Food Chain Management Using GPS Tracking

Akila Akter
Student Id: 012162008

A Project

in

The Department

of

Computer Science and Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Science in Computer Science and Engineering

United International University

Dhaka, Bangladesh

December 2019

Approval Certificate

This project titled "**Food Chain Management Using GPS Tracking**” submitted by **Akila Akter**, Student ID: **012162008**, has been accepted as Satisfactory in fulfillment of the requirement for the degree of Master of Science in Computer Science and Engineering on December 2019.

Board of Examiners

1.

________________________________ Supervisor

Professor Dr. Mohammad Nurul Huda
Professor & Director - MSCSE
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka-1212, Bangladesh

2.

________________________________ Examiner

Rubaiya Rahtin khan
Assistant Professor
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka-1212, Bangladesh

3.

________________________________ Ex-Officio

Professor Dr. Mohammad Nurul Huda
Professor & Director - MSCSE
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka-1212, Bangladesh

Declaration

This is to certify that the work entitled "**Food Chain Management Using GPS Tracking**" is the outcome of the project accepted out by me under the supervision of Dr. Mohammad Nurul Huda, Professor & Director - MSCSE, United International University (UIU), Dhaka, Bangladesh on December, 2019.

Akila Akter
Department of Computer Science and Engineering
MSCSE Program
Student ID: 012162008
United International University (UIU)
Dhaka-1212, Bangladesh.

In my capacity as supervisor of the candidate's project, I certify that the above statements are true to the best of my knowledge.

Professor Dr. Mohammad Nurul Huda
Professor & Director - MSCSE
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka-1212, Bangladesh

Abstract

With recent technological advancement of modern science people are now expecting the information about the location of any object for tracking purposes. Presently, we want more location based services for being advanced and to save time and money also. GPS is a system which is already implemented and everyone can access it without any restriction. Having the facility of GPS to develop this system we need a GPS device to calculate the location from the information taken from GPS. Hence, we have chosen GPS based device to perform this calculations because GPS based device is cost effective and offers multidimensional purposes having some special built-in features like GPS service. Thus, this system is developed for location tracking of a group of people with a proximity alert system using various latest demanding tools and technology like JSF, Java, Bootstrap etc.

This project ‘Food Chain Management Using GPS Tracking‘ is an GPS based web application which helps the user to locate the restaurant nearby the current location and choice best restaurant based on good rating available in the respective restaurant.

The application tracks the current location of the user through the google maps api and it provides the nearest restaurants available from that location on the basis of nearest distance .The application shows the nearest restaurants within 5 Km distance from the current location in the nearest restaurant page. The user can view the restaurant short information on the map view.

User can also view the restaurant location in the map and find the distance and route to the particular restaurant selected. Apart from this, application has the features of putting rate of specific restaurant.

This projects generally gives the easier tool for searching the restaurants that are available nearby and saves time with its modest and actual.

Acknowledgement

I would like to start by expressing my deepest gratitude to the Department of Computer Science & Engineering, MSCSE, United International University for providing me with an opportunity to work on a Project as a part of our syllabus. It is an honor for me to express our gratitude to our project supervisor Prof. Dr. Mohammad Nurul Huda for his constant support and guidance throughout the project. We are thankful to Prof. Dr. Mohammad Nurul Huda and Rubaiya Rahtin Khan for providing the constant feedbacks throughout the development of this report. Their suggestion throughout the development of this project helped us to cope up with different obstacles.

We are grateful to all our teachers for their suggestions and inspirational lectures that paved the towards the completion of this project.

Finally, I would like to thank our classmates for their valuable comments and their suggestion during this project. Any kind of suggestion or criticism will be highly appreciated and acknowledged.

Table of Contents

LIST OF FIGURES

Chapter 1

Introduction

1.1 Introduction

The “Food Chain Management Using GPS Tracking” is a GPS-based web application which helps people to find the closest restaurants based on the user’s current position and other specification like price, restaurant type, dish and more.

1.2 Background

Restaurant owners can provide their restaurant information using the web-portal. This information will act as the bases for the search results displayed to the user. An administrator also uses the web-portal in order to administer the system and keep the information accurate. The administrator can, for instance, verify restaurant owners and manage user information.

Furthermore, the software needs both Internet and GPS connection to fetch and display results. All system information is maintained in a database, which is located on a web-server. The software also interacts with the GPS-Navigator software which is required to be an already installed application on the user’s mobile phone or similar searching devices. By using the GPS-Navigator, users can view desired restaurants on a map and be navigated to them. The application also has the capability of representing both summary and detailed information about the restaurants. The application should be automatic adjust of fully responsive to browse from either a mobile phone application or similar services.

1.3 Objective

The Prime objective is to create a web-based application which could locate a list of restaurants. The user not only finds the nearest restaurant in the city, but also he can make a choice of the best restaurant based on the rating. The user has the facility to make a call directly and get notification of the restaurant as a list view after login to the system.

Chapter 2

Literature Review

2.1 Introduction

This chapter first describe about key concepts regarding to this the restaurant information store and dynamic menu create and the Google Location API followed by existing related work on social web application and their respective features and limitations will be discussed. Also discussed the functional and non-functional feature of the web application.

2.2 Restaurant Information Service

Restaurant information is the base data for this web application. All type of restaurant data is store by the admin/restaurant owner user. During restaurant information insert also insert the address, latitude and longitude for calculate the distance between user current positions.

Restaurant type metadata need to store for categorized the restaurant. Admin/Restaurant owner can input their desired restaurant category like Barbeque, Fast food, Chinese etc. This feature is special because of admin user don't need any bindings to add restaurant of different type.

Restaurant details like link, discount offers, discount prices etc are preserve for the use of public user. User can get notification regarding restaurant discount offer and price. Also user search and get the list view based on different criteria.

This project include the dynamic menu creation feature. The admin user can add the menu from the UI. Also include the authentication and authorization using role add and menu assign to role.

2.3 Nearest Restaurant Service

The admin user input all the restaurant desired information. Those input are used as a output for the public user. For nearest restaurant service need the following service:

2.3.1 Places Based Facilities

Places based services are services which gives the location information like latitude and longitude of the user. With over the 770 million GPS-enabled devices, places based services can be retrieved from mobile/laptop/tab devices. Places based facilities give the access to find the user and also find the user nearest restaurants. By incorporating the places based facilities with the web applications, some attractive feature related with the restaurants are:

- *Places of the restaurant*. Track the user current position and find the registered nearest restaurant can be notified to users.
- *Rating of the restaurant.* Restaurant rating feature user can choose the best restaurant from their nearest distance.
- *Discount offer of Restaurant.* Discount offer of different restaurant notified the user.

2.3.2 Google Map APIs

Google map API is the most popular application programming interface by which user can view their desired destination. Along with the destination they can find the path or nearest path where they want to reached. Google map API has the attributes to show the specific location. To represent the location API use latitude and longitude point. User can zoom in/out by clicking the specific point. Infowindow is another attributes of Google map. User can view the short description of the restaurant in map infowindow. User can easily access the registered nearest restaurant by view the user position and restaurant position in map.

2.4 Non Function Requirements of “Food Chain Management Using GPS Tracking”

1. Noticeable search facility
2. Search feature practice
3. Preview results in the list view
4. Preview results in the map view
5. Navigate to the information link
6. Get reply time from server
7. System extensibility, reliability
8. Responsiveness of the web application
9. Internet and GPS Connection
10. Communication Security
11. Restaurant Owner and Admin Login Account Security
12. Restaurant Owner , Admin and User Account Security
13. Restaurant Owner Create Account Security
14. Application extendibility, testability, portability

2.5 Limitations

This project has some limitations that are:

- User can only find the registered restaurant.
- Registered restaurant does not approved the address is correct or not.
- Mail sending service does not include.

2.6 Conclusion

Project detail life cycle have been discussed in this chapter. Besides, there have been information different service. After analysis the application features and limitation found gap which is, only registered restaurant can show on map not all restaurant. Next chapter discussed about the system architecture and development procedure for the system visual outcome.

Chapter 3

Project Development Procedure

3.1 Introduction

In this chapter describe the development and design procedure. And also the architectural life cycle presents in this chapter. Application lifecycle management (ALM) tools and technology is also describe the in this chapter.

3.2 Application Explanation

The main objective regarding the web application is to find the nearest restaurant from the user current position. The GPS device is used track the user current position. When user login into the web application, user's current position will be traced then user can search the nearest restaurants which is registered into the database. All the restaurants are displayed into the map and also displayed into the grid or list view. User can explore restaurants from the UI.

User review is the useful feature for this web application. By using this review other user can choose the best restaurant. Specific rating show into the map infowindow. Which is calculated by average rating calculation for specific restaurant.

Application server search the nearest restaurants which is stored in database. Then, from the search UI user can explore their desired restaurants based on different criteria.

