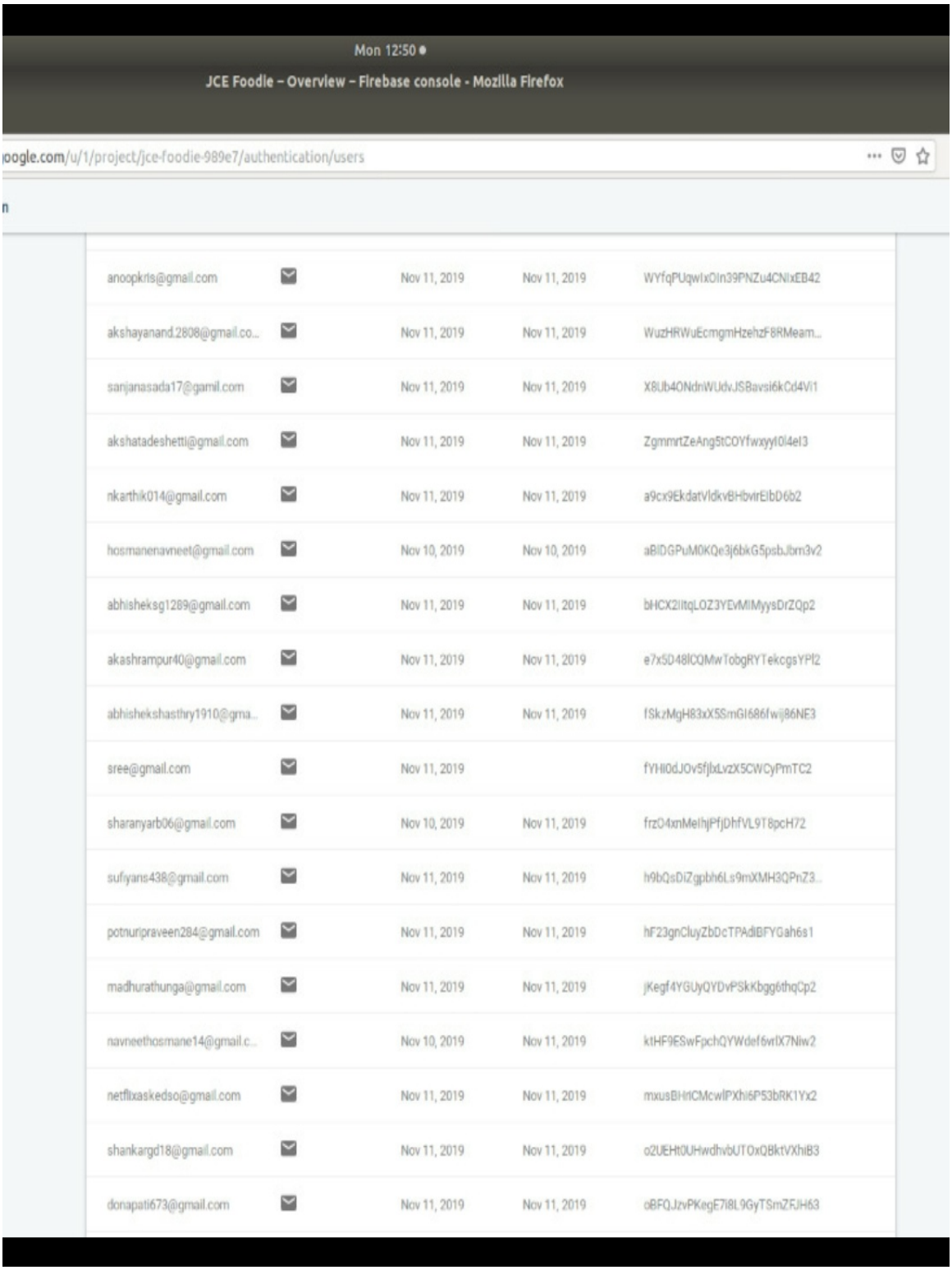

Figure 16: Authentication of various users

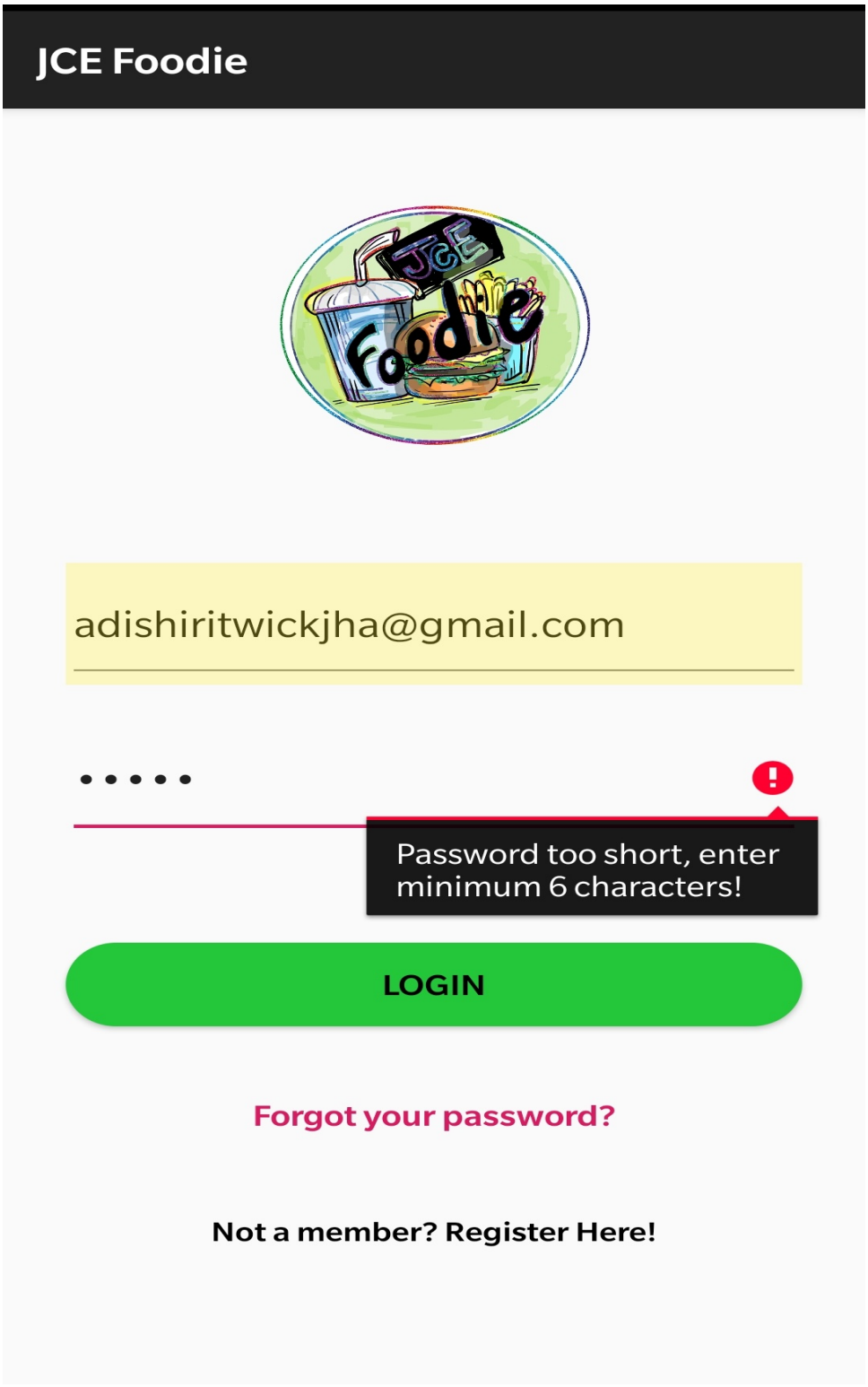

Figure 17: Short password

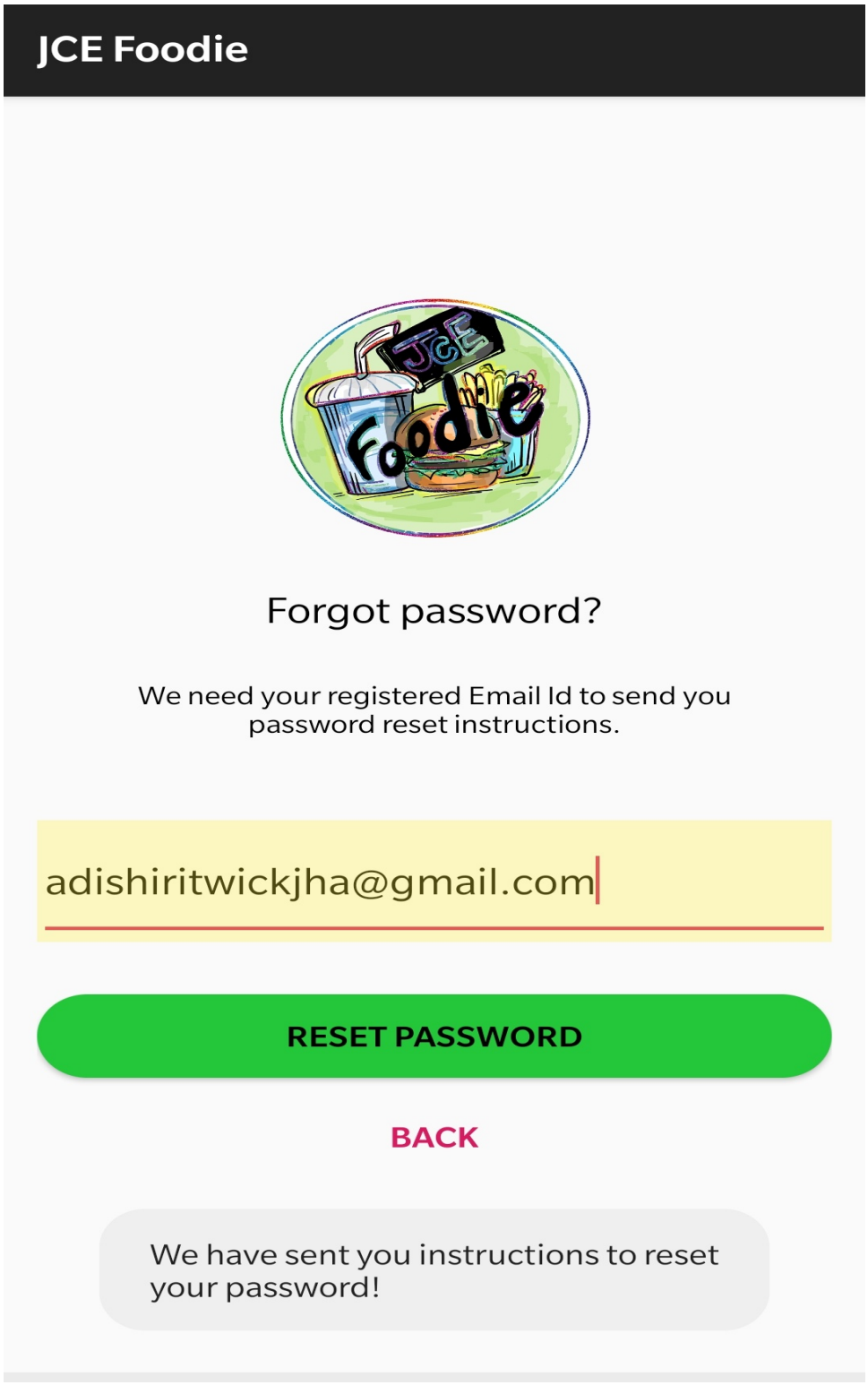

Figure 18: Forgot password

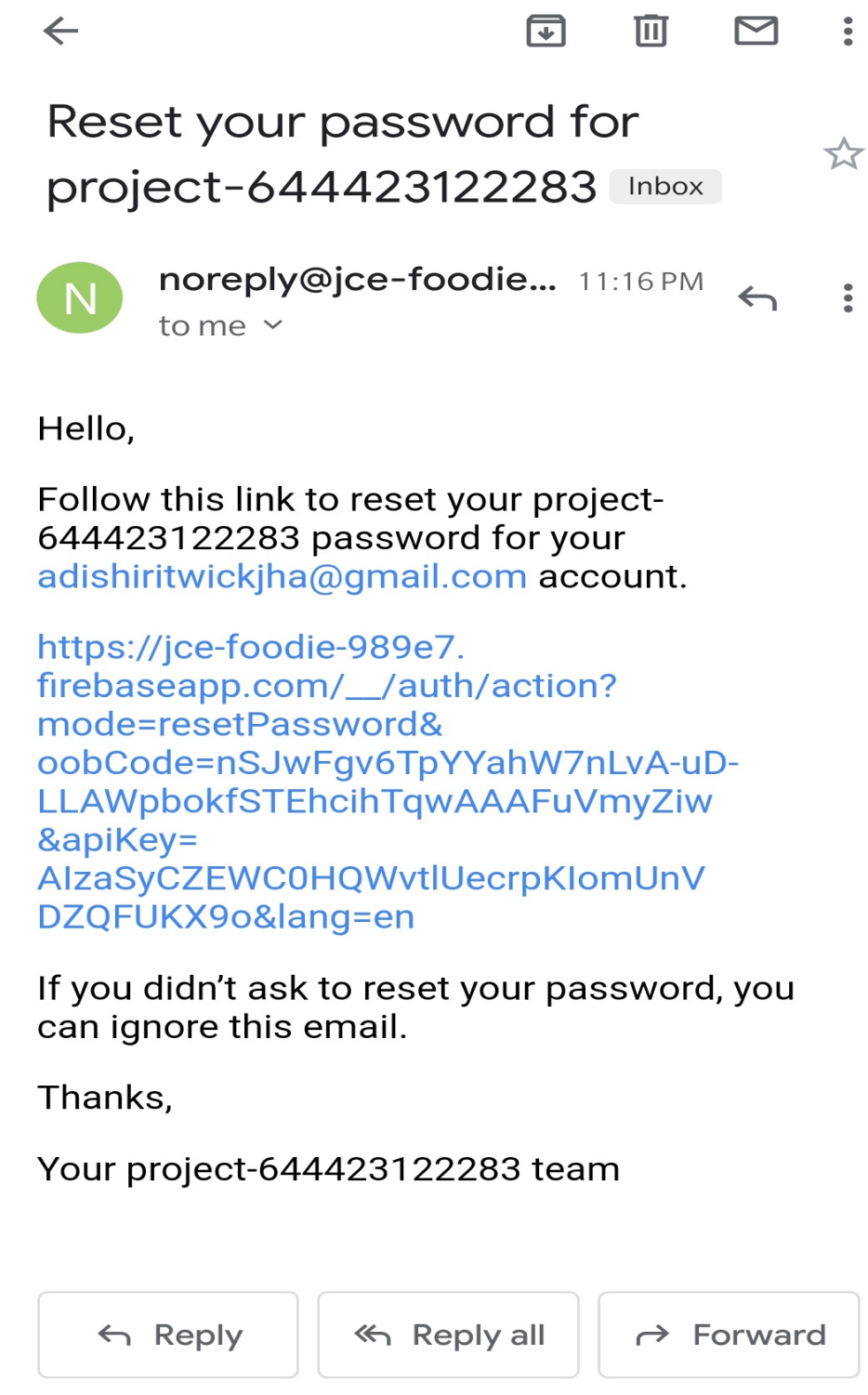

Figure 19: Password reset link

