



**CHRIST**  
(DEEMED TO BE UNIVERSITY)  
B A N G A L O R E • I N D I A

## **Student Support System**

**(newbie.com)**

By

Adarsh Verma (2147102)

C Saranya (2147142)

Reuben Mohan Kurian(2147129)

Under the guidance of

Dr. Sagaya Aurelia P

A Project report submitted in partial fulfillment of the  
requirements for the award of degree of Master of  
Computer Applications of CHRIST (Deemed to be  
University)

May – 2022



**CHRIST**  
(DEEMED TO BE UNIVERSITY)  
BANGALORE · INDIA

## CERTIFICATE

*This is to certify that the report titled **Newbie.com** is a bonafide record of work done by **Reuben Mohan Kurien (2147129)**, **C Saranya(2147129)**, and **Adarsh verma (2147102)** of **CHRIST (Deemed to be University)**, Bengaluru, in partial fulfillment of the requirements of **III Trimester MCA** during the year 2022.*

**Head of the Department**

**Faculty In-charge**

Valued-by: Dr. Sagaya Aurelia P

Names : Reuben Mohan Kurien  
C Saranya  
Adarsh vema

Register Number(s) : 2147129  
2147142  
2147102

Examination Centre : CHRIST (Deemed  
to be University)

Date 22/05/22

## **I. Acknowledgments**

We want to take this opportunity to express our gratitude to all the people who have, in various ways, helped in the successful completion of our project. We must convey our gratitude to our guide **Dr. Sagaya Aurelia P** for giving us a constant source of inspiration and help in preparing the project, personally correcting our work, and providing encouragement throughout the project. We also thank all our friends and other faculty members for steering us through the tough and easy phases of the project in a result-oriented manner with concerned attention.

## II. List of Figures

Figure Number	Figure Description	Page Number
2.2.1	Use case Diagram	10
3.1.1	Architectural Diagram	17
3.2.1	ER Diagram	19
3.3.1	Relationship Schema	20
3.4.1	Table Description	21
3.5.1	Interface Design	27
6.1.1	Screen Shots	53

### III. List of Table

Table Number	Table Description	Page Number
3.4.1	Education	21
3.4.2	Food	21
3.4.3	Places	22
3.4.4	Product	22
3.4.5	Shopping	23
3.4.6	Stay	24
3.4.7	Travel	25
3.4.8	Users	26
3.4.9	Payment	26

## IV. TABLE OF CONTENT

Serial No.	Topics	Page Number
1	<b>Introduction</b> 1.1 Overview 1.2 Existing System and Disadvantages 1.3 Proposed System and Advantages	1
2	<b>Requirement Specifications</b> 2.1. Requirement Description 2.2. Analysis Modelling	4
3	<b>Detailed Design</b> 3.1. Architectural Diagram 3.2. Entity Relationship Diagram 3.3. Relationship Schema 3.4. Table Description 3.5. Interface Design	17
4	<b>Construction</b> 4.1. Software and Hardware Environments	29
5	<b>Testing</b> 5.1. Test Cases 5.2. Module Integration Plan 5.3. Test Results 5.4 Defects	34
6	<b>Screenshots</b> 6.1. Home page 6.2. Products	53
7	<b>Conclusion</b>	59
8	<b>References</b>	60

## **V. ABSTRACT**

When a new student arrives in the new city to study for his dream degree, faces many issues. The new environment is not easy, mainly for students who come from different places and cultures, to adapt. The main issues students face when they arrive in a new city are finding the best educational shop, restaurant, PG, shopping and places near to visit. Our aim is to make student life easier and more memorable. Our website is mainly targeted to students in particular locations (near college environment), where they can find the best and nearest educational shop, restaurant, PG, shopping and places near to visit. The most unique objective of our website is where one can sell or buy products. In a city the cost of living is usually high mainly students who are dependent on their parents most of the time are may not able to buy required things like furniture, electronic gadgets, etc. When a student completes his education and wants to move to another city may not want the furniture or gadgets, he has which may be of good quality, our website connects these two users where they can exchange the products in a nominal rate. We wish to help the College students, so furthermore we plan to expand our target from a particular area to a particular city and then the other states. We also plan on including some more features which would help the students. As different states have different environments and different ways of living, hence we will be focusing on Customizing the website for each state and city. Modules that are still in progress are User profiles, User reviews, and sending notifications to users

# **1.Introduction**

## **1.1 Overview:-**

Focuses on helping students who are new in the city with essential needs. As we know many students leave their homes and go to different cities to complete their education, and are new to the city, area, etc. And don't know many people or places around them.

The aim of our website is to help these students and help them get familiar with the new city. The website provides them with information such as description, photos, location, best-selling items, avg prices, timings, etc. and user reviews about that store/location, etc. Based on the user reviews of the place that store or location will be trending on our website. In our website the user can list his/her own place. For example if a user owns a café and wants to list it on the website for the students to visit he/she can do that too. Our website would help students get reviews about good places around them and help them make better decisions. This would help the students have a stress-free and much more enjoyable college life. In future we hope that most institutions suggest our system for the betterment of the students.



---

## 1.2 Existing System and Disadvantages:

The existing system which has similar features like our system is the studentrate.com and imstudent.com is a website where users can find great discounts and steal deals that range from textbooks to technology. The difference between our website and studentrate.com and imstudent.com is our website is not specific for selling products for students, only the shop owners in the city can sell the products. Students mostly don't earn during their college period, for that our website also allows second-hand products selling. The second-hand product is sold mostly by students, so it is still more beneficial.

The meetup.com provides people to identify other people who have similar interests like them, like in education, food, sports, politics, etc. This website is not student-specific, whereas ours is completely student specific.

bookmypg.co.in provides details of the pg's in most of the metropolitan cities, they also gives the user the permission to choose the type of the pg but the only issue in this website is that they don't consider the real time genuine experiences of the user and they don't provide the user to choose the pg they require based on the users budget and the requirements.

**Disadvantages:**

Does not sell second-hand products, delivery time will be more,  
It is not student specific(meetup.com).

**1.3 Proposed system and their Advantages:-**

Newbie.com is a student support system which would help those students who leave home and go to other cities or states to study. Our system would help them find places of various categories such as education,travel,etc. In and around their college location. It will also help students sell and buy things they own such as tables,chairs,books,etc.

This would help students save money. Using our system the students would get knowledge about the various hotels, restaurants, places to visit,etc near their area along with user reviews, photos ,location,rating,etc. So they can visit the place if they like. Students can also meet-up with people of common interests and do some activities like photographers would go for a group photo walk.

Students can also look for roommates or flatmates by checking the profiles of other students.

Newbie.com is also useful to people who want to run a business as they can market their shop,place,etc on the website and thereby attract more crowds to their stores and increase their sales or business.

This way our aim with newbie.com is to help all students who leave their homes and come to different cities to study have a

smooth transition into the new place,by helping them find whatever they need by just clicking a few buttons.

### **Advantages:-**

- Helping students have a smooth transition into the new city/place.
- Find essential stores.
- Getting them familiar with common hangout places.
- Helping businesses market to their target audiences.
- Helps students save time.
- Easy to use and understand
- Help students buy or sell items easily.
- Meeting people of the same interests and doing activities together thereby increasing contacts.
- Helps in finding roommates or flatmates.

## **2.1 Requirements Specification:-**

### **2.1 Requirement Description:-**

#### **2.1.1 Users(student):-**

##### **Registration:-**

The user fills all the required information and then submits the information and registers successfully.

##### **Login:-**

The user enters his/her credentials and logs into the system.

##### **Reset Password:-**

User can reset his/her password.

**View profile:-**

Users can view profiles of other users also.

**2.1.2 Product:-(user)**

**View Product: -**

The users can view products listed by other users.

**Buy/Rent Product:-**

The users can buy/rent the products listed by other users if they like it.

**Add Product:-**

The user can sell/rent their own products if any such as furniture, tv, etc.

**Update Product:-**

The user can update the product he/she has added to the website.

**2.1.3 Product:-(admin)**

**Remove Product:-**

The admin can remove a product from the website, if the product is inappropriate.

### **2.1.4 Stay:-(user)**

#### **View Stay:-**

The user can view the pg or the hostels.

#### **Add Stay:-**

The user can add a pg if he/she owns one.

#### **Remove Stay:-**

The user can remove the pg from the website if he doesn't want to list it.

#### **Edit Stay:-**

The user can edit the pg listed on the website.

### **2.1.5 Pg/Hostel:-(admin)**

#### **Remove Pg/Hostel:-**

The admin can remove the pg from the website.

### **2.1.6 Education:-(user)**

#### **View Stores:-**

The user can view the educational stores such as xerox, stationary, etc.

#### **Add store:-**

The user can list the stores to the website.

**Edit stores:-**

The user can edit the shops or the stores in the website.

**2.1.7 Education:-(admin)****Remove the store:-**

The admin can remove the store from the website.

**2.1.8 Travel(user):-****View The travel:-**

The user can view all the routes and the best travel medium to get there.

**Add the travel:-**

The user can add the travel route.

**Delete the travel:-**

The user can delete the travel route.

**2.1.9 Travel(admin):-****Remove the travel:-**

The admin can remove the travel route.

---

### **2.1.10 Shopping(user):-**

#### **View The shopping stores:-**

The user can view all the stores.

#### **Add the shopping stores:-**

The user can add shopping stores.

#### **Delete the the stores:-**

The user can delete the shopping stores.

### **2.1.11 Shopping(admin):-**

#### **Remove the store:-**

The admin can remove the store.

### **2.1.12 Places to visit(user):-**

#### **Add places:-**

The user can add the places to visit to the visit.

#### **Remove Places:-**

The user can remove the places entered by him.

#### **Edit Place:-**

The user can edit the places entered by him

---

### **2.1.13 Places to visit(admin):-**

#### **Remove places:-**

The admin can remove the places added to the website.

### **2.1.14 Food (user):-**

#### **Add place:-**

The user can add restaurants or cafes if he owns any. Which will be visible to others.

#### **Edit place:-**

The user can edit places added by him in the website.

#### **Delete the place: -**

The user can delete the places added by him in the website.

#### **View and review the place:-**

The user can view and review cafes and restaurants which are already there on the website.

### **2.1.15 Food(Admin):-**

#### **Remove the place:-**

The admin can remove a place from the website if any issues occur.