3.3 Architectural Overview of the System

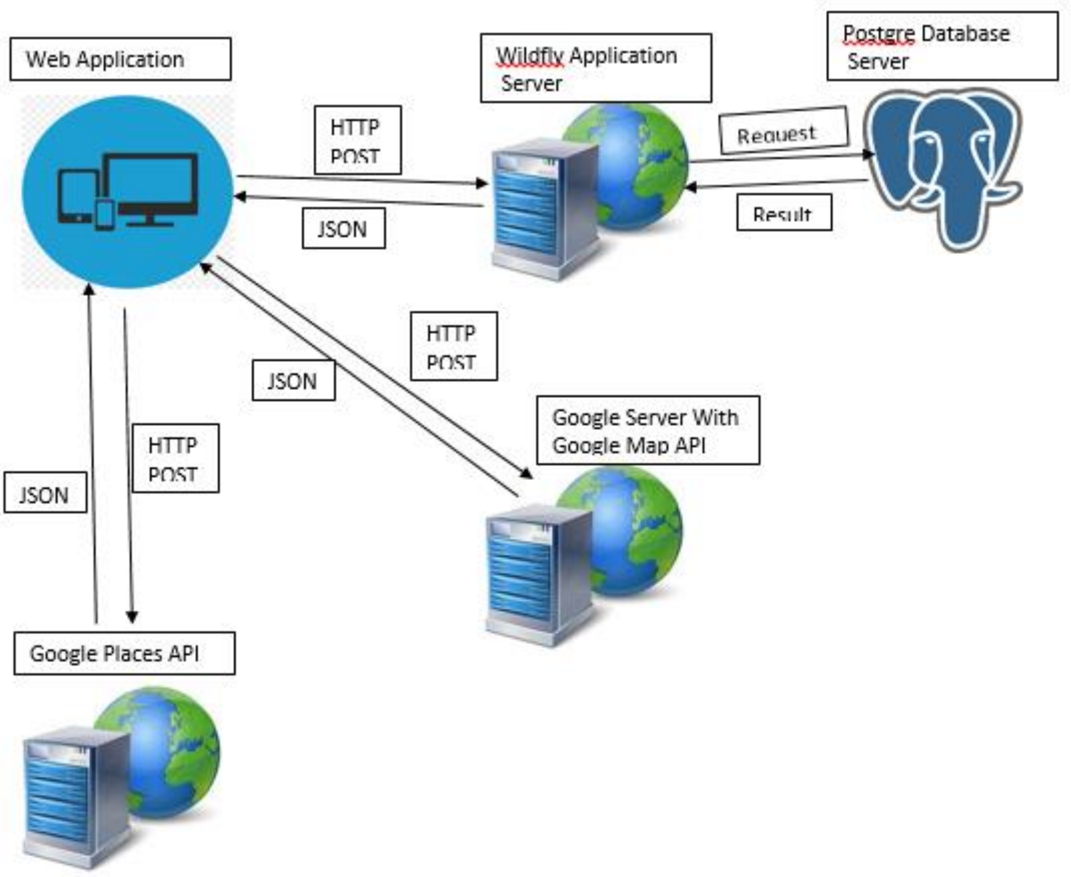


Figure 3.1: Architectural Overview of the System

From the above figure 3.1 presents the architectural overview of the system. There are mainly two server need for this projects. Web application communicates with the application server. Web application send HTTP request get result from application server in JSON format. And application server request send to database and retrieve data from database. Web application get the data from another two server. Google places server used to get address name, latitude and longitude data. By using this API we can get exact places with latitude and longitude data. Google map API server is used to show the user and restaurants position in the map. Both API server get HTTP request and send back response as a JSON format.

3.4 Application Lifecycle Management Process

Application lifecycle management is the process of project full lifecycle. This cycle start from development to deployment steps. The project need to develop as an organized way which is achieved by Agile methodologies. Following are some key point regarding Agile:

i) All the phases and communication are managed by the tools
ii) Documentation about project managed by tools
iii) Customer interaction increased during development lifecycle
iv) Frequently response after customer feedback

Agile is the process by which development task divide into small story. Task is developed with the small module. So that, story include into the specific sprint. By using agile technology module development and delivery process running fast.

Agile methodology has four types of activity during the sprint. First is daily scrum it is the daily activity for the project update discuss among the team. Sprint planning, review and retrospective are agile process also the activities of agile methodologies.

The concept of this project development is modularized. All the features are divide into separate module. As the application developed by modularized concept so, development, testing and deliver perform faster. It is incremental process so the testing process running effectively during the project lifecycle. For this incremental process project fault or issue identify is easy.

3.5 Tools and Technologies Choose for Development

Most of the latest technologies are used in this projects

3.5.1 Oracle Java JDK

Oracle Java JDK used for the application development on JAVA. All of the advanced API and development libraries are included in the JDK. Business logic code are compile and build using JAVA.

3.5.2 Java Server Faces (JSF)

JAVA Server Faces is the server side language and JSF is most commonly used language for front end UI development. In JSF technology UI and business logic are store in the server side. JSF incorporate most of the features like Component Based Framework, Integration with Expression Language, Support HTML5 and Bean Annotations. JSF can run on the different application server platform such as Oracle Weblogic server, Glasshfish client edition and JBoss. It is easily deployable front end language into those application server. It has a servlet container which act as a library named Tomcat and Jetty. Other than that, development guideline and help can easily found in the JSF documentation. Developer study curve is smooth because of availability resource on the websites or internet. JSF also work as a controller by binding with the JAVA code automatically. It handle the HTTP request and response. Bean is used as a controller class to redirect HTTP method GET, PUT, POST and DELETE request to server.

3.5.3 Front End and Back Technology

The Hypertext Markup Language (HTML5) is used for front end GUI functionality. Incorporate HTML5 components with the JSF we can use both HTML5 and JSF feature at a time. During rendering in browser all the JSF code will transform to HTML5 code.

Bootstrap is the CSS framework for designing web application fully responsive and auto adjustable with mobile and other devices. Bootstrap framework is used for UI design and style added in the pages. For adding Bootstrap, pages are fully responsive and auto adjustable with the desktop or mobile view.

Hibernate is used for interaction between object and database. In this project used Hibernate as an Object Relational Mapping (ORM). It's a JAVA based ORM to communicates with the RDBMS. Hibernate create a model class to integrate with the database tables. Also integrate other functionality of database like functions, procedures and other query services.

In the following figure shows that Hibernate communicate between RDBMS and Java Objects. It has a persistence file to configure database tables with Java model classes.

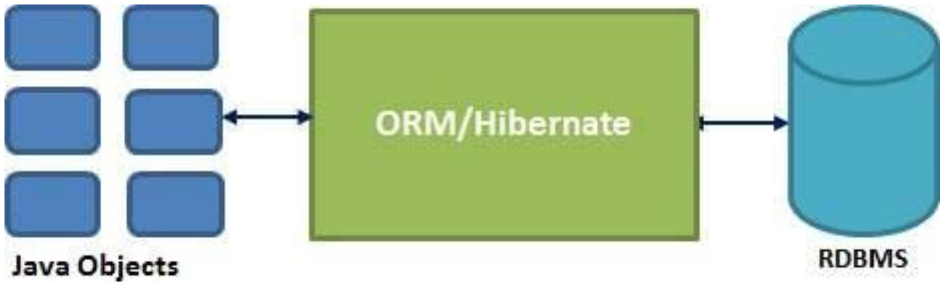


Figure 3.2: Hibernate Communication

3.5.4 PostgreSQL Database

I this project for structured query language (SQL) choose PostgreSQL. It is client edition which user can access as a relational database. It incorporate all the relational database features based on SQL standards. This database is required to save application information which is used by user and admin. Different tables are user, restaurant, restaurant type, rating, description, and discount offers and discount prices are created into the database. From this data user can view the application registered restaurants in the map view or others fature.

3.5.5 Google Places API

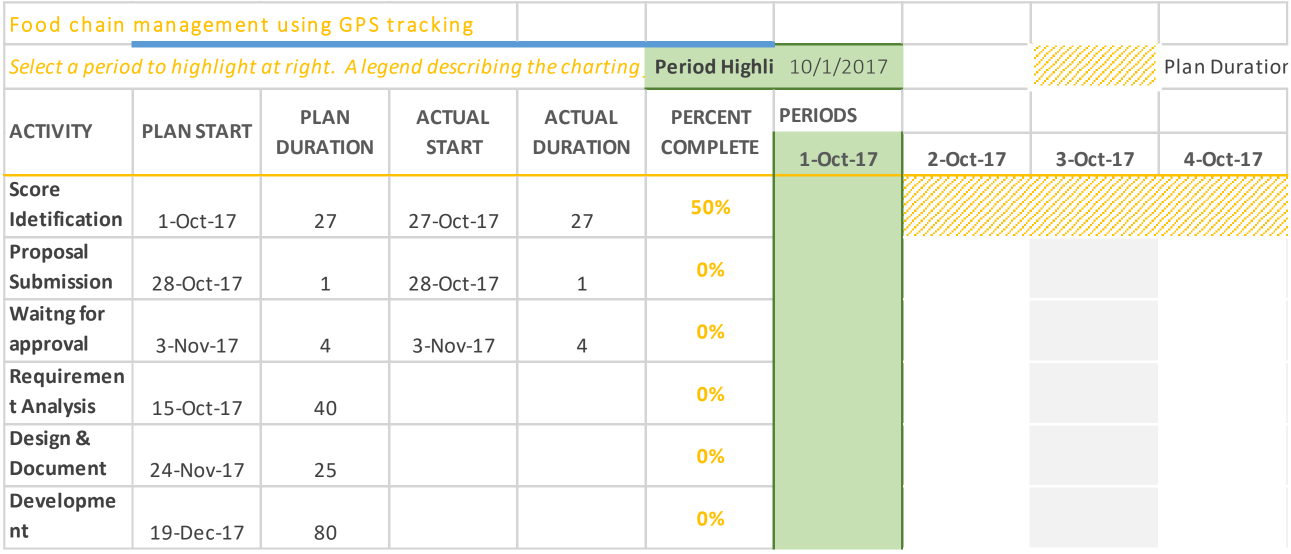
This project includes the Google places application programming interface (API). If any admin user search in the places Google places API shows the respective address relative to search address. Along with the addresses it gives the position of the address. The position attributes point's two data one is latitude and another is latitude. User may be not interested to entry the address and manually. So that, Google map API used in this project. By using this API address and other attributes can retrieve easily. Google places API, used for find the address and store the address, latitude and longitude.

3.5.6 Google Maps API

Google Maps Android API provides the API libraries to create interactive and feature rich maps in Web application. All the restaurant within an area can be anchored to specific positions on the map using the API libraries provided. Google Maps Android API is used to show the map of the restaurant outlets in the web application. Besides showing, user also can zoom and pan the map.

