

PROJECT REPORT ON

Tasty Eats

Submitted by

ASWATHI k

Under the guidance of

Sreerag



FUTURA LABS

ENABLE • TRANSFORM • ACCELERATE

Futura Labs Kochi, Kerala, India

PH: 9946325888,

<https://thefuturalabs.com>

CONTENTS

SL.NO.	CONTENTS	PAGE NO.
01	INTRODUCTION	03
02	CONCEPTUAL MODEL	05
03	DATABASE DESIGN	09
04	BIBLIOGRAPHY	10
05	APPENDIX	11
06	ANNEXURE	12

INTRODUCTION

ABSTRACT

This ecommerce website aims to make food delivery easy and free. One of its unique features is that it delivers the product exactly when the needy need it, no matter what time it is.

This website has two main modules, user and admin. It includes features that allow users to view and update their profile, add items they like to their cart, and view individual items.

The frontend of this web application is built using versatile and dynamic ReactJs library, while the backend will leverage the robust capabilities of the NodeJs runtime along with the Express.js framework. MongoDB, a NoSQL database, will be employed to store and manage data efficiently, providing scalability and flexibility for the e-commerce platform.

Through the implementation of a MERN stack, the project seeks to deliver a responsive and feature-rich e-commerce solution. Users will experience a user-friendly interface for seamless navigation, product exploration, and secure transaction processing. Administrators, on the other hand, will have access to tools for effective product management, order tracking, and user account management.

In summary, Mern based this project is user friendly so there will be less difficulties for the users. This efficient platform will only give good experience to the users.

MODULES

The complete project is divided into two modules. And the modularization is based on the users and products. The different modules based on the system are:

USER

ADMIN

ADMIN

Login
Dashboard
Admin Profile View
User Details
Add Items
Manage Items
View Order Details

User

Register
Login
User Profile
Add to Cart
View Items
Order

CONCEPTUAL MODELS

REQUIREMENT MODELING DATA FLOW DIAGRAM

Data Flow Diagram (DFD) is used to define the flow of the system and its resources such as information. Data flow diagrams are the way of expressing system requirements in a graphical manner. DFD represents one of the most ingenious tools used for structured analysis. A DFD is also known as a bubble chart. It has the purpose of clarifying system requirements identifying major transformations that will become programs in system design. In the normal convention, logical DFD can be completed using only 4 notations.



Represents source/destination data.



Represents data flow.

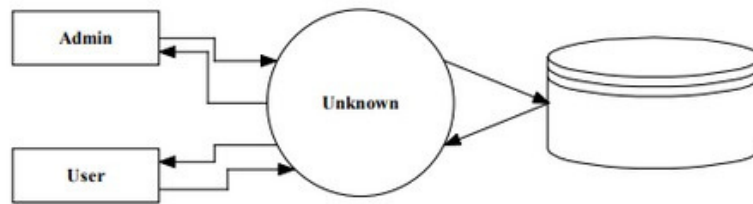


Represents a process that transforms incoming data into outgoing flow.