## 2.2 Analysis Modelling:-

### 2.2.1 Use case Diagram:-

Travel

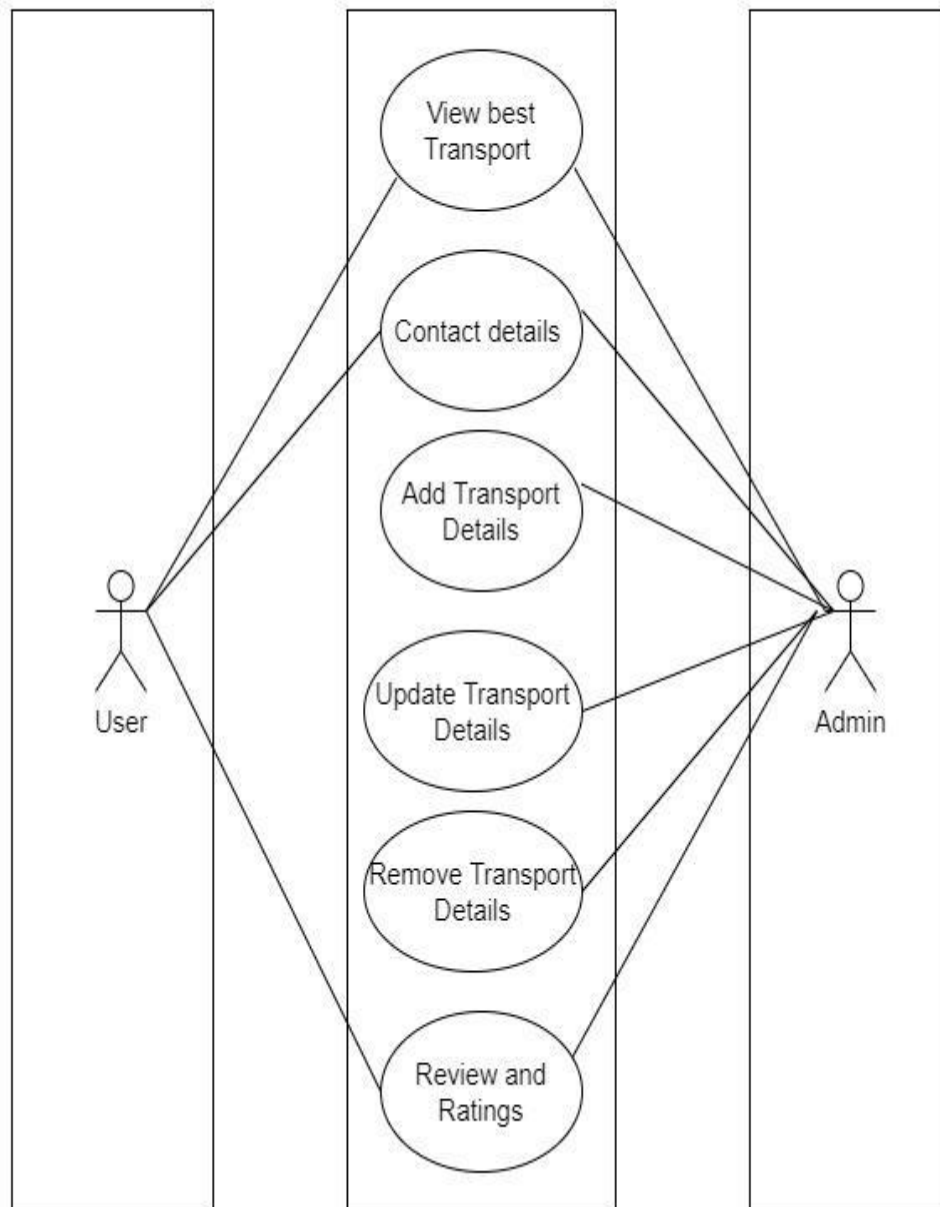


Fig 2.2.1 Travel

## Places to Visit

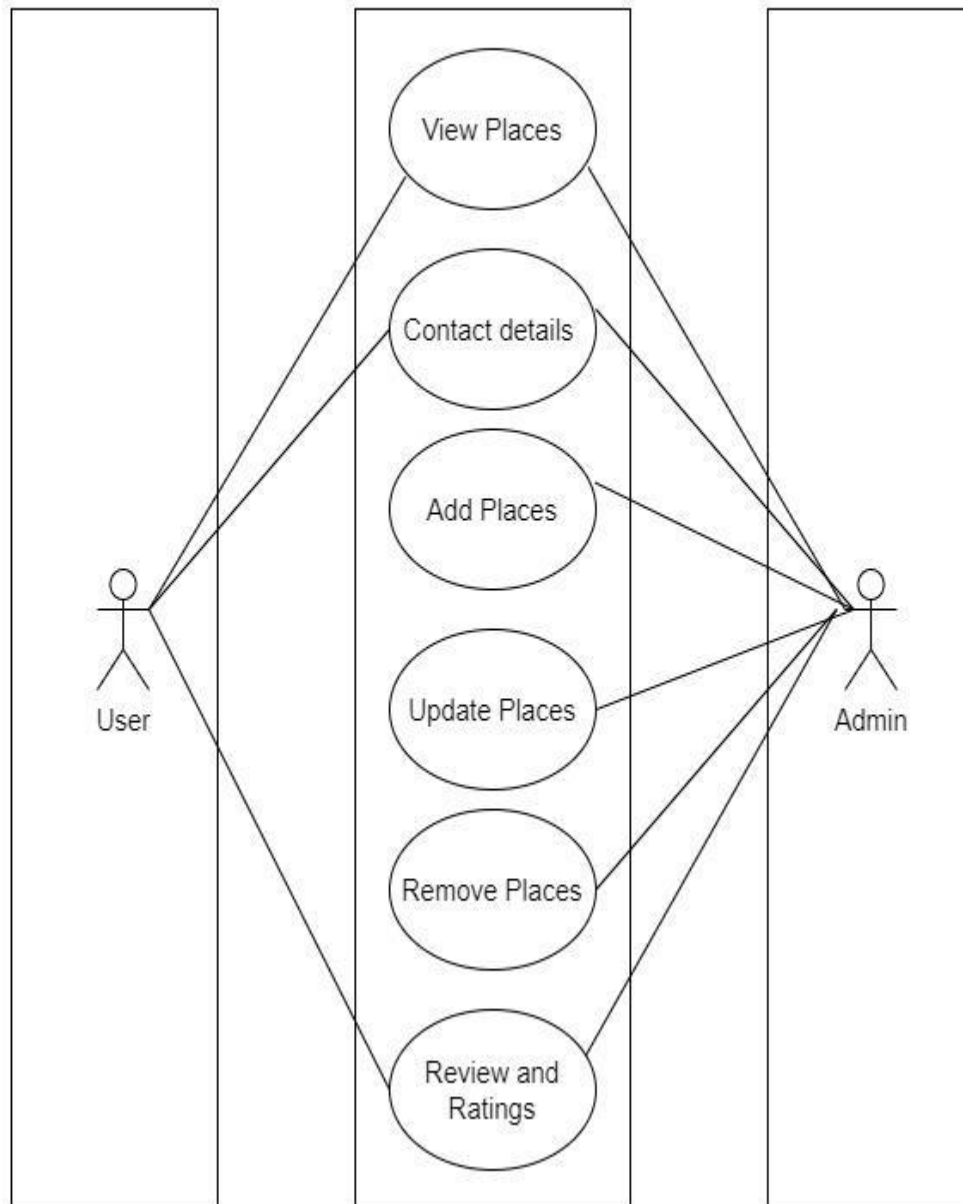


Fig 2.2.2 Places to visit

## Paying Guest

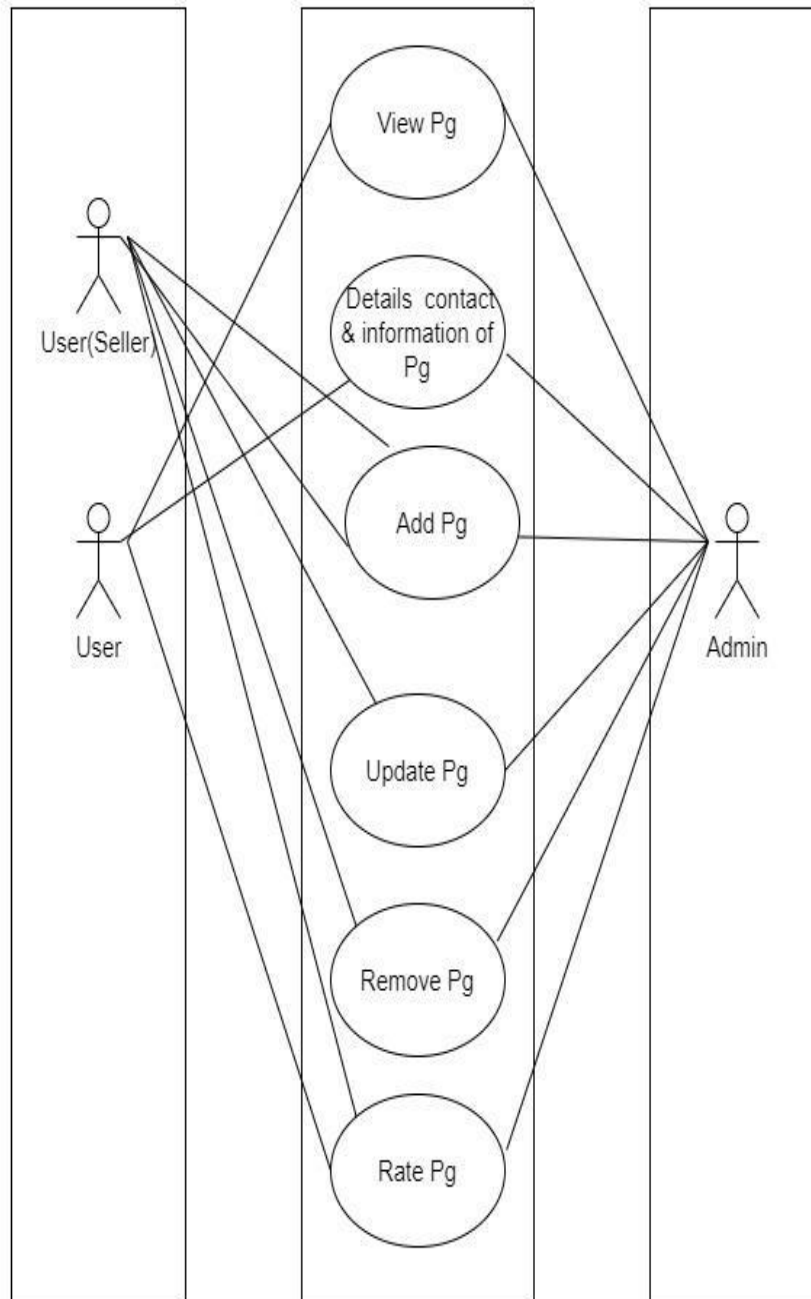


Fig 2.2.3 Paying Guest

## Rent

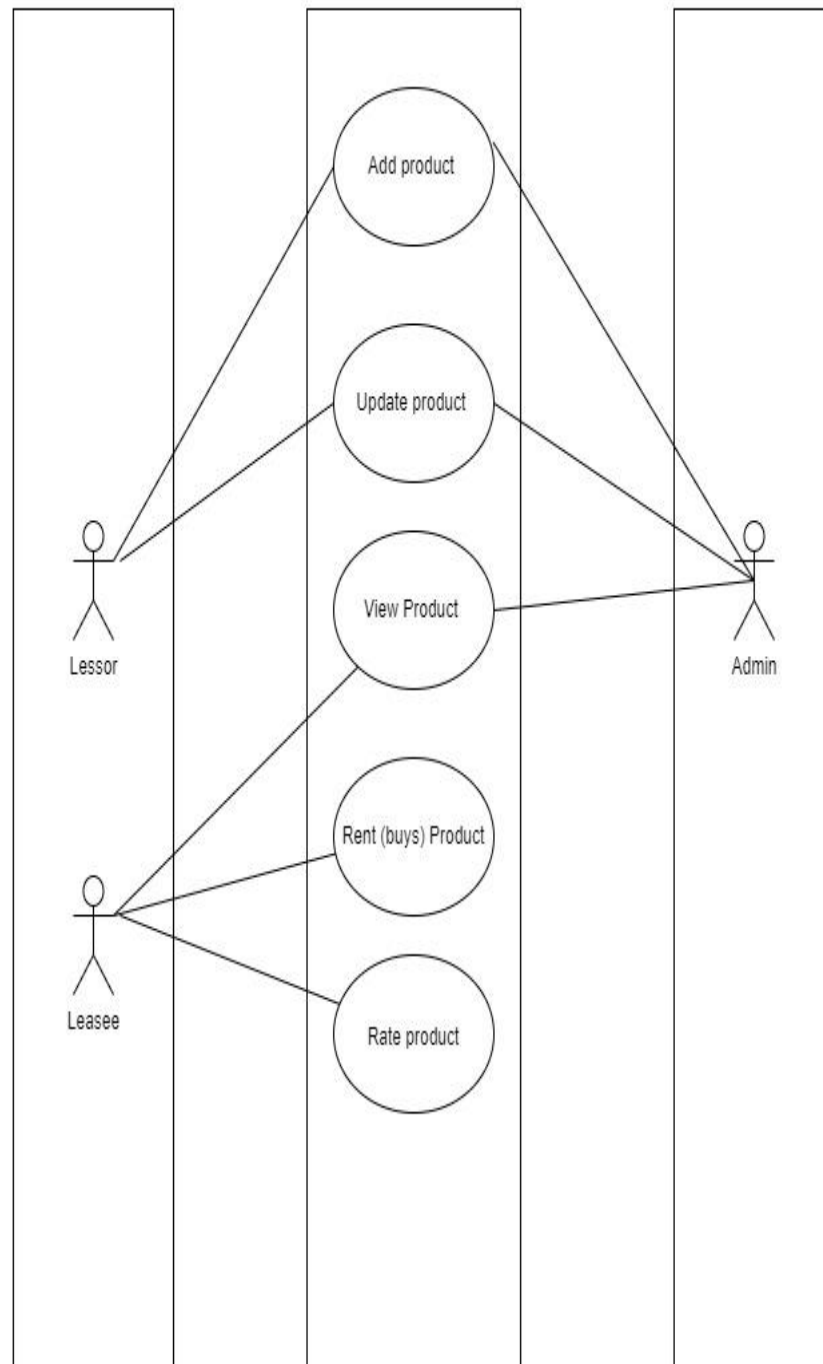


Fig 2.2.4 Rent

## Shopping

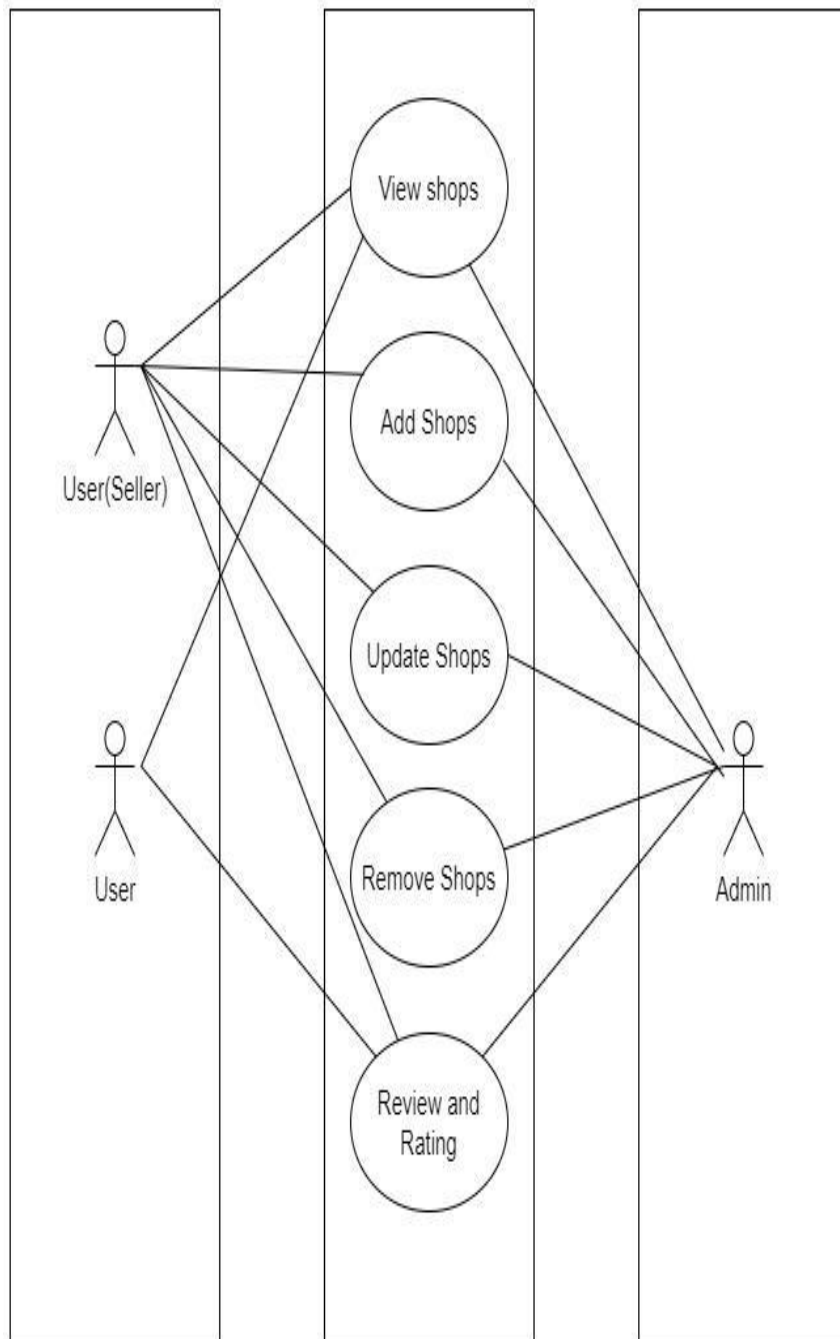


Fig 2.2.5 Shopping

## Product

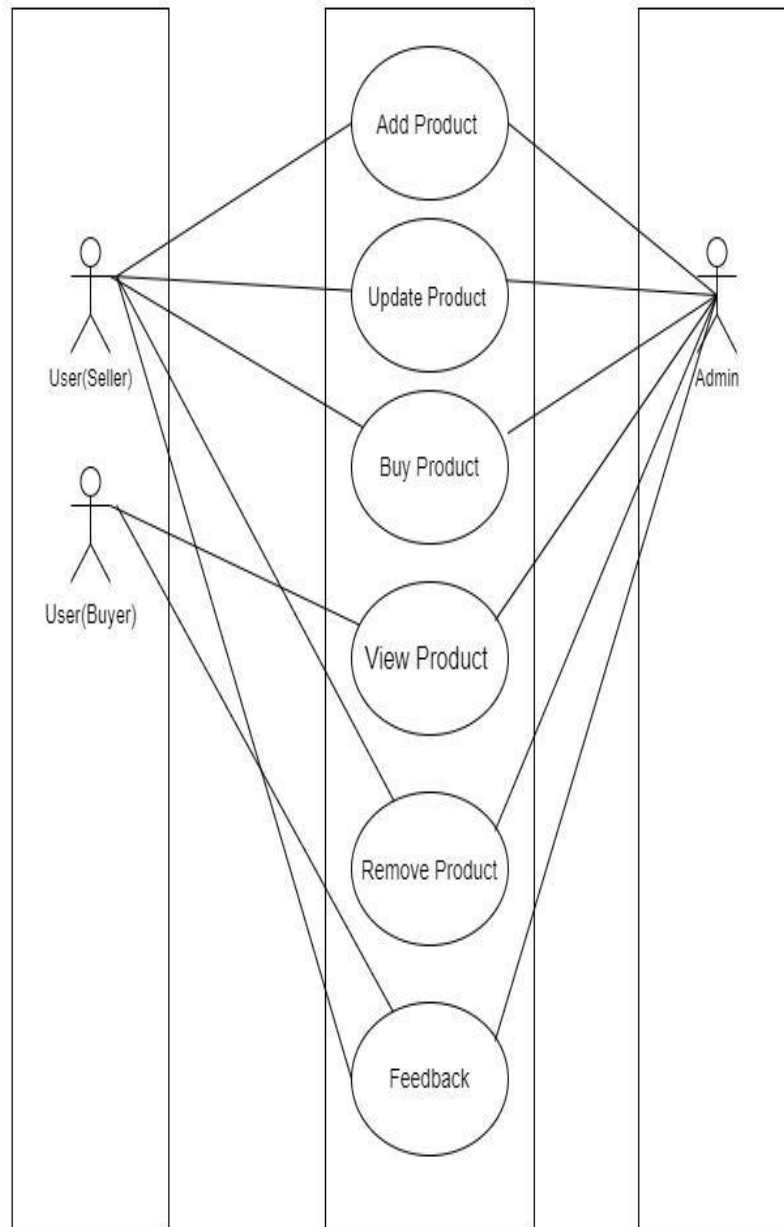


Fig 2.2.6 Product