3.6 Project Scheduling

The project analysis and design schedule generate according to the following Figure 3.3



Food chain management using GPS tracking

Select a period to highlight at right. A legend describing the charting Period Highli 10/1/2017 ▨ Plan Duratior

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE	PERIODS 1-Oct-17	2-Oct-17	3-Oct-17	4-Oct-17
Score Idetification	1-Oct-17	27	27-Oct-17	27	50%		▨	▨	▨
Proposal Submission	28-Oct-17	1	28-Oct-17	1	0%				
Waitng for approval	3-Nov-17	4	3-Nov-17	4	0%				
Requirement Analysis	15-Oct-17	40			0%				
Design & Document	24-Nov-17	25			0%				
Development	19-Dec-17	80			0%				

Figure 3.3: Project Schedule

According to application lifecycle this project divide into different phases:

i) Preparation for Proposal Submission

Proposal preparation is the first phase of this project. After the topic selection, the next task is scope identify and declaration. Then need to prepare project proposal with highlights point of this project. Then need to submit this project proposal to accept the concept of this project

ii) Requirement Analysis

In this phase, all the requirements and features need to identify. Project scope divide into several features. All the feature have to break down into specific task. Here project functional and non-functional requirements are included.

iii) Tools and Technology Selection

For this project developments many other tools and technology compared. As this project is include fully responsive feature so that, modern tools are selected for this project. Both of front-end and back-end technology are selected in this phase. Here choose, UI, UX, application server, database and application logic development language.

iv) Design, development and testing

Here choose the project whole architecture and also design different diagram to present this project. All type of diagram like use case diagram, class diagram, sequence diagram and others are design in this phase. After design complete, development and testing start. All the user experience (UX) or UI and project object choose. This object represents as a database tables. After that, business logic and database logic developed in this phase. With the writing of business logic have to developed unit test case. After development is completed all the unit test have to pass. Testing is also the part of this project. Test case and test plan need to write down. After completing development testing cycle start. Some bugs arise in this phase. After fixing all the bugs this project is ready to deployment.

v) Deployment

Application deployment procedure is last phase of a system. In this project WildFly application server is used to deploy the web application. WildFly is platform independent application server.

3.7 Conclusion

The application lifecycle management of this project is used by agile methodology. By using Agile process the application development, testing and deployment phases is more smooth and faster. The project development procedure and other operation is described in this chapter. Following chapter describe the application design technique and strategy.

Chapter 4

STRATEGY AND DESIGN

4.1 Introduction

This chapter presents the database entities and application model or objects relationship. Here class diagram shoe the system object communication between one object to another. And the sequence diagram show the functional sequence of the system. Also discussed about use case diagram and data flow diagram. Model class is replica of the database entities relationship.

4.2 Class Diagram

Following figure 4.1 presents the object relationship. This objects are also known as class. There are different class use in the system and major 8 class or objects functionality is maximum. The objects are:

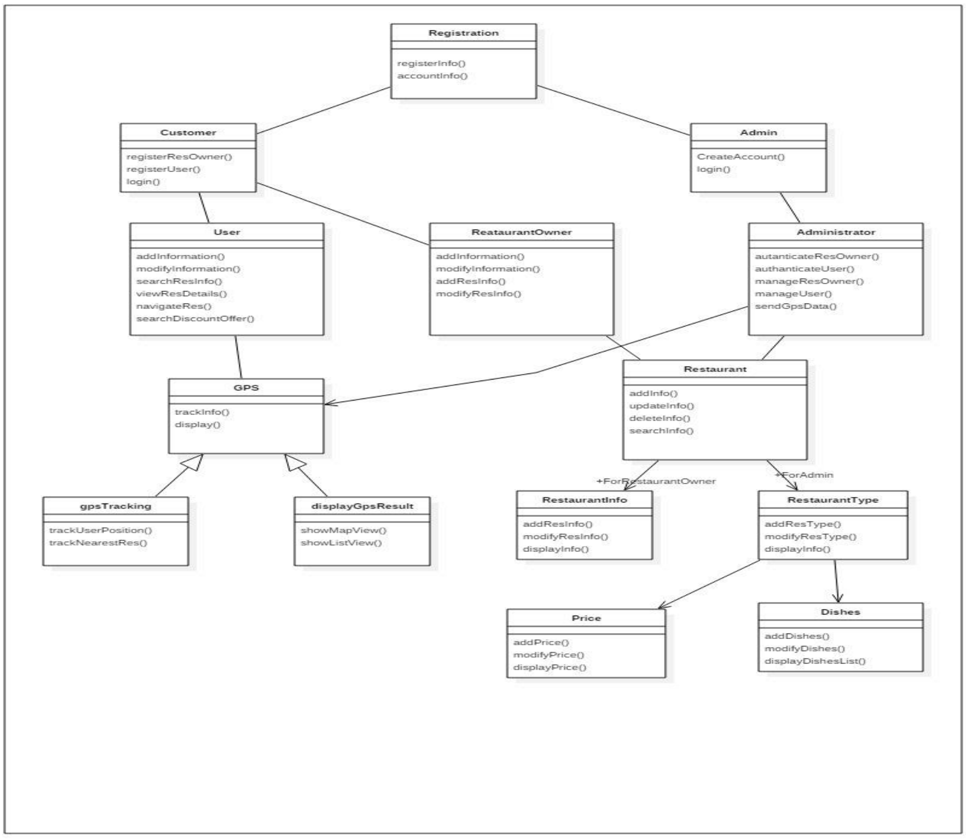


Figure 4.1: Object/Class Interaction Diagram

4.3 Database Design

Database tables are:

i) user

Preserve the user login information like name, email, address, city, Zip code and last update date. Table also include the user role number to specify the user role. Based on role menu assigned to user.

ii) restaurant

Preserve the initial data of restaurant like name, type, address etc. For identify category of restaurant type_id is insert in the table for identify the type of the restaurant.

iii) restaurant_type

Store the restaurant category. Which is use as a metadata, during the restaurant save operation perform.

iv) restaurant_detail

Store the details information of restaurant. The details information like link, offer and price are store for further search by the user.

v) restaurant_rating

Store the restaurant rating. Restaurant rating stored separate data based on user and restaurant. For average rating calculation perform by data group of specific restaurant

vi) feature

Store the menu of the system. Both user and admin can access the assigned menu after login. Here menu and sub-menu data store as a parent child relation based.

vii) role_feature

Store the role and menu information. This table show the information of which menu assigned in which role.

viii) role

Store the role list which is use for user create and menu assign.

ix) location

Store the restaurant latitude and longitude. Latitude and longitude insert based on restaurant.

4.3 Data Flow Diagram

“Food Chain Management Using GPS Tracking” is GPS based application. Administrator, Restaurant owner and Public users data interact with the GPS for track the current location. Data of the location automatically load from the Google map and places api.

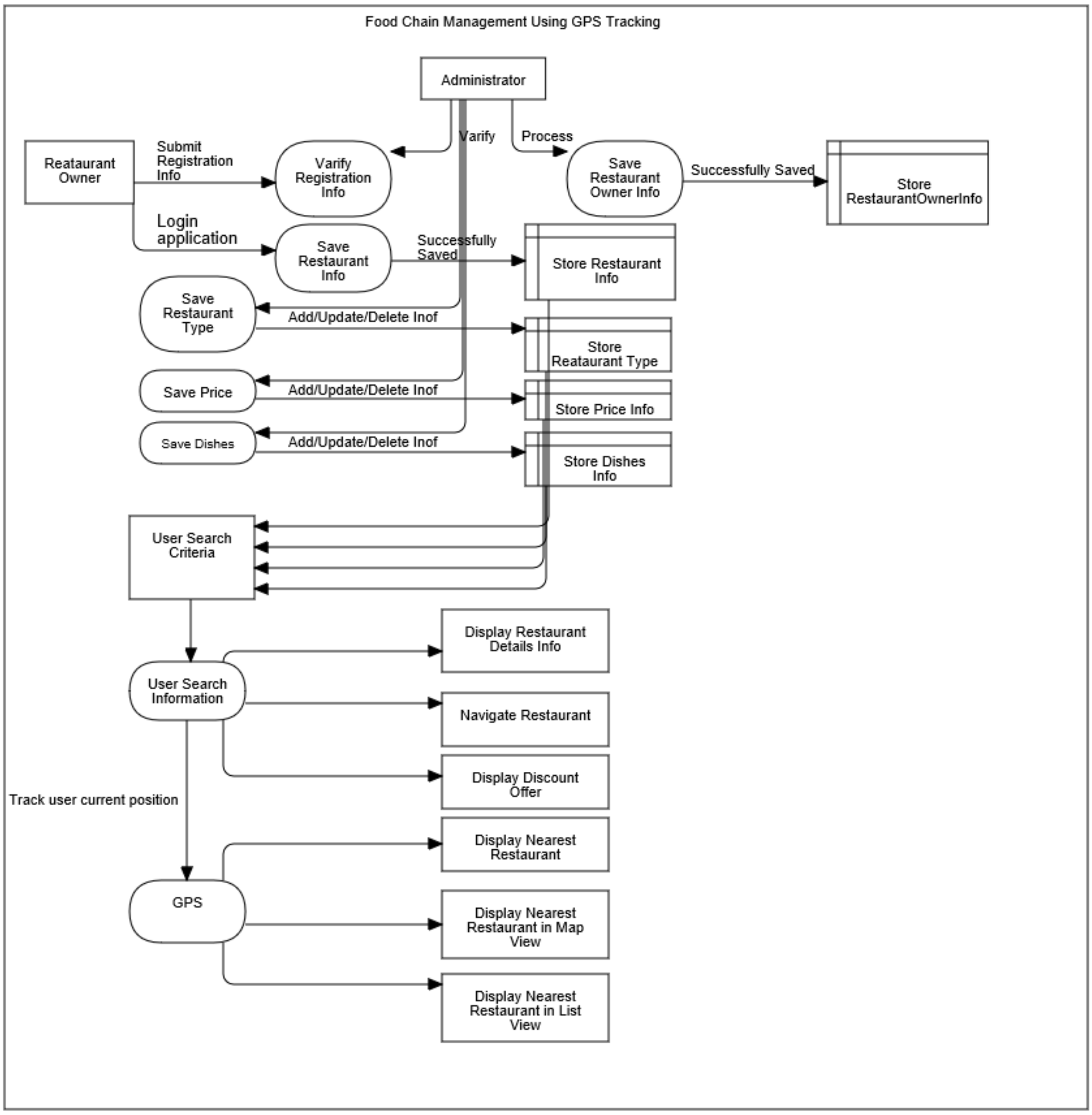


Figure 4.2: Data Flow Diagram

4.4 Physical Architectural Design

Physical architectural diagram has different section to communicate one to another server. System divide into different modules. The system modules communicate with the web server and database server by sending HTTP request. Web server interact with the database server through LAN connection. GPS enable device connected with the modules. Modules retrieve data from GPS devices and parse data according to system requirements. Whole system required the GPS device, modules, web server and database server. External two API use for this system. Both API data pass using HTTP request and response. Google places API gives the location information and related latitude and longitude data. System pass the data to Google Map API as JSON format for smooth interaction between system and API.