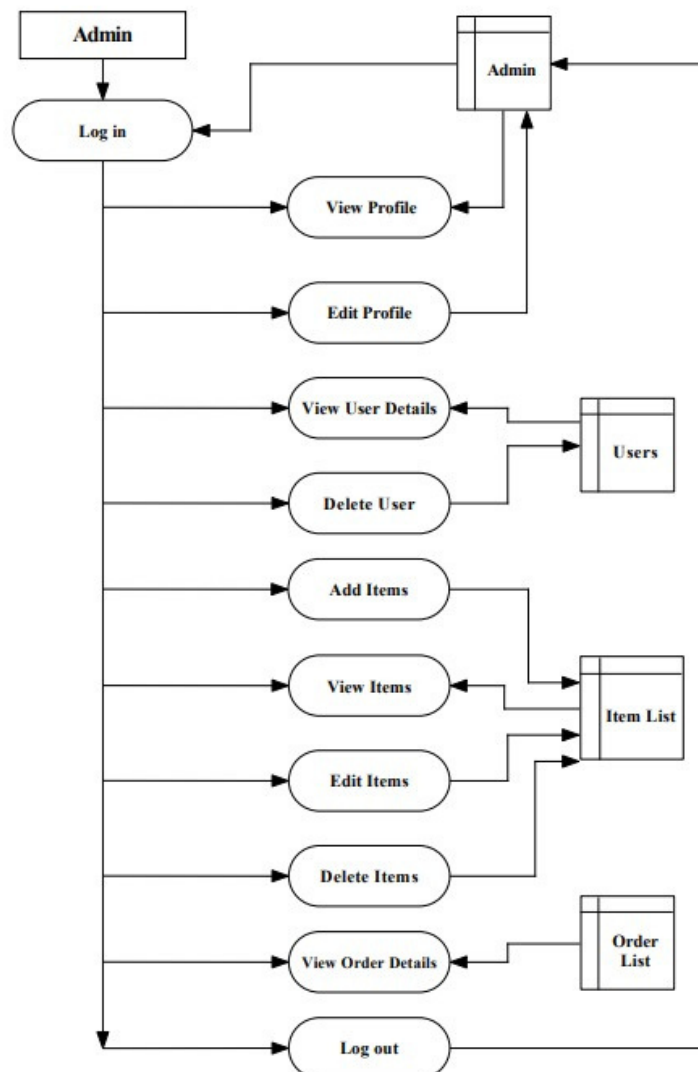


Represents data storage/internal storage.

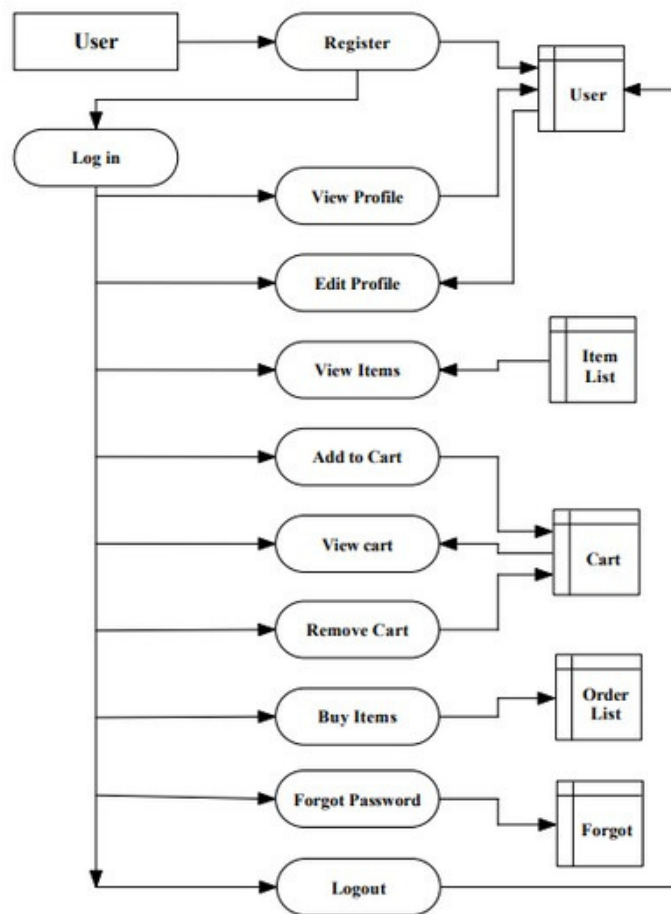
LEVEL 0 DFD



LEVEL 1 DFD - ADMIN



LEVEL 1 DFD - USER



FUNCTIONS

The system after careful analysis has been identified to be presented with the following modules:

Admin

Login – Admin can login to the application by entering username and password. **Dashboard**–Admin can view the total stat of the website.

Admin Profile– Admin can upload their profile details.

View User Details – Admin can view registered user details.

Add Items – Admin can add items.

Manage Items – Admin can update and delete items.

View Order Details – Admin can view the details of the item ordered by the user and the details of the person who made the order.

User

Login - User can login to the application by entering username and password. **User Profile** - User can upload their profile details.

Add To Cart - User can add items to cart.

View Items - User can view individual items.

Order - User can view all their progress and updates.

Remove from cart - User can remove items from cart.

DATABASE DESIGN

A database is a collection of records. The main objective of database design is to provide effective auxiliary storage without any applications and to contribute to the overall efficiency of the computer program. components of the whole system. The organization of data in the database aims to achieve the following objectives.

Controlled redundancy

Ease of learning in use

Data independence

More information in low cost

Accuracy and integrity

Recovery from failures

Privacy and security

Performance

The design should be done in a way the information stored in the database can be retrieved quickly whenever necessary. The general theme behind a database is to handle information as an interfered whole. A database is a collection of interrelated data stored with minimum redundancy to serve users quickly and efficiently. Database design runs parallel without application design. As we collect information about what is to be done, we will obviously collect information about data needed to enter, stored messages and printed reports. The designing of the database is done with utmost care and security during the designing phase of the system. Special care was taken to develop a minimum number of databases for the maximum efficiency of the system.