## Food

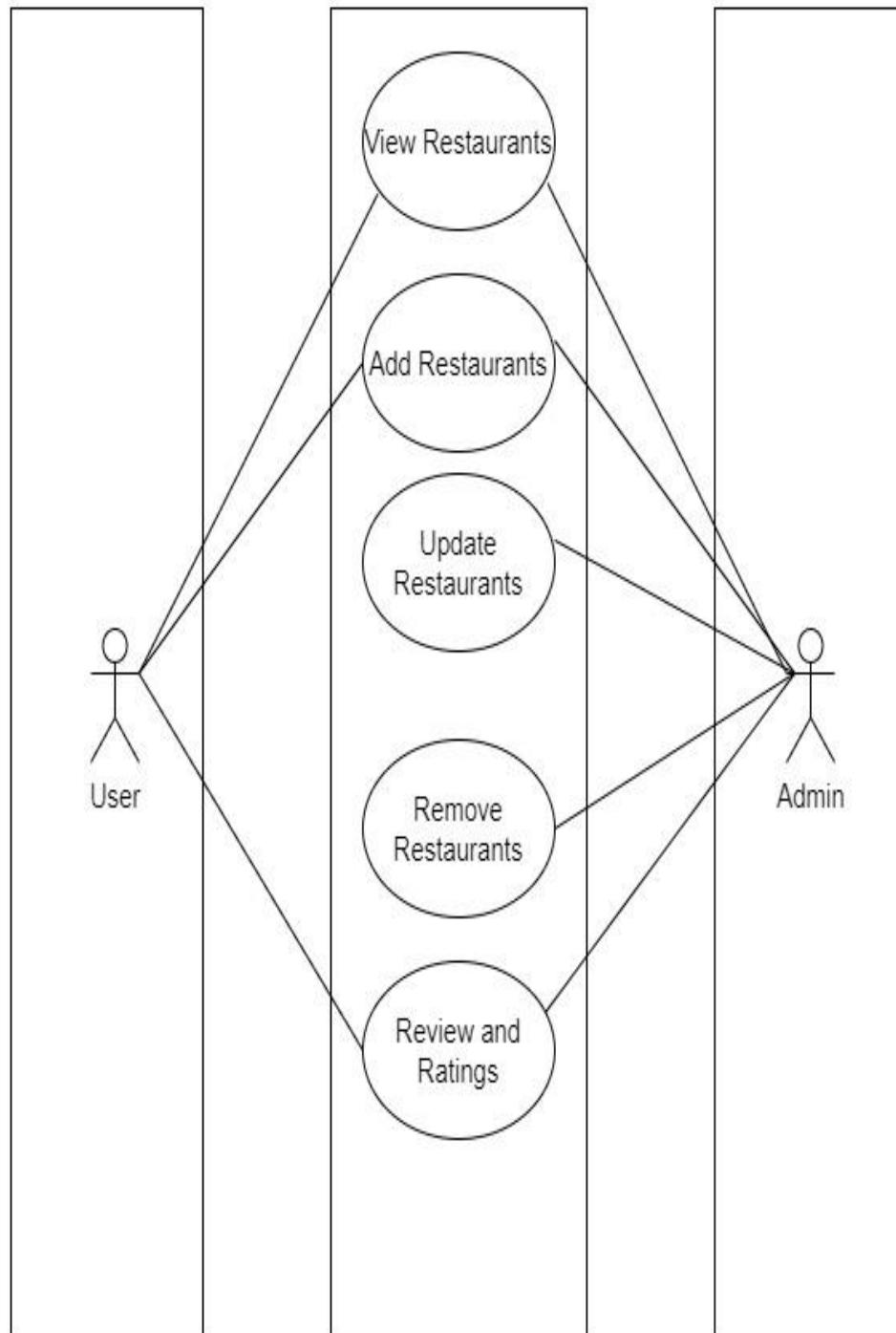


Fig 2.2.7 Food

### 3. Detailed Design:-

#### 3.1 Architecture Diagram:-

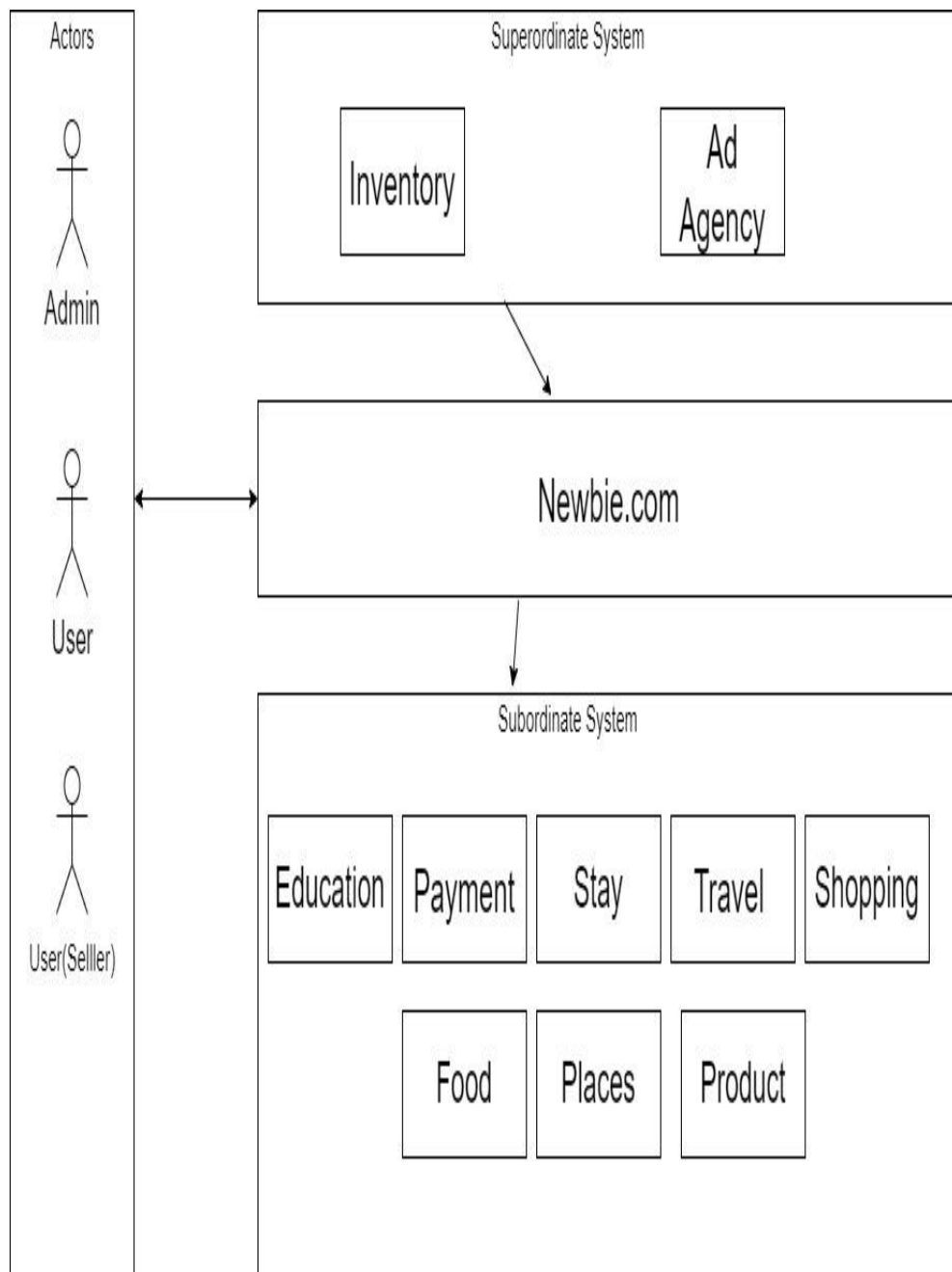


Fig 3.1.1 Architectural Diagram



### 3.2 ER Diagram:-

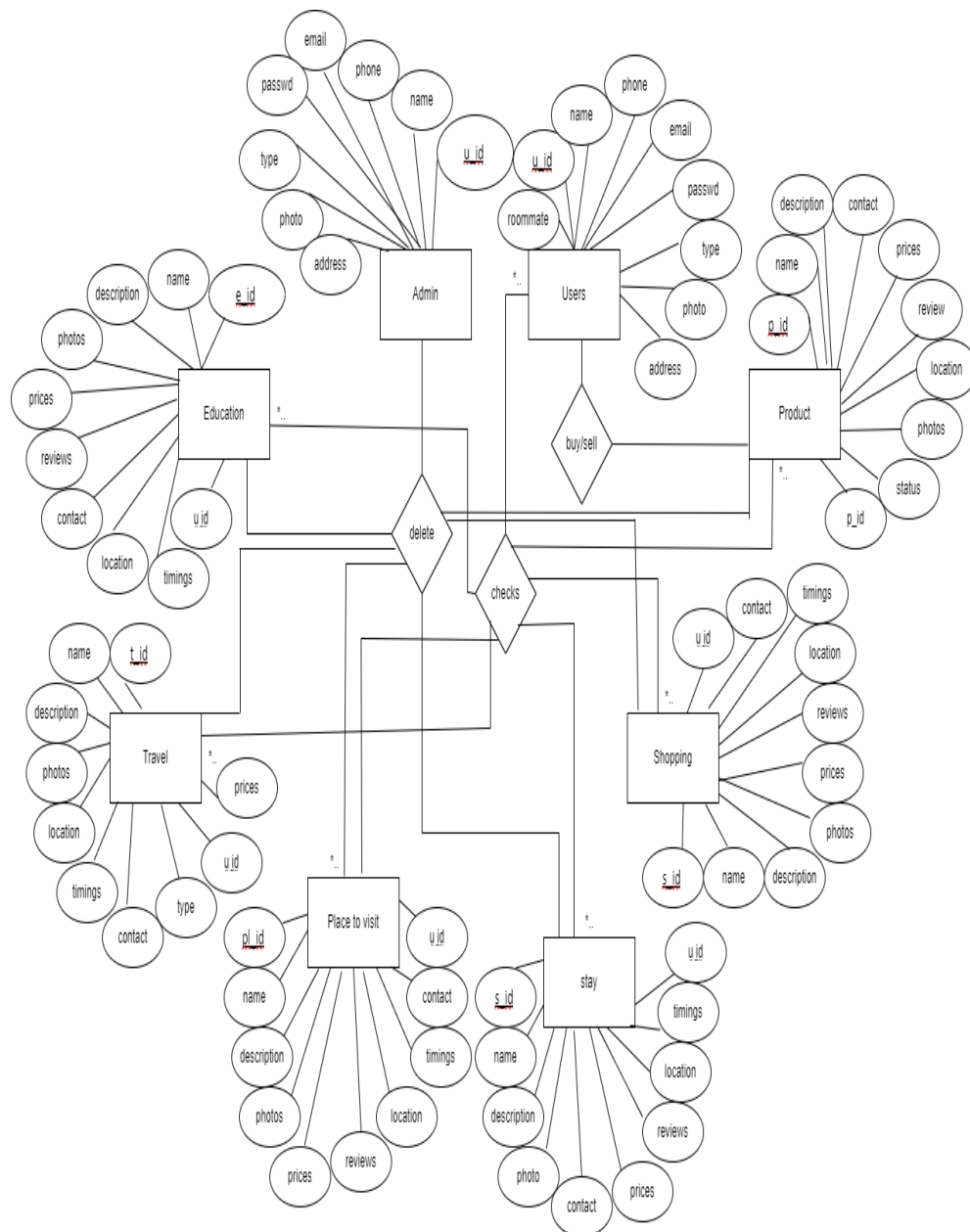


Fig 3.2.1 ER Diagram

### 3.3 Relationship Schema:-

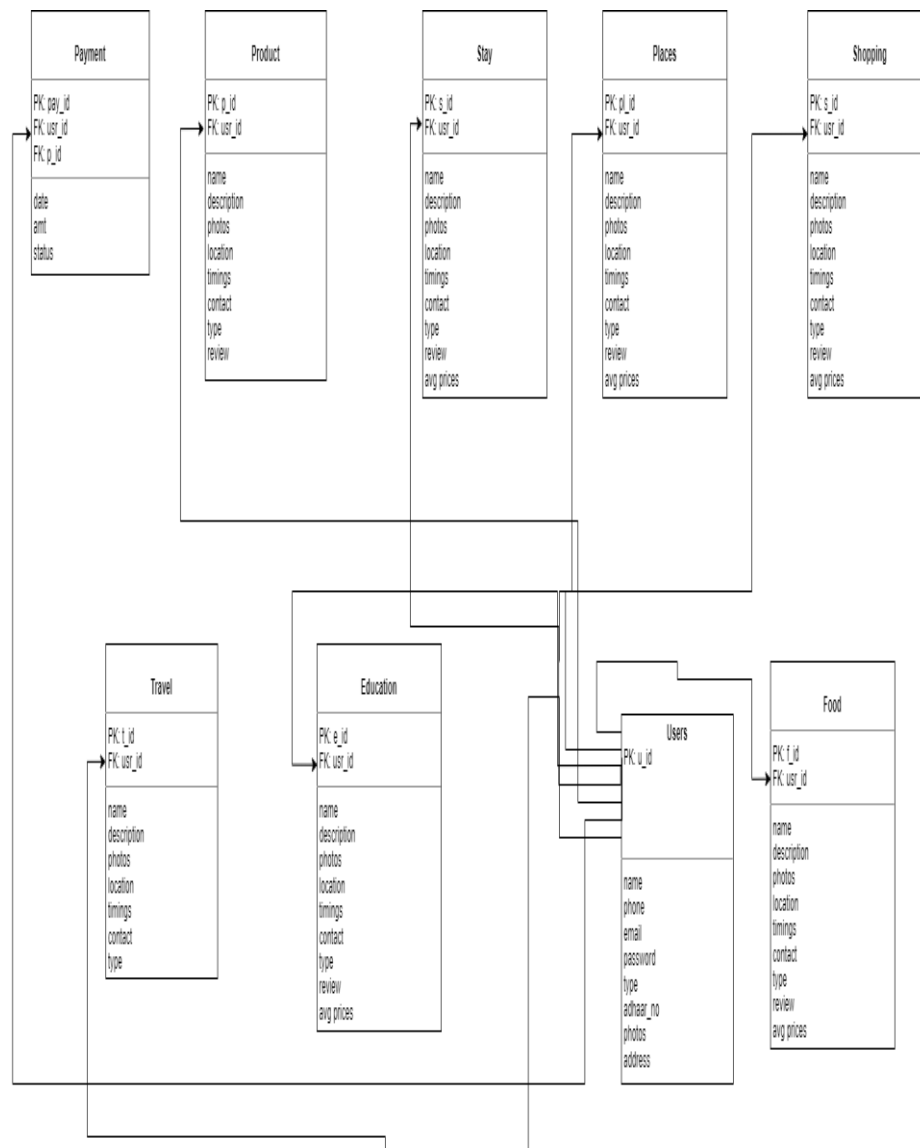


Figure 3.3.1

### 3.4 Table Description:-

#### 3.4.1 Education

Table 3.4.1

Column	Type	Null	Default	Comments
e_id (Primary)	int(11)	No		The id given to each of the education store entered in the table.
name	varchar(100)	No		The name of the store.
description	varchar(500)	No		A short description about the store and what all you get there.
photos	varchar(500)	No		A couple of photos of the store.
avg prices	int(11)	No		The average prices at the store.
reviews	varchar(500)	No		The reviews given by the customer.
contact	int(20)	No		The contact details of the shop.
location	varchar(500)	No		The exact location of the shop.
timings	date	No		The opening and closing timing of the shop
usr_id	int(11)	No		Foreign key of the user