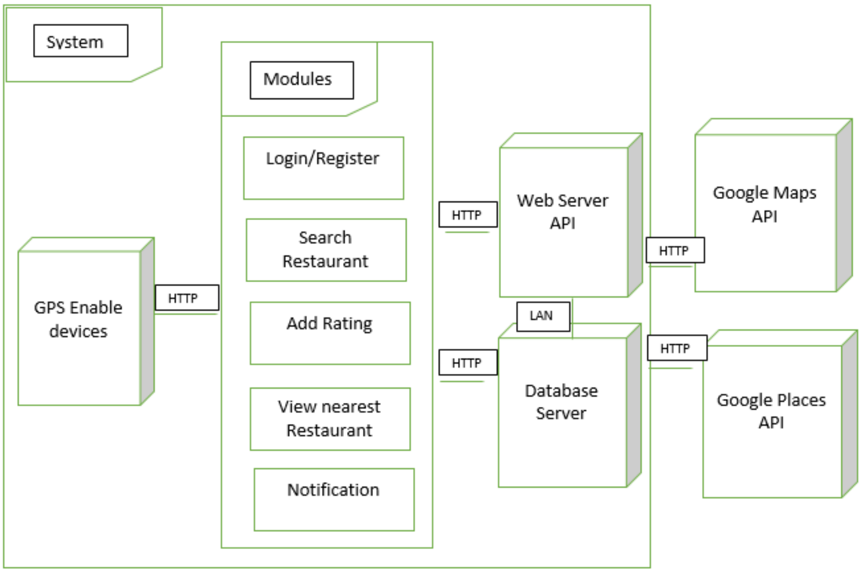


Figure 4.3: System Physical Architectural Diagram

4.5 User Interaction Diagram

User interaction diagram or Use case diagram is the primary diagram of the system. Here we need to identify the system user and their functionality. Those functionality which are user perform. All the function presents by using circle. By using straight line between user and circle show the user wise operation. This system has two category user one is public user and another is admin user. Public user can view the web portal part.

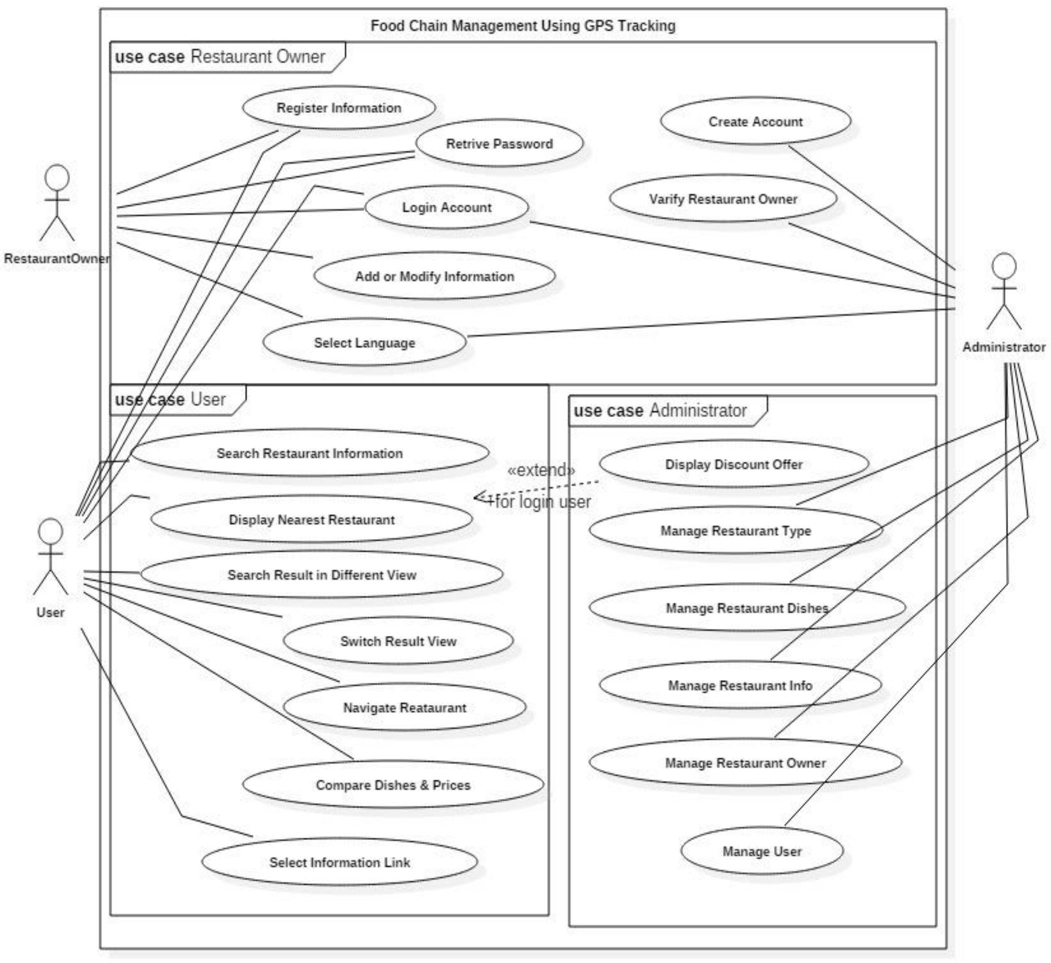


Figure 4.4: User Interaction Diagram

4.6 Calculate Nearest Restaurant

To calculate the nearest restaurant distance from user current position we use Haversine formula. For calculate two latitude and longitude position this formula is needed. By determining the user current latitude and longitude and also the restaurant latitude and longitude this formula calculated latitude one and latitude two distance, on the other hand calculate the longitude one and longitude two distance. Formula of the distance between two points on the earth is given bellow:

$$= \quad 2 - \quad 1$$

$$= \quad 2 - \quad 1$$

$$= (\sin (\quad \overline{\qquad\qquad}))^2 + \cos(\quad 1) * \cos(\quad 2)$$

$$2 \quad * \quad (\sin (\quad \overline{\qquad\qquad} \, 2 \quad))$$

$$= 2 * \quad 2(\quad (\,), \quad (1 - \,)) \quad 2$$

$$= \quad *$$

From the above formula R is radius of the earth. Which radius is 6378 kilometers and based on miles is 3958

Time complexity of this formula is optimized. Individual calculation of distance between user and restaurant is time consuming. This formula only calculate the distance inside the rectangle box, so that the time is faster. The following figure represents the calculating mechanism of this formula. Here, in figure 4.5 show red and yellow color where red is restaurants and yellow show the filtered restaurants surrounded by the rectangle box. Using these filtered restaurants calculate the distance between user and restaurants. This technique is faster than other distance calculation technique.

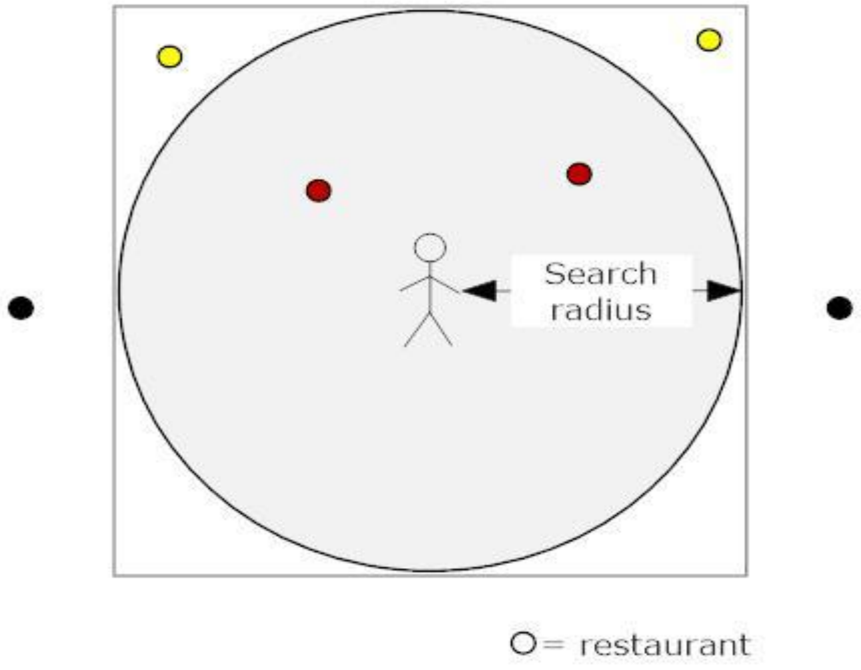


Figure 4.5: Get Nearby Restaurants

4.7 Function Time and Order Sequence Diagram

Sequence diagram of a system described the function sequence and function to function communication. It is provide the structure of operation order. In a sequence diagram rectangle box represent the system activity or function. The vertical box show the time duration of a specific task. By using arrow request order sequence to a function and get   response within a time duration can visualize. Following are the two sequence diagram:

4.7.1 For User

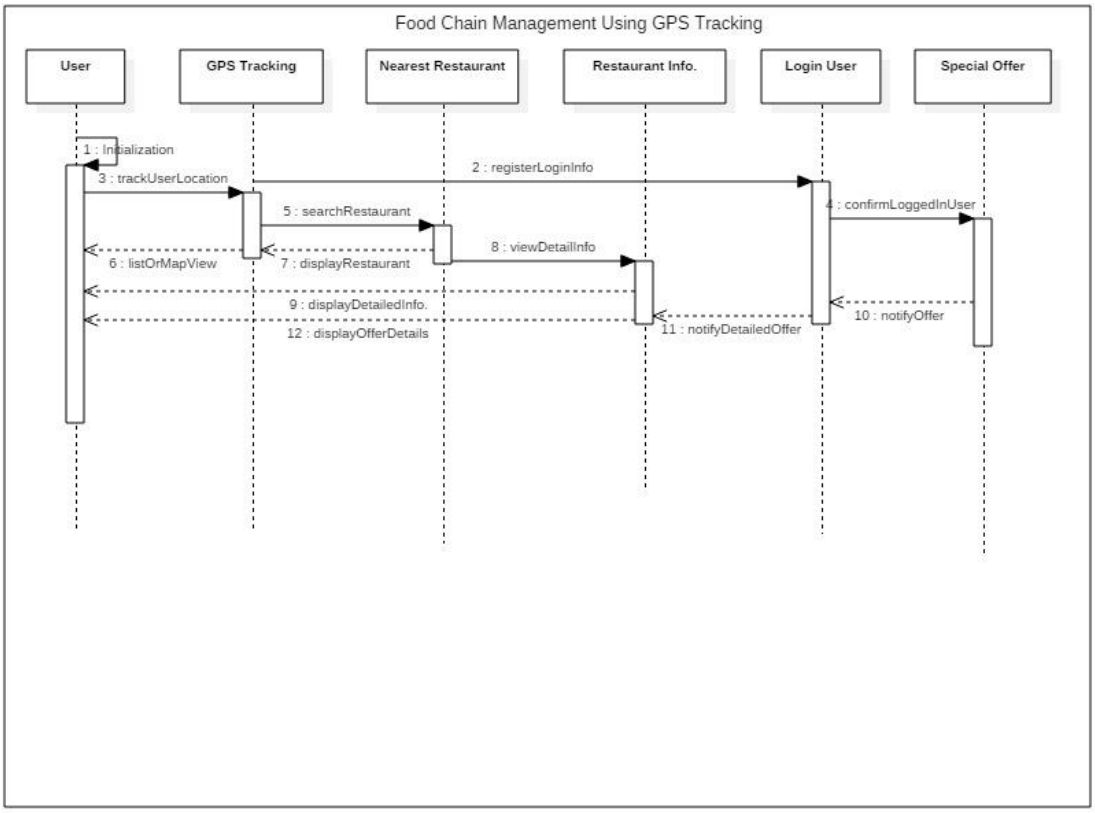


Figure 4.6: Sequential order of functions for User

4.7.2 For Restaurant Owner/Administrator

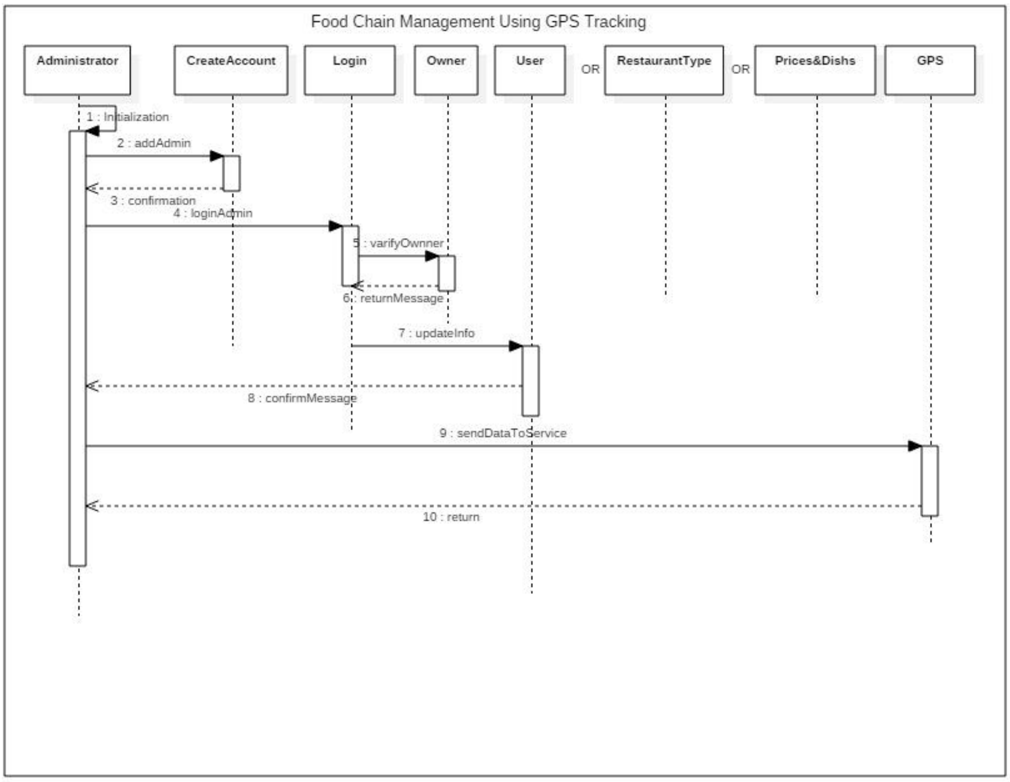


Figure 4.6: Sequential order of functions for Restaurant/Admin

4.8 Conclusion

This chapter described all type of modeling diagram. The use case diagram, sequence diagram, class diagram, data flow diagram, physical architectural diagram are required to understand the system requirements, flow and functionality. Before development start this diagram is the visual view of whole system. The next chapter discussed about system outcome after development perform. UI functionality and behavior describe in the following section.

Chapter 5

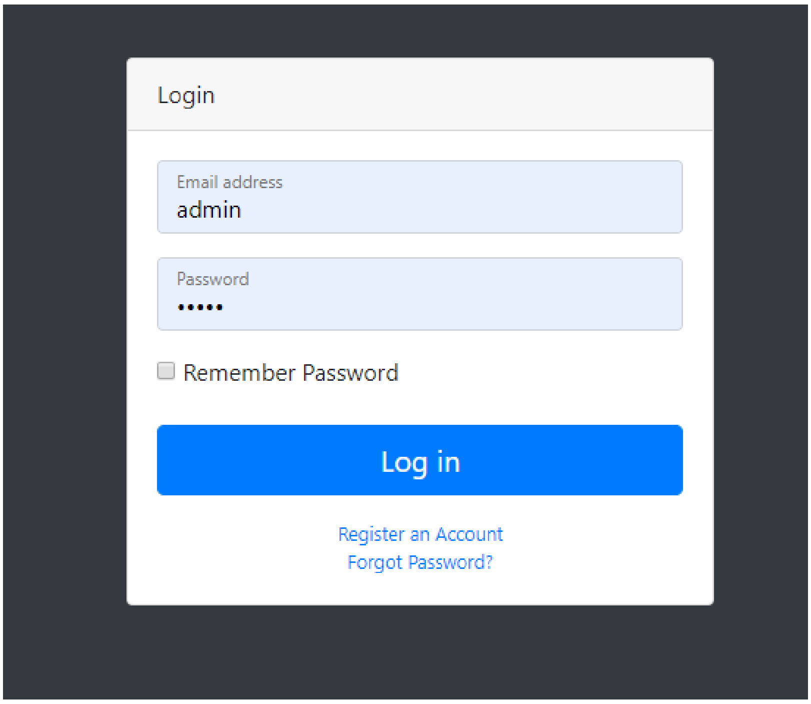
FINAL OUTCOME AND USER GUIDE

5.1 Introduction

Throughout this section, show the final outcome of the web application. After development complete testing cycle was performed to justify the system act properly. All the features are clarified by include screenshots of the web application.

5.2 Login Page

The admin and user can login using this page. Also user and admin login information remember for future use.



Login

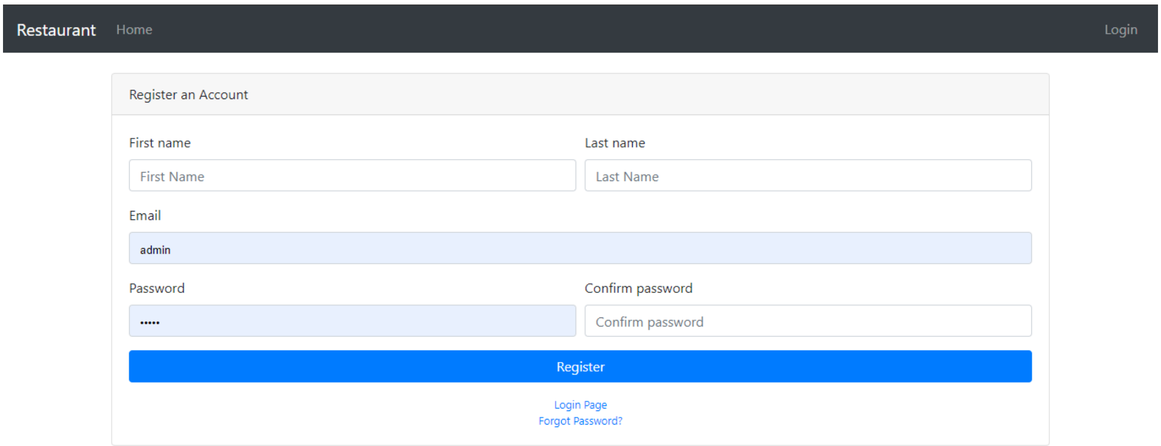
Email address: admin

Password: •••••

□ Remember Password

5.3 User Create

Public user access the 5.3.1 user create page for save user the information.