Atlas ADMATH's Access Manager Billing All Clusters Get Help Asaphth

Project 0 Data Services App Services Charts

Overview

DEPLOYMENT

Database

Data Lake

SERVICES

Device Sync

Triggers

Data API

Data Federation

Atlas Search

Stream Processing

Migration

SECURITY

Backup

Database Access

Network Access

Advanced

Docs

+ Create Database

Search Namespaces

test

adminusers

books

cardetails

conformoders

forgotpasses

itemdetails

movies

nodemallers

orderdetails

orderusers

requesters

users

users985

test.adminusers

STORAGE SIZE: 3MB LOGICAL DATA SIZE: 3MB TOTAL DOCUMENTS: 3 INDEXES TOTAL SIZE: 3MB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

File

Type a query: { field: "value" }

RESET Apply Options

INSERT DOCUMENT

```

{
  "_id": "ObjectID('55a6c27b6b9b5286c6726')",
  "Name": "asaphth",
  "Email": "asaphth@gmail.com",
  "Password": "Q2Pw8Xk3Lm9g12aw32bdcg7K9YKVS2RZV3-",
  "createdAt": "2024-05-18T06:51:37.771-00:00",
  "updatedAt": "2024-05-18T06:51:37.771-00:00",
  ...x: 0
}

{
  "_id": "ObjectID('55a6c34d8b5e2f9695235')",
  "Name": "x00tra 8",
  "Email": "x00tr@gmail.com",
  "Password": "Q2Pw8Xk3Lm9g12aw32bdcg7K9YKVS2RZV3-",
  "createdAt": "2024-05-18T05:27:09.226-00:00",
  "updatedAt": "2024-05-18T05:27:09.226-00:00",
  ...x: 0
}

{
  "_id": "ObjectID('55a6c4a28b5a438c2bed1')",
  "Name": "Pranav 1P",
  "Email": "pranavgp@gmail.com",
  "Password": "Q2Pw8Xk3Lm9g12aw32bdcg7K9YKVS2RZV3-",
  "Images": "202403090551_2M_8578.JPG",
  "createdAt": "2024-05-18T05:12:32.825-00:00",
  "updatedAt": "2024-05-27T04:11:22.254-00:00",
  ...x: 0
}

```

System Status: All Good

©2024 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Atlas ADMATH's Access Manager Billing All Clusters Get Help Asaphth

Project 0 Data Services App Services Charts

Overview

DEPLOYMENT

Database

Data Lake

SERVICES

Device Sync

Triggers

Data API

Data Federation

Atlas Search

Stream Processing

Migration

SECURITY

Backup

Database Access

Network Access

Advanced

Docs

+ Create Database

Search Namespaces

test

adminusers

books

cardetails

conformoders

forgotpasses

itemdetails

movies

nodemallers

orderdetails

orderusers

requesters

users

users985

test.conformoders

STORAGE SIZE: 3MB LOGICAL DATA SIZE: 2.2KB TOTAL DOCUMENTS: 4 INDEXES TOTAL SIZE: 3MB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

File

Type a query: { field: "value" }

RESET Apply Options

INSERT DOCUMENT

```

{
  "_id": "ObjectID('55a6c67795b67437b3c565a')",
  "Details": {
    "userData": {
      "_id": "55a6c3c54d2bda4bda5",
      "HouseName": "Araksh House",
      "HouseNo": "180",
      "Pincode": "673732",
      "Landmark": "Araksh",
      "City": "Calicut",
      "phone": "907945223",
      "ProfileId": "55a612a63a787f6488863a8",
      ...x: 0
    },
    "timestamp": {
      "0": {
        "0": {
          "order": "55a612a63a787f6488863a8",
          ...x: 0
        }
      }
    }
  }
}

{
  "_id": "ObjectID('55a6c6b85b67437b3c5667')",
  "Details": {
    "userData": {
      "timestamp": {
        "0": {
          "order": "55a612a63a787f6488863a8",
          ...x: 0
        }
      }
    }
  }
}

