A PROJECT REPORT ON

# TIFFIN WALA

### An Online Tiffin Delivery Service

SUBMITTED IN PARTIAL FULFILMENT OF

#### PG DIPLOMA IN ADVANCED COMPUTING (PG-DAC)



UNDER THE GUIDANCE OF

Mr. Jitesh Bafna

PRESENTED BY

|  |  |
| --- | --- |
| 230340120061 | Rohan Dehankar |
| 230340120130 | Nikhita Upare |
| 230340120148 | Suraj Patil |
| 230340120153 | Pradip Borade |
| 230340120181 | Sajid Patel |
| 230340120234 | Yogesh Pawar |
|  |  |

AT

CENTER FOR DEVELOPMENT OF ADVANCED COMPUTING C- DAC, PUNE

# ACKNOWLEDGEMENT

The project “TIFFIN WALA” was a great learning experience for us and we are submitting this work to Advanced Computing Training School (C- DAC ACTS, Pune).

We are very glad to mention the name of Mr. Jitesh Bafna for his valuable guidance to work on this project.

We are highly grateful to Ms. Risha P. R., Manager of ACTS Training Centre, CDAC, for her guidance and support whenever necessary during the course of our journey to acquire PG-Diploma in Advanced Computing (PG-DAC) through CDAC ACTS, Pune.

Our heartfelt thanks go to Mrs. Priyanka Ranade (Course Coordinator, PG-DAC) who gave us all the required support and kind coordination to provide all the necessities to complete the project and throughout the course up to the last day of the course.

From:

230340120061 Rohan Dehankar

230340120130 Nikhita Upare

230340120148 Suraj Patil

230340120153

230340120181

Pradip Borade

Sajid Patel

230340120234 Yogesh Pawar

##### **TABLE OF CONTENTS**

|  |  |
| --- | --- |
| CONTENT | PAGE NO. |
| 1. Introduction | 2 |
| 1.1 Purpose | 2 |
| 1.2 Project Background | 2 |
| 1.3 Goals of the project | 2 |
| 2. Overall Description | 3 |
| 2.1 Proposed Methodology | 3 |
| 2.2 S/W and H/W Requirements | 3 |
| 2.3 Technology platform used for project | 4 |
| 3. Requirements Specification | 5 |
| 3.1 External Interface Requirements | 5 |
| 4. System Diagram | 7 |
| 4.1 Use Diagram | 8 |
| 4.2 Data Flow Diagram | 10 |
| 4.3 ER Diagram (MySQL Auto Generated) | 13 |
| 4.5 Project Screenshot | 14 |
| 5. Table Structure | 27 |
| 5.1 Customers | 27 |
| 5.2 Vendors | 27 |
| 5.4 Tiffin | 28 |
| 5.5 Orders | 28 |
| 5.8 Address | 29 |
| 5.9 Logins | 29 |
| 6. Conclusion | 30 |
| 6.1 Future Scope | 30 |
| 7. References | 31 |
|  |  |

**LISTS OF FIGURE**

|  |  |
| --- | --- |
| FIGURE | PAGE NO. |
| Figure 1 Use Case Diagram Admin | 7 |
| Figure 2 Use Case Diagram Vendor | 8 |
| Figure 3 Use Case Diagram Customer | 9 |
| Figure 4 Vendor Level 1 Data Flow | 10 |
| Figure 5 Customer Level 1 Data Flow Diagram | 11 |
| Figure 6 Admin Level 2 Data Flow Diagram | 12 |
| Figure 7 ER Diagram (MySQL Auto Generated) | 13 |
| Figure 11 Project Screenshots | 14 |
|  |  |

**ABSTRACT**

Tiffin Wala is a website that provides a platform for tiffin service providers and customers to buy tiffins from tiffin walas.The goal of the project is to provide hygienic and high-quality food services to customers, particularly students and office workers.The project aims to make the food exploration and booking experience easier for users.The tiffin service industry has seen an increase in demand due to the current demand for healthy, preservative-free, and tasty food options.The project provides an opportunity for tiffin service providers to expand their customer base and reach a larger audience.The project utilizes a user-friendly interface for customers to browse the catalogue and order food online.The combination of frontend and backend technologies allows for the development of a full-stack web application that provides a seamless experience for both tiffin vendors and customers.The project can be used as a model for other tiffin service providers looking to expand their customer base and reach a larger audience.

1. **INTRODUCTION**
   1. **Purpose**

The Online tiffin service 'TiffinWala' website is intended to provide complete solution for Vendors, Customer as well as Internal users (Staff) as a single Gateway using internet. Vendors could be anyone who want to setup their tiffin service center but don’t have platform for the same specially housewife’s who wish have a source of side income. It will enable vendors to provide tiffin service online, consumers to browse through all the available tiffin service and order tiffin without physically visiting the tiffin service center.

* 1. **Project Background**

In the current competitive world, many youths travel to different unknown locations for their basic education or jobs. The main problem they face is the food they get, and they crave for homemade food, but it is difficult to find it. On the other hand, some housewives wish to work and earn money to gain financial independence. It is difficult for these ladies to reach customers and market their products.

* 1. **Goals of the project**

The main objective of this project is to give a common platform for the customers and service provider. This system will help consumers from various places to communicate with various providers (tiffin service providers) and ease their searching efforts. The main interest of the Project is to create a central service system that will act as a bridge between providers and consumers.

1. **OVERALL DESCRIPTION**
   1. **Proposed Methodology**

* Er Ghar Ka Dabba System is a web application.
* There are mainly two types of users. One is the vendor (Mess Owner) and the other is the customer.
* Customers can search for mess menus, and special dishes at their convenience.
* Vendors can get more customers.
* Ghar Ka Dabba System provides the functions which connect the customers and the vendors through the portal.
* The Ghar Ka Dabba System will be administrated by Admin.
  1. **S/W and H/W Requirements**

**Server Side:**

* HDD: 500 GB or above
* Processor: Intel corei5or above
* RAM: 4GB or above
* Database: MySQL

**Client Side (minimum requirement):**

* Processor: Intel Dual Core
* HDD: Minimum 80GB Disk Space
* RAM: Minimum 2GB
* OS: Windows 7, Linux
  1. **Technology platform used for project**

###### HTML , Bootstrap

###### JavaScript

###### ReactJS

###### Spring Boot REST API

###### Hibernate

###### JPA

###### MySQL

###### GITHUB

* 1. **Reason for using specific technologies:**
* HTML and Bootstrap basic formatting and rendering on browser
* JavaScript makes web pages dynamic
* ReactJS allowed us to manage routing, state, components ,html pages

,toggling, navigation with ease

* Spring boot REST API allows us to create REST APIs with minimal configurations
* Hibernate helped my mapping entities , their state and deals with database
* JPA managed relational data in entities
* MySql allowed to store data and perform CRUD operations in them
* Github Helped us to store, maintain version control, and manage the source code between the team.

1. **REQUIREMENTS SPECIFICATION**
   1. **External Interface Requirements**

User Interfaces:

* All the users will see the same page when they enter in this website. This page asks the users a username and a password.
* After being authenticated by correct username and password, user will be redirect to their corresponding profile where they can do various activities.
* The user interface will be simple and consistence, using terminology commonly understood by intended users of the system. The system will have simple interface, consistence with standard interface, to eliminate need for user training of infrequent users.

Hardware Interfaces:

* No extra hardware interfaces are needed.
* The system will use the standard hardware and data communication resources. This includes, but not limited to, general network connection at the server/hosting site, network server and network management tools.

**Application Interfaces:**

Web Browser:

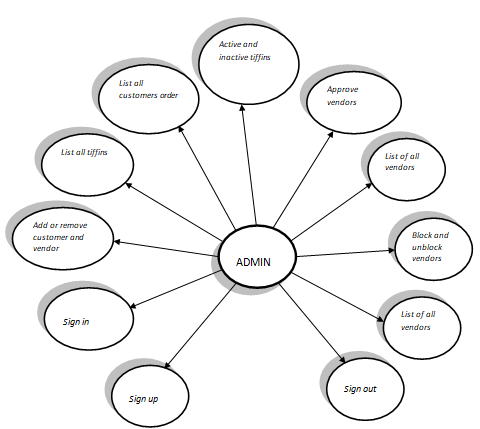
The system is a web-based application; clients need a modern web browser such as Mozilla Firebox, Internet Explorer, Opera, and Chrome. The computer must have an Internet connection in order to be able to access the system.

Communications Interfaces:

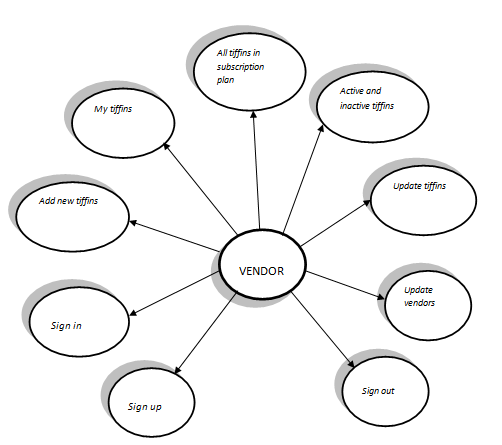
* This system uses communication resources which includes but not limited to, HTTP protocol for communication with the web browser and web server and TCP/IP network protocol with HTTP protocol.
* This application will communicate with the database that holds all the booking information. Users can contact with server side through HTTP protocol by means of a function that is called HTTP Service. This function allows the application to use the data retrieved by server to fulfil the request fired by the user.

1. **SYSTEM DIAGRAM**
   1. Use Diagram

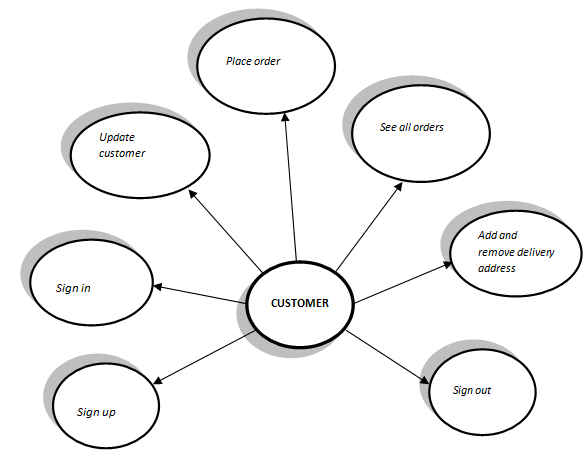
**Figure 1.Use Case Diagram Admin Activity**