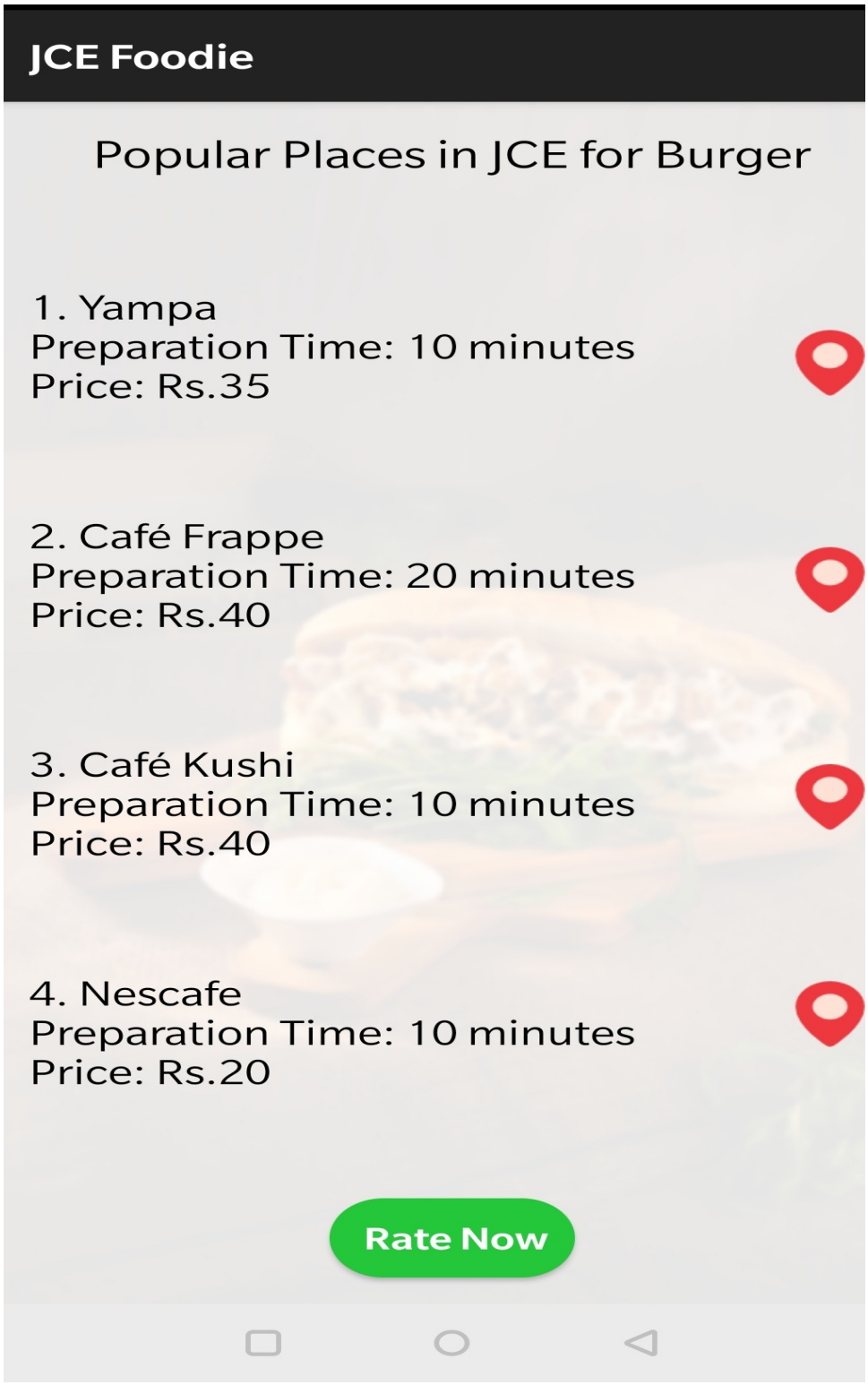

Figure 20: Rank page

Figure 21: Rate page

Figure 22: Before rating

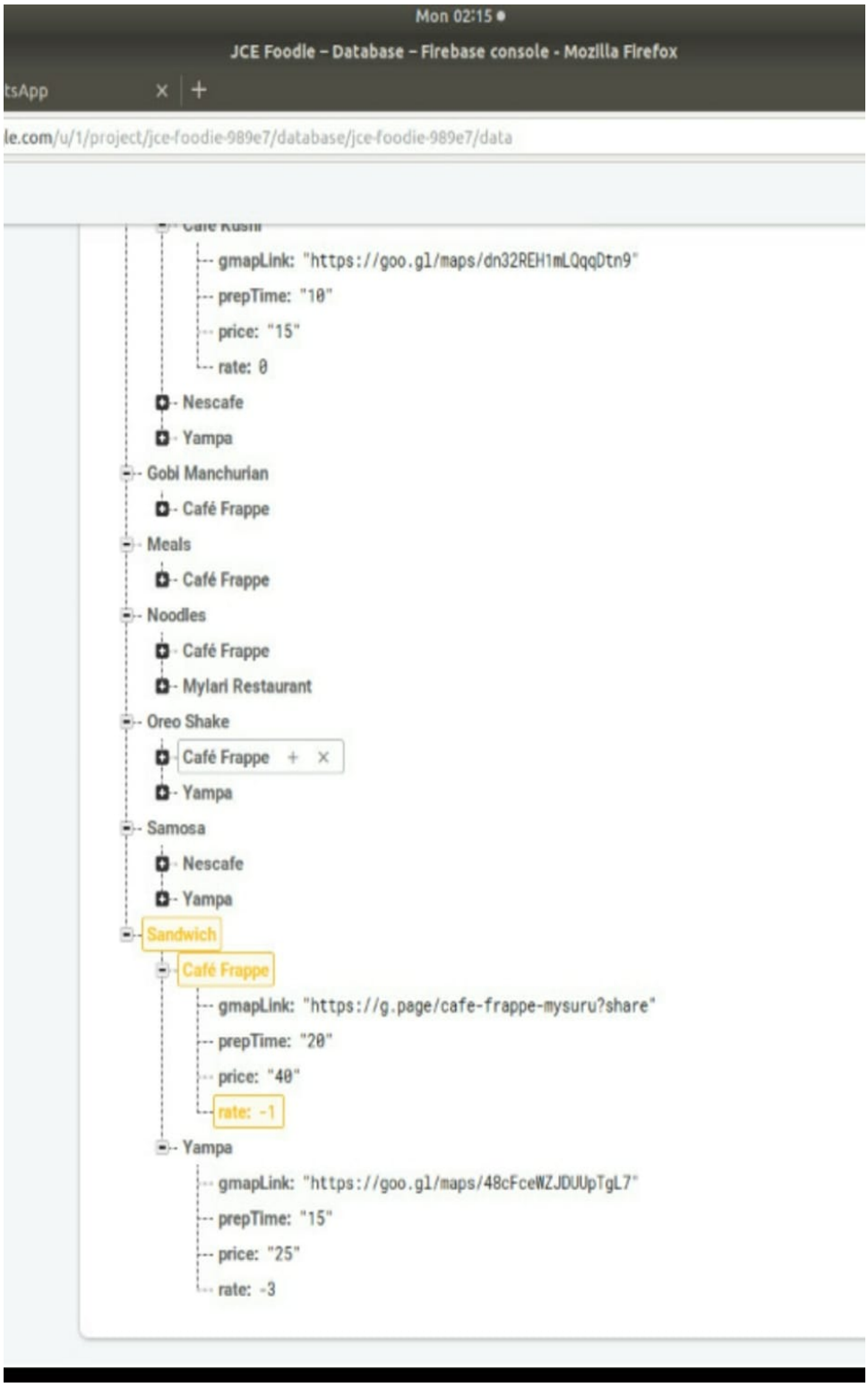

Figure 23: After rating

23 Integration Testing and Result Analysis

Integration Testing is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. All the units of the software are implemented after integrating and the output of the software is tested for correctness of the results. Sometimes when the the units are integrated, the interaction between the various units may pose errors due to which the desired or expected output may not be obtained. Integration testing is done by giving a variety of inputs to the integrated unit or final product and testing the working of the software.

It can be done in two ways:

i. Incremental approach

ii. Big bang approach

We used incremental approach of Integration testing.In Incremental testing, modules are integrated one by one. We integrated login, rate and rank module of our application and connected it to Firebase.

When all the modules of our project were integrated to build the complete android application the behaviour of the modules with each other were observed. All the modules behaved as expected and there weren't any problems faced. The votes given by the users were reflected in the rank of the restaurant. And any input given to JCE Insider was reflected too.

RESULT ANALYSIS: The application successfully went through Integration testing.