```

System Status: All Good

©2024 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

Atlas ADMATH's Access Manager Billing All Clusters Get Help Asaphth

Project 0 Data Services App Services Charts

Overview

DEPLOYMENT

Database

Data Lake

SERVICES

Device Sync

Triggers

Data API

Data Federation

Atlas Search

Stream Processing

Migration

SECURITY

Backup

Database Access

Network Access

Advanced

Docs

+ Create Database

Search Namespaces

test

adminusers

books

cardetails

conformoders

forgotpasses

itemdetails

movies

nodemallers

orderdetails

orderusers

requesters

users

users985

test.adminusers

STORAGE SIZE: 3MB LOGICAL DATA SIZE: 3MB TOTAL DOCUMENTS: 3 INDEXES TOTAL SIZE: 3MB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

File

Type a query: { field: "value" }

RESET Apply Options

INSERT DOCUMENT

```

{
  "_id": "ObjectID('55a6c27b6b9b5286c6726')",
  "Name": "asaphth",
  "Email": "asaphth@gmail.com",
  "Password": "Q2Pw8Xk3Lm9g12aw32bdcg7K9YKVS2RZV3-",
  "createdAt": "2024-05-18T06:51:37.771-00:00",
  "updatedAt": "2024-05-18T06:51:37.771-00:00",
  ...x: 0
}

{
  "_id": "ObjectID('55a6c34d8b5e2f9695235')",
  "Name": "x00tra 8",
  "Email": "x00tr@gmail.com",
  "Password": "Q2Pw8Xk3Lm9g12aw32bdcg7K9YKVS2RZV3-",
  "createdAt": "2024-05-18T05:27:09.226-00:00",
  "updatedAt": "2024-05-18T05:27:09.226-00:00",
  ...x: 0
}

{
  "_id": "ObjectID('55a6c4a28b5a438c2bed1')",
  "Name": "Pranav 1P",
  "Email": "pranavgp@gmail.com",
  "Password": "Q2Pw8Xk3Lm9g12aw32bdcg7K9YKVS2RZV3-",
  "Images": "202403090551_2M_8578.JPG",
  "createdAt": "2024-05-18T05:12:32.825-00:00",
  "updatedAt": "2024-05-27T04:11:22.254-00:00",
  ...x: 0
}

```

System Status: All Good

©2024 MongoDB, Inc. Status Terms Privacy Atlas Blog Contact Sales

BIBLIOGRAPHY

WEBSITES

1.MongoDB Documentation:

Author: MongoDB

URL: <https://docs.mongodb.com/>

2.Express.js Documentation:

Author: Express.js

URL: <https://expressjs.com/>

3.React Documentation:

Author: React.js

URL: <https://reactjs.org/docs/getting-started.html>

4.Node.js Documentation:

Author: Node.js

URL: <https://nodejs.org/en/docs/>

5.Full Stack Open (Course by University of Helsinki):

Authors: Matti Luukkainen, et al.

URL: <https://fullstackopen.com/en/>

6.MERN Stack Front To Back: Full Stack React, Redux & Node.js:

Author: Brad Traversy

Book URL: <https://www.amazon.com/MERN-Stack-Front-Back-React-Node/dp/1789954117>

7.Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node:

Authors: Vasan Subramanian

Book URL: <https://www.apress.com/gp/book/9781484243909>

8.Learning React: Functional Web Development with React and Redux:

Authors: Alex Banks, Eve Porcello

Book URL: <https://www.oreilly.com/library/view/learning-react-2nd/9781492051718/>

9.Node.js Design Patterns:

Author: Mario Casciaro

Book URL: <https://www.packtpub.com/product/node-js-design-patterns-third-edition/9781803237918>

10.MongoDB in Action:

Authors: Kyle Banker, Peter Bakkum, Shaun Verch, Douglas Garrett

Book URL: <https://www.manning.com/books/mongodb-in-action-second-edition>

CODE

[illegible][illegible]