**Figure 2.Use Case Diagram Vendor Activity**

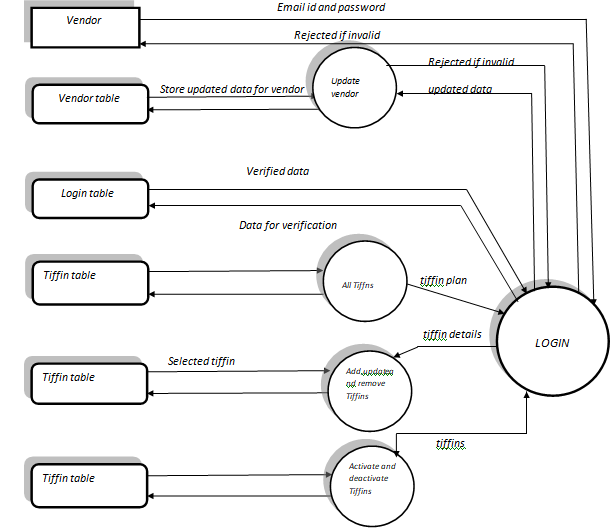


**Figure 3.Use Case Diagram Customer Activity**

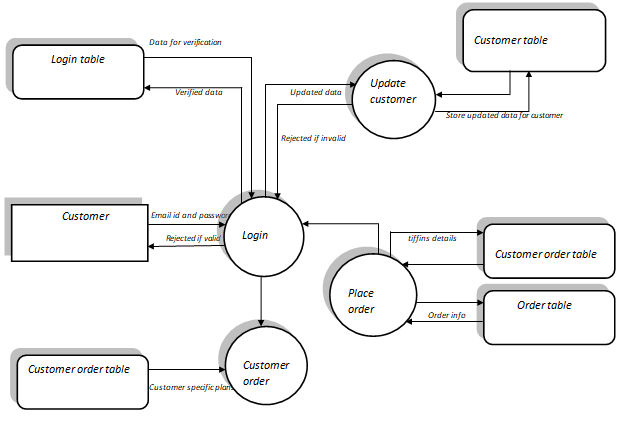


* 1. Data Flow Diagram

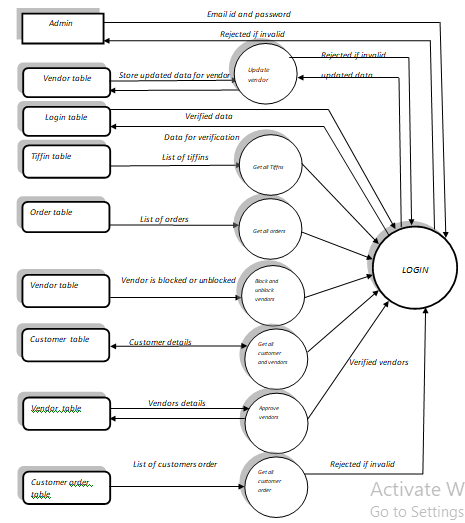
**Figure4.Vendor Level 1DFD**



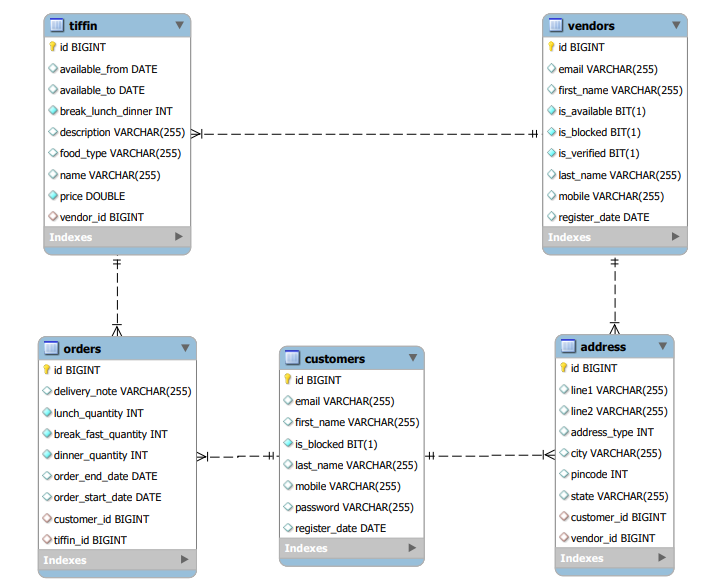
**Figure5.Customer Level 1 DFD**

****

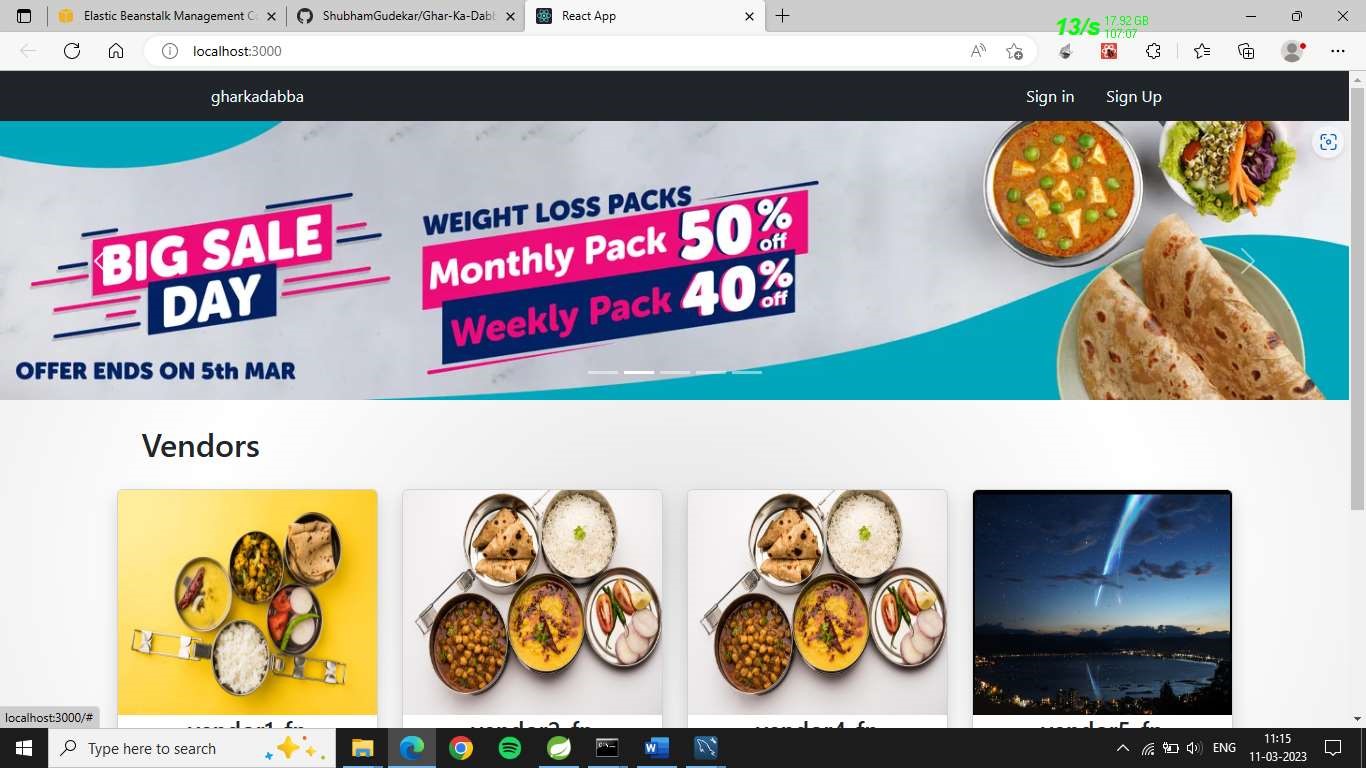
**Figure6.Admin Level 1 DFD**

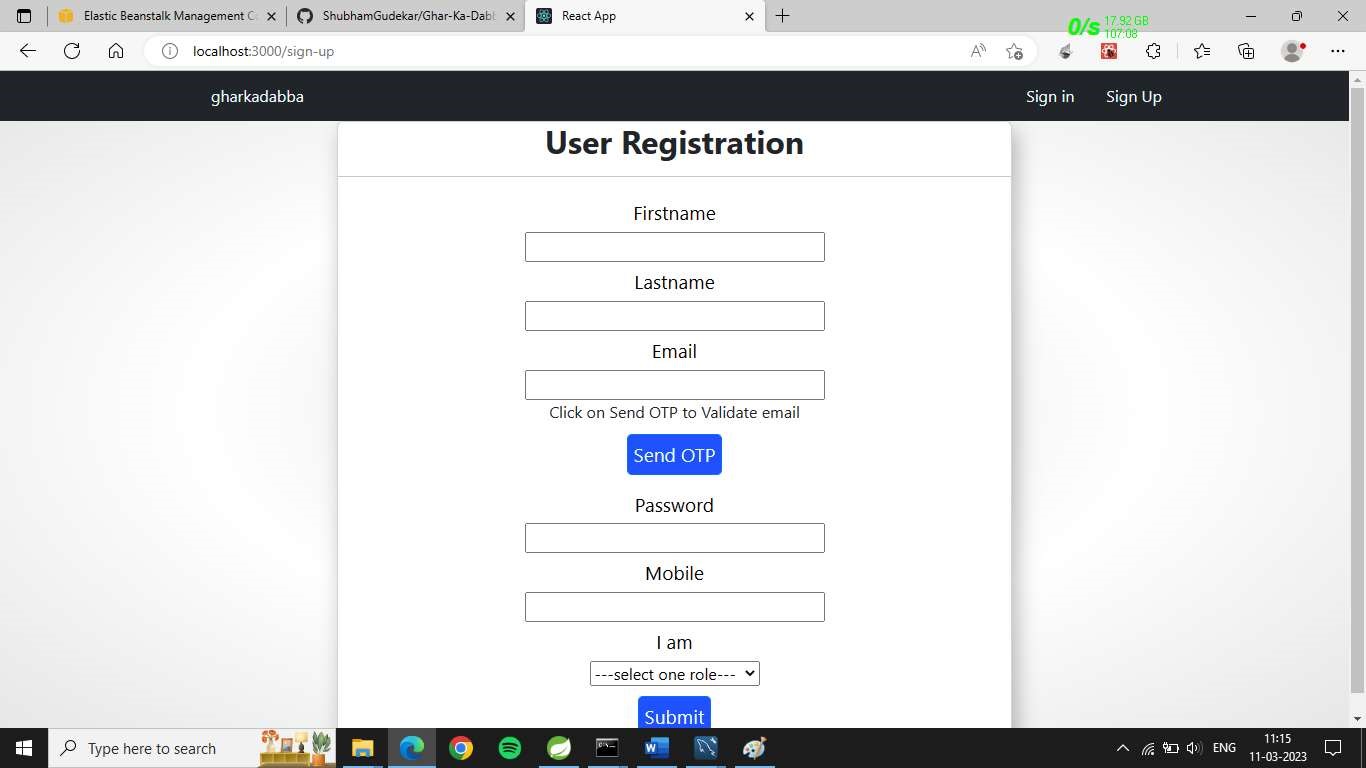
****

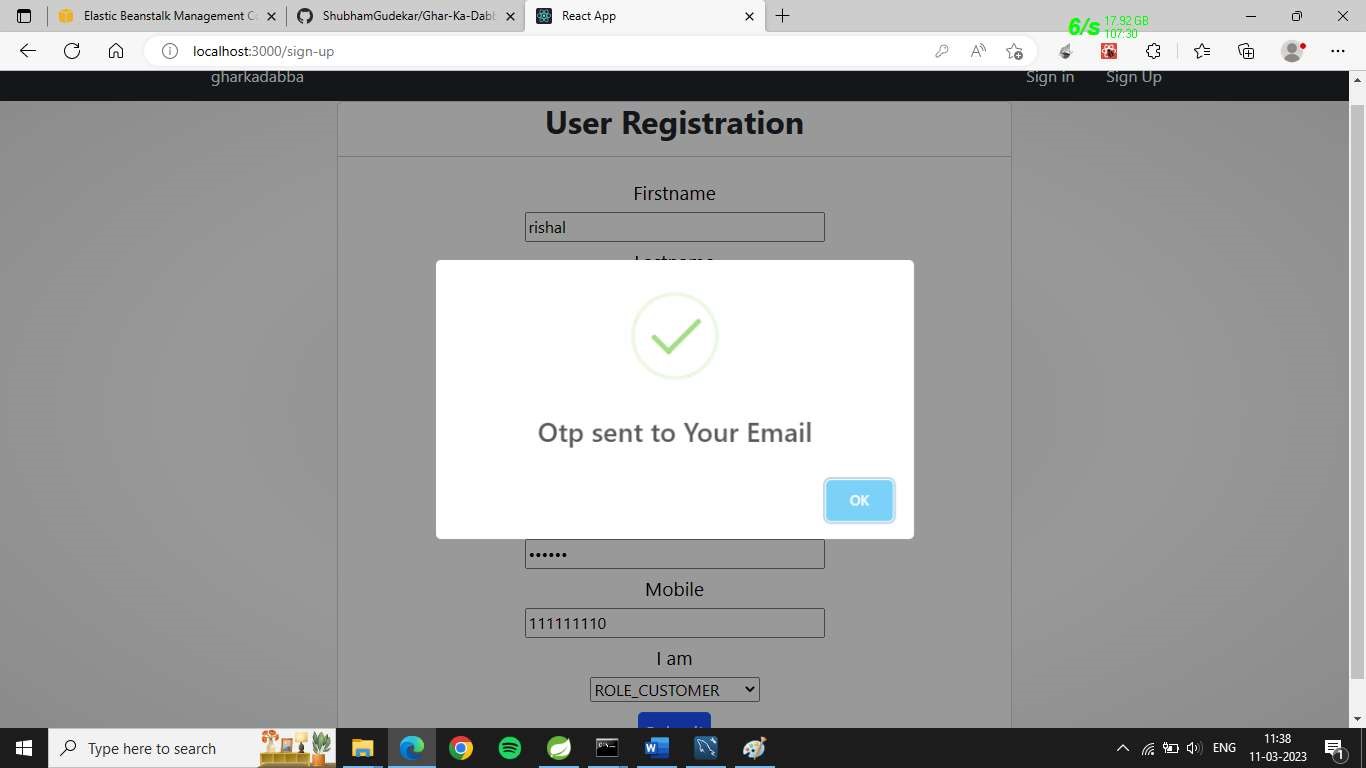
* 1. **ER Diagram (MySQL Auti Generated)**

