

**A  
Project Report  
on  
OMG optical  
Developed By  
*Harshad Prajapati [2205112110078]*  
*Snehal Roy [2205112110157]*  
*Rutik Khairnar [2205112070033]*  
developed at  
Company Name - BharatHelp.Info  
as  
Partial Fulfillment of IV<sup>th</sup> Semester of  
Master of Computer Applications  
for A.Y. 2023 - 2024**

**Under The Guidance of  
Prof. Faruk Abdulla**

**Submitted To  
Department of MCA  
Faculty of IT & Computer Science**





## **CERTIFICATE**

This is to certify that Mr. Harshad Prajapati, Enrollment No. **2205112110078**, Ms. Snehal Roy, Enrollment No. **2205112110157**, Mr. Rutik Khairnar, Enrollment No. **2205112070033** student of Master of Computer Applications has satisfactorily completed the Major Project on "**OMG optical**" at **BharatHelp.Info** as fulfillment of MCA Semester IV.

Seat No. \_\_\_\_\_

Date of Submission: \_\_\_\_\_

---

Internal Guide

---

Project Coordinator

---

Director - MCA

**Department of MCA  
Faculty of IT & Computer Science  
PARUL University, Vadodara**

# INDEX

No.	Description	Page No.
1.	<b>About Department of MCA</b>	04
2.	<b>Company Profile</b>	05
3.	<b>Project Profile</b> <ul style="list-style-type: none"> <li>3.1 Project Definition</li> <li>3.2 Project Description</li> <li>3.3 Existing System / Work Environment</li> <li>3.4 Problem Statements</li> <li>3.5 Need for New System</li> <li>3.6 Proposed System &amp; Features</li> <li>3.7 Scope</li> <li>3.8 Outcomes</li> <li>3.9 Tools &amp; Technology used</li> <li>3.10 Project Plan</li> </ul>	07
4.	<b>Requirement Analysis</b> <ul style="list-style-type: none"> <li>4.1 Feasibility Study</li> <li>4.2 Users of the System</li> <li>4.3 Modules</li> <li>4.4 Process Model</li> <li>4.5 Hardware &amp; Software Requirements</li> <li>4.6 Use Cases</li> <li>4.7 Use Case Diagram</li> </ul>	12
5.	<b>Design</b> <ul style="list-style-type: none"> <li>5.1 Use Case Scenarios</li> <li>5.2 Diagrams <ul style="list-style-type: none"> <li>5.2.1 UML / DFD</li> <li>5.2.2 Entity Relationship Diagram</li> </ul> </li> <li>5.3 Data Dictionary</li> </ul>	19
6.	<b>Implementation</b> <ul style="list-style-type: none"> <li>6.1 Form Layouts</li> <li>6.2 Report Layouts</li> <li>6.3 Coding Convention</li> </ul>	29
7.	<b>Testing</b> <ul style="list-style-type: none"> <li>7.1 Test Strategy</li> <li>7.2 Test Cases</li> </ul>	37
8.	<b>Future Enhancement</b>	38
9.	<b>Bibliography</b>	39

## About Department of MCA

### **PARUL University**

Parul University is a legitimate university established under Gujarat Private University Act 2009, after legislation passed by the Government of Gujarat on 26<sup>th</sup> March 2015 giving University status to Parul Group of Institutes functioning under the aegis of Parul Arogya Seva Mandal Trust.

### **Faculty of IT & Computer Science**

Faculty of IT and Computer Science, Parul University has materialized as one of the prime IT education providers at global level. Various departments under Faculty of IT and Computer Science strive in preparing IT-industry ready professionals by means of various skill development courses, vocational courses, co-curricular & extra-curricular activities, industry visits and expert lectures.

### **MCA Department**

The Department of Master of Computer Application at Parul University emphasizes on building professionals in the domain of computer applications by providing necessary environment by means of facilitating suitable blend of technical and non-technical learning experience. The department cultivates students in various curricular, co-curricular and extra-curricular activities in order to produce future system analysts, system designers, system programmers, application programmers, testing professionals, system managers, project managers, researchers and other leading positions in systems/IT department.

The department offers various subjects from diversified technical/non-technical areas such as – core IT domain, management, communication skills, mathematics & logic building and rich pool of elective subjects.

The department of MCA focuses on project-based learning, and hence students are motivated to work on tiny hands-on projects in practical oriented subjects to get better exposure. Moreover, throughout their MCA studies, students are required to work on around 3 mini/major projects in individual/team to get enough confidence on software-development and thereby become industry-ready.

## 2. Company Profile

Company Profile: Bharathelp.info

UDYAM Registration Details:

UDYAM Registration Number: UDYAM-BR-33-0004006

Name of Enterprise: Bharathelp.info

Type of Enterprise: Services

Major Activity: Information Technology and Computer Services

Date of UDYAM Registration: 26.06.2020

Date of Incorporation/Registration of Enterprise: 15/08/2023

National Industry Classification Code(s):

62012 - Web-page designing

62099 - Other information technology and computer service activities

63112 - Web hosting activities

63121 - Operation of websites that use a search engine to generate and maintain extensive databases of internet addresses and content in an easily searchable format

Description:

Bharathelp.info is a dynamic and innovative information technology services company based in Vadodara, Gujarat. Since our inception in June 2020, we have been dedicated to providing cutting-edge solutions in the field of computer programming, consultancy, and related activities.

Mission Statement:

Our mission is to empower businesses and individuals by leveraging technology to unleash their full potential. We strive to deliver high-quality, customized IT solutions that drive efficiency, productivity, and growth.

Services:

Web-page designing

Other information technology and computer service activities

Web hosting activities

Operation of websites utilizing search engines to maintain extensive databases of internet addresses and content

**Target Market:**

Bharathelp.info caters to a diverse clientele, including small businesses, startups, and enterprises across various industries. We specialize in providing tailored IT solutions to meet the unique needs and challenges of each client.

**Future Outlook:**

As we continue to grow and expand our offerings, Bharathelp.info remains committed to staying at the forefront of technological innovation. We aim to forge long-lasting partnerships with our clients, delivering value-driven solutions that drive sustainable success.