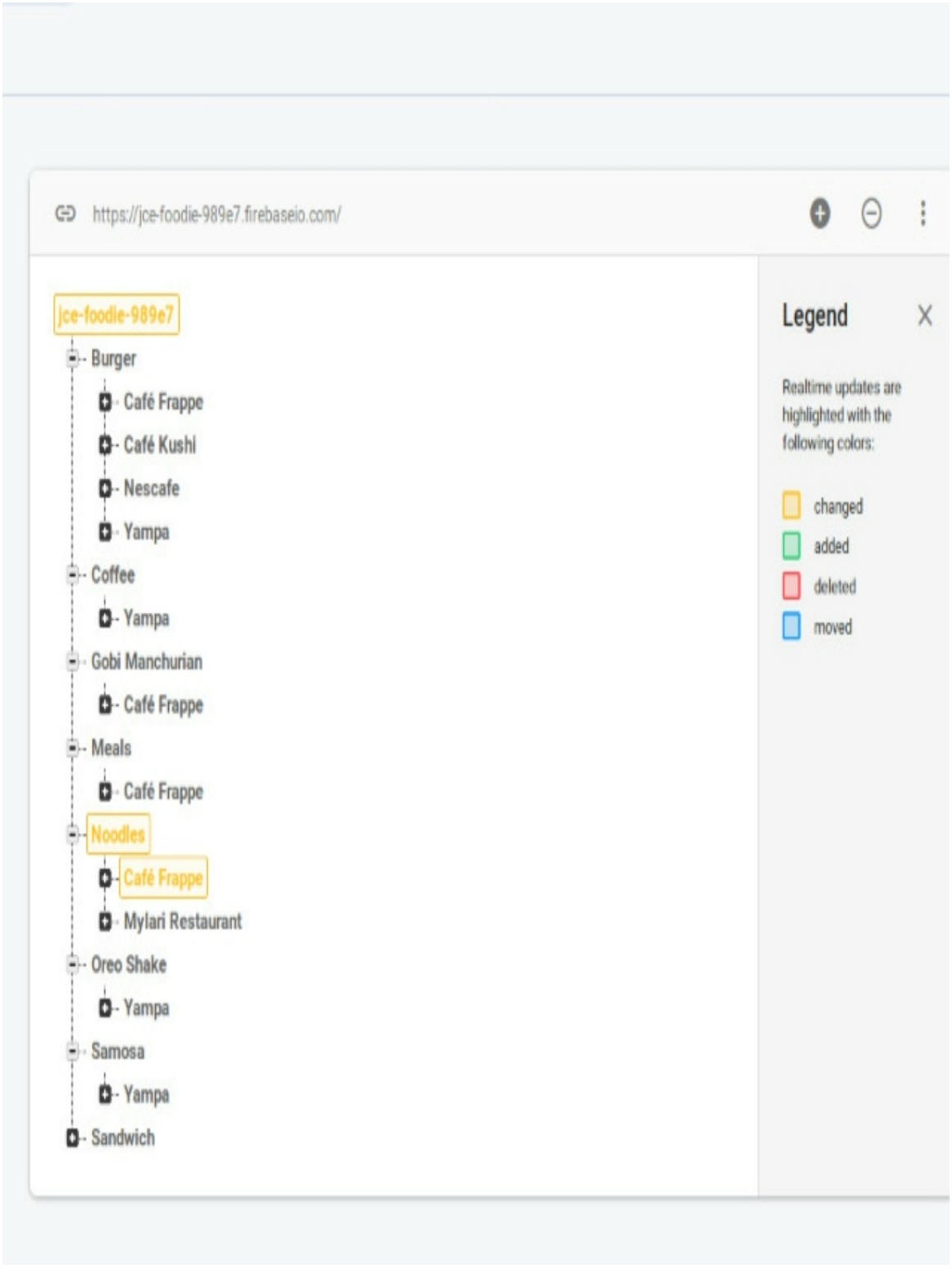

Figure 24: Data gets updated in Firebase

24 System Testing and Result Analysis

System testing is performed on a complete integrated system to evaluate compliance of the system with corresponding requirements.

i. **Recovery:** Recovery testing is a system that forces the software to fail in a variety of ways.

ii. **Security:** Our software is secure because we provide Email and password login.

iii. **Usability:** We have ensured that our software works properly for next 4-5 years

iv. **Stress:** Stress testing executes in a manner that demands resources in abnormal quality, frequency or volume.

v. **Deployment:** It is sometimes called Configuration testing, exercises software in each environment in which to operate. It runs fine on all android phones.

RESULT ANALYSIS: As per our current knowledge our software works fine and all data is displayed appropriately.