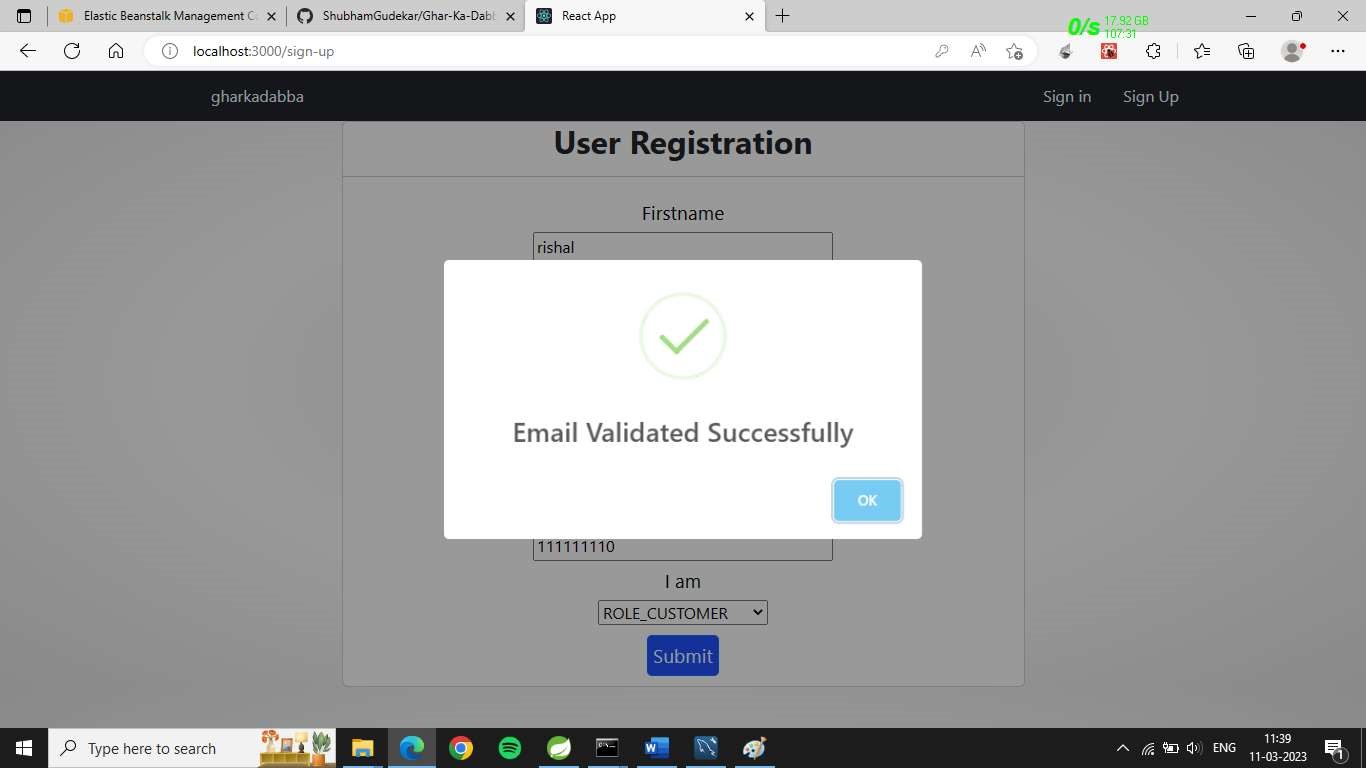


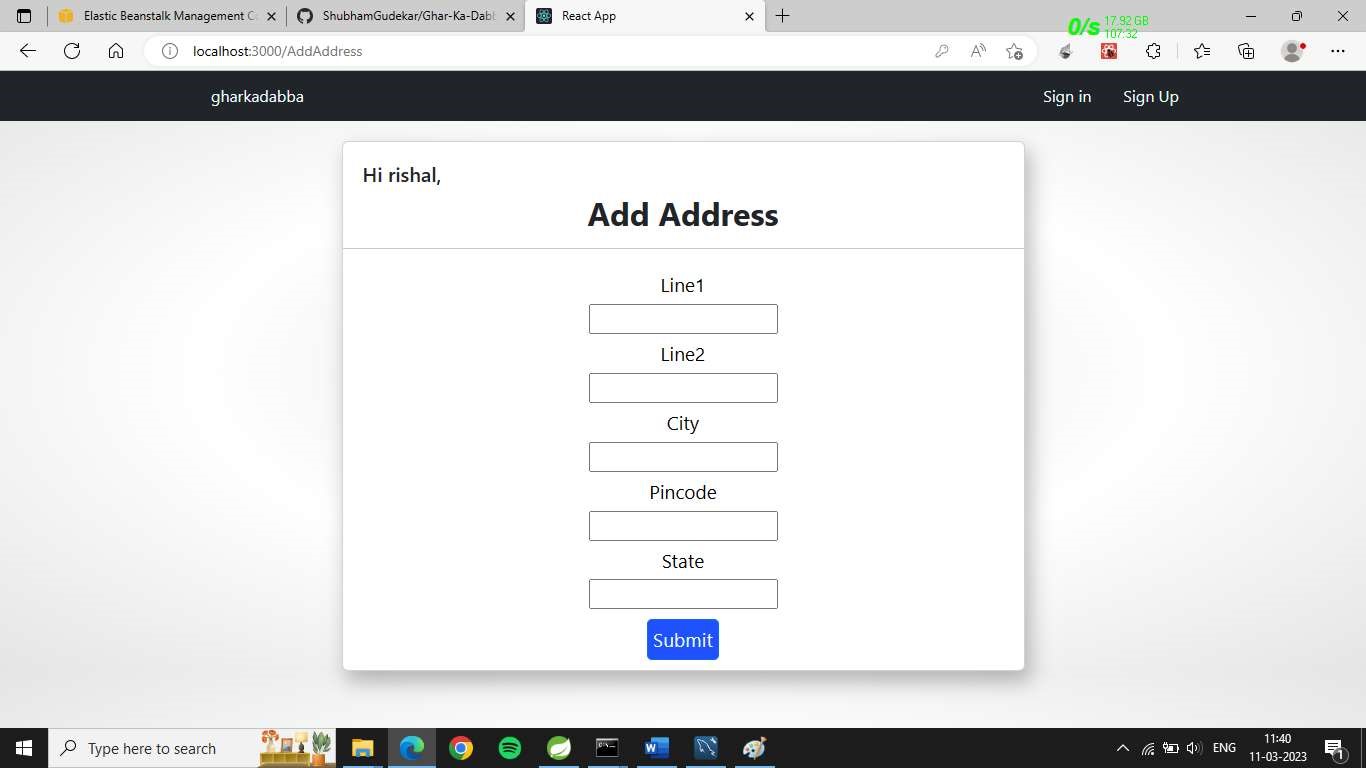
* 1. **Project Screenshots**

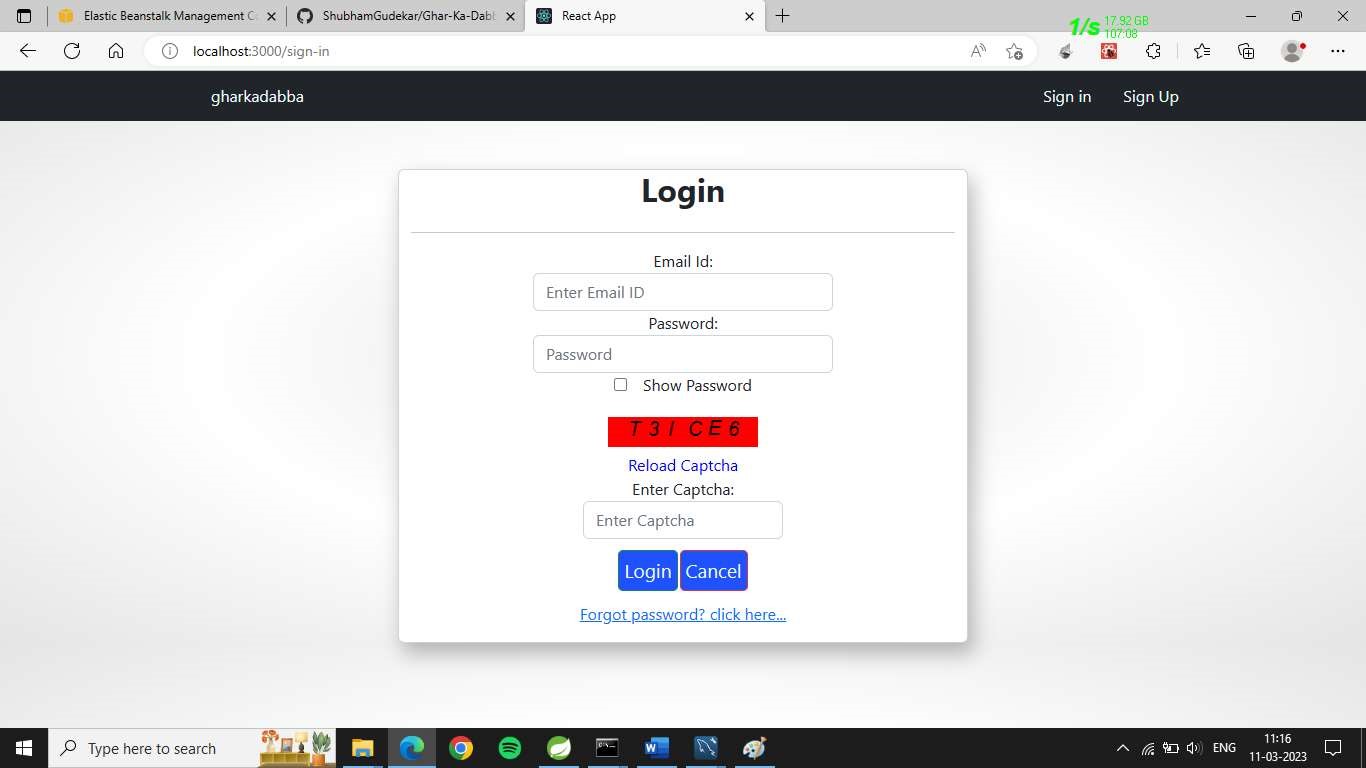


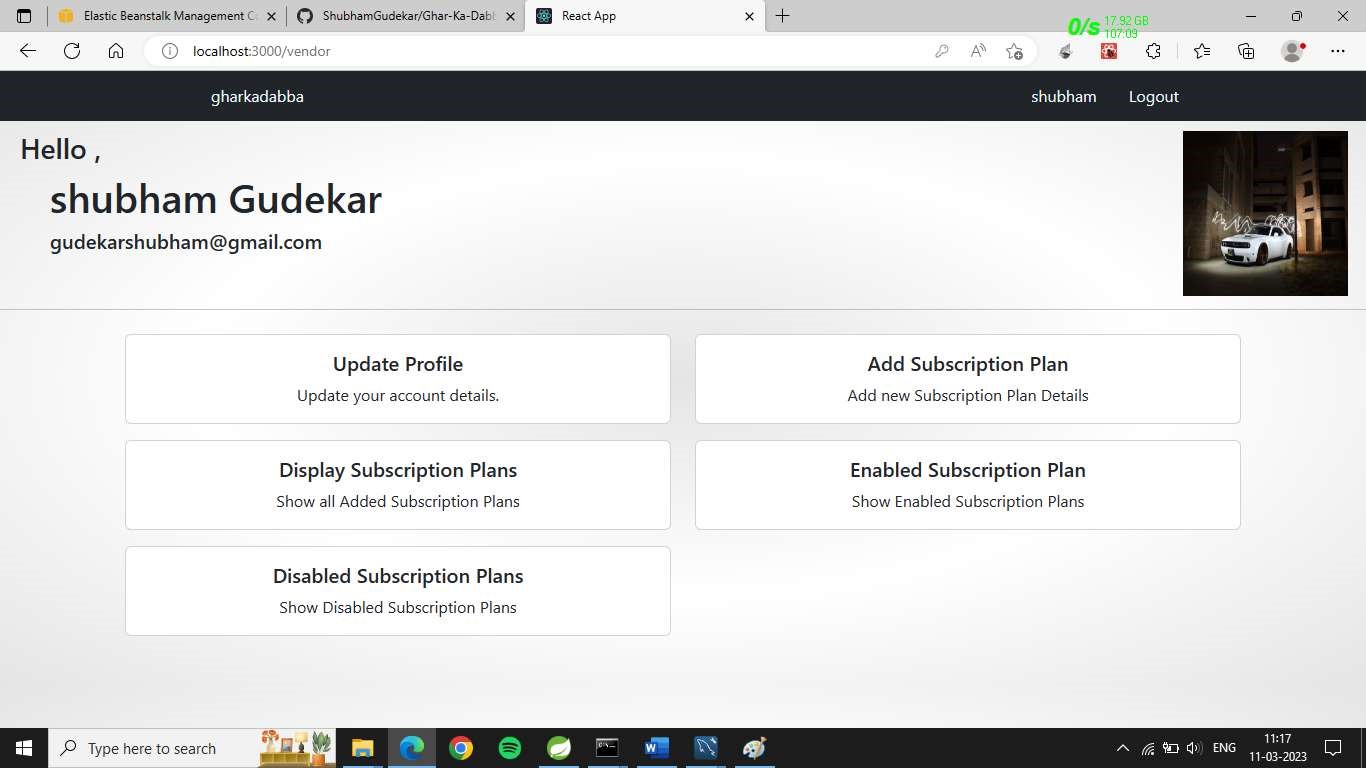


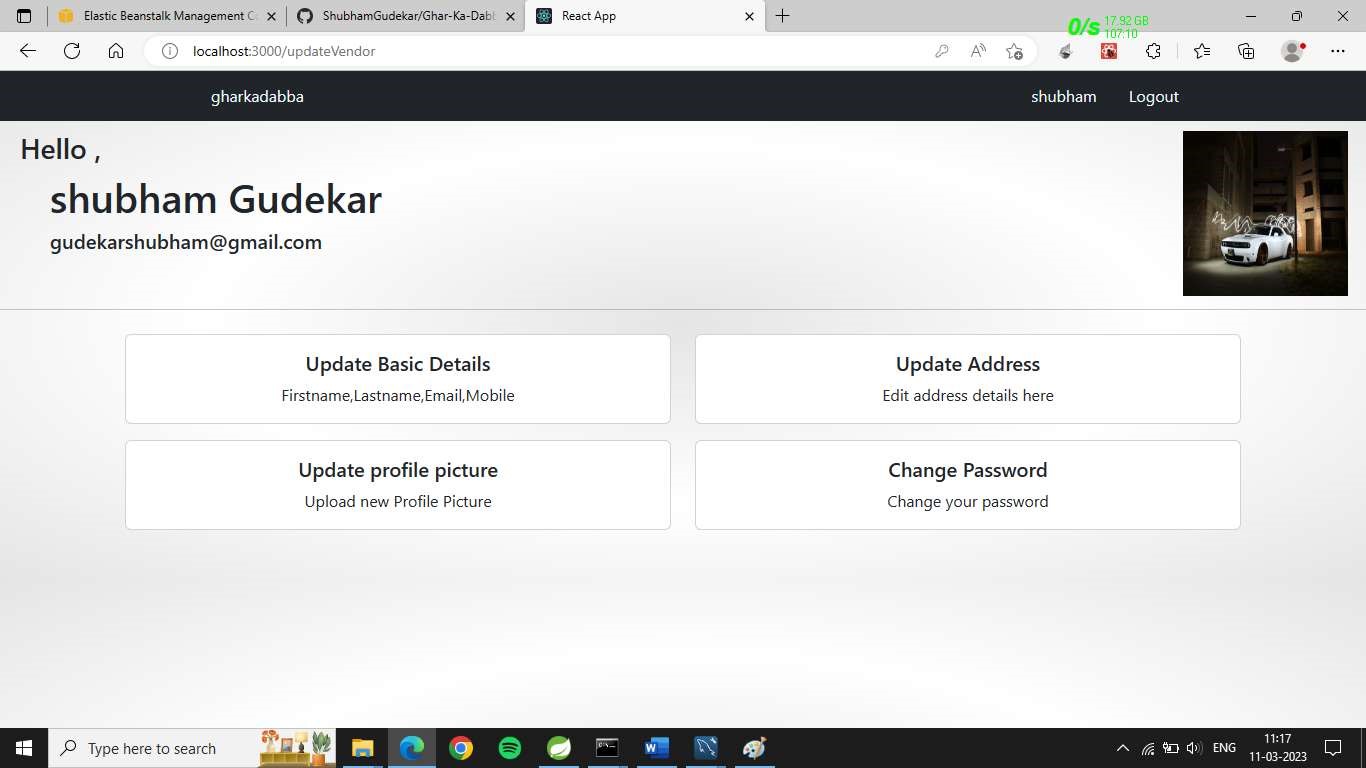


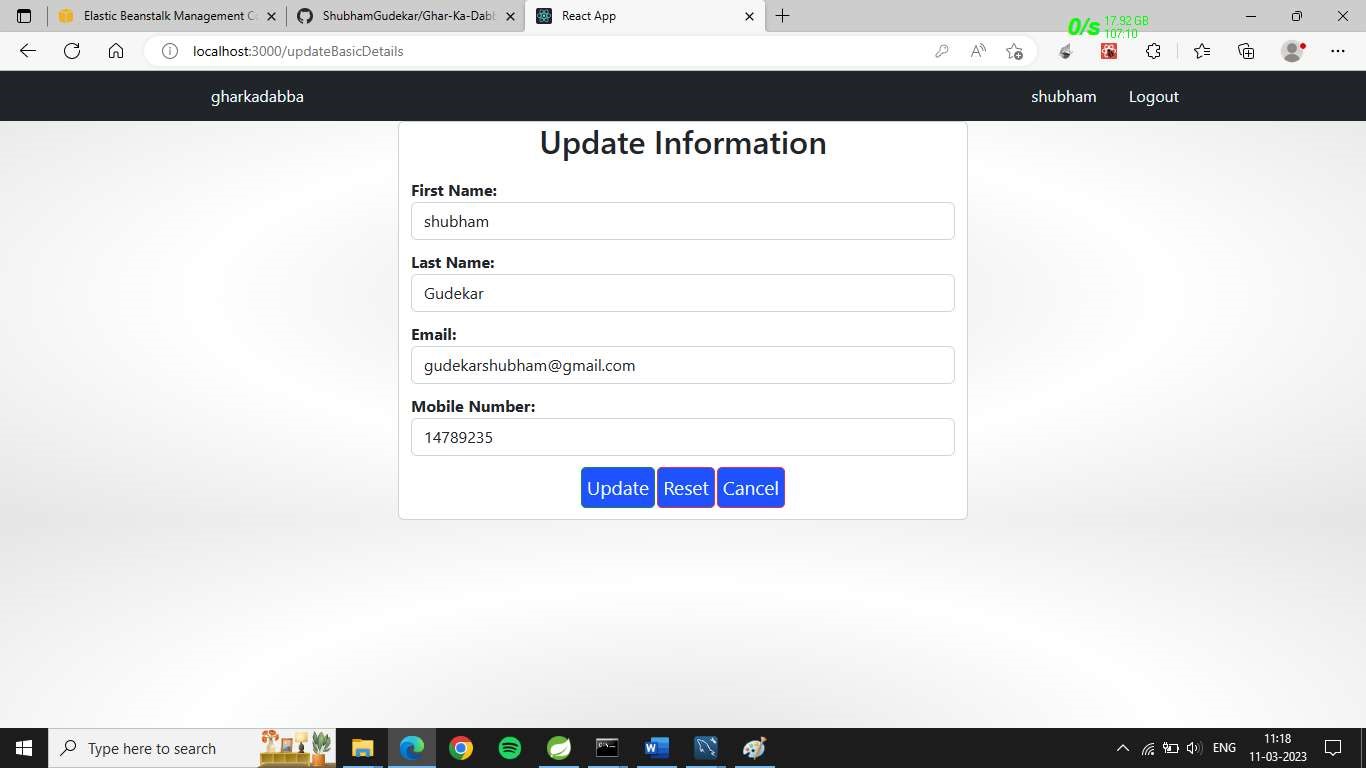


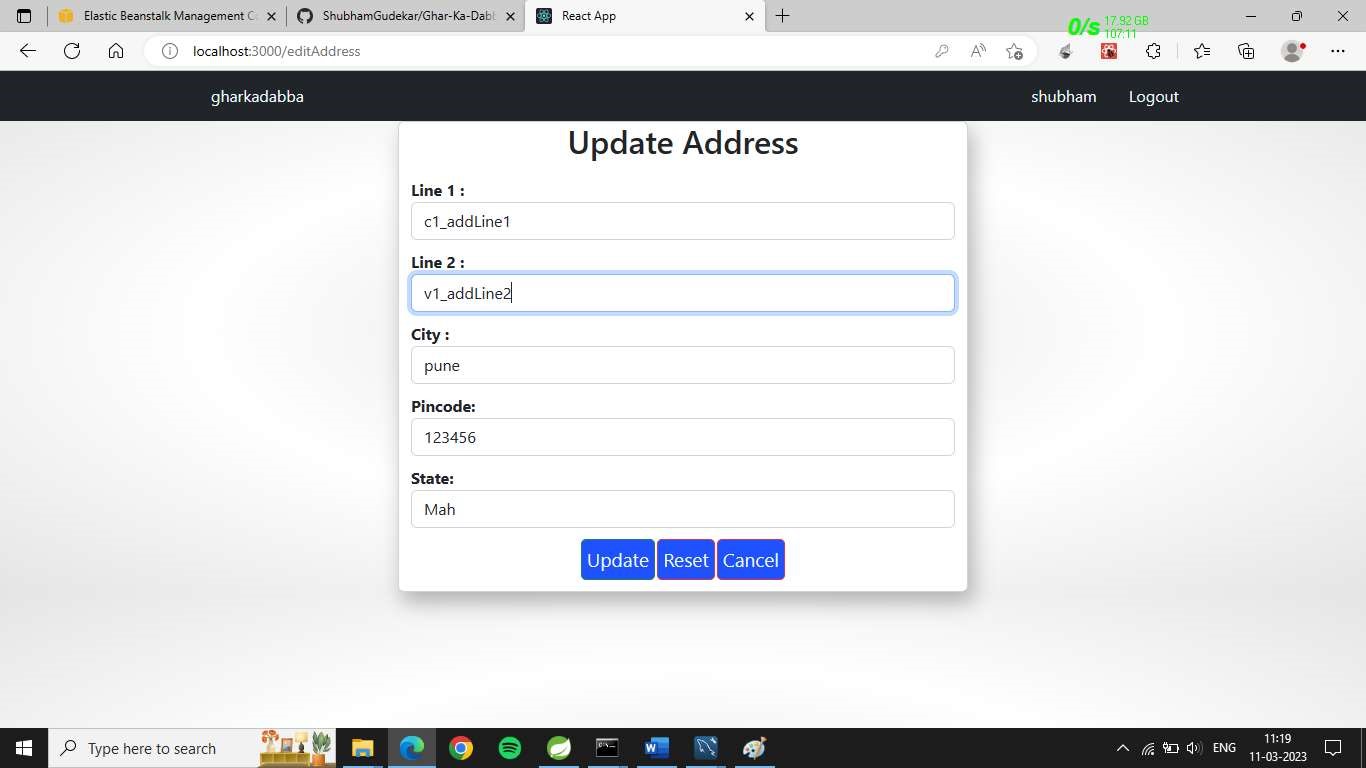


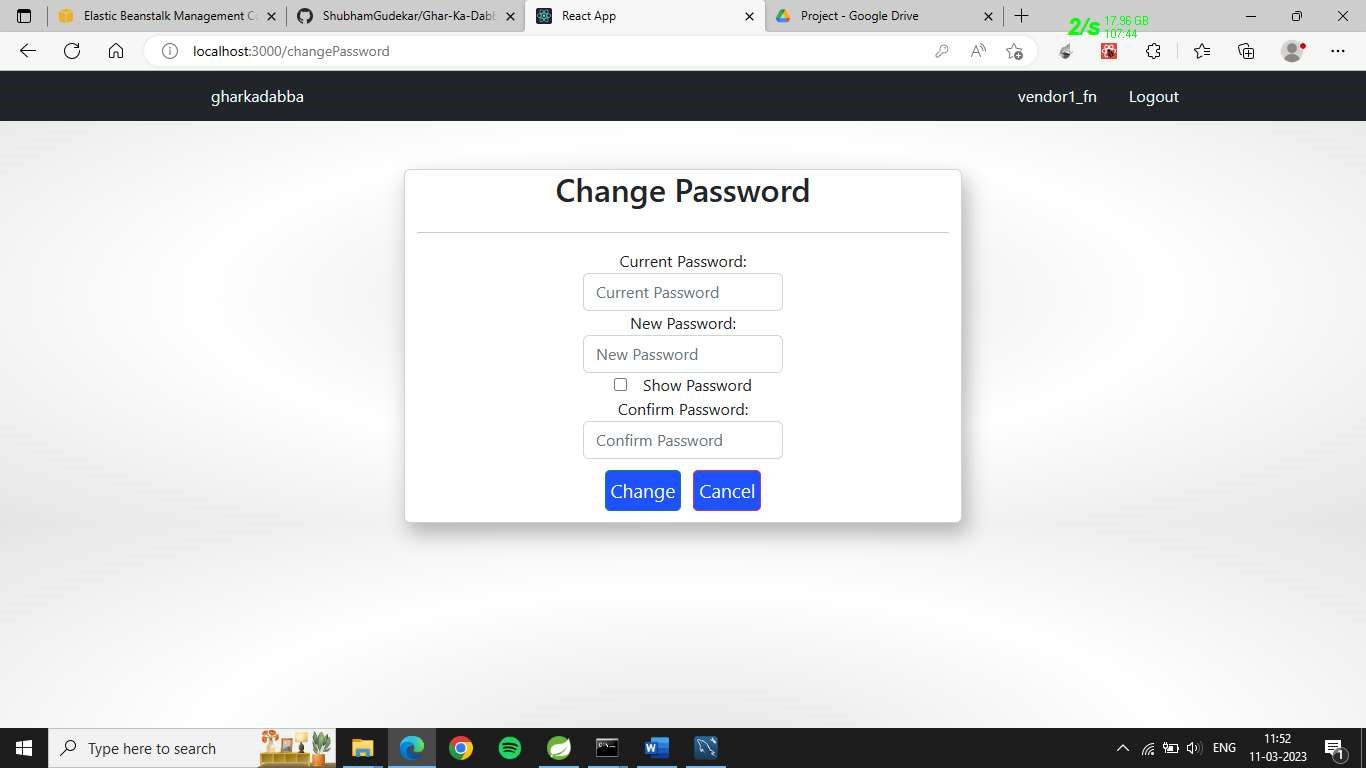


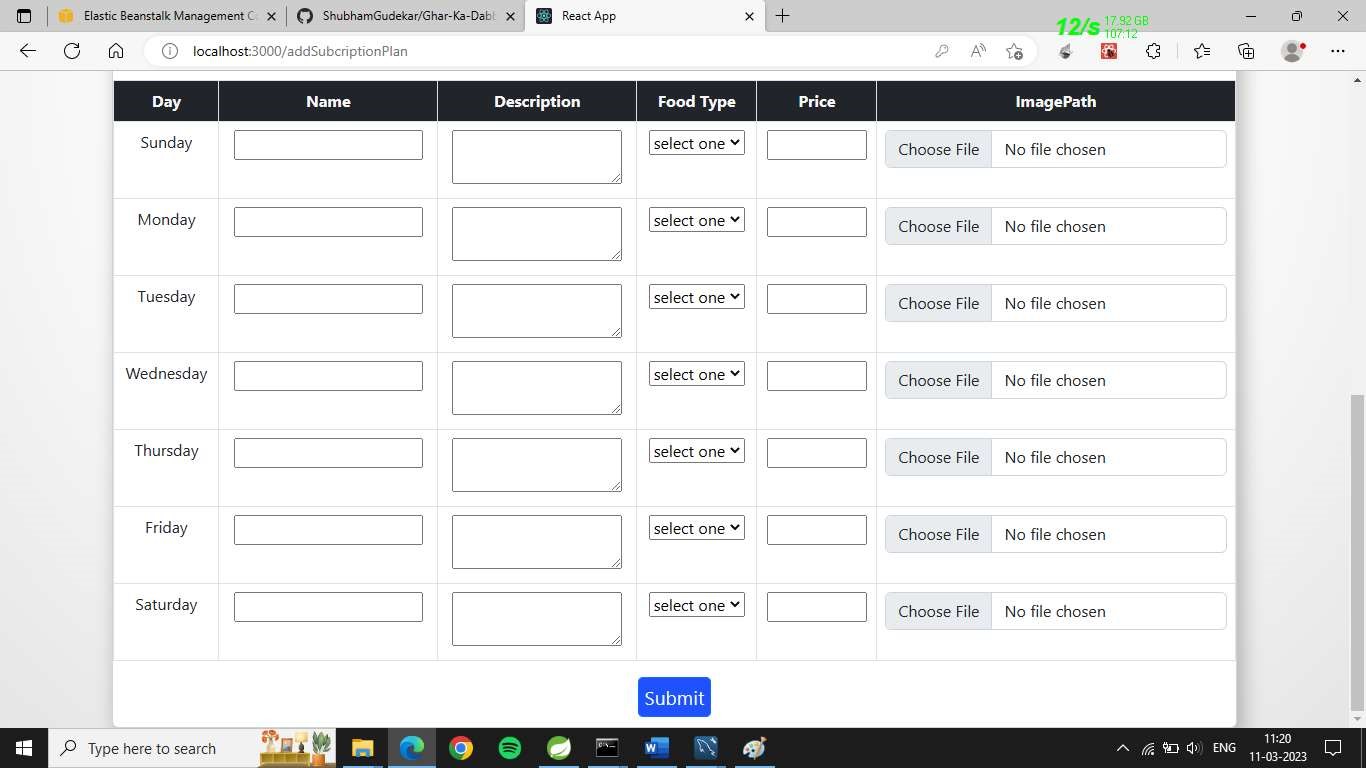


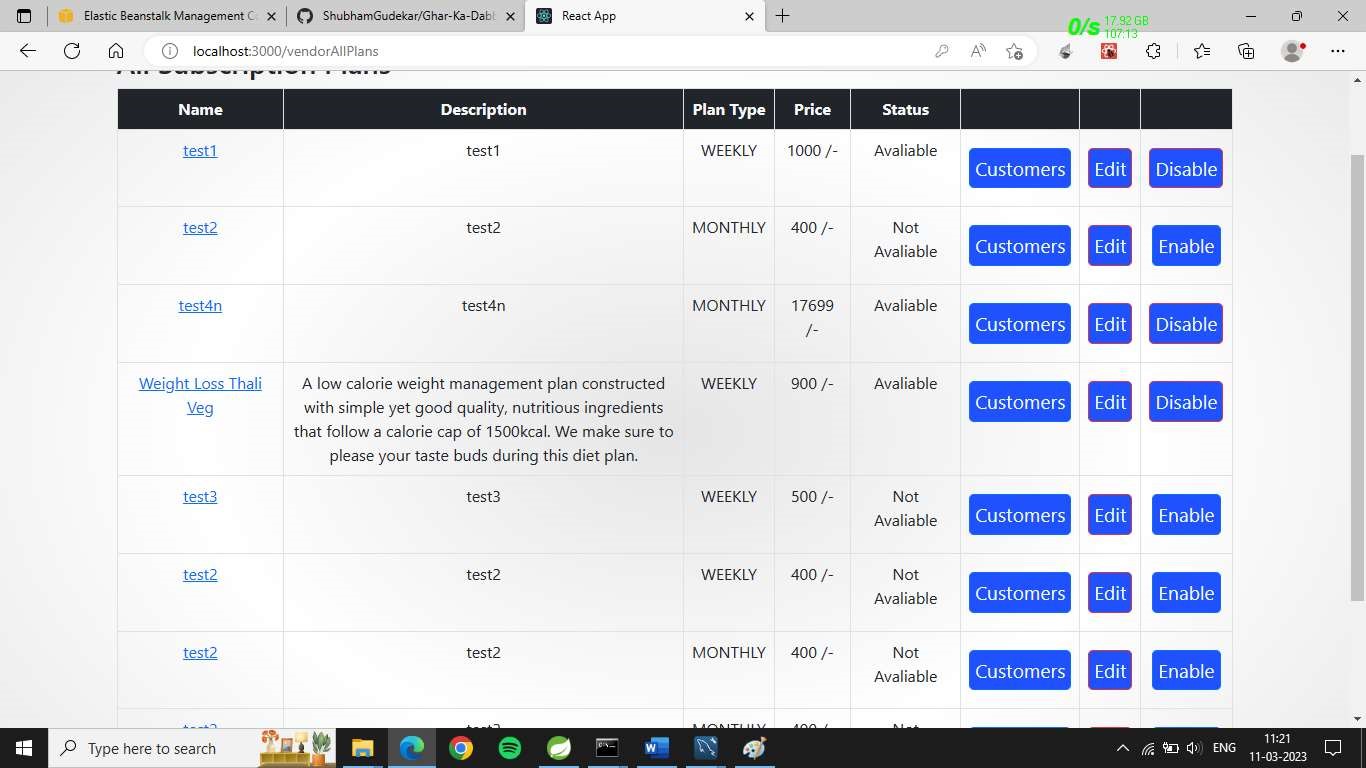


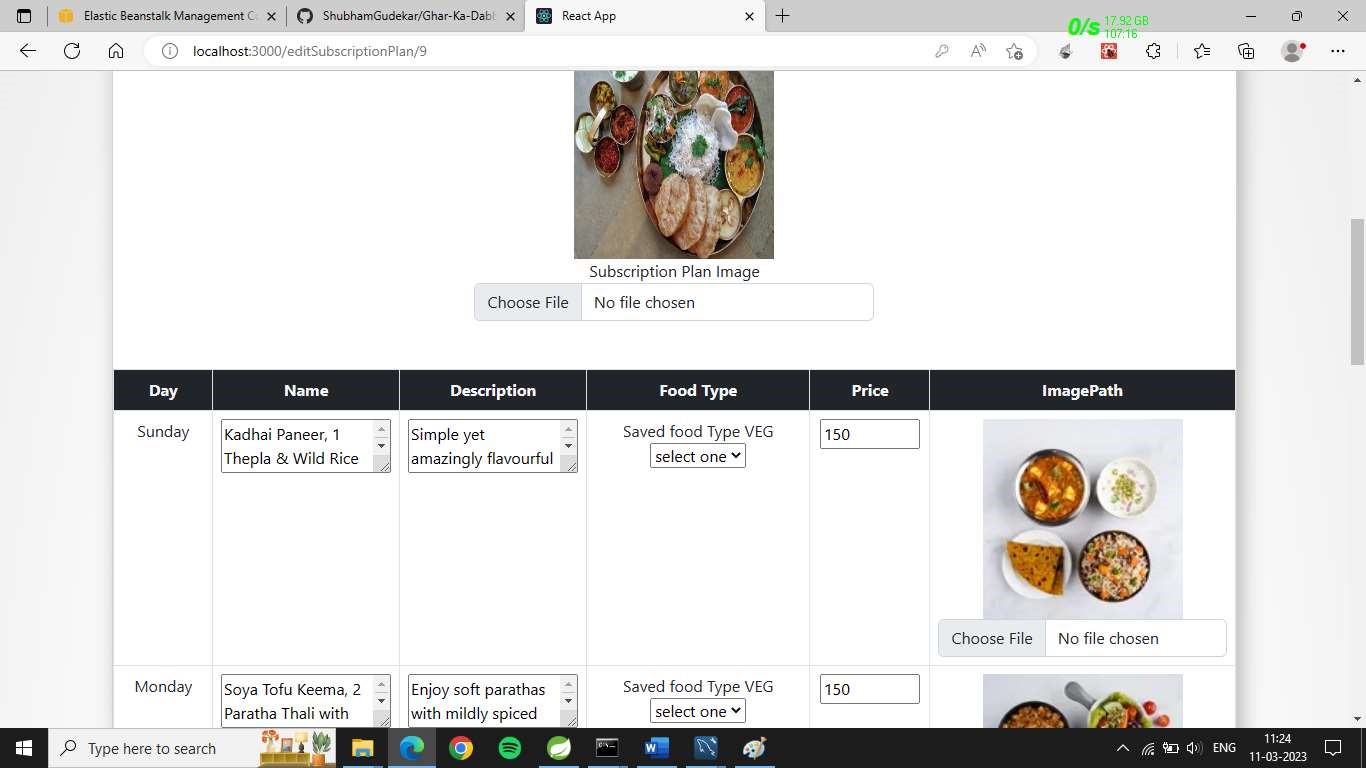


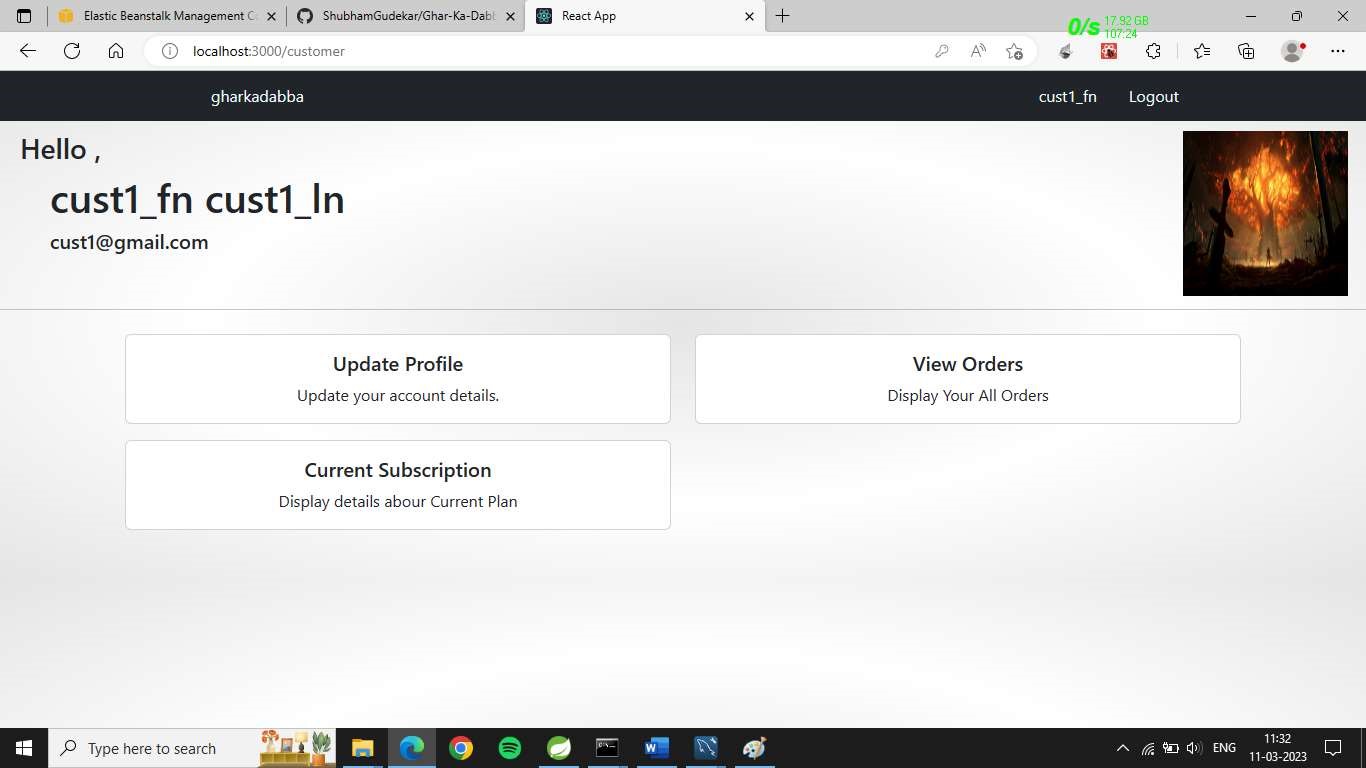


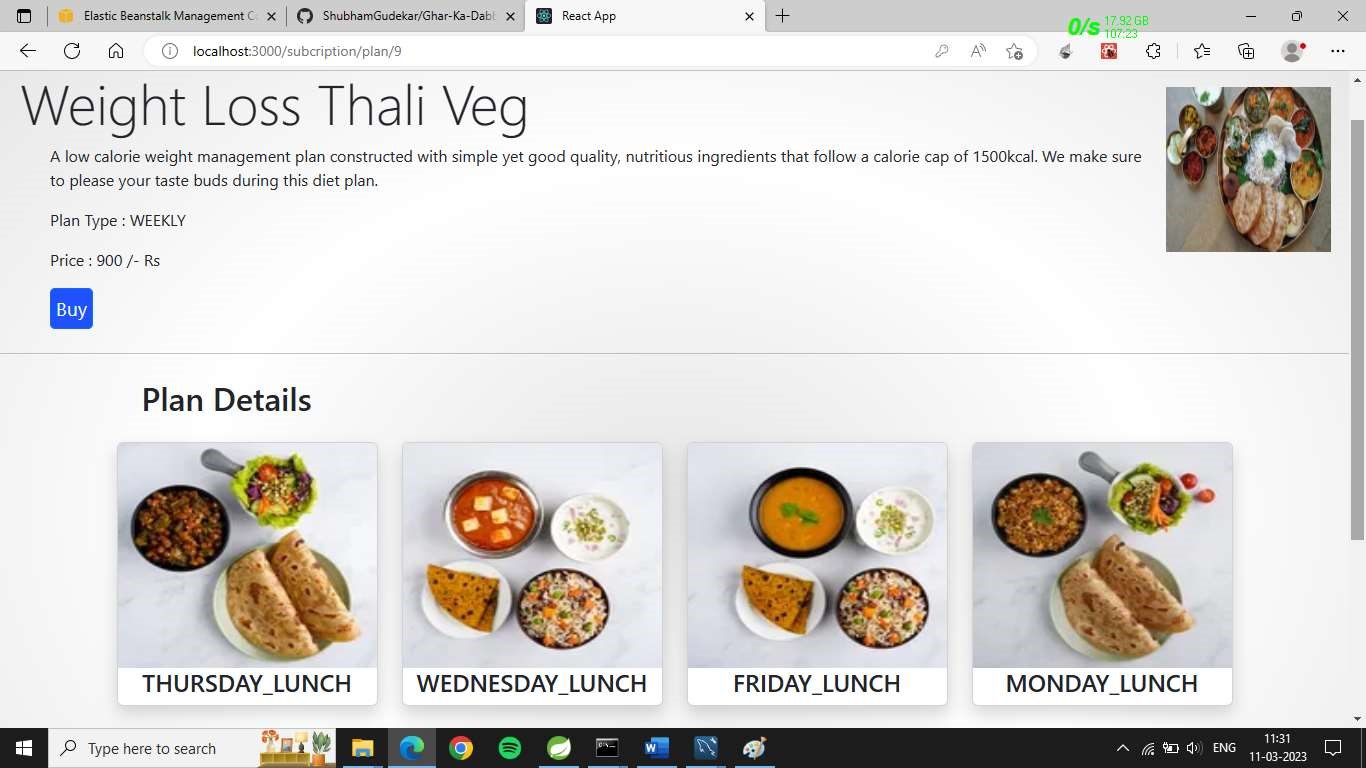


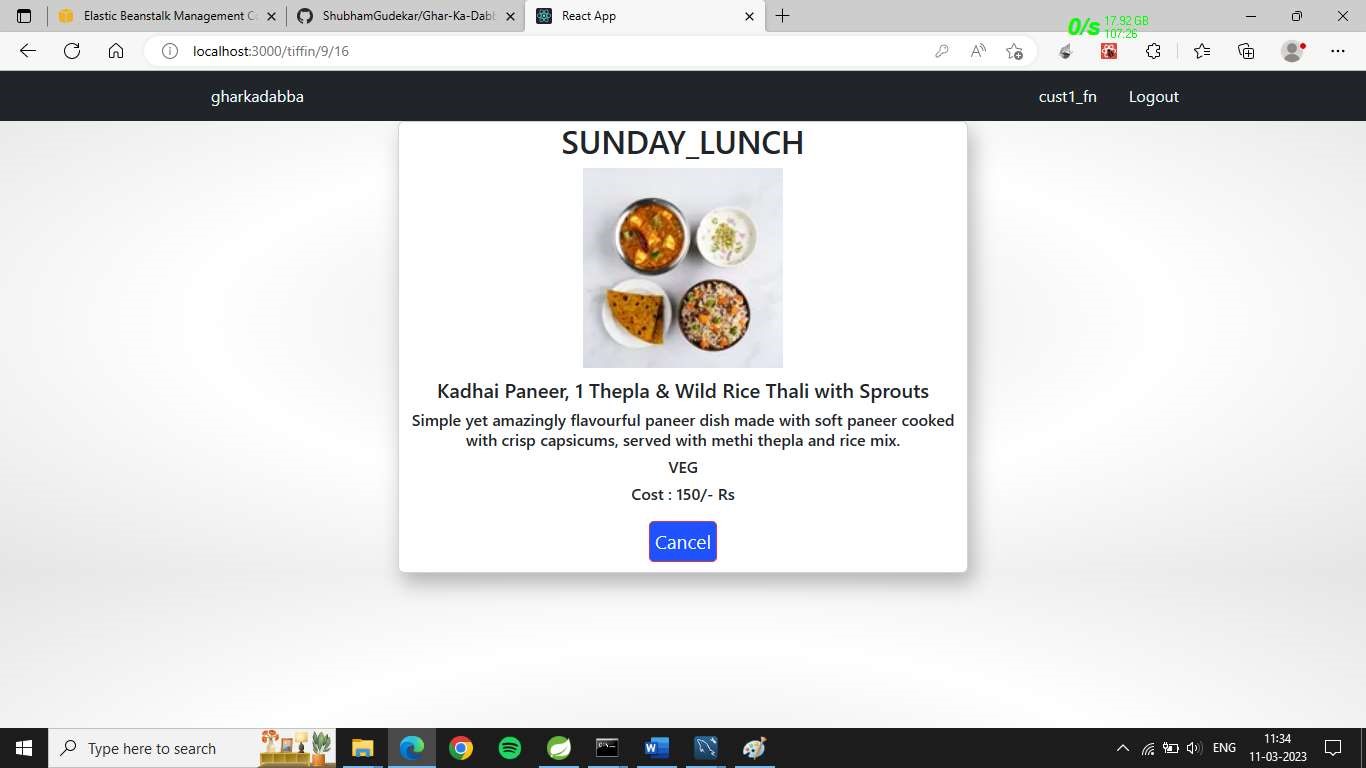


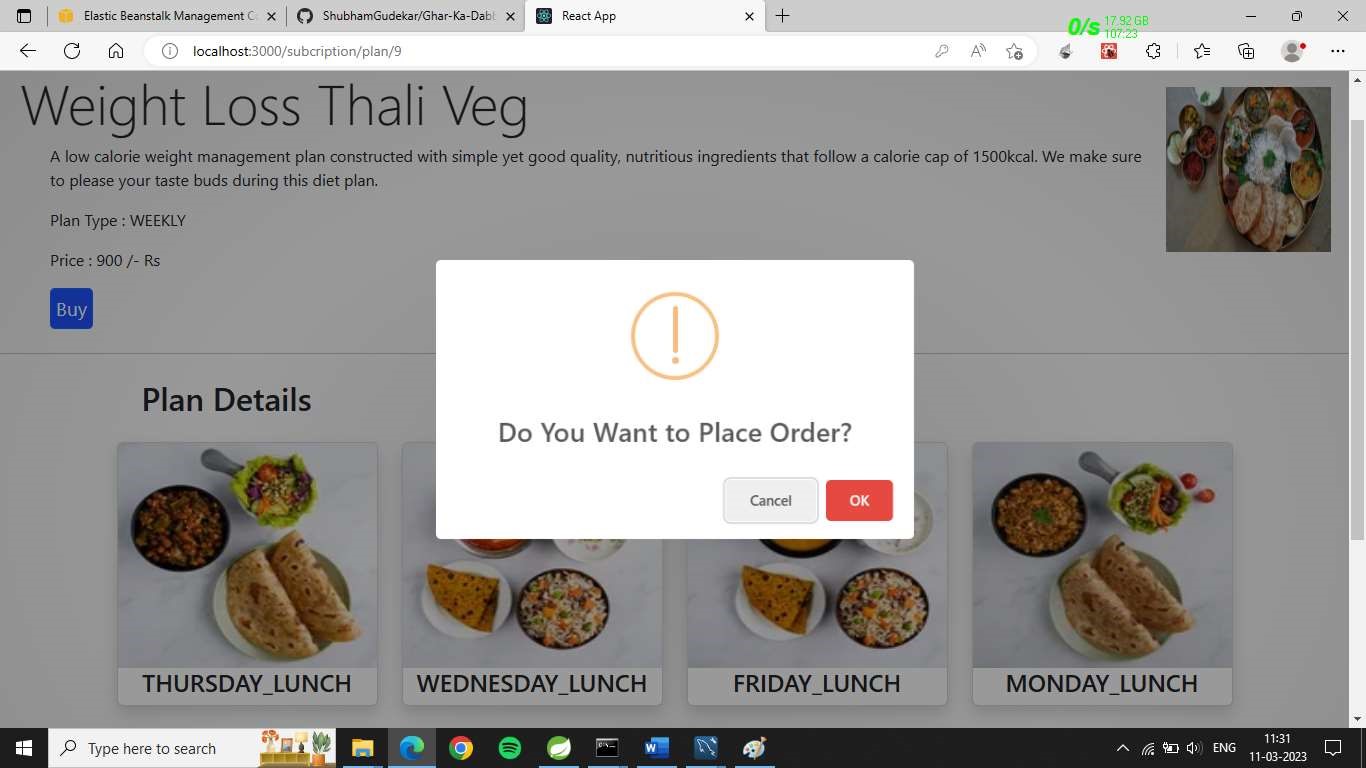


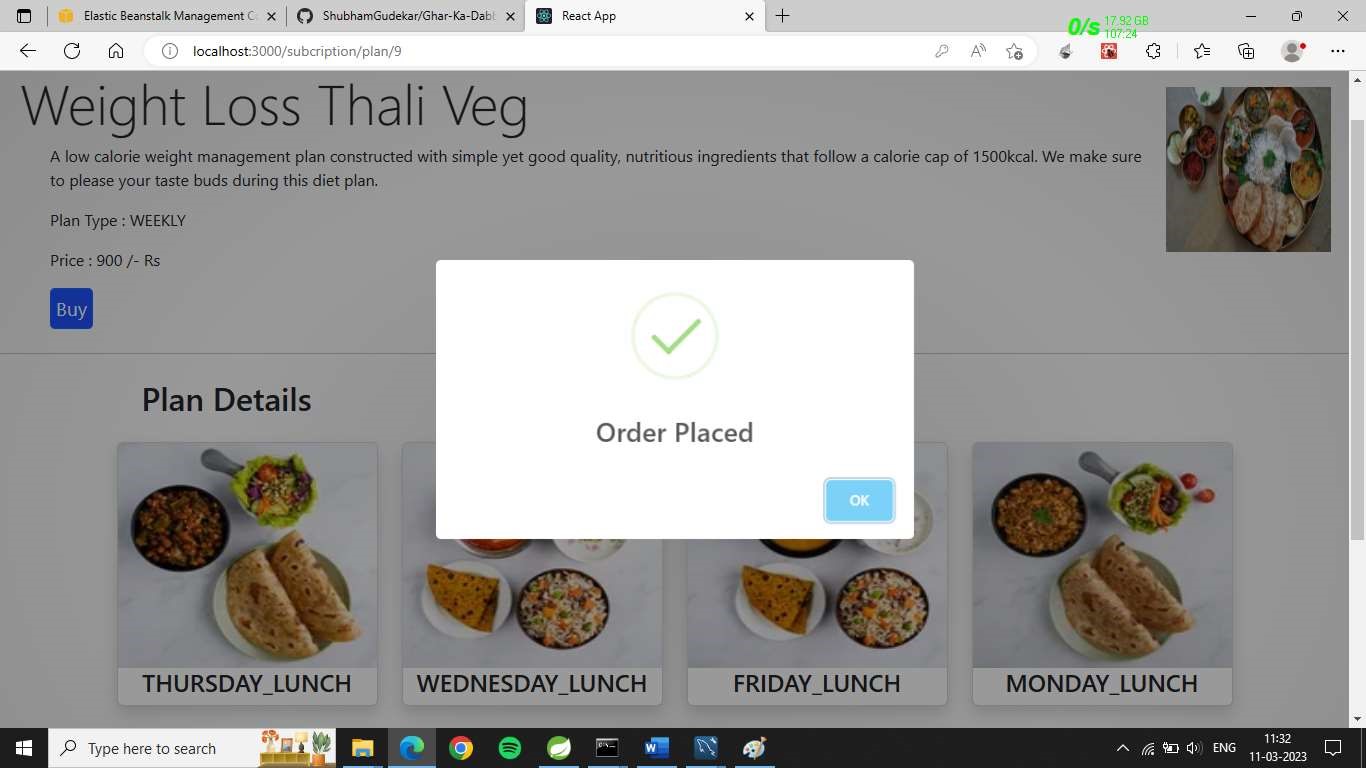


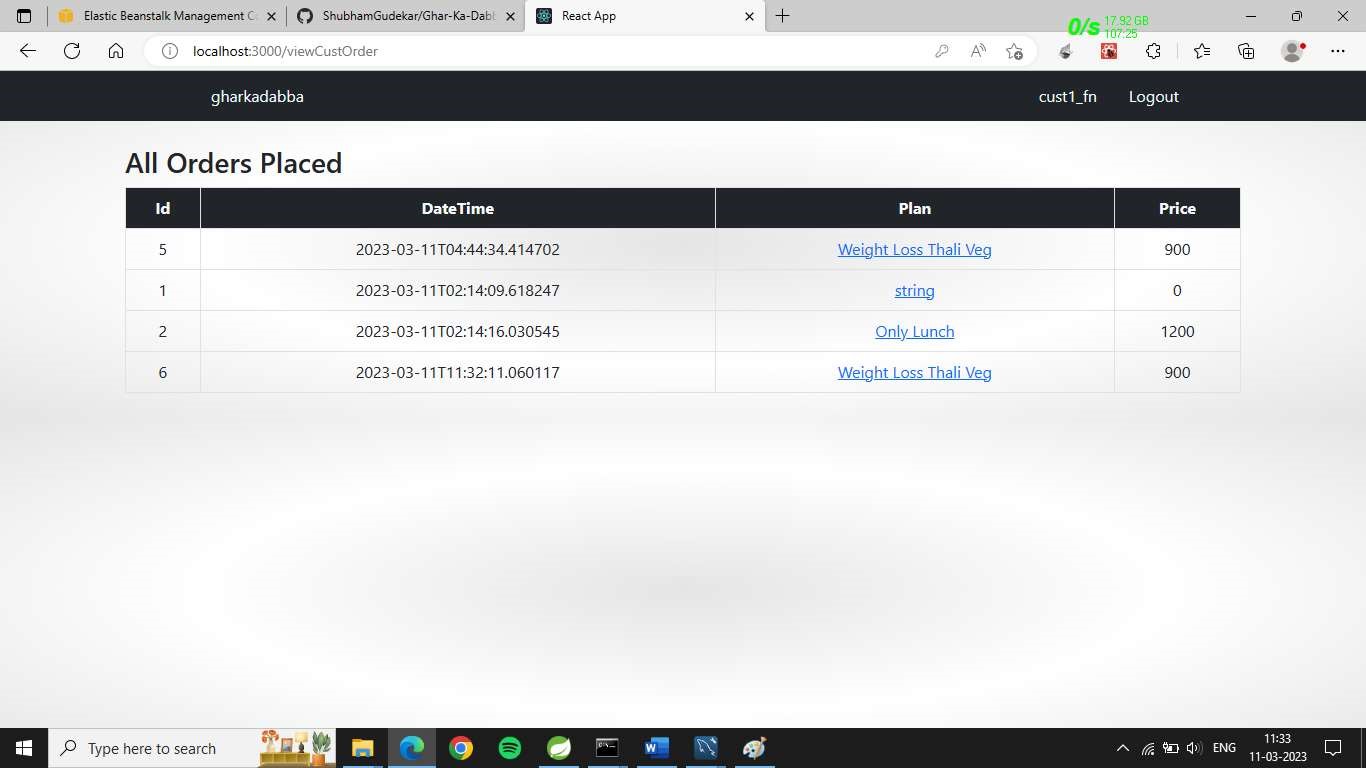


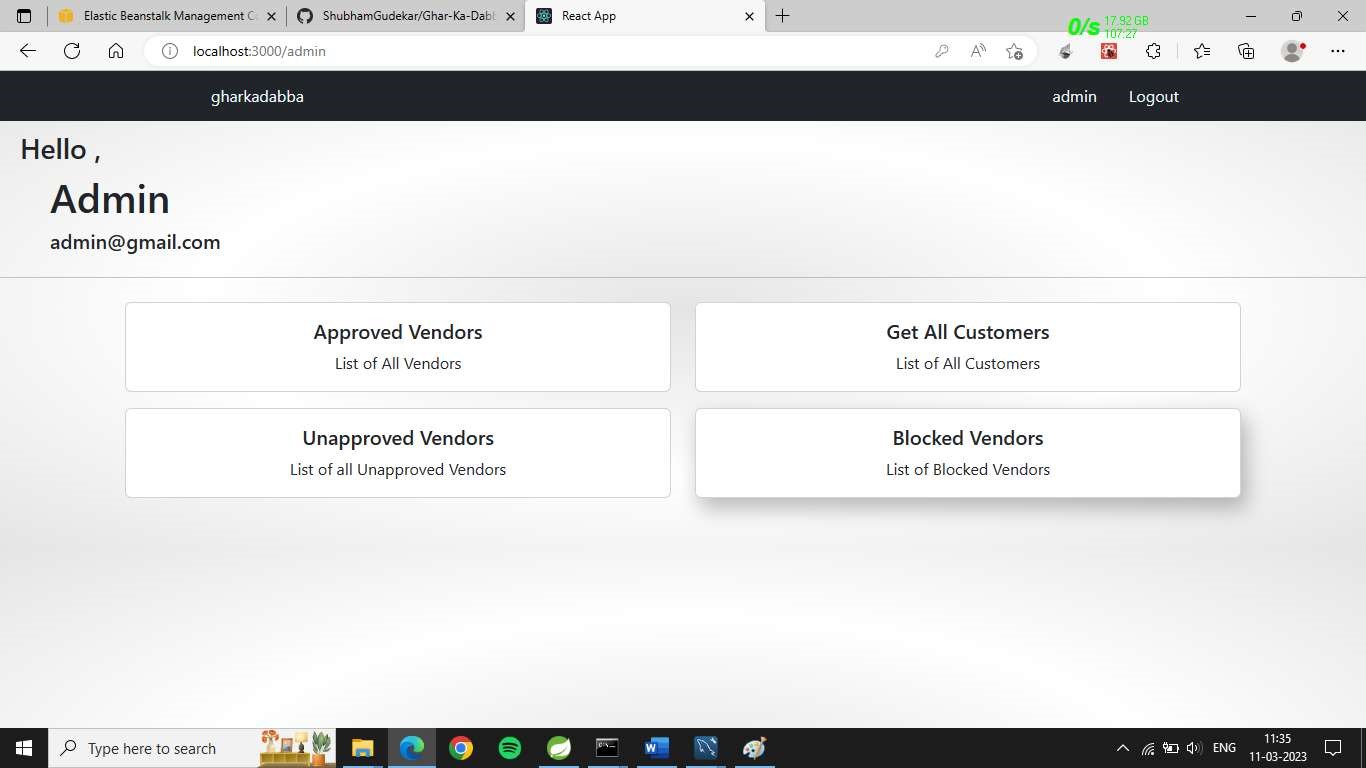


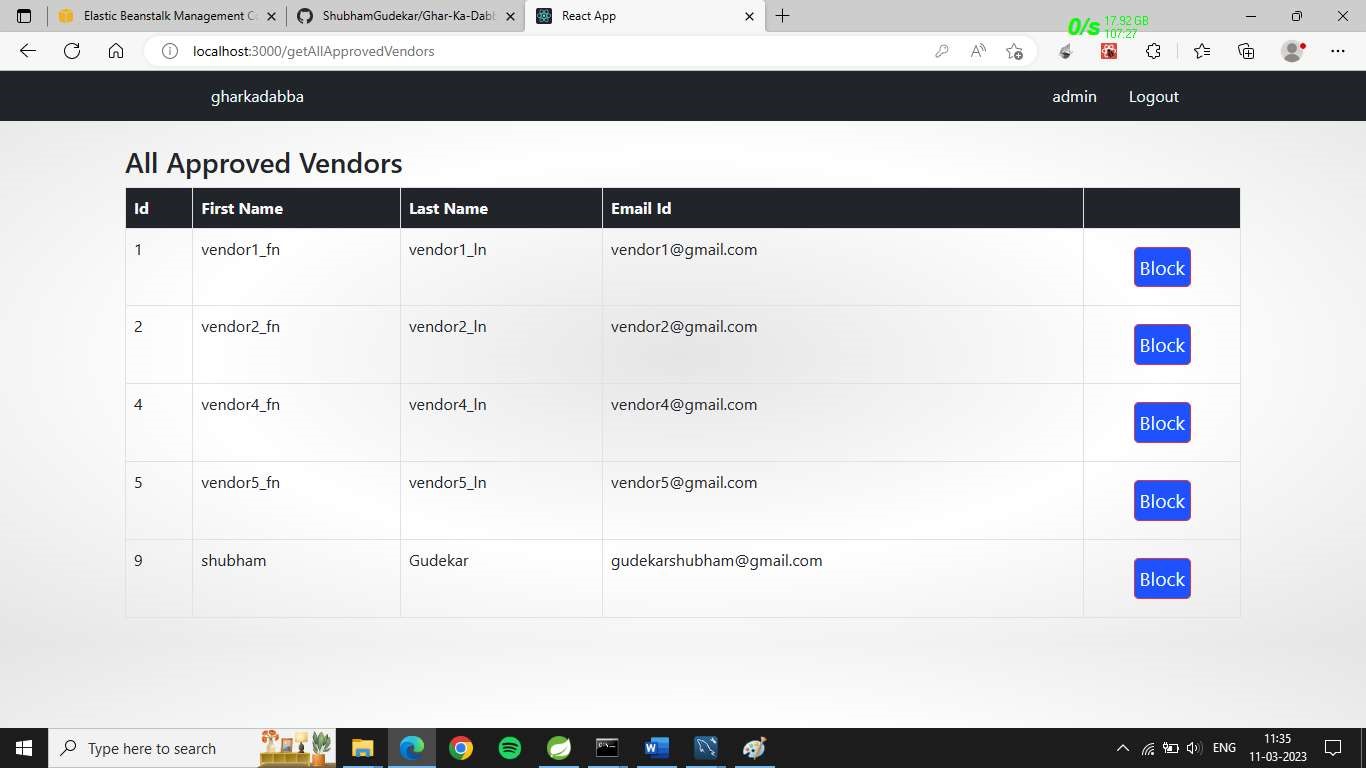