ANNEXURE

FOR USERS

Login Page

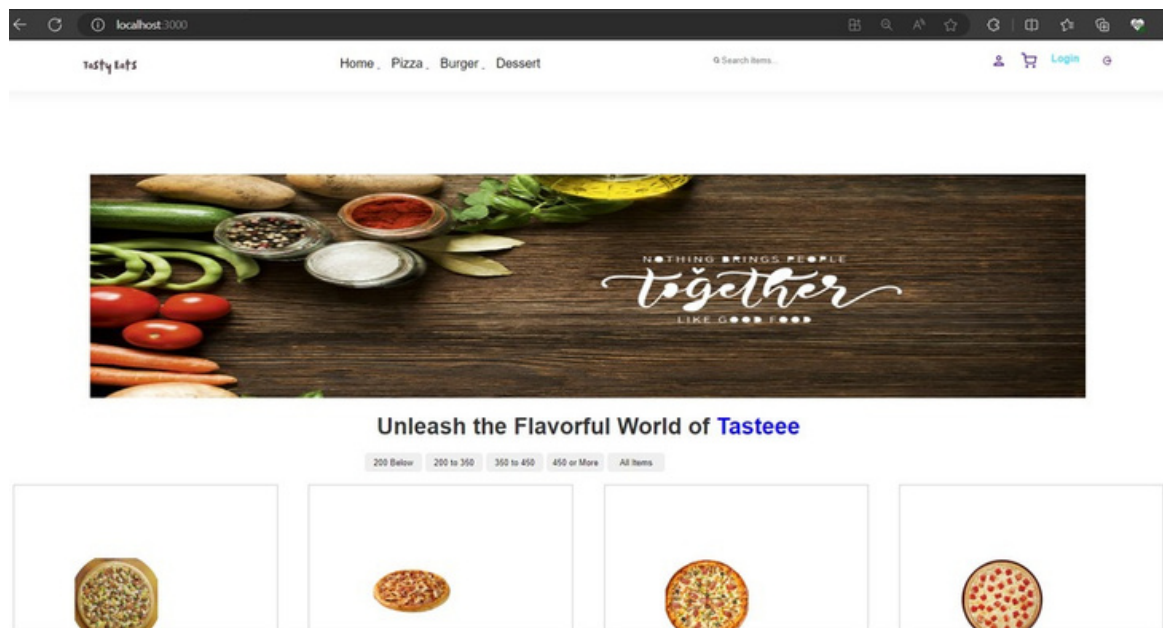
LOGIN

Login

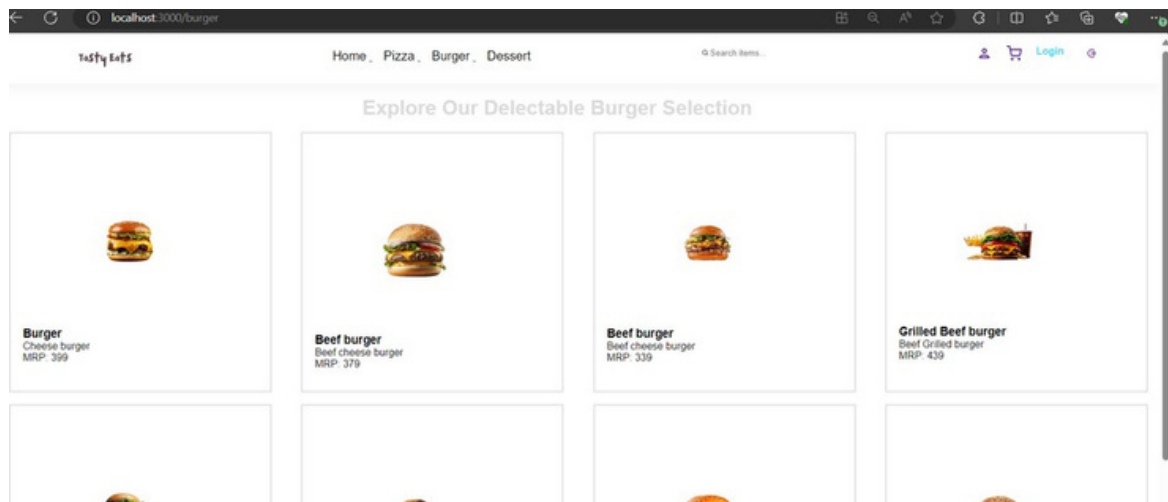
Forgot Password? [Click](#)

Don't have an account? [Register](#)