### 3. Project Profile

#### 3.1 Project Definition :

OMG Opticals & Gift Articles is an online shopping platform specializing in optical products and accessories like watches, wallets, and belts. Offering a collection of stylish eyewear and fashion items, our goal is to provide a seamless shopping experience for individuals looking to enhance their style. OMG Opticals & Gift Articles focuses on quality, user-friendly navigation, and personalized choices, making it a go-to destination for those seeking a blend of fashion and functionality.

#### 3.2 Project Description :

##### Objective

**1 .Diverse Product Range:** Offer a broad selection of optical products, including eyewear and accessories, to cater to diverse preferences and needs of customers.

**2. Interactive Shopping Experience:** Create an engaging online environment with immersive product displays, interactive features, and helpful guides for customers to make informed choices and enhance their shopping experience.

**3.Order Tracking and Quality Assurance:** Implement features allowing customers to track their orders, receive updates, and ensure product quality, ensuring a seamless and confident shopping journey.

**4.Customer Support and Guidance:** Facilitate meaningful interactions between customers and support staff, establishing a helpful and supportive community where customers can seek guidance, ask questions, and receive expert advice on optical products.

**5.Accessibility and Inclusivity:** Prioritize website accessibility, adhering to standards to ensure all users, including those with disabilities, can easily navigate and make purchases. Provide a diverse range of optical options to accommodate various styles, preferences, and vision needs.

**6.User-Friendly Shopping Platform:** Design an intuitive and easy-to-navigate e-commerce platform that enhances the overall user experience, making it seamless for customers to explore, select, and purchase optical products.

### 3.3 Existing System

- **User Interface (UI):** The front-end of the system includes a user-friendly interface for customers to browse menus, select items, and place orders. This can be a website or a mobile app.
- **Shopping cart Management:** A system for managing menus is crucial. It allows restaurants to update their menu items, descriptions, and prices easily. It may also support the addition of images for each item.
- **Order Management:** This includes the ability to view and manage incoming orders. Restaurant staff should be able to accept or reject orders, update the order status, and communicate with customers if necessary.
- **Payment Processing:** Integration with various payment gateways to facilitate online payments securely. Common payment methods include credit/debit cards, digital wallets, and other online payment options.
- **User Authentication and Profiles:** Users should be able to create accounts, log in securely, and manage their profiles. This feature enables order history tracking, personalized recommendations, and loyalty programs.

### 3.4 Problem Statements :

#### **Inefficient Order Processing:**

- Problem: The current Online Optics Shopping system lacks efficiency in processing and managing orders, leading to delays, errors, and poor customer satisfaction.
- Objective: Improve order processing to enhance the speed and accuracy of order fulfillment.

#### **Limited Menu Visibility:**

- Problem: Users find it challenging to navigate and explore the menu due to its complexity or poor organization, resulting in a suboptimal user experience.
- Objective: Enhance menu visibility and user-friendliness to make it easier for customers to find and select items.

#### **Payment Processing Issues:**

- Problem: Customers encounter difficulties during the payment process, such as payment failures, limited payment options, or concerns about security.
- Objective: Optimize the payment processing system to ensure a seamless and secure transaction experience.

### 3.5 Need for New System :

#### Improved User Experience:

- **Issue:** The current system offers a suboptimal user experience, leading to customer dissatisfaction and decreased user engagement.
- **Need:** Develop a new system with a more intuitive and user-friendly interface to enhance the overall user experience, making it easier for customers to navigate and place orders.

#### Security Concerns:

- **Issue:** Security vulnerabilities in the existing system pose risks to user data, payment information, and overall system integrity.
- **Need:** Implement a new system with robust security measures, including encryption, secure payment gateways, and compliance with industry standards.

### 3.6 Proposed System & Feature :

#### All module and Feature:

- User management
- Home
- Cart management
- Order management
- Payment Processing
- Notification
- Delivery

### 3.7 Scope :

- **Shopping cart Management:** The system allows to create, update, and manage their menus, including adding new items, adjusting prices, and adding descriptions.
- **Order Processing:** This includes order confirmation, preparation, dispatch, and delivery tracking. Order processing may also involve communication channels between customers and Seller.
- **Payment Processing:** The system supports secure online payment transactions through various methods, ensuring a smooth and reliable payment process.
- **Feedback and Ratings:** Users can provide feedback and ratings for Shopping Experince and their offerings, contributing to the reputation and quality assessment of the system.

**3.8Outcomes :**

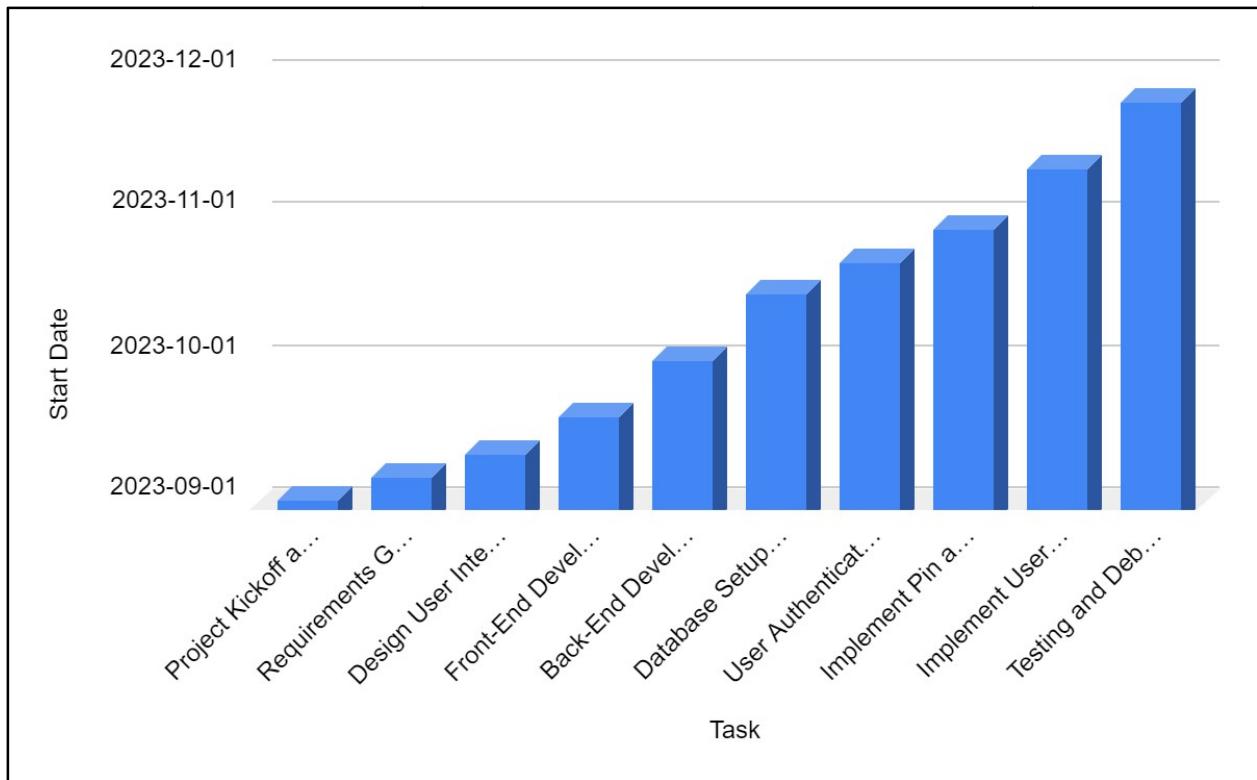
- **Enhanced User Experience:**
- **Outcome:** Users should have a seamless and enjoyable experience when using the platform, leading to increased customer satisfaction and loyalty.
- **Increased Sales and Revenue:**
- **Outcome:** A well-designed and user-friendly system can attract more customers, leading to increased order volumes and revenue for restaurants.