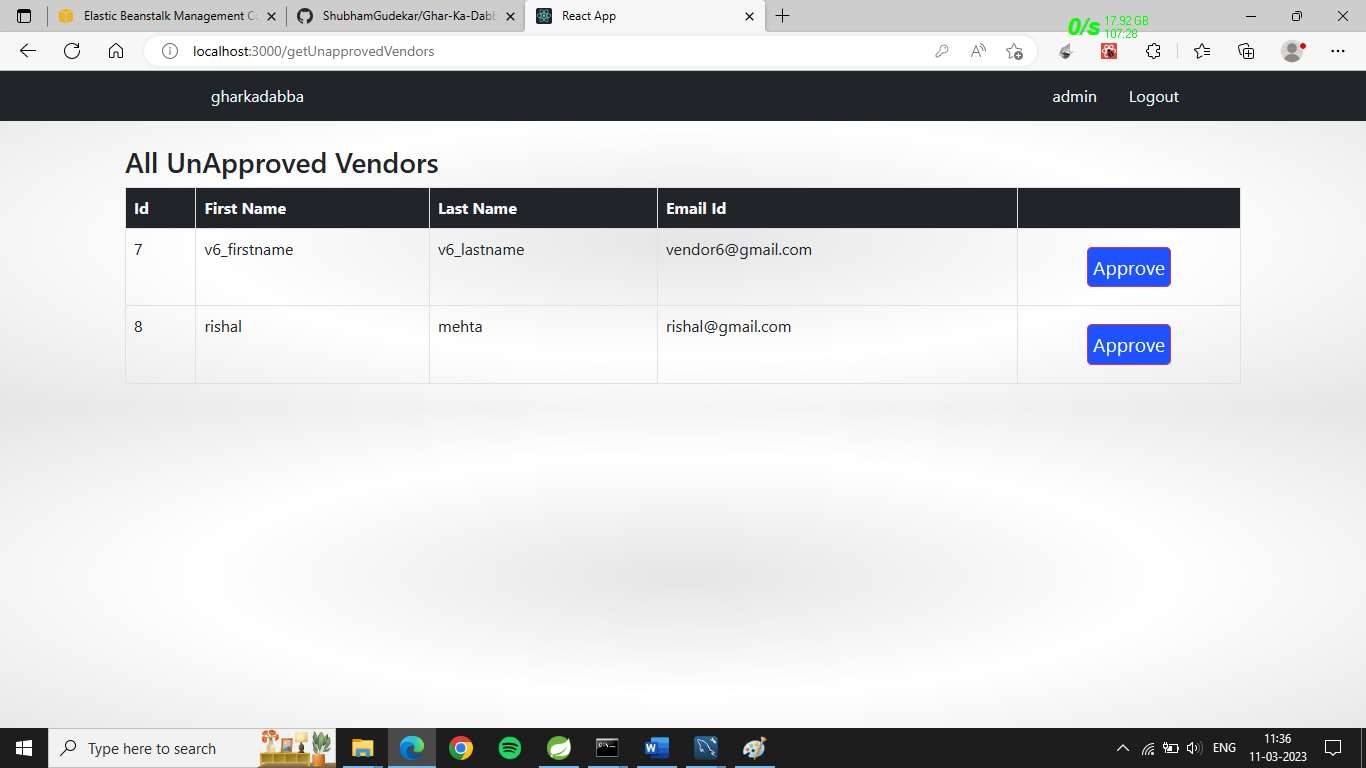


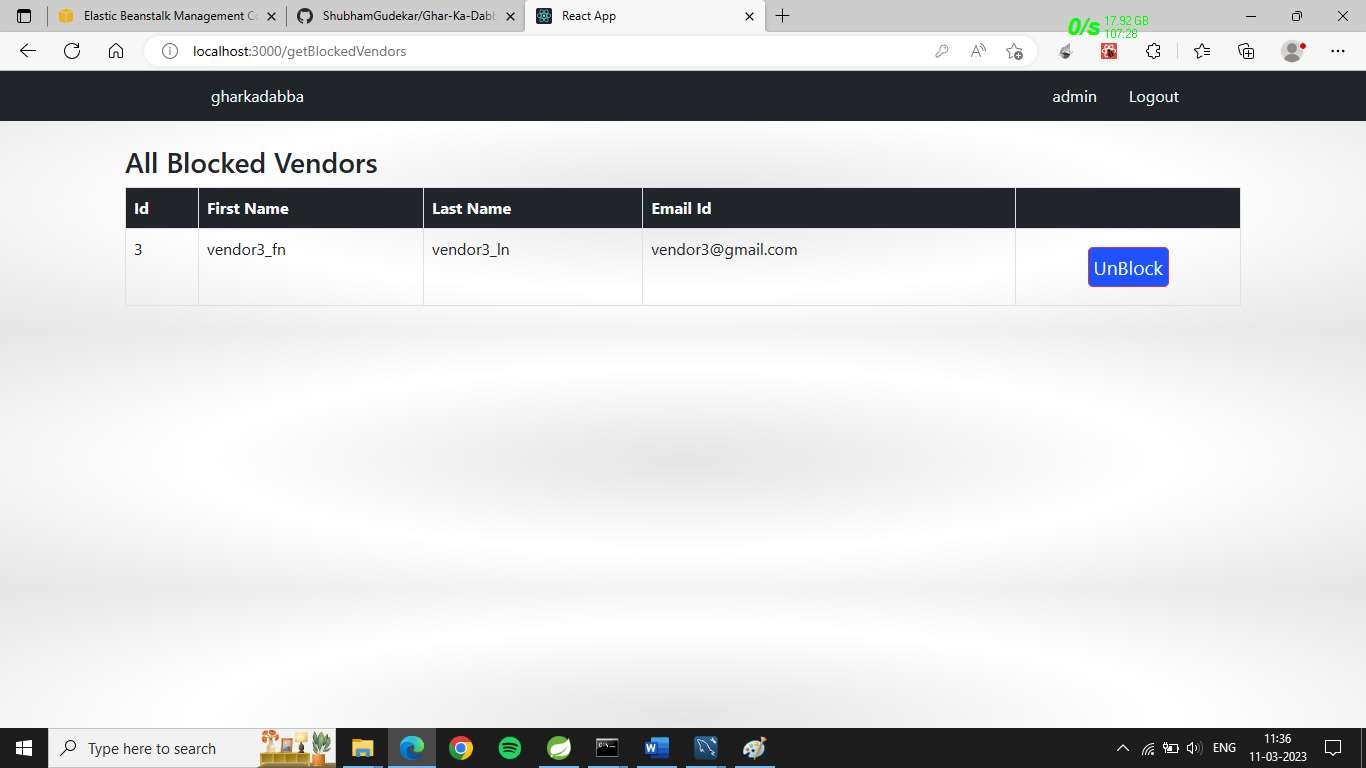


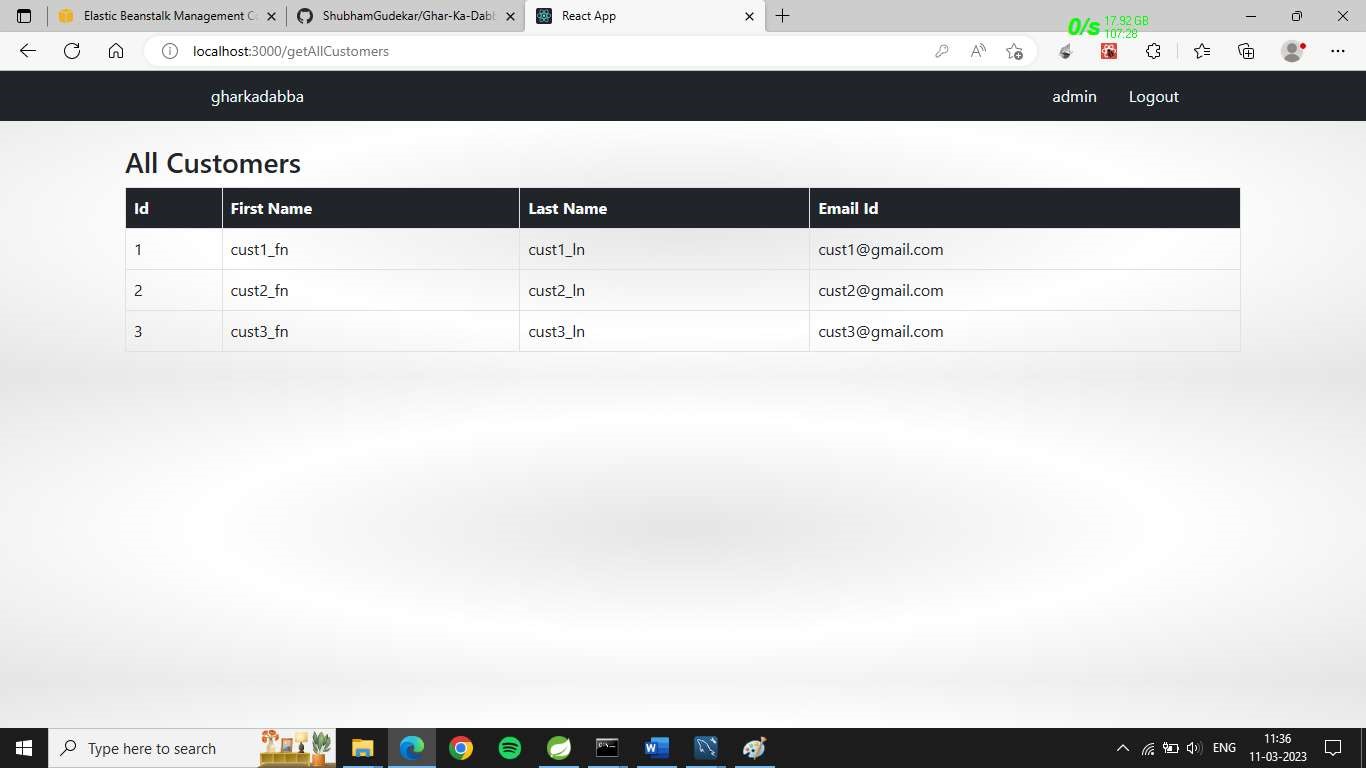












1. **TABLE STRUCTURE**
   1. **Customers**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | bigint | NO | PRI | NULL | auto\_increment |
| email | varchar(50) | YES | UNI | NULL |  |
| first\_name | varchar(50) | YES |  | NULL |  |
| last\_name | varchar(50) | YES |  | NULL |  |
| mobile | varchar(10) | YES | UNI | NULL |  |
| register\_date | Datetime(6) | YES |  | NULL |  |
| is\_blocked | Bit(1) | NO |  | NULL |  |
| password | varchar(255) | YES |  | NULL |  |

* 1. **Vendors**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | bigint | NO | PRI | NULL | auto\_increment |
| email | varchar(50) | YES | UNI | NULL |  |
| first\_name | varchar(50) | YES |  | NULL |  |