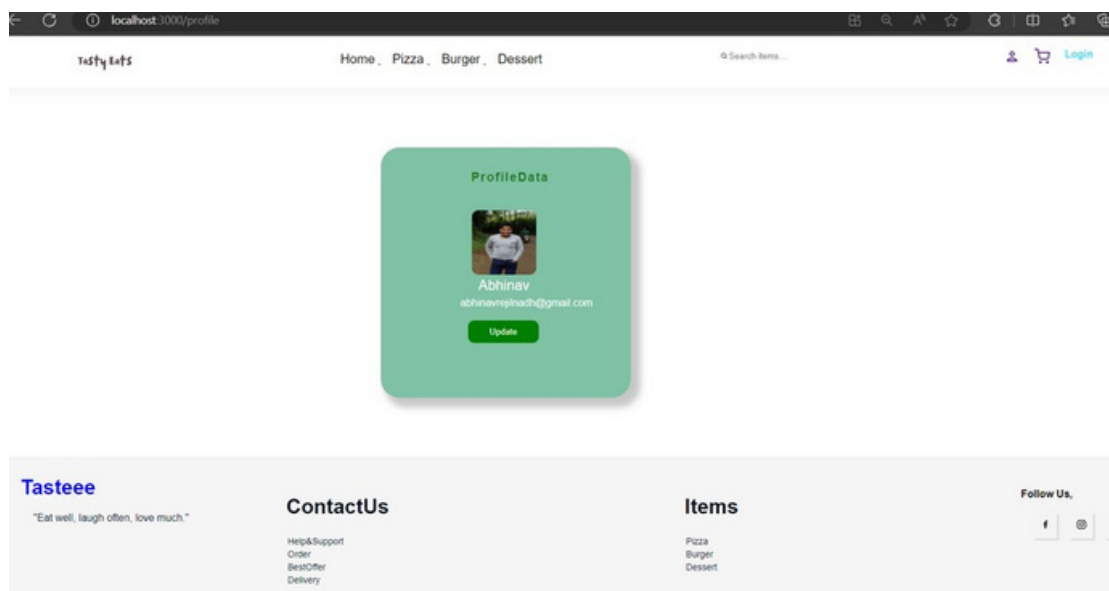
HomePage



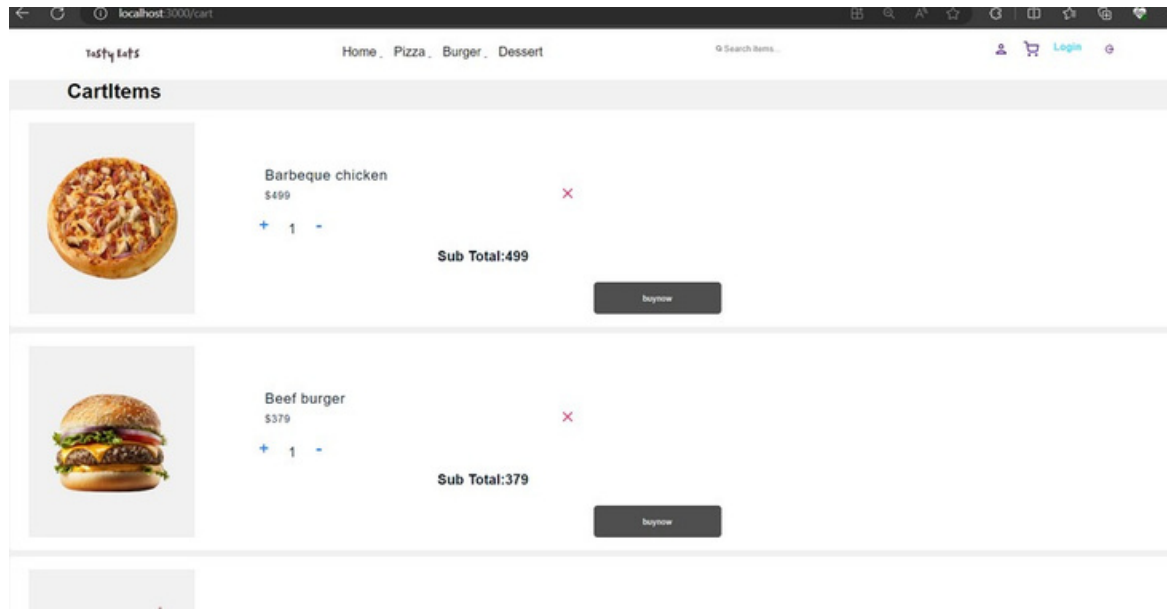
Burger Page



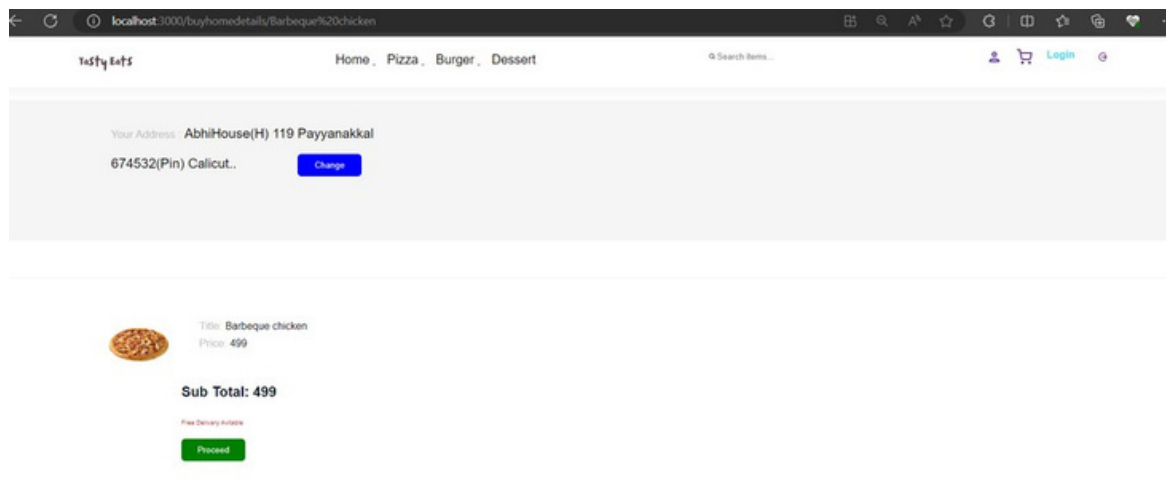
User Profile



Cart Page