**3.9 Tools & Technology used :****Software Requirements:-**

- **Operating System :** Windows
- **Front-End:** HTML, CSS, BOOTSTRAP ,JAVASCRIPT
- **Back-End:** NODE JS ,MONGODB
- **Web Server:** MONGODB SERVER

**Hardware Requirements:-**

- I3(Processor) Recommended.
- 256 MB Ram
- 512 KB Cache Memory
- 512 GB SSD / 1 TB Hard Drive
- Microsoft Compatible 101 or more Key Board



## 4. Requirement Analysis

**4.1 Feasibility Study :** A feasibility analysis involves a detailed assessment of the need value and practically of a proposed enterprise such as system development. The process of designing and implementing record keeping system has significant accountability and resource implementing for an organization . feasibility analysis will help you make informed and transparent decisions at crucial points during the developmental process to determine whether it is operationally economically and technically realistic to process with a particular course of action .

Most feasibility studies are distinguished for both users and analysts. First the study often presupposes that when the feasibility document is being prepared the analyst is in a position to evaluate solution .second most studies tend to overlook the confusion inherent in system development – the constraints and the assumed attitudes.

**4.1.2 Operational Feasibility :** people are inherently resistant to change and computers have been known to facilitate change .An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a computerized system. It is common knowledge that computer installation have something to do with turnover transfers retraining and changes in employee job status .therefore it is understood that the introduction of a candidate system requires special effort to educate, sell and train the staff op new ways of conducting business.

**4.1.3 Technical feasibility :** technical feasibility centers around the existing computer system and to what extend it can support the proposed addition. For example if the current computer is operating at 80 per cent capacity an arbitrary ceiling – then running another application could overload the system or require additional hardware .this involves financial considerations to accommodate technical enhancements . if he budget is a serious constraint then the project is judged not feasible

### 4.2 User of the System :

- Admin
- Customer

### 4.3 Modules of the System :

- **User Management Module:**  
Allows users to register, log in, and manage their profiles.  
Includes features for password recovery, account settings, and user authentication.
- **Product Management Module:**  
Enables restaurants to create, update, and manage their menus.

Supports the addition of new items, modification of existing items, and menu categorization.

- **Order Management Module:**

Handles the entire order lifecycle, from order placement to fulfillment.

Includes features for order confirmation, order preparation, dispatch, and delivery tracking.

- **Payment Processing Module:**

Integrates with various payment gateways to facilitate secure online transactions.

Supports multiple payment methods, including credit/debit cards, digital wallets, and other online options.

- **Notification Module:**

Sends push notifications and alerts to users, notifying them of order confirmations, updates, and promotions.

Keeps users informed about the status of their orders in real-time.

- **Admin Dashboard Module:**

Provides a backend dashboard for restaurant owners and administrators to manage orders, update menus, and monitor performance. Includes features for analytics, reporting, and user management.

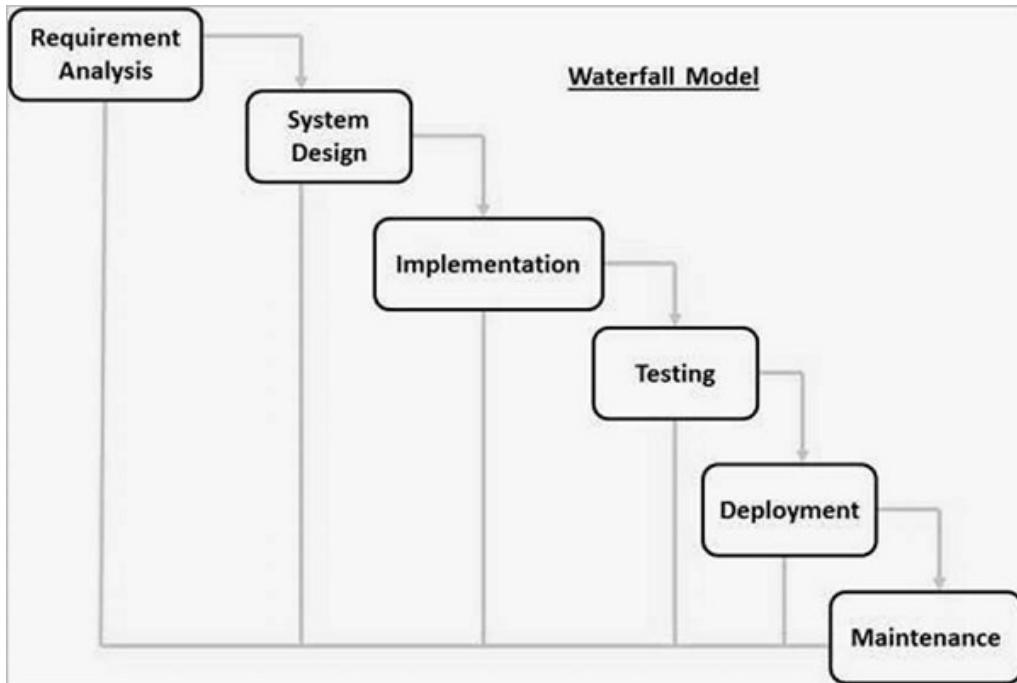
- **Delivery Logistics Module:**

Optimizes delivery logistics, including route planning, assignment of delivery personnel, and tracking delivery status.