#### 3.4.1.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	e_id	0	A	No	

### 3.4.2 Food

Table 3.4.2

Column	Type	Null	Default	Comments
f_id ( <i>Primary</i> )	int(11)	No		The id given to each of the café and restaurant entered in the table.
name	varchar(20)	No		The name of the café or restaurant.
description	varchar(500)	No		A small description about the café or restaurant.
photos	varchar(500)	No		Photos of the café or the restaurant.
location	varchar(500)	No		The exact location of the café or restaurant.
timings	date	No		The opening and the closing timings of the restaurant.
contact	int(20)	No		The contact details of the cafes and the restaurant.
avg prices	int(100)	No		The average prices of the restaurants.
usr_id	int(11)	No		Foreign key of the user

#### 3.4.2.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	f_id	0	A	No	

### 3.4.3 Places

Table 3.4.3

Column	Type	Null	Default	Comments
pl_id ( <i>Primary</i> )	int(11)	No		The id given to each of the place entered in the table.
name	varchar(20)	No		The name of the place.
description	varchar(500)	No		A short description about the place entered.
photos	varchar(500)	No		Photos of the place entered.
avg prices	int(100)	No		Average prices of the place entered.
reviews	varchar(500)	No		The reviews entered by the user.
location	varchar(500)	No		The location of the place.
timings	date	No		The opening and closing times of the place.
contact	int(20)	No		The contact details of the place.
usr_id	int(11)	No		Foreign key of the user

#### 3.4.3.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	pl_id	0	A	No	

### 3.4.4 Product

Table 3.4.4

Column	Type	Null	Default	Comments
p_id (Primary)	int(11)	No		The id given to each of the product entered in the table.
name	varchar(100)	No		The name of the product entered by the user.
description	varchar(500)	No		The description of the product entered by the user.
photos	varchar(500)	No		The photos of the product entered by the user.
contact	int(20)	No		The contact of the user.
prices	int(20)	No		The price of the item entered.
reviews	varchar(500)	No		The reviews of the user about the seller.
location	varchar(500)	No		The location of where the item is.
usr_id	int(11)	No		Foreign key of the user

#### 3.4.4.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	p_id	0	A	No	

### 3.4.5 Shopping

Table 3.4.5

Column	Type	Null	Default	Comments
s_id (Primary)	int(11)	No		The id given to each of the product entered in the table.
name	varchar(200)	No		The name of the shops entered in the table.
description	varchar(500)	No		A short description about the shops.
photos	varchar(500)	No		The photos of the shop.
avg prices	int(100)	No		The average prices at the shop.
reviews	varchar(500)	No		The reviews entered by the user.
location	varchar(500)	No		The location of the place.
timings	date	No		The opening and closing times of the place.
contact	int(20)	No		The contact details of the place.
usr_id	int(11)	No		Foreign key of the user

#### 3.4.5.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	s_id	0	A	No	

### 3.4.6 Stay

Table 3.4.6

Column	Type	Null	Default	Comments
s_id (Primary)	int(11)	No		The id given to each of the pg or flat entered in the table.
name	varchar(200)	No		The name of the pg or flat entered.
description	varchar(500)	No		A short description about the pg or flat.
photos	varchar(500)	No		Photos of the pg or flat.
contact	int(20)	No		The contact details of the place.
avg prices	int(100)	No		The average prices at the pg or flat.
reviews	varchar(500)	No		The reviews entered by the user.
location	varchar(500)	No		The location of the place.
timings	date	No		The opening and closing times of the place.
usr_id	int(11)	No		Foreign key of the user

#### 3.4.6.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	s_id	0	A	No	



### 3.4.7 Travel

Table 3.4.7

Column	Type	Null	Default	Comments
t_id (Primary)	int(11)	No		The id given to each of the mode of transport entered in the table.
name	varchar(20)	No		The name of the travel mode.
description	varchar(500)	No		A short description about the mode of transport.
photos	varchar(500)	No		Photos of the mode of transport.
location	varchar(500)	No		The location of the transport area.
timings	date	No		The timings of the mode of transport
contact	int(20)	No		Contact details of the mode of transport.
type	varchar(500)	No		The type of transport eg:- Bus, Flight,etc.
usr_id	int(11)	No		Foreign key of the user

#### 3.4.7.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	t_id	0	A	No	

### 3.4.8 User

Table 3.4.8

Column	Type	Null	Default	Comments
u_id (Primary)	int(11)	No		The user id given to each user.
name	varchar(20)	No		The name of the user.
phone	int(20)	No		The phone number of the user.
email	varchar(100)	No		The email id of the user.
password	varchar(100)	No		The password of the user.
type	varchar(100)	No		The type of the user i.e. admin or student.
adhaar_no	int(100)	No		The aadhar number of the user.
photos	varchar(500)	No		The photo of the user.
address	varchar(500)	No		The address of the user.