| last\_name | varchar(50) | YES |  | NULL |  |
| mobile | varchar(10) | YES | UNI | NULL |  |
| register\_date | Datetime(6) | YES |  | NULL |  |
| is\_verified | bit(1) | NO |  | NULL |  |
| is\_blocked | bit(1) | NO |  | NULL |  |
| is\_verified | bit(1) | NO |  | NULL |  |

* 1. **Tiffins**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | bigint | NO | PRI | NULL | auto\_increment |
| available\_from DATE | date | YES |  | NULL |  |
| available\_to DATE | date | YES |  | NULL |  |
| break\_lunch\_dinner | int | YES |  | NULL |  |
| description | varchar(255) | YES |  | NULL |  |
| food\_type | varchar(255) | YES |  | NULL |  |
| name | double | NO |  | NULL |  |
| price | double | NO |  | NULL |  |
| vendor\_id | bigint | NO |  | NULL |  |

* 1. **Orders**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | bigint | NO | PRI | NULL | auto\_increment |
| delivery\_note | Varchar(255) | NO |  | NULL |  |
| lunch\_quantity | int | NO |  | NULL |  |
| break\_fast\_quantity | int | NO |  | NULL |  |
| dinner\_quantity | int | NO |  | NULL |  |
| order\_end\_date | DATE | NO |  | NULL |  |
| order\_start\_date | DATE | NO |  | NULL |  |
| customer\_id | bigint | NO |  | NULL |  |
| tiffin\_id | bigint | NO |  | NULL |  |

* 1. **Address**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | NULL |  | Default | Extra |
| line1 | varchar(255) | YES |  | NULL |  |
| line2 | varchar(255) | YES |  | NULL |  |
| city | varchar(255) | YES |  | NULL |  |
| pincode | int | YES |  | NULL |  |
| state | varchar(255) | YES |  | NULL |  |
| customer\_id | bit(1) | NO |  | NULL |  |
| Vendor\_id | bit(1) | NO |  | NULL |  |
| address\_type | int | YES |  | NULL |  |