Manages coordination between restaurants, delivery personnel, and users.

## 4.4 Process Model

**Waterfall model :**



- **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

- **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

## 4.5 Hardware & Software Requirement

### Hardware Requirements

Hardware Components	Specification
Processor	Intel core I3,/I5
RAM	4GB/8GB
Hard disk	512GB/1TB
Monitor	15.6 colour monitor or advance
Device	Keyboard, Mouse

### Software Requirements:

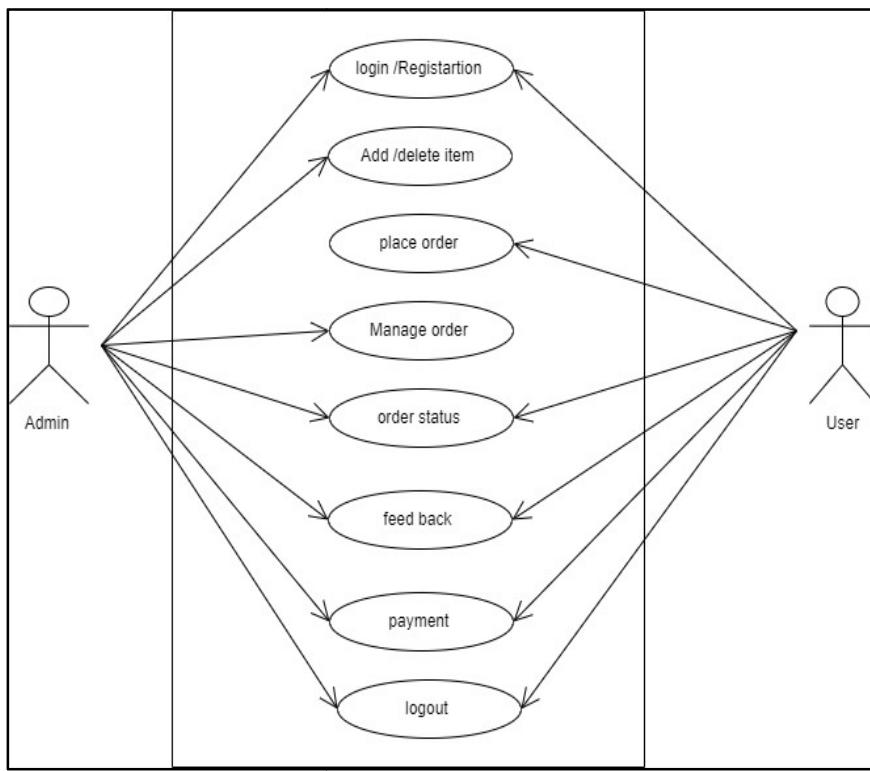
Name of component	Specification
Operating System	WindowsXP,windows10
Control Panel	Xampp Control Panel/Wamp Control Panel
Software development kit	Google Chrome, Internet Explorer, Mozilla Firefox ( any appropriate or suitable browser)
Programming Language	HTML, CSS,JAVASCRIPT,NODE JS

## 4.6 Use Cases :

- Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor.

- Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do.
- Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.
- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system.
- A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, Secondary Actor Receiver.

## 4.7.USE CASE DIAGRAM



## 5. Design

### 5.1 Use Case Scenarios:

#### 1. User Registration

- Scenario: A new user visits the e-commerce website and registers for an account.
- Steps
  - 1. The user navigates to the registration page and fills out the required fields such as name, email, and password.
  - 2. After submitting the registration form, the system validates the information and creates a new user account.
  - 3. Upon successful registration, the user receives a confirmation email with a verification link to activate the account.
  - 4. The user can now log in using the registered credentials and access the full functionality of the website.

#### 2. Admin Dashboard:\*

- Scenario: An admin accesses the backend dashboard to manage the e-commerce platform.
- Steps
  - 1. The admin logs in to the admin panel using their credentials.
  - 2. Upon successful login, the admin is redirected to the dashboard displaying various management options.
  - 3. The admin can view statistics such as sales figures, site traffic, and user engagement.
  - 4. The admin can manage user accounts, including adding, editing, or deactivating accounts if necessary.
  - 5. The admin can monitor and manage product listings, orders, inventory, and payments.

#### 3. Product Addition:

- Scenario: An admin adds a new product to the e-commerce platform.
- Steps:
  - 1. The admin navigates to the product management section of the admin dashboard.
  - 2. The admin selects the option to add a new product and fills out the product details such as name, description, price, and images.
  - 3. The admin assigns relevant categories and attributes to the product, such as gender (male, female, kids).
  - 4. Optionally, the admin sets inventory levels, SKU (stock-keeping unit), and other product-specific information.
  - 5. After reviewing the details, the admin saves the product, making it available for users to purchase.

#### 4. View All Products:

- Scenario: A user or admin accesses the product catalog to view all available products.

- Steps:

1. The user/admin navigates to the product catalog page on the website.
2. The system displays a list/grid view of all products available for purchase.
3. Users/admins can browse through the products, filter by category (e.g., men, women, kids), and sort by various parameters (price, popularity, etc.).
4. Each product listing includes essential information such as name, price, thumbnail image, and a brief description.
5. Users can click on individual product listings to view detailed information and proceed with the purchase scenario: A user filters products based on categories such as gender (male, female, kids).

- Steps:

1. The user navigates to the product catalog or applies a category filter on the website.
2. The system presents options to filter products by various categories, including gender.
3. The user selects the desired category (e.g., male, female, kids) to narrow down the product listings.
4. The system refreshes the product catalog, displaying only products that belong to the selected category.
5. The user can further refine the search results by applying additional filters or sorting options.