Register an Account

First name: First Name

Last name: Last Name

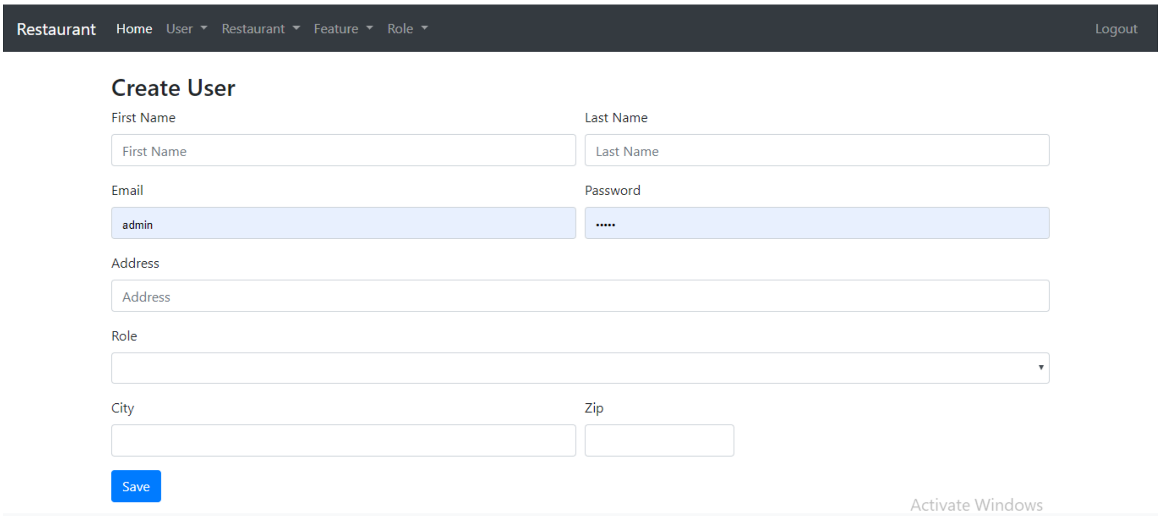
Email: admin

Password: •••••

Confirm password: Confirm password

Figure 5.3.1: Public user create

Admin user use the different 5.3.2 page for save admin /restaurant owner information.



Create User

First Name: First Name

Last Name: Last Name

Email: admin

Password: •••••

Address: Address

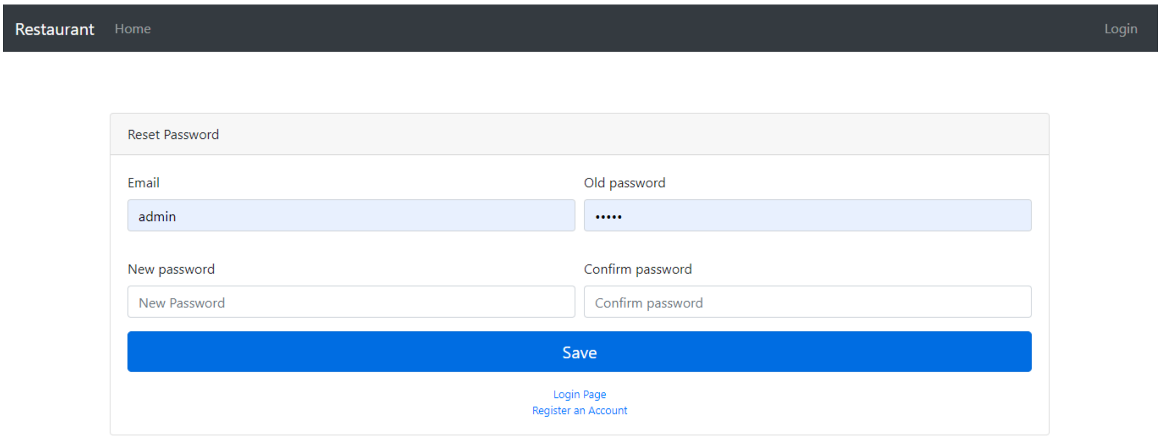
Role:

City:

Zip:

Figure 5.3.2: Admin user create

Public user and admin user retrieve their password using 5.3.3 password change screen.



Reset Password

Email: admin

Old password: •••••

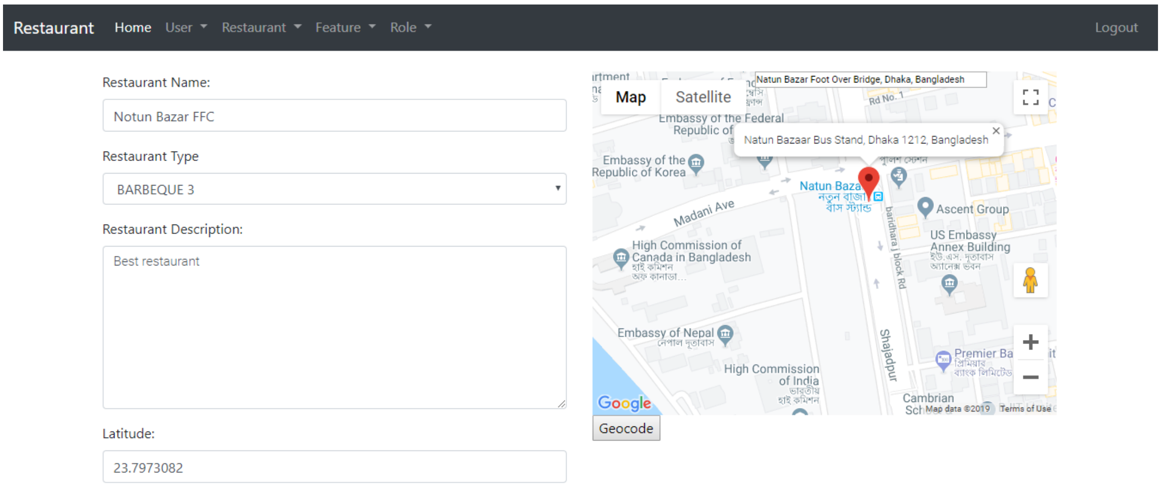
New password: New Password

Confirm password: Confirm password

Figure 5.3.3: Change Password

5.4 Features

Admin/Restaurant owner can save the restaurant information using 5.4.1 following UI



Restaurant
Home
User
Restaurant
Feature
Role
Logout
Map
Satellite
Natun Bazar Foot Over Bridge, Dhaka, Bangladesh
Natun Bazaar Bus Stand, Dhaka 1212, Bangladesh
Embassy of the Federal Republic of
Embassy of the Republic of Korea
Natun Bazar
Madani Ave
Ascent Group
US Embassy Annex Building
High Commission of Canada in Bangladesh
Embassy of Nepal
High Commission of India
Shahjadpur
Premier Ba
Cambrian Sch
Google
Map data ©2019
Terms of Use
Geocode

Restaurant Name: Notun Bazar FFC

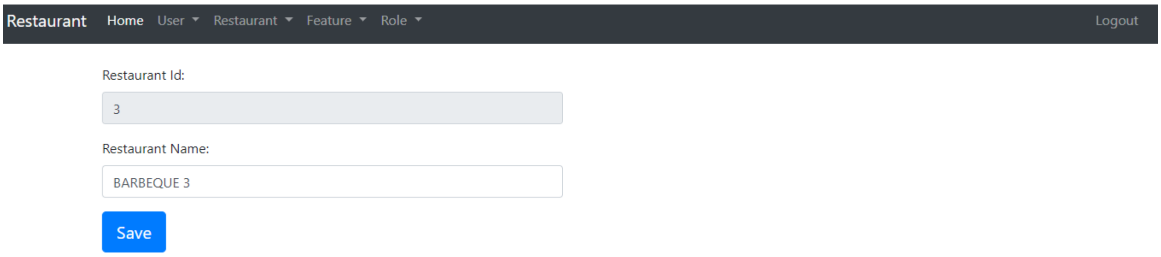
Restaurant Type: BARBEQUE 3

Restaurant Description: Best restaurant

Latitude: 23.7973082

Figure 5.4.1: Restaurant information create

Restaurant category or restaurant type metadata can be save by admin/restaurant owner using 5.4.2 restaurant type create page

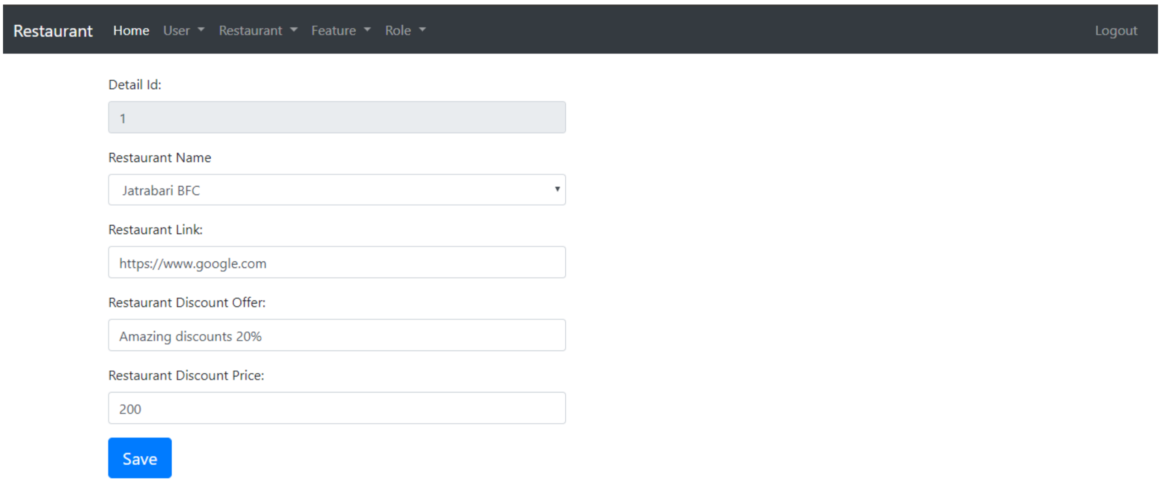


Restaurant Id: 3

Restaurant Name: BARBEQUE 3

Figure 5.4.2: Restaurant type create

Save information of restaurant details like link, discount offer, discount price using 5.4.3 UI