Buy Now Page



ANNEXURE

FOR ADMIN

Login Page

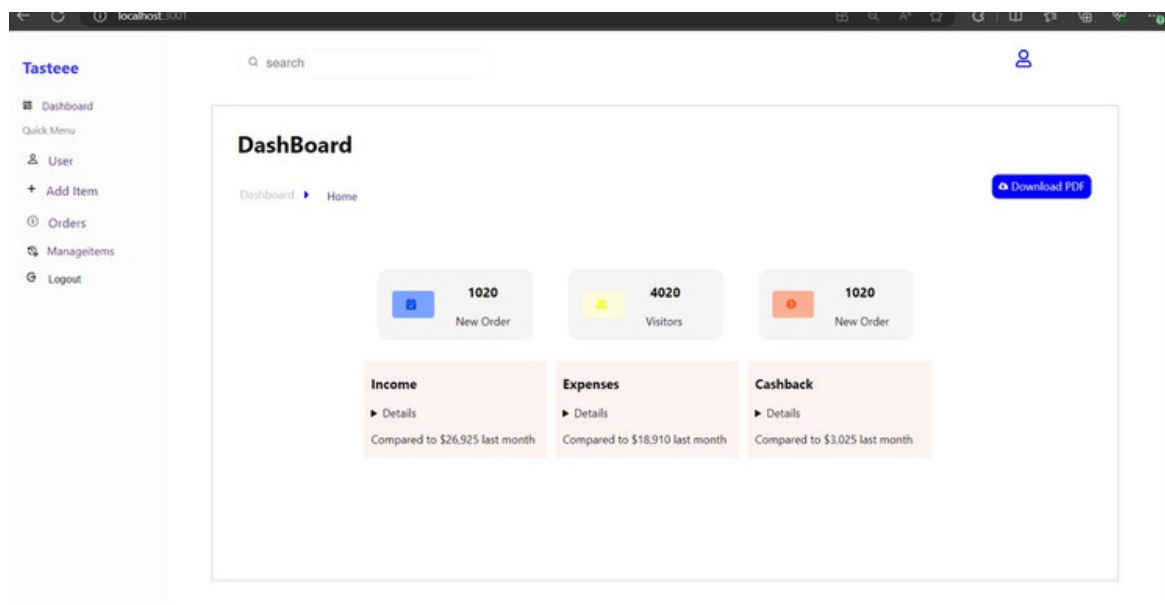
LOGIN

Login

Forgot Password? [Click](#)

Don't have an account? [Register](#)