## 6. Payment Gateway Integration

- Scenario :A user completes a purchase and makes payment through the integrated payment gateway.

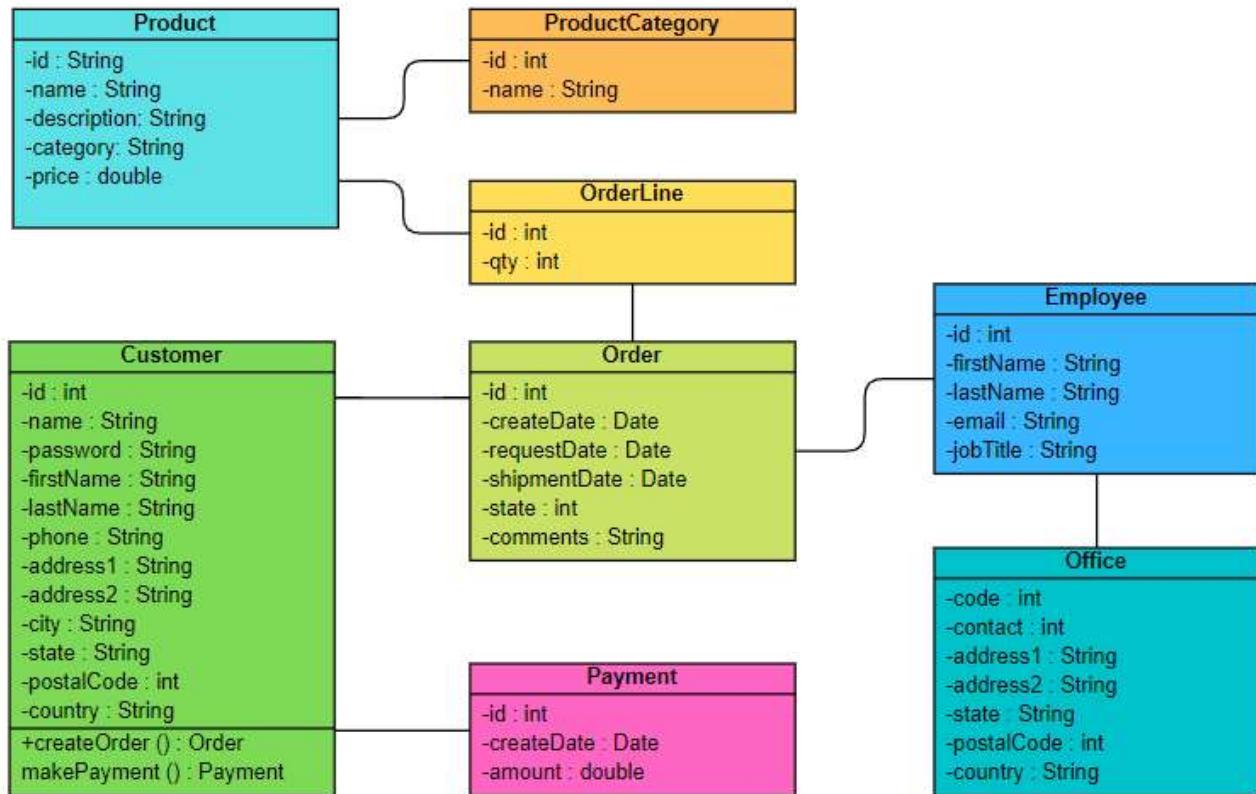
- Steps:

1. After selecting products for purchase, the user proceeds to the checkout page.
2. The system presents various payment options, including credit/debit card, PayPal, or other supported payment methods.
3. The user selects the preferred payment method and enters the necessary payment details (e.g., card number, expiration date, CVV).
4. The system securely processes the payment transaction using the integrated payment gateway.
5. Upon successful payment authorization, the user receives a confirmation of the order, and the transaction is recorded in the system.
6. If applicable, the system triggers order fulfillment processes such as inventory deduction and shipping arrangements.

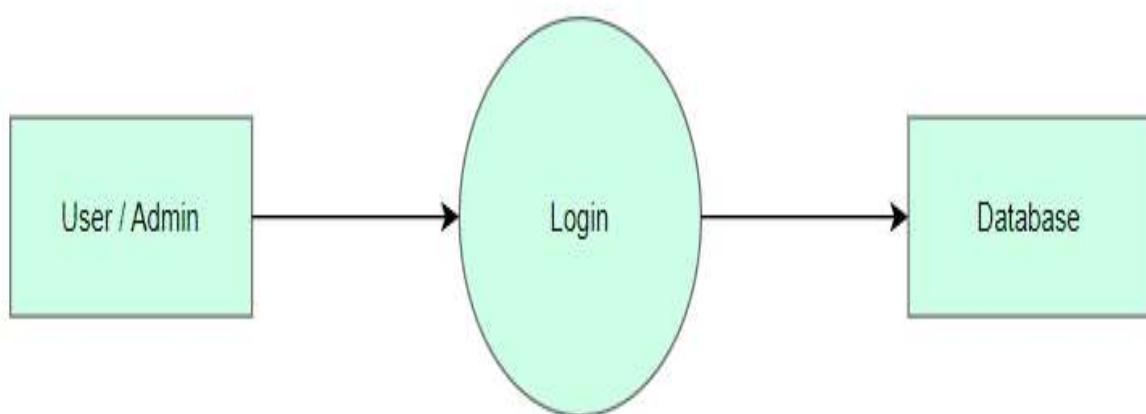
These use case scenarios cover the core functionalities of user registration, admin dashboard management, product addition/viewing, category-based filtering, and payment gateway integration in an e-commerce website. Each scenario illustrates typical interactions and workflows involved in operating and using the platform.