Detail Id: 1

Restaurant Name: Jatrabari BFC

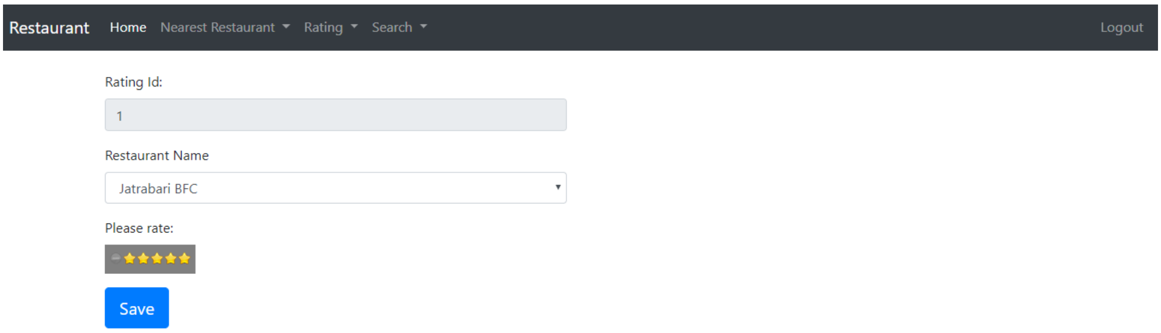
Restaurant Link: https://www.google.com

Restaurant Discount Offer: Amazing discounts 20%

Restaurant Discount Price: 200

Figure 5.4.3: Restaurant detail create

User can rate the best restaurant using 5.4.4 restaurant rating screen



Rating Id: 1

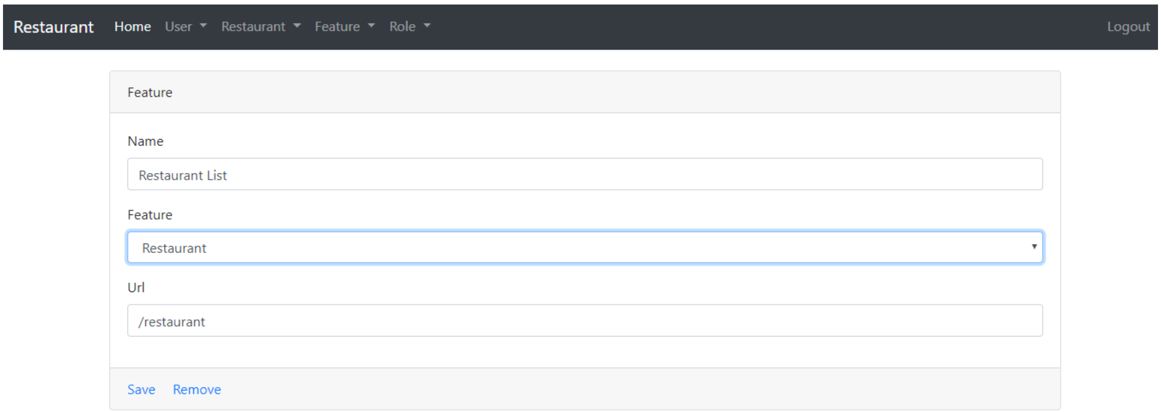
Restaurant Name: Jatrabari BFC

Please rate: ★★★★★

Figure 5.4.4: Restaurants rating create

5.5 Feature Permission

Admin user can insert dynamic feature (5.5.1) for this application also create the role list (5.5.2). After role list create menu can be assign in specific role (5.5.3).



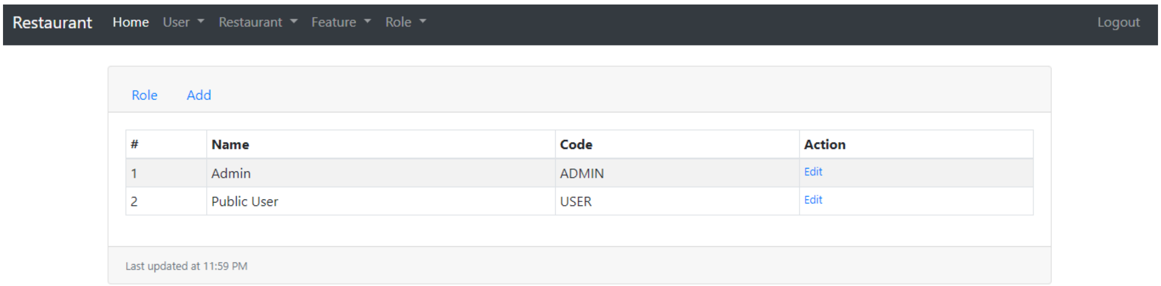
Feature

Name: Restaurant List

Feature: Restaurant

Url: /restaurant

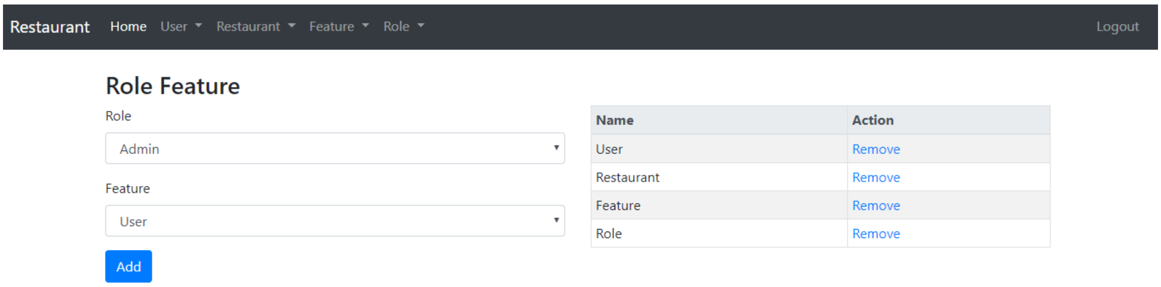
Figure 5.5.1: Application feature create



#	Name	Code	Action
1	Admin	ADMIN	Edit
2	Public User	USER	Edit

Last updated at 11:59 PM

Figure 5.5.2: Application role list



Role Feature

Role: Admin

Feature: User

Name	Action
User	Remove
Restaurant	Remove
Feature	Remove
Role	Remove

Figure 5.5.3: Application feature assign

5.6 Nearest Restaurant

After admin or public user can view the home page after login the application

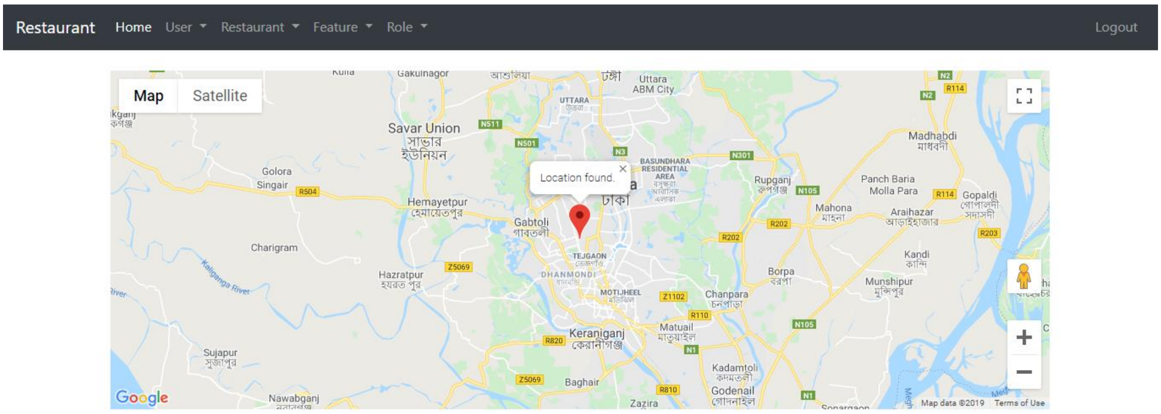


Figure 5.6.1: Application home after login

Following UI show the nearest restaurant from the current user position track using GPS tracker

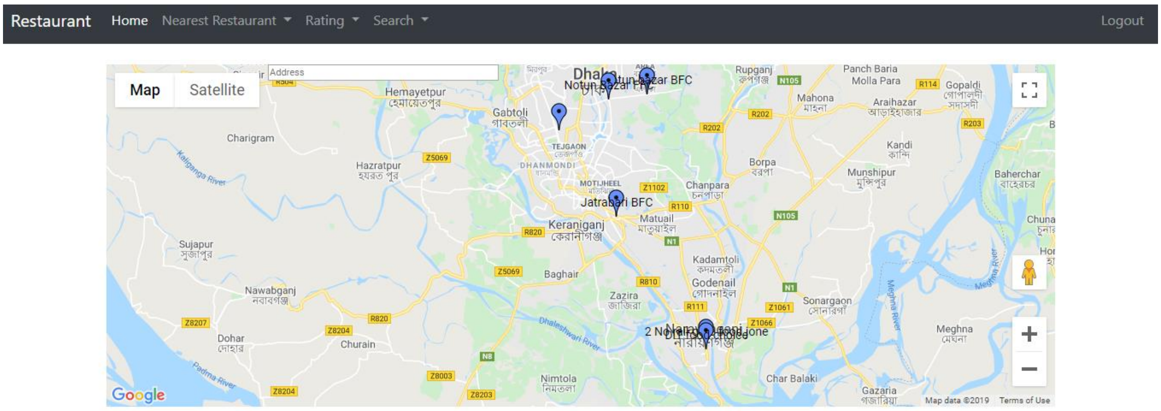
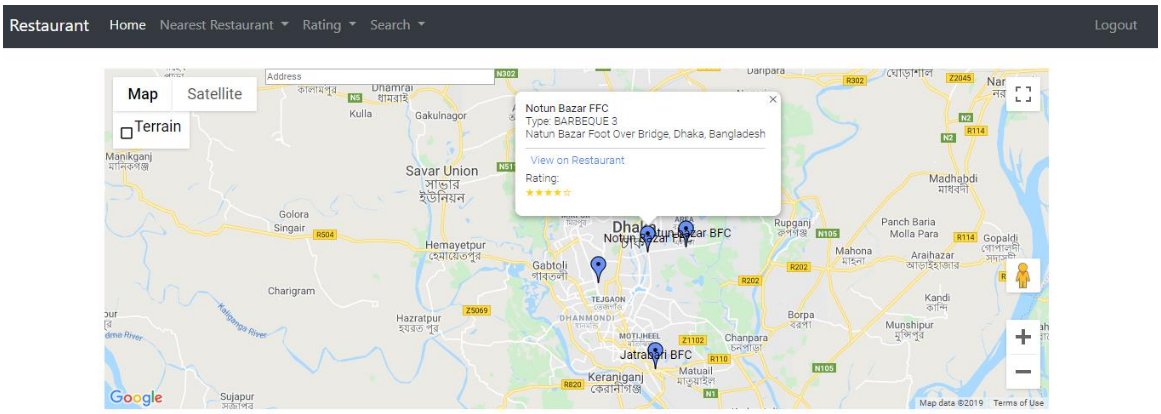