#### 3.4.8.1 Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	u_id	0	A	No	

### 3.5 Interface Design:-

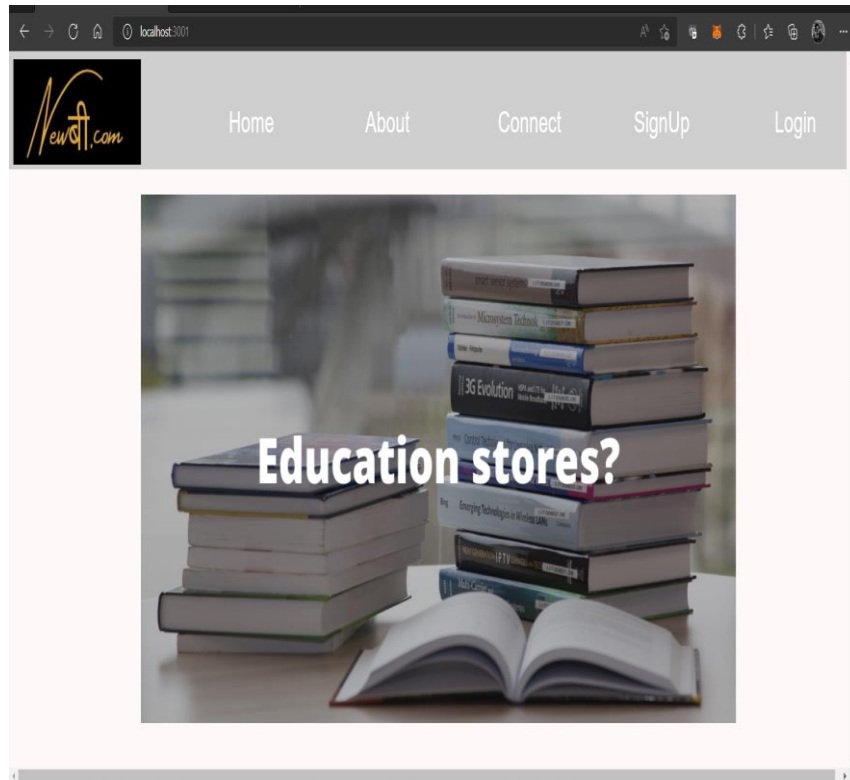


Fig 3.5.1 Home page

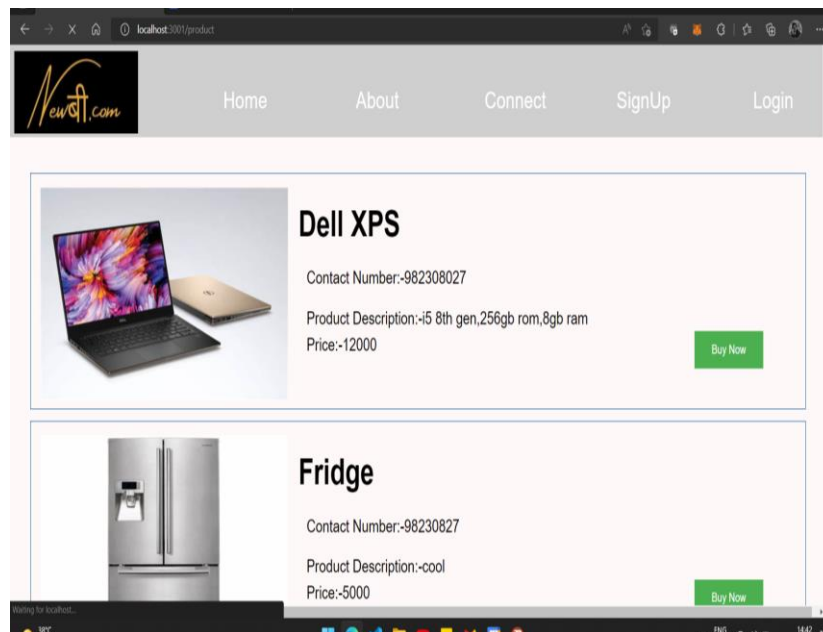


Fig 3.5.2 Products

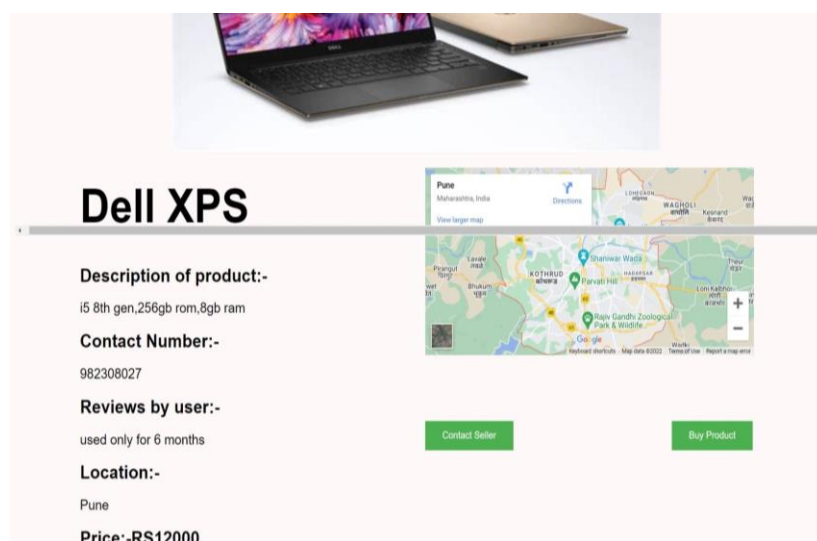
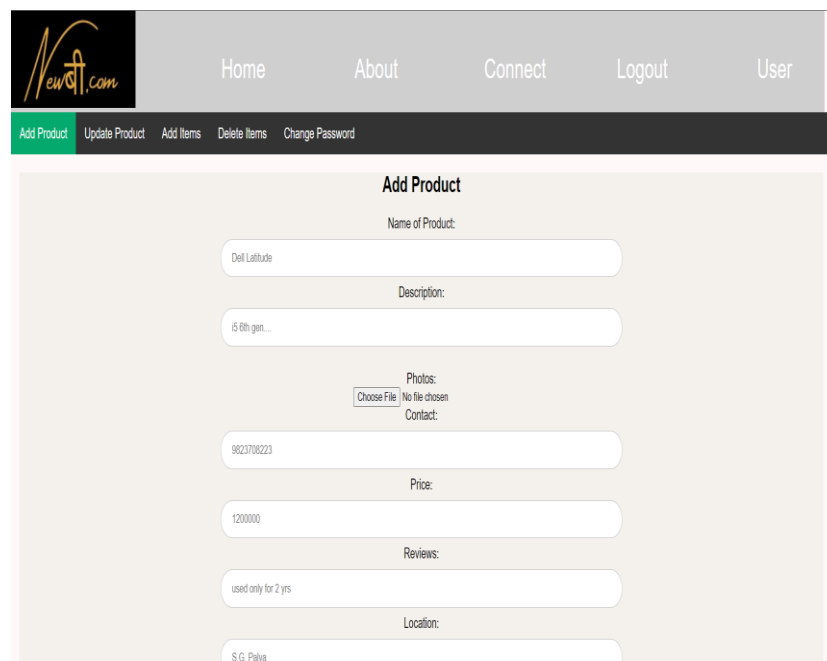


Fig 3.5.3 Product Details



The screenshot shows the 'Add Product' form on the Newbie.com website. The form is titled 'Add Product' and contains several input fields with sample data:

- Name of Product:** Dell Latitude
- Description:** i5 6th gen...
- Photos:** Choose File | No file chosen
- Contact:** 9823708223
- Price:** 1200000
- Reviews:** used only for 2 yrs
- Location:** S.G. Palaya

Fig 3.5.4 Add Product

## 4. Construction

### 4.1 Software and Hardware Environments

#### 4.1.1 Node.js

Node.js is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine). Node.js was developed by Ryan Dahl in 2009 and its latest version is v0.10.36.

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js

applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

We have selected Node.js as the backend because Node.js works well with real-time handling of large amounts of information. Node.js enables the collection and visualization of data to be presented as dashboards. Node.js backend development is concise, as it provides many features that can be impossible to have with any event-based environments. Node.js is fast as it stores its data in native JSON. It is supported by a large community of developers that are constantly working on improving it. It performs high-level data streaming. It is also suitable for developing software for different platforms regardless of the underlying operating system. Applications developed in Node.js can be scaled and offer relatively higher performance.

#### **4.1.2 EJS with HTML and JAVA**

EJS or Embedded Javascript Templating is a templating engine used by Node.js. Template engine helps to create an HTML template with minimal code. Also, it can inject data into HTML template at the client side and produce the final HTML. EJS is a simple templating language which is used to generate HTML markup with plain JavaScript. It also helps to embed JavaScript to HTML pages

### **4.1.3MySQL**

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and an easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications. MySQL is a Relational Database Management System (RDBMS) software that provides many things, which are as follows: It allows us to implement database operations on tables, rows, columns, and indexes. It defines the database relationship in the form of tables (collection of rows and columns), also known as relations. It provides the Referential Integrity between rows or columns of various tables. It allows us to updates the table indexes automatically. It uses many SQL queries and combines useful information from multiple tables for the end-users.

### **4.1.4 Php myAdmin**

phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement. It has Intuitive web interface Support for most MySQL features:

- Browse and drop databases, tables, views, fields and indexes
- Create, copy, drop, rename and alter databases, tables, fields and indexes

- Maintenance server, databases and tables, with proposals on server configuration
- Execute, edit and bookmark any SQL-statement, even batch-queries
- Manage MySQL user accounts and privileges
- Manage stored procedures and triggers
- Import data from CSV and SQL
- Export data to various formats: CSV, SQL, XML, PDF, ISO/IEC 26300 - OpenDocument Text and Spreadsheet, Word, LATEX and others
- Administering multiple servers
- Creating graphics of your database layout in various formats
- Creating complex queries using Query-by-example (QBE)
- Searching globally in a database or a subset of it Transforming stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link
- Intuitive web interface

#### **4.1.5 VScode**

Visual Studio Code (famously known as VS Code) is a free open source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times. VS Code supports a wide array of programming languages from Java, C++, and Python to CSS, Go, and Dockerfile. Moreover, VS Code allows you to add on and even creating new extensions including code linters, debuggers, and cloud and web development support.



We have selected VS Code as our platform as it Supports multiple programming languages, It can detect if any snippet of code is left incomplete. Also, common variable syntaxes and variable declarations are made automatically. It also has Cross-Platform Support. Usually supports all the programming languages but, if the user/programmer wants to use the programming language which is not supported then, he can download the extension and use it. And performance-wise, the extension doesn't slow down the editor as it runs as a different process. With the ever-increasing demand for the code, secure and timely storage is equally important. It is connected with Git or can be connected with any other repository for pulling or saving the instances.

## **5. Testing:-**

### **5.1 Test cases:-**

Requirements:

1. The purpose is to login successfully into Newbie.com.
2. Select a product from products.
3. Contact the Seller and see his location.
4. After knowing the Seller's location Proceed to the payment page.
5. Adding an Item to the Product list.
6. Updating an Item in the Product list.
7. Deleting an Item in the Product list.
8. Logout successful.

Test ID	Test Name	Description	Req ID	Procedure	Expected O/P	Actual Output
1.1	whether the site is loading or not	Check if the website Script is for localhost:3001(newbie.com)	1	1. Login to the machine  2. Login to the website	1. Site should be loaded successfully.  2. Home page of Newbie.com should be loaded.	1. Website was loaded successfully.  2. Homepage of Newbie.com was available.

1.2	Whether username and password is specific or any user name is accepted.	Check if the login page accepts any random username for the login or it only accepts the specified username and password	1	<p>1. Login to the machine</p> <p>2. Login with a random username and password and hit enter.</p>	<p>1. Username and password column should be empty with a message displaying the wrong username or password.</p>	<p>1 Login was not successful and a message was displayed to login again.</p>
1.3	Whether the user can login.	Check if the user can login to the website or not	1	<p>1. Load the website</p> <p>2. Login with the correct credentials and hit enter</p>	<p>1. Website should be changed from a generic one to the User's personal Website.</p>	<p>1. After the Successful login user could see on the top right corner with the login as User.</p>

1.4	Whether the user can change the password	Check if the user can change the password or not.	1	<ol style="list-style-type: none"><li>1. Navigate to the login page.</li><li>2. Select the forgot password.</li><li>3. add the current password and confirm it and hit enter.</li></ol>	<ol style="list-style-type: none"><li>1. User should be able to add the new password.</li></ol>	<ol style="list-style-type: none"><li>1. User was able to add the new password .</li></ol>
-----	--	---	---	---	---	--

2.1	whether the user can select the product he/she needs to buy.	Check if the user has permission to select the product he/she wishes to buy.	2	<p>1. Login to the website.</p> <p>2. Navigate to the products section.</p> <p>3. Select the product.</p>	1. User should successfully select the product.	1. User could successfully select the product.
-----	--	--	---	---	---	--

2.2	Whether the user can buy the see the seller's location and details after selecting the product.	Check if the user can see the location of the seller after selecting the product and also the other details about the seller.	2 .	<p>1. Select the product.</p> <p>2. Click on the map location on the right hand side.</p> <p>3. check if the other details of the seller are mentioned or not.</p>	<p>1. User should be able to see the location of the seller in the maps and also the contact details.</p>	<p>1. User should be able to see the location of the seller in the maps and also the contact details.</p>
-----	---	---	-----	--	---	---

2.3	Whether the user is able to pay for the selected product.	Check if the user is able to pay for the selected product.	2	<p>1. Select the product.</p> <p>2. Proceed for the payment.</p> <p>3. Enter the transaction ID.</p>	<p>1. User should be able to select the product.</p> <p>2. User should be able to proceed to the payment page and enter the Transaction ID.</p>	<p>1. User could select the product.</p> <p>2. User could enter the Transaction ID but it is not reflecting in table.</p>
-----	---	--	---	--	---	---

3.	Whether the User could add products.	Check if the User is able to add the products in the website	3	<p>1. Login to the website.</p> <p>2. Navigate to products.</p> <p>3. Click on add products.</p>	1. User should be able to navigate and proceed to the option named add product.	1. User was able to add the product.
3.1	Whether the User could add the image of the product.	Check if the user can add the image of the product in the add product list.	3	<p>1. Navigate to the add products window.</p> <p>2. add the image of the product.</p>	1. User should be able to add the image of the product.	1. User was able to add the image successfully but after adding the product the image was not displayed correctly.



3.2	Whether the Product can be deleted.	Check if the user can delete the product.	3	<ol style="list-style-type: none"><li>1. Navigate to products.</li><li>2. select the product to be deleted.</li><li>3. confirm the Product to be deleted.</li></ol>	<ol style="list-style-type: none"><li>1. user should be able to delete the product successfully, and it should be reflected in the products window.</li></ol>	<ol style="list-style-type: none"><li>1. user was able to delete the product successfully and it was reflected in the products window.</li></ol>
-----	-------------------------------------	---	---	---	---	--

3.3	Whether the Product can be updated.	Check if the user can update the product.	3	<ol style="list-style-type: none"><li>1. Navigate to products.</li><li>2. select the product to be updated.</li><li>3. confirm the Product to be updated.</li></ol>	<ol style="list-style-type: none"><li>1. user should be able to update the product successfully, and it should be reflected in the products window.</li></ol>	<ol style="list-style-type: none"><li>1. user was not able to update the product.</li></ol>
-----	-------------------------------------	---	---	---	---	---

4.	Whether the main logo helps in navigating back to the home page.	Check if the main logo on the left hand side of the website is able to navigate back to the homepage of the website.	4	<p>1. Load the website.</p> <p>2. Click on the logo.</p>	1. user should be able to navigate back to the home page of the website.	1. user was able to navigate back to the home page of the website.
5.	Whether the user is able to logout successfully	Check whether the user is able to logout successfully.	5	<p>1. Click on the top right part of the website i.e user.</p> <p>2. click on logout.</p>	1. user should be able to logout successfully.	1. user was able to logout successfully.

---

## 5.2 Module Integration Plan

Newbie.com has a total of 9 modules in it. Where each module has its own purpose with different types of data. Other than that there are two types of users whose functionalities differ from one another name, namely the two users are Admin and Buyer/Seller, Admin has the most of the functionalities which buyer or seller doesn't have with them. Here for the Integration of all the modules we have used the approach of Incremental Testing.

Incremental testing is one of the approaches used during the testing phase of integration testing. Incremental testing uses several stubs and drivers to test the modules one by one. This helps us discover any errors and defects in our modules.

After the Integration of Buyer/Seller user and Admin user we could analyze that after login and logout from Seller/Buyer user we could switch to Admin user by logging in with the correct login details and after switching we could find that the functionalities for Admin are different from Seller/Buyer user. This proves that this part of Integration was successful. Similarly with the other modules we could analyze that each module is connected with most of the module and the sharing of data is reflected back in the other modules too.