## 5.2 Diagrams

### UML

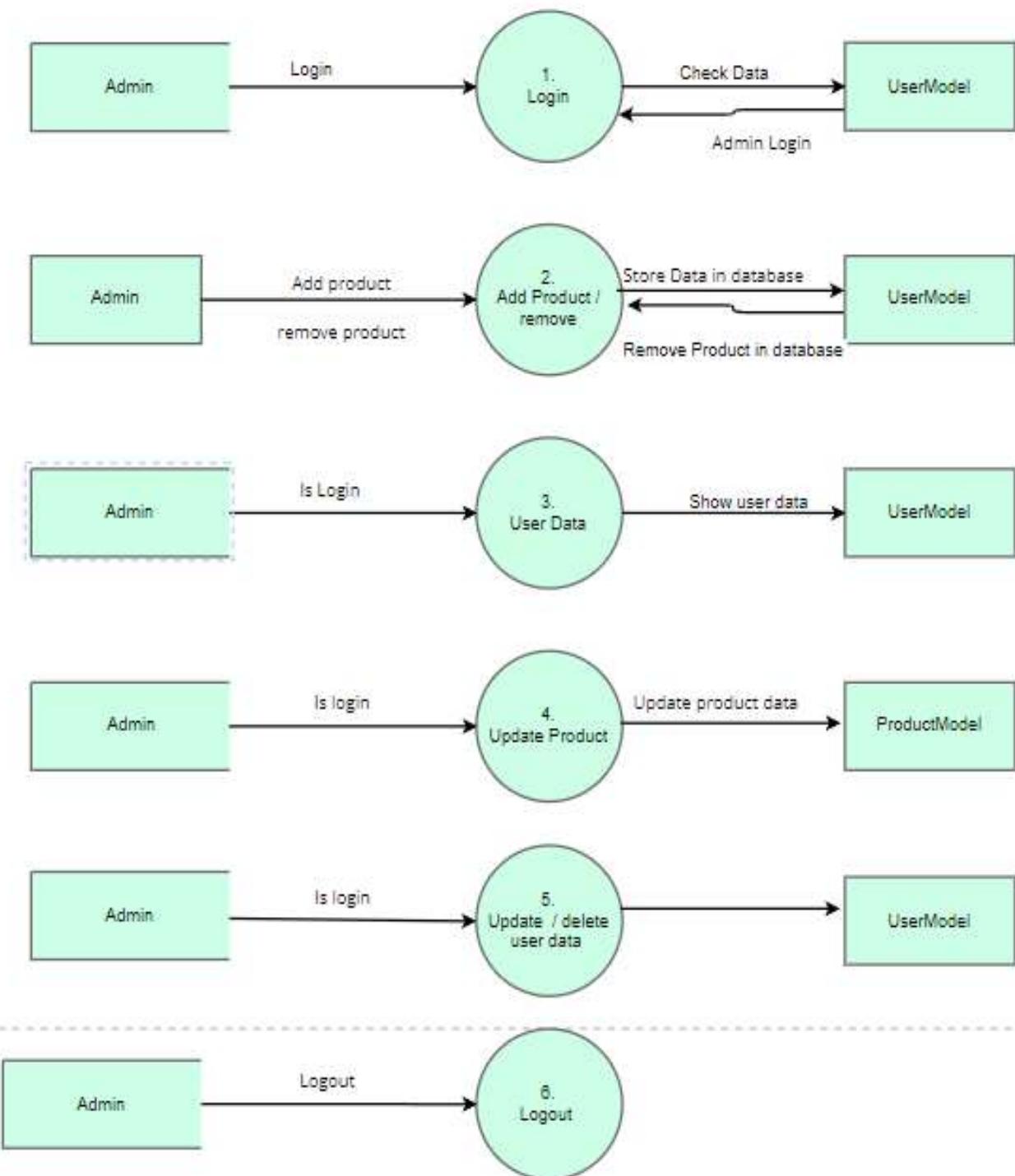


5.1 Order Processing Class Diagram

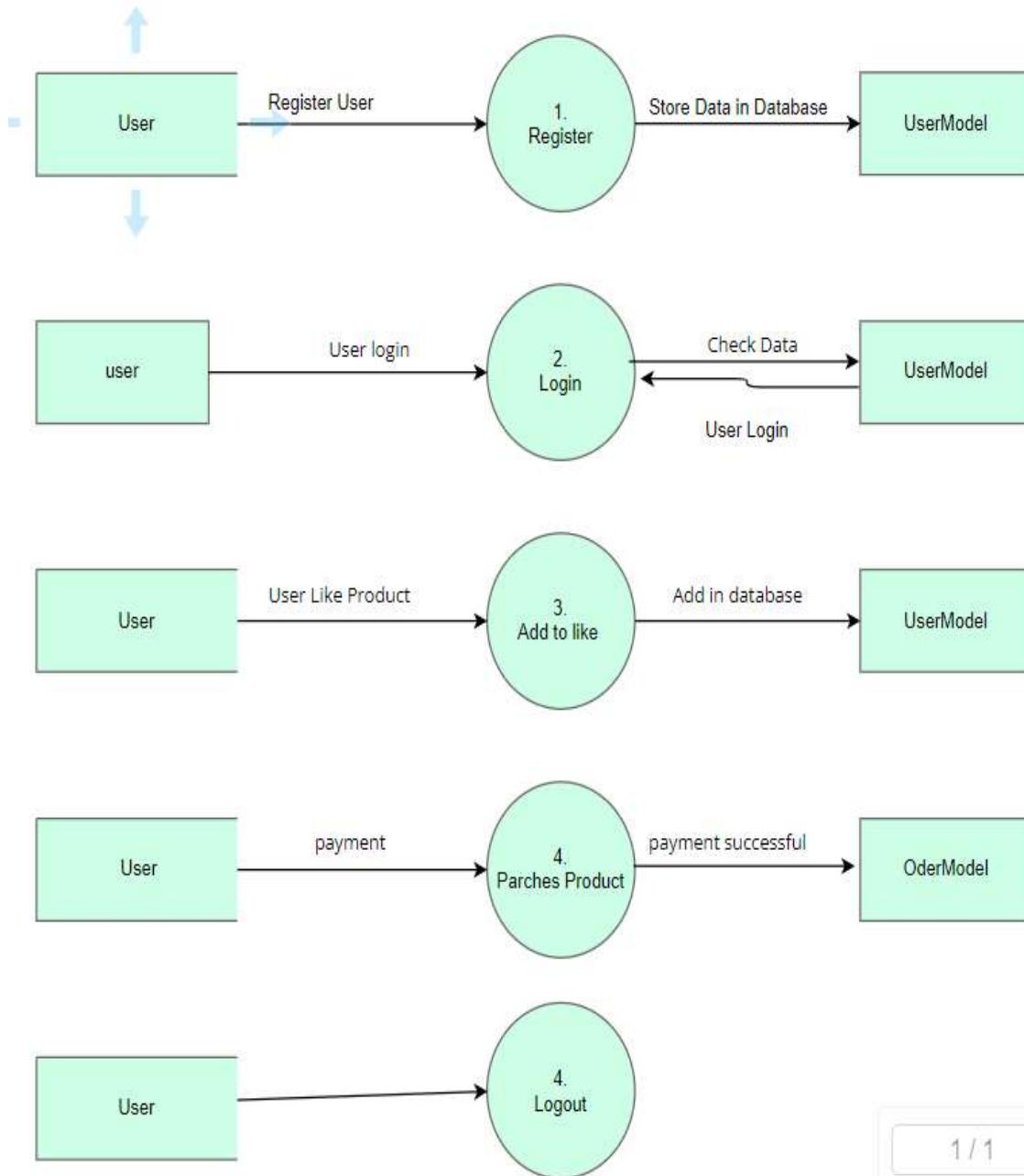
**DFD****5.2 : 0 Level DFD**

0.0 Level DFD

## 5.3 : Level 1 DFD for Admin



## 5.4: level 1 For user DFD



### 5.3: Data Dictionary:

MongoDB is a popular open-source NoSQL database management system that provides a flexible and scalable approach to storing and managing data. Here are five advantages of using MongoDB:

#### 1. Flexible Data Model:

- MongoDB uses a flexible document-based data model, storing data in JSON-like documents. This model allows developers to store data in a way that mirrors their application's object model, making it easier to work with and reducing the need for complex object-relational mapping (ORM) layers. This flexibility enables developers to adapt to changing data requirements more easily.

#### 2. Scalability and Performance:

- MongoDB is designed to scale horizontally across multiple servers, making it suitable for handling large volumes of data and high throughput applications. By distributing data across multiple nodes in a cluster, MongoDB can handle increased load and provide better performance compared to traditional relational databases. Additionally, MongoDB's sharding capabilities allow for automatic and transparent data partitioning, further enhancing scalability.

#### 3. High Availability and Fault Tolerance:

- MongoDB provides built-in support for replica sets, which are clusters of MongoDB nodes that maintain copies of the same data. Replica sets offer automatic failover and data redundancy, ensuring high availability and fault tolerance. In the event of a node failure, MongoDB can automatically promote a secondary node to primary, minimizing downtime and data loss.