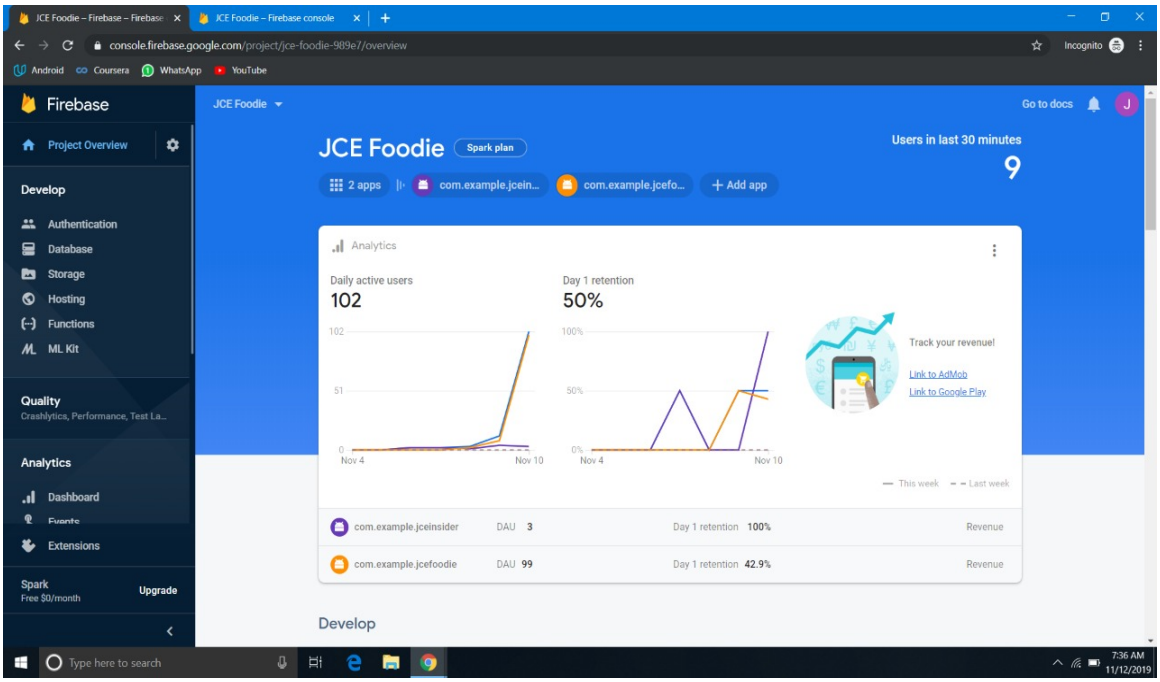

Figure 25: Firebase

25 Acceptance Testing and Result Analysis

Once we did the Unit and Integration Testing, we asked our stakeholders to perform system testing on our application. We went to the Yampa's owner and 1st year Fresher, Vaibhav. They saw the effective working of our project and appreciated the hardwork.

Yampa's owner also told us about some useful enhancements for our software :

i. He wants our software to inform everyone if there is any discount on any day.

ii. He wants detailed feedback which can help him to do the exact changes in his food items.

Vaibhav wanted few more features to be included:

i. He wanted an Order option which can save his time during the lunch break.

ii. In addition to that, he wanted personalised recommendations.

Figure 26: Interview with Vaibhav

Figure 27: Meeting with Students

Figure 28: Interview with a Faculty

RESULT ANALYSIS: Overall, our stakeholders were satisfied. The software was tested successfully.

Part VI
Conclusion and Future work

In the application developed, users are able to find the better option for a food product available in more than one cafe and also more data about it. This leads to saving a lot of time and money. Veterans or existing users can put up their point of view which is a neccesity. One of the areas of improvement of this app is the scale. Due to time and complexity the app only has cafes in the campus of SJCE. In the future, a lot more places can be added. Places like Kannan Bakery, Lassi Shop, Aroma the Bakers, etc. can be integrated in the app, as they will be providing more options.

Neural networks can be integrated with the app. The algorithm can be trained to find the users' mindsets based on their choices made in the past. This will be applied to give them likeable suggestions. Also machine learning algorithms can be applied to predict a user's taste in food. This can be done by training the algorithm on the first few choices made by the user.

Part VII
References and Remarks

26 References

- Roger S Pressman: Software Engineering - A Practitioners approach, 7th edition, McGraw-Hill Publication, 2015.
- `www.tutorialspoint.com`
- `www.geeksforgeeks.org`
- `www.stackoverflow.com`
- Seniors, CSE, JSS S&TU

27 Professor's Remarks