### 5.3 Test Results

#### 5.3.1 Test Cycle 1

Test ID	Test Name	Description	Req ID	Procedure	Expected O/P	Actual Output	Result
1.1	whether the site is loading or not	Check if the website Script is for localhost:3001(newbie.com)	1	1. Login to the machine  2. Login to the website	1. Site should be loaded successfully.  2.Home page of Newbie.com should be loaded.	1. Website was loaded successfully.  2. Homepage of Newbie.com was available.	Pass
1.2	Whether username and password is specific or any user name is accepted.	Check if the login page accepts any random username for the login or it only accepts the specified username and password	1	1. Login to the machine  2. Login with a random username and password and hit enter.	1. Username and password column should be empty with a message displaying the wrong username or password.	1 Login was not successful and a message was displayed to login again.	Fail

<b>1.3</b>	Whether the user can login.	Check if the user can login to the website or not	1	1. Load the website  2. Login with the correct credentials and hit enter	1. Website should be changed from a generic one to the User's personal Website.	1. After the Successful login user could see on the top right corner with the login as User.	Pass
<b>1.4</b>	Whether the user can change the password	Check if the user can change the password or not.	1	1. Navigate to the login page.  2. Select the forgot password .  3. add the current password and confirm it and hit enter.	1. User should be able to add the new password.	1. User was able to add the new password.	Pass
<b>2.1</b>	whether the user can select the product he/she needs to buy.	Check if the user has permission to select the product he/she wishes to buy.	2	1. Login to the website.  2. Navigate to the products section.  3. Select the product.	1. User should successfully select the product.	1. User could successfully select the product.	Pass

2.2	Whether the user can buy the see the seller's location and details after selecting the product.	Check if the user can see the location of the seller after selecting the product and also the other details about the seller.	2.	1. Select the product. 2. Click on the map location on the right hand side. 3. check if the other details of the seller is mentioned or not.	1. User should be able to see the location of the seller in the maps and also the contact details.	1. User should be able to see the location of the seller in the maps and also the contact details.	Pass
2.3	Whether the user is able to pay for the selected product.	Check if the user is able to pay for the selected product.	2	1. Select the product. 2. Proceed for the payment. 3. Enter the transaction ID.	1. User should be able to select the product. 2. User should be able to proceed to the payment page and enter the Transaction ID.	1. User could select the product. 2. User could enter the Transaction ID but it is not reflecting in table.	Fail
3.	Whether the User could add products.	Check if the User is able to add the products in the website	3	1. Login to the website. 2. Navigate to products.	1. User should be able to navigate and proceed to the option named add product.	1. User was able to add the product.	Pass

				3. Click on add products.			
3.1	Whether the User could add the image of the product.	Check if the user can add the image of the product in the add product list.	3	1. Navigate to the add products window.  2. add the image of the product.	1. User should be able to add the image of the product.	1. User was able to add the image successfully but after adding the product the image was not displayed correctly.	Fail
3.2	Whether the Product can be deleted.	Check if the user can delete the product.	3	1. Navigate to products.  2. select the product to be deleted.  3. confirm the Product to be deleted.	1. user should be able to delete the product successfully, and it should be reflected in the products window.	1. user was able to delete the product successfully and it was reflected in the products window.	Pass
3.3	Whether the Product can be updated.	Check if the user can update the product.	3	1. Navigate to products.  2. select the product	1. the user should be able to update the product successfully, and it should be reflected in the	1. the user was not able to update the product.	Fail



				to be updated.  3. confirm the Product to be updated.	products window.		
4.	Whether the user is able to view the Places	Check if the user is able to see the Pg and flat on clicking the Stay option	4	1. Navigate to stay.  2. Click on stay.	1. the user should be able to view the information about the flat/PG.	1. User was able to successfully see the information about Sapthagiri PG on clicking Stay.	Pass
5.	Whether the main logo helps in navigating back to the home page.	Check if the main logo on the left hand side of the website is able to navigate back to the homepage of the website.	5	1. Load the website.  2. Click on the logo.	1. users should be able to navigate back to the homepage of the website.	1. the user was able to navigate back to the homepage of the website.	Pass
6.	Whether the password is getting encrypted	Check if the password is getting encrypted or not.	6	1. Login into the website.  2. Check the password in database if it's getting converted to hash values	1. hashed password should be visible.	1. Password was hashed and it was converted to the hash values.	Pass

7.	Whether the user is able to logout successfully	Check whether the user is able to logout successfully.	5	1. Click on the top right part of the website i.e. user.  2. click on logout.	1. the user should be able to logout successfully.	1. the user was able to logout successfully.	Pass
----	---	--	---	---	--	--	------

### 5.3.2 Test Cycle 2

Test ID	Test Name	Description	Req ID	Procedure	Expected O/P	Actual Output	Result
1.1	Whether username and password is specific or any user name is accepted.	Check if the login page accepts any random username for the login or it only accepts the specified username and password	1	1. Login to the machine  2. Login with a random username and password and hit enter.	1. Username and password column should be empty with a message displaying the wrong username or password.	1 Login was successful and a message was displayed to login again.	Pass
2.1	Whether the user is able to pay for the selected product.	Check if the user is able to pay for the selected product.	2	1. Select the product.  2. Proceed for the payment.  3. Enter the transaction ID.	1. User should be able to select the product.  2. User should be able to proceed to the payment page and enter the	1. User could select the product.  2. User could enter the Transaction ID and it is reflected in the table.	Pass

					Transaction ID.		
<b>3.1</b>	Whether the User could add the image of the product.	Check if the user can add the image of the product in the add product list.	3	1. Navigate to the add products window.  2. add the image of the product.	1. User should be able to add the image of the product.	1. User was able to add the image successfully after adding the product the image was displayed correctly.	Pass
<b>3.2</b>	Whether the Product can be updated.	Check if the user can update the product.	3	1. Navigate to products.  2. select the product to be updated.  3. confirm the Product to be updated.	1. the user should be able to update the product successfully, and it should be reflected in the products window.	1. the user was able to update the product.	Pass

#### 5.4 Defects

1. In the Products module the user review part was not working due to some issues.
2. Updating the product has some issues.

## 6. Screenshots

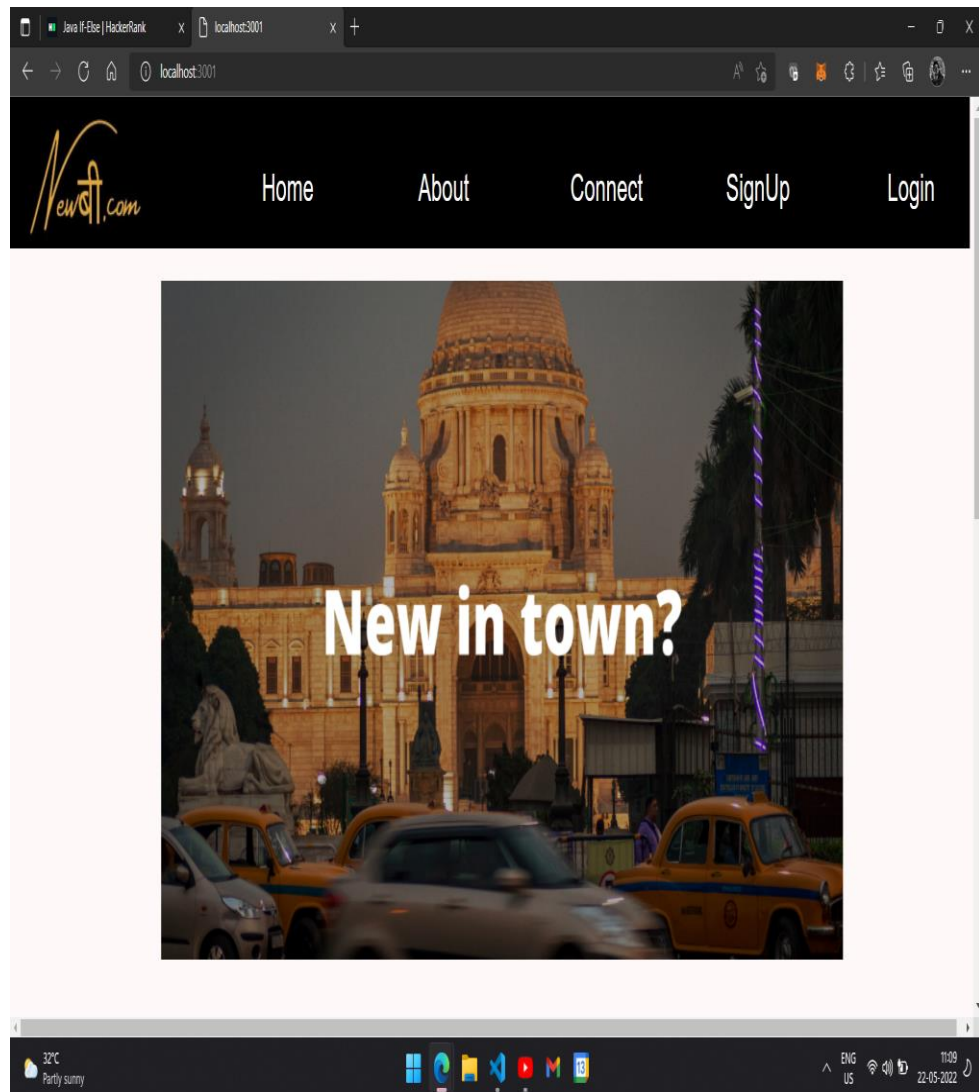


Figure 6.1.1 Home page

The screenshot displays the Newbie.com website with a navigation bar containing links for Home, About, Connect, SignUp, and Login. A 'Sign up' modal is centered on the screen, featuring input fields for Name, Enter Phone Number, Address, Enter Aadhar, Enter Email, and Enter Password. Below these fields is a checkbox labeled 'I am searching for a roommate' and a red 'Submit' button. The background of the website shows various category cards like Products, Food, Shopping, Places to visit, Travel, and Roommate. The footer contains the copyright notice: Copyright©2022 Newbie. All Rights Reserved. Developed & Designed by: Adarsh Verma, C.Saranya, Rouben Mohan Kurian.

Figure 6.1.2 Signup

The screenshot shows the Newbie.com website with the same navigation bar. A 'Login' modal is centered on the screen, with input fields for Enter Email and Enter Password, followed by a red 'Submit' button and a link for 'Forgot password?'. The background shows the same category cards as in the previous figure. The browser window at the bottom indicates the page is running on localhost:3001.