#### 4. Rich Query Language and Indexing:

- MongoDB supports a powerful query language that allows developers to perform complex queries, aggregations, and data manipulations. It provides support for a wide range of query operators, allowing for efficient data retrieval and manipulation. Additionally, MongoDB supports indexing on various fields, which can significantly improve query performance by enabling faster data lookup and retrieval.

#### 5. Schema Evolution and Agile Development:

- MongoDB's schema-less nature allows for schema evolution without requiring downtime or complex migrations. Developers can easily add or modify fields within documents, making it well-suited for agile development methodologies where requirements may change frequently. This flexibility accelerates the

development process and reduces the overhead associated with schema changes in traditional relational databases.

**User Model :**

```
{  
  "user_id": "String (5 characters)",  
  "username": "String",  
  "mobile": "Number",  
  "password": "String",  
  "role": "String ('user' or 'admin')",  
  "like": ["ObjectId"],  
  "userAddressId": ["ObjectId"],  
  "createdAt": "Date"  
}
```

**Product Model:**

```
{  
  "productName": "String",  
  "description": "String",  
  "category": "String",  
  "price": "Number",  
  "stockQuantity": "Number",  
  "gender": ["String"],  
  "images": ["String"],  
  "availability": "Boolean",  
  "manufacturer": "String",  
  "variants": [  
    {  
      "size": ["String"],  
      "color": "String",  
      "price": "Number",  
      "stockQuantity": "Number",  
      "availability": "Boolean"  
    }  
  ],  
  "createdAt": "Date"  
}
```



User Address:

```
{  
  "userId": "ObjectId",  
  "address": {  
    "street": "String",  
    "city": "String",  
    "state": "String",  
    "zipCode": "String",  
    "NewMOB": "Number"  
},  
  "createdAt": "Date"  
}
```