* 1. **Logins**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default | Extra |
| id | bigint | NO | PRI | NULL | auto\_increment |
| version | int | NO |  | NULL |  |
| email | varchar(255) | YES | UNI | NULL |  |
| password | varchar(350) | YES |  | NULL |  |
| user\_role | varchar(255) | YES |  | NULL |  |

1. **CONCLUSION**

In the current competitive world, many youths travel to different unknown locations for their basic education or jobs. The main problem they face is the food they get, and they crave for homemade food, but it is difficult to find it. On the other hand, some housewives wish to work and earn money to gain financial independence. It is difficult for these ladies to reach customers and market their products. So this online tiffin service will provide the common platform for those mess owner and students or youths to register mess and spend money according to service provided by owner. And also reduces the searching efforts of messes for youths and also giving financial help to vendor

* 1. Future Scope

Future extensions for step-up of project.

* Association with Google maps
* Payment mode
* Discount /offer management
* Billing Estimated time of implementation.
* 4 weeks

Benefits of future extension.

* Growth for registered messes through advertisements
* Attractive UI
* Richer user experience
* Robustness in application

1. **REFRENCES**

* <https://reactjs.org/docs/getting-started.html>
* <https://react-bootstrap.github.io/components/carousel/>
* https://www.geeksforgeeks.org/reactjs-tutorials/
* <https://javaee.github.io/javaee-spec/javadocs/>
* https://reactjs.org/docs/getting-started.html