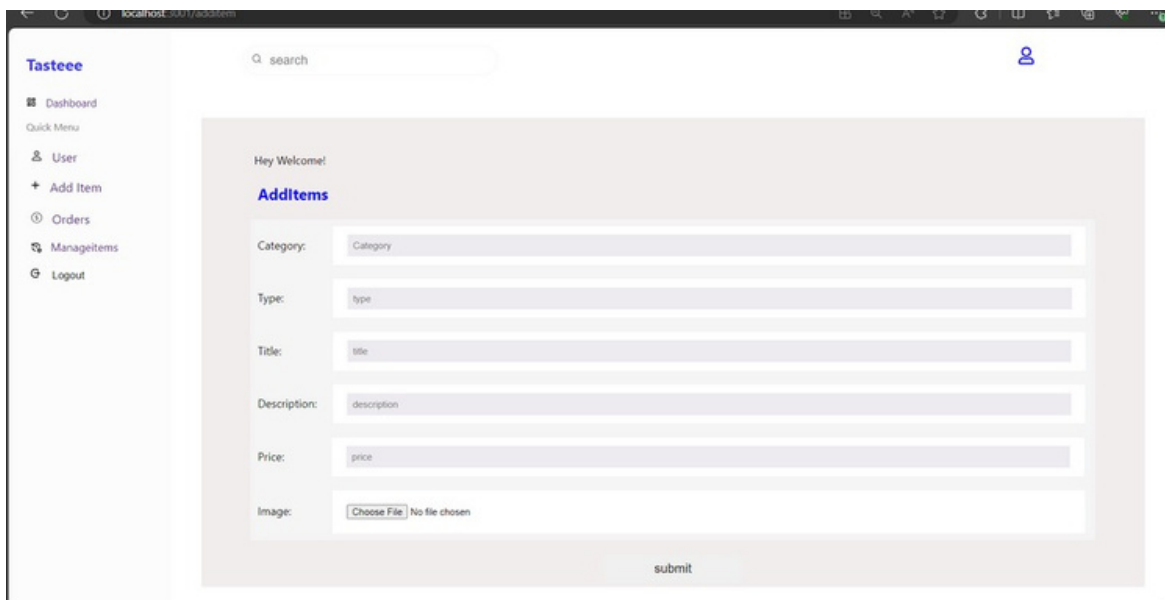
Dashboard



Admin Profile



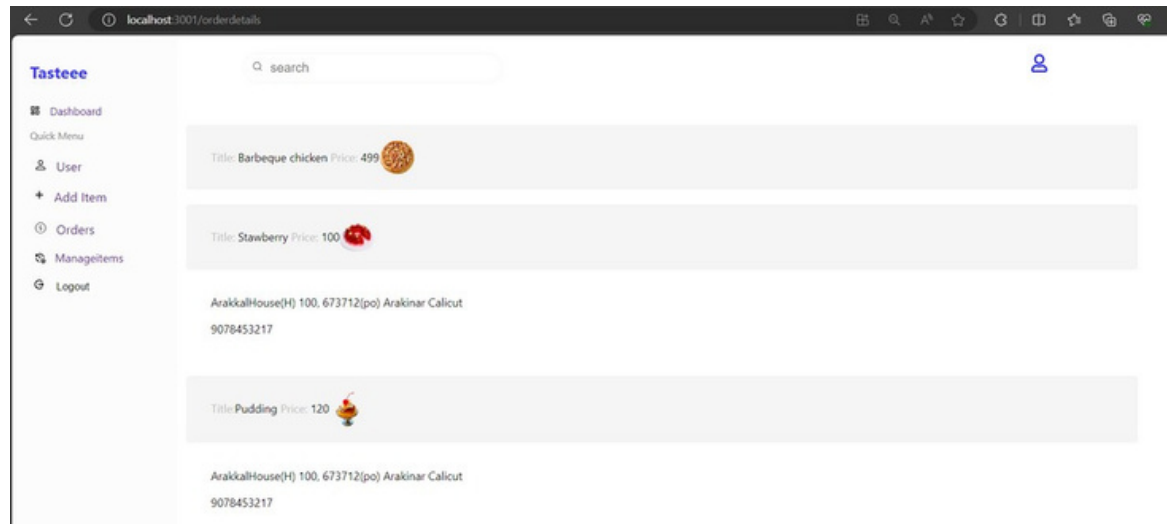
Add Items



Manage Items

ProductDetails						
Image	Title	Description	Price	Update	Delete	
	Tandoori pizza	Tasty Tandoori Pizza	420	<button>update</button>	<button>delete</button>	
	Barbeque chicken	Barbeque chicken with spicy	499	<button>update</button>	<button>delete</button>	
	Mushroom Pizzaa	Cheese mushroom pizza	599	<button>update</button>	<button>delete</button>	
	Tomoto Pizza	Cheese Tomoto pizza	529	<button>update</button>	<button>delete</button>	
	Chicken Pizza	Spicy Chicken pizza	459	<button>update</button>	<button>delete</button>	
	Corn pizza	Cheese Corn Pizza	499	<button>update</button>	<button>delete</button>	
	Cheese pizza	Tomoto Cheese Pizza	299	<button>update</button>	<button>delete</button>	
	veg pizza	vegetable Cheese Pizza	199	<button>update</button>	<button>delete</button>	
	Burger	Cheese burger	399	<button>update</button>	<button>delete</button>	

View Order



User Details