Figure 6.1.3 Login page

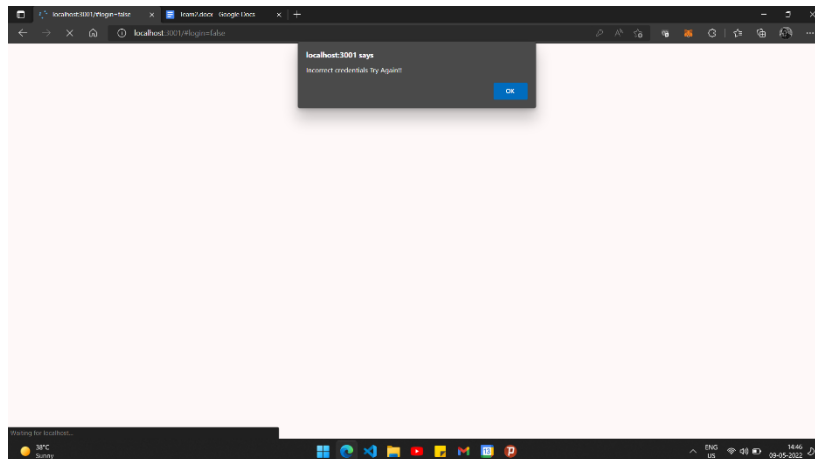


Figure 6.1.4 Incorrect Credentials

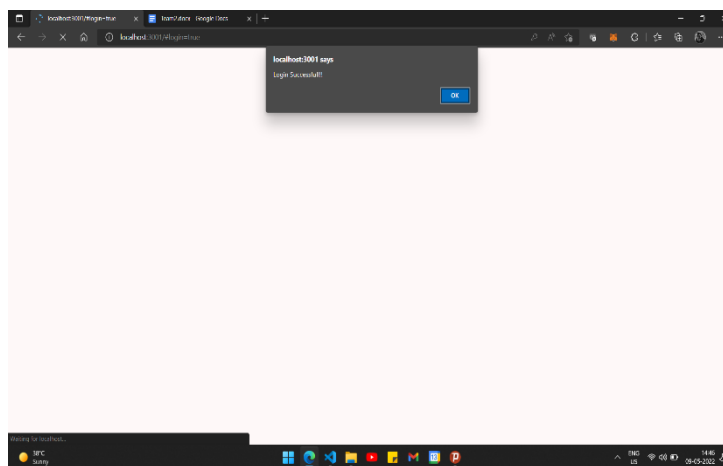


Figure 6.1.5 Successful login

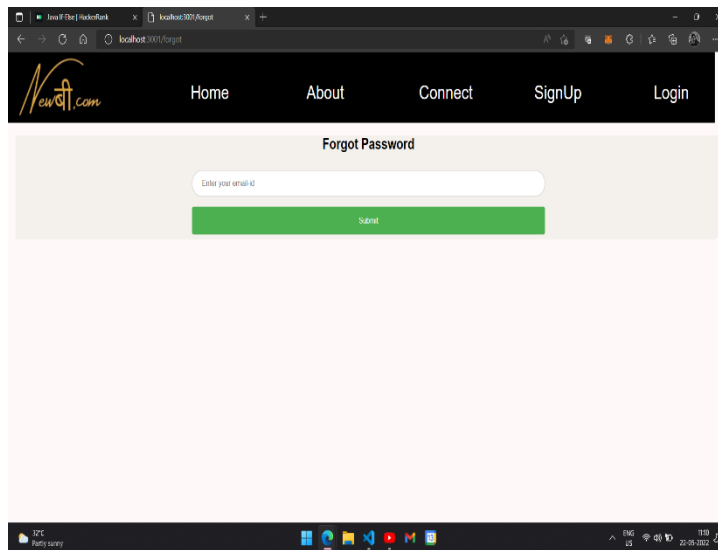


Figure 6.1.6 Successful login

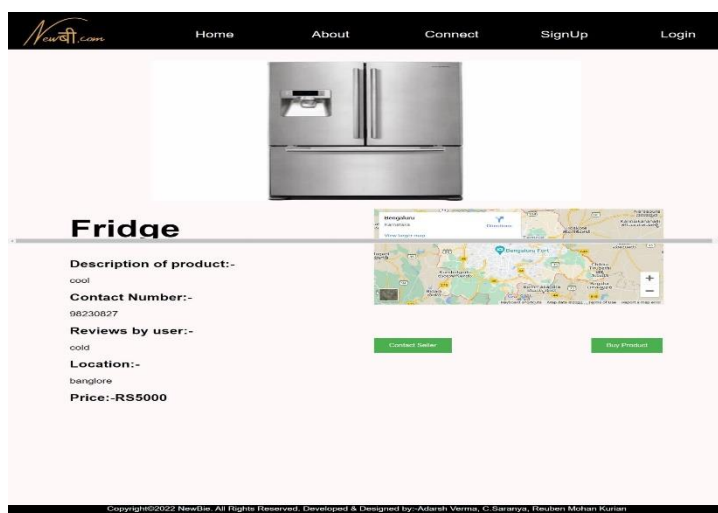


Figure 6.1.7 Product

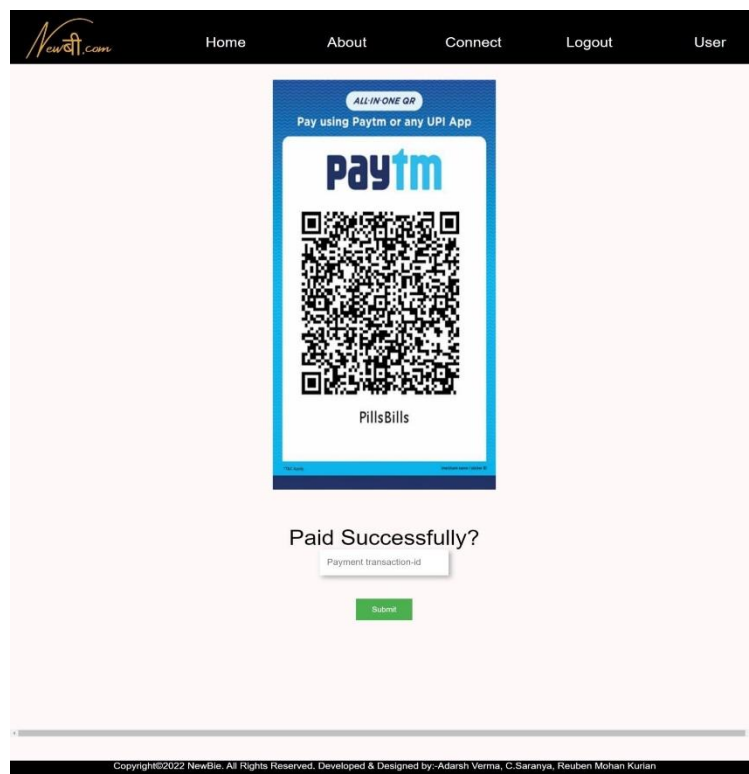


Figure 6.1.8 Payment

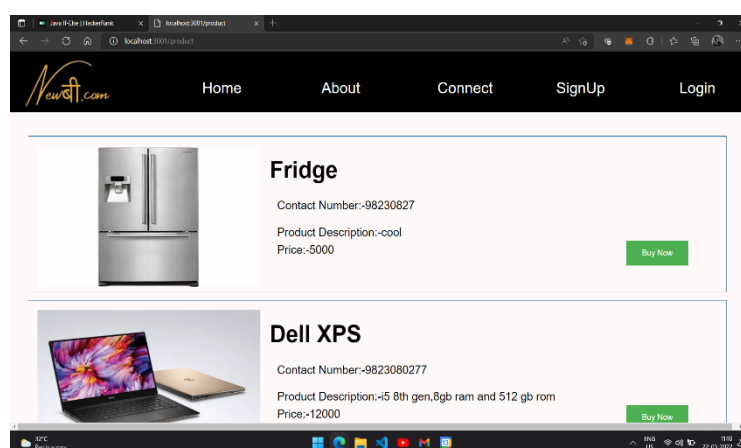


Figure 6.1.9 Products



The screenshot shows a web browser window with the URL `localhost:3001/contact seller?theme=Fridge&cat=product`. The page has a dark header with the `Newbie.com` logo and navigation links: `Home`, `About`, `Connect`, `Logout`, and `User`. The main content area is titled `Contact Person` and contains a form with the following fields:

- `Enter your email:` with the value `abc@gmail.com`
- `Your Message:` with the value `Hey`
- A green `Submit` button.

The browser's taskbar at the bottom shows the system clock as 12:00 on 22-05-2022.

Figure 6.1.10 Contact Seller

The screenshot shows the `Payments` page of the Newbie.com application. The header is the same as in Figure 6.1.10. Below the header, there is a navigation bar with links: `Add Product`, `Update Product`, `Add Items`, `Delete Items`, `Change Password`, and `Payments` (which is highlighted). The main content area displays a table with payment records.

Payment Id	Transaction Id	Date of payment	Payment Amount	Payment Status(Will show completed after verified by the admin)	Product Name
1	reuben	2022-05-12 01:17:29	12000	completed	Deil XPS
4	navin	2022-05-12 14:13:56	5000	completed	Fridge

The browser's taskbar at the bottom shows the system clock as 12:07 on 22-05-2022.

Figure 6.1.11 Payment Status

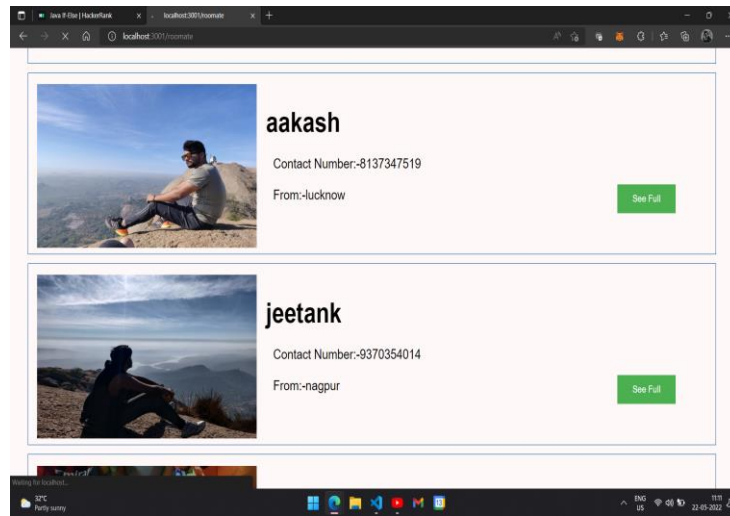


Figure 6.1.12 Users

## 7. Conclusion

Newbie.com helped in improving many things which are directly related to the real world environment. We inculcated the Coding guidelines which was known to us in prior but it was implemented properly in this project, we could learn about the different programming languages software tools, frameworks and databases while the initial stages of the project and after a successful completion of the planning stage we could select the best resources and platforms for our project. After this phase we learnt about making the structure of the website by making the Er diagrams, architectural diagrams, use case diagrams e.t.c. After knowing the structure of our website we started coding and created a database with the tables for the project, this helped us to learn about MySQL, Nodejs, EJS, PHPmyAdmin and Vscod. After the phase of coding we began testing our project which was a unique concept for us as it was just theoretical before this but

in this project we could work with it practically. We learnt about different softwares and tools and their workings and could implement most of it in our project. Other than the technical skills we could work on team building and communicational skills which helped us boost our confidence in taking up any task at any time and in the end with all these skills we managed to understand and learn about the timely completion of the project we could finish most of the modules in the given time.

## References

1. <https://www.w3schools.com/js/>
2. <https://nodejs.org/>
3. <https://www.w3schools.com/nodejs/>
4. <https://www.javatpoint.com/nodejs-tutorial>
5. <https://www.youtube.com/watch?v=BLI32FvcdVM>
6. [https://www.youtube.com/watch?v=YFmgNiimfyk&list=PLobAq7hWqZWGTfhj4jNQAVzJd\\_v6iTErQ](https://www.youtube.com/watch?v=YFmgNiimfyk&list=PLobAq7hWqZWGTfhj4jNQAVzJd_v6iTErQ)
7. [https://www.youtube.com/watch?v=7H\\_QH9nipNs](https://www.youtube.com/watch?v=7H_QH9nipNs)
8. <https://www.myroomie.in/>
9. <https://www.npmjs.com/package/nodemailer>
10. <https://ejs.co/>