Figure 5.6.2: Nearest restaurant

Restaurant details information show on map info window or list view. User can search specific restaurant based on different criteria

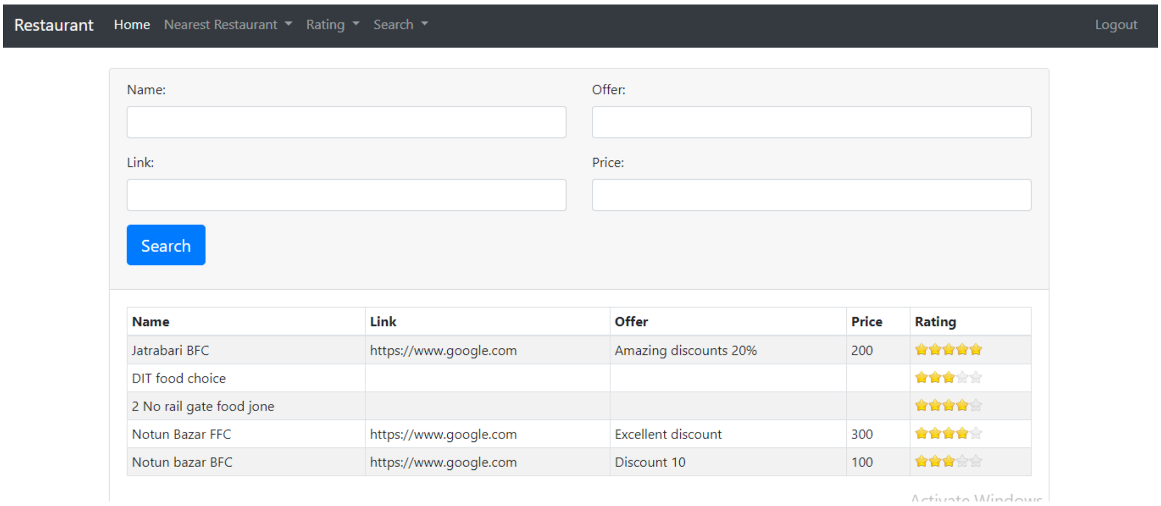


Notun Bazar FFC
Type: BARBEQUE 3
Natun Bazar Foot Over Bridge, Dhaka, Bangladesh

View on Restaurant

Rating:
★★★★☆

Figure 5.6.3: Nearest restaurant information



Name	Link	Offer	Price	Rating
Jatrabari BFC	https://www.google.com	Amazing discounts 20%	200	★★★★★
DIT food choice				★★★☆☆
2 No rail gate food jone				★★★★☆
Notun Bazar FFC	https://www.google.com	Excellent discount	300	★★★★☆
Notun bazar BFC	https://www.google.com	Discount 10	100	★★★☆☆

Figure 5.6.4: Restaurant search list view

Application collect all users rating from restaurant rating page then the average ratings show in the search page (5.6.4). Based on rating user can choose the best or their desired restaurant.

5.7 Conclusion

Web application different or dynamic features are like user navigate the specific restaurant from the restaurant list view page or they can navigate the restaurant from info window of map view. Screenshots of the web application we can visualize the features and functionality of this project. After registration of restaurant user can view, navigate and choose the best nearest registered restaurants.

Chapter 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion

This project is simplify the searching technique of the registered nearest restaurants from user current location. User can give their feedback using restaurants rating .The average review of the restaurants helps the other user to choose the best restaurant. This project use the GPS tracker from which device user access the application. GPS enabled device is chosen because of the application can easily track the user current location from satellite and most of the modern device have this built-in GPS feature. GPS connection is required for this project to track user current position. Also required internet connection to get data from Google map API and send the API data to application server. If connection problem occur wrong data will send to the server. Cannot retrieve data from map API, So that the map cannot show the restaurant position or may be show the incorrect location.

6.1 Limitations

As this is my first project related to restaurant domain and work with the google map API. I have faced some obstacles to parse the API data. So that I cannot implement some features those are-

1. Display all restaurant which are registered by this application owner or admin.
2. The restaurant approved system does not incorporate in this application.
3. Mail sending option does not incorporate.

6.2 Future Work

This project has some limitation, so that there are some option to improve the application. In future I want to incorporate all type of improvement in my restaurant database which restaurant registered by restaurant owner or admin and also develop in-built registered restaurant in my restaurant database. Another future I have to incorporate in my project that is, my application get the feedback from user

And also gather knowledge from user experience to show the nearest restaurant. Based on user experience like user type or which type of restaurant they are interested, according this criteria registered user get notification and send email by the application.

References

[1] “Class Diagram”, https://en.wikipedia.org/wiki/Class_diagram.

[2] “Use case”, http://en.wikipedia.org/wiki/Use_case 6.

[3] “Data Model”, http://en.wikipedia.org/wiki/Data_model.

[4] “User Interface Design”, http://en.wikipedia.org/wiki/User_interface_design.

[5] “Software Testing”, http://en.wikipedia.org/wiki/Software_testing.

[6] “Solid Principle”, https://scotch.io/bar-talk/s-o-l-i-d-the-first-five-principles-of-object-oriented-design

[7] “Front End”, https://www.tutorialspoint.com/jsf/index.html.

[8] “Primefaces Map API”, https://www.primefaces.org/showcase/ui/data/gmap/basic.xhtml.

[9] “Google Map Platform”, https://developers.google.com/maps/documentation/javascript/tutorial

[10] “Primefaces Component”, https://www.primefaces.org/showcase/

[10] “Postgree Database”, http://www.postgresqltutorial.com/

[10] “Application Server”, https://docs.wildfly.org/17/Getting_Started_Developing_Applications_Guide.html

Appendix A

This section includes hardware and software specification and required tools and technology for web application.

Hardware Specification:

1. CPU – i3
2. Disk Space – 5 GB
3. CPU Speed – 1GB RAMs

Software Specification:

1. Operating System (Windows-10,8,7)
2. JVM (Java Virtual Machine)
3. PostGreSQL Server
4. WildFly Application Server
5. Internet Connection
6. Installed GPS application

Software Architecture:

1. Eclipse or Netbeans IDE for application service development
2. Brackets or Notepad++ IDE for front end development
3. PostGreSQL for Database
4. pgAdmin development IDE for database development
5. HTML, Bootstrap, JavaScript API library
6. GPS(API) application service
7. RESTful service with JAX-RS(API)
8. Primefaces for UI component like rating and gMap functionality add
9. **Identity management and security** supported by Auth0 and JSON Web Tokens (JWT)
10. JavaScript library(JSF and HTML5) for UI (User Interface) component functionality
11. Supporting library for web service communication
12. Domain knowledge on Restaurant features and discount offer

13. Information about specific restaurant dishes and prices
14. Knowledge about Users search criteria
15. Knowledge about Users display result mode(List, Map etc)
16. Domain knowledge of Restaurant owner about restaurant information

Some Code from the project:

Sample code of user register screen. Following code include HTML5, JSF, CSS3

```
<ui:composition
   template="#{template.name}"
            xmlns="http://www.w3.org/1999/xhtml"
            xmlns:h="http://xmlns.jcp.org/jsf/html"
            xmlns:ui="http://xmlns.jcp.org/jsf/facelets"
            xmlns:jsf="http://xmlns.jcp.org/jsf">
   <ui:param name="title" value="Register" />
   <ui:define name="breadcrumb">
      <ol class="breadcrumb">
         <li class="breadcrumb-item">
            <a href="#">User</a>
         </li>
         <li class="breadcrumb-item active">Registration</li>
      </ol>
   </ui:define>
   <ui:define name="content">
      <div class="card card-register mx-auto">
         <div class="card-header">Register an Account</div>
      <div class="card-body">
        <form jsf:id="formRegister">
          <div class="form-group">
            <div class="form-row">
              <div class="col-md-6">
                <div class="form-label-group">
                  <label for="firstName">First name</label>
                  <input jsf:id="firstName" name="firstName" type="text" jsf:value="#{userBean.user.firstName}" class="form-control" placeholder="First Name"/>

                </div>
              </div>
              <div class="col-md-6">
                <div class="form-label-group">
                  <label for="lastName">Last name</label>
                  <input jsf:id="lastName" name="lastName" type="text" class="form-control" jsf:value="#{userBean.user.lastName}" placeholder="Last Name"/>

                </div>
              </div>
            </div>
          </div>
          <div class="form-group">
```

```
            <div class="form-label-group">
              <label for="inputEmail">Email</label>
              <input jsf:id="email" name="email" type="email" class="form-control"
jsf:value="#{userBean.user.email}" placeholder="Email"/>

            </div>
          </div>
          <div class="form-group">
            <div class="form-row">
              <div class="col-md-6">
                <div class="form-label-group">
                  <label for="inputPassword">Password</label>
                  <input jsf:id="password" name="password" type="password"
class="form-control" jsf:value="#{userBean.user.password}" placeholder="Password"
required="required"/>

                </div>
              </div>
              <div class="col-md-6">
                <div class="form-label-group">
                  <label for="confirmPassword">Confirm password</label>
                  <input type="password" jsf:id="confirmPassword"
name="confirmPassword" class="form-control"
jsf:value="#{userBean.confirmPassword}" placeholder="Confirm password"
required="required"/>
```

```
                                </div>
                            </div>
                        </div>
                    </div>
                    <button class="btn btn-primary btn-block" jsf:action="#{userBean.savePublicUser}" type="submit">Register</button>
                </form>
                <div class="text-center">
                    <a class="d-block small mt-3" href="login.html">Login Page</a>
                    <a class="d-block small" href="forgot-password.html">Forgot Password?</a>
                </div>
            </div>
        </div>
    </ui:define>

</ui:composition>
```

```
                                </div>
                            </div>
                        </div>
                    </div>
                    <button class="btn btn-primary btn-block" jsf:action="#{userBean.savePublicUser}" type="submit">Register</button>
                </form>
```

```
            <div class="text-center">
              <a class="d-block small mt-3" href="login.html">Login Page</a>
              <a class="d-block small" href="forgot-password.html">Forgot
Password?</a>
            </div>
          </div>
        </div>
      </ui:define>


</ui:composition>
```