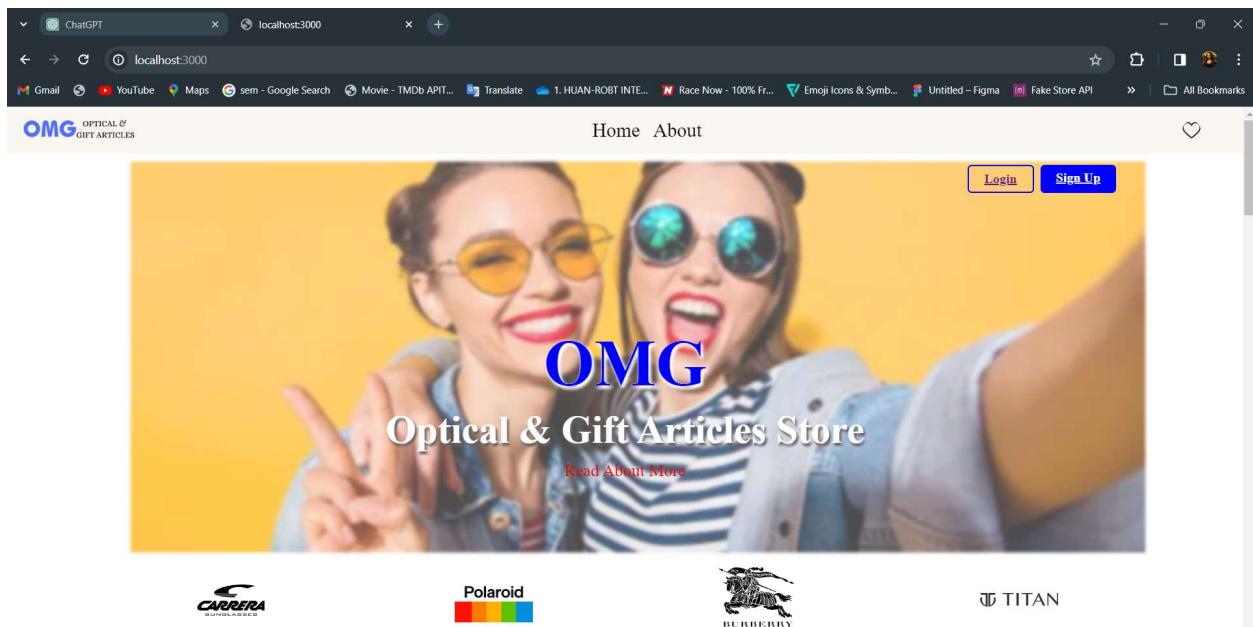
User Order:

```
{  
  "order_id": "String (5 characters)",  
  "user": "ObjectId",  
  "product": "ObjectId",  
  "userAddress": "ObjectId",  

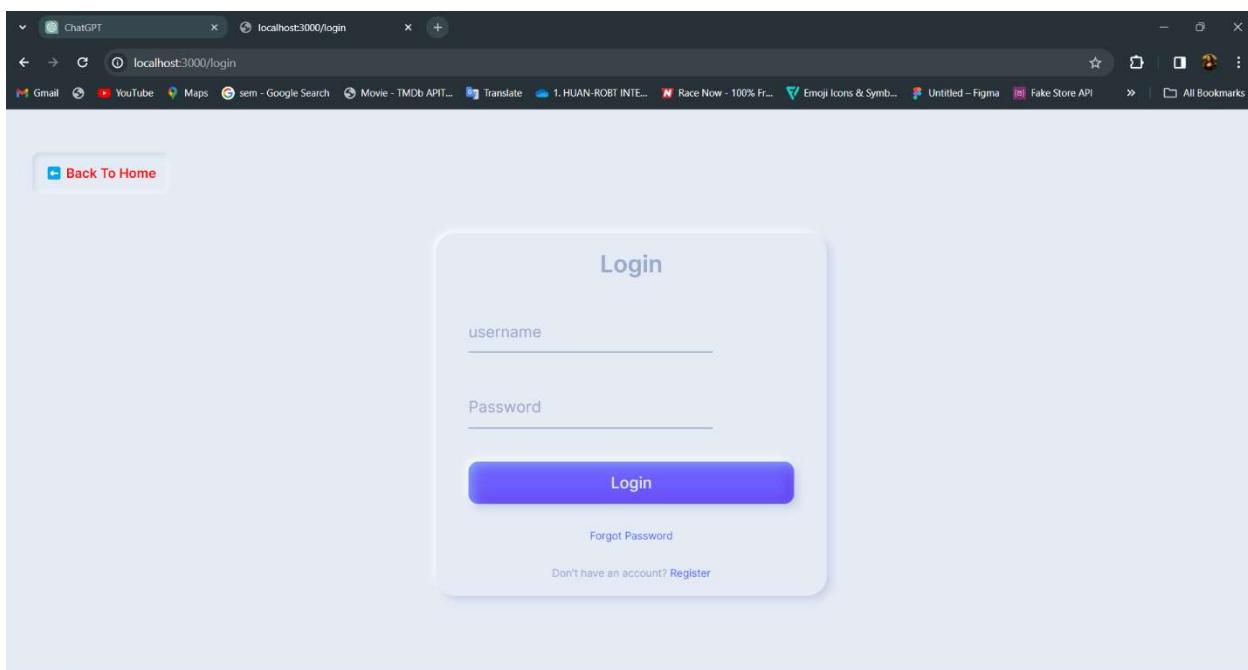
```

## 6.Implementation

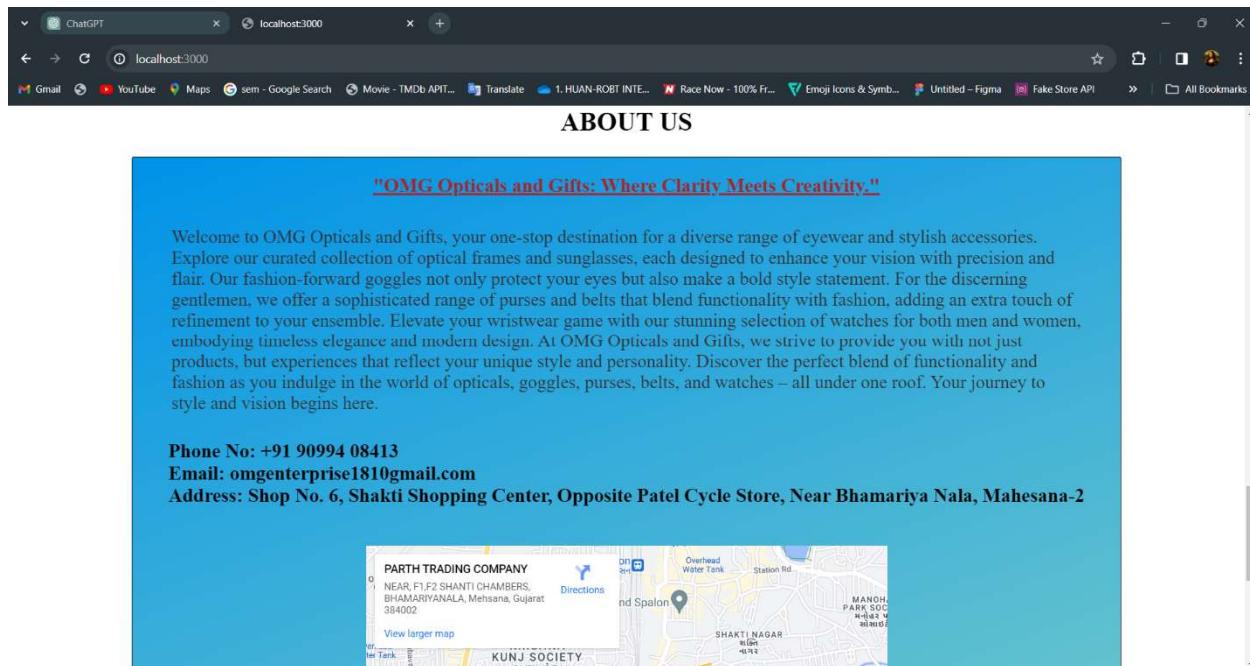
### 6.1 Form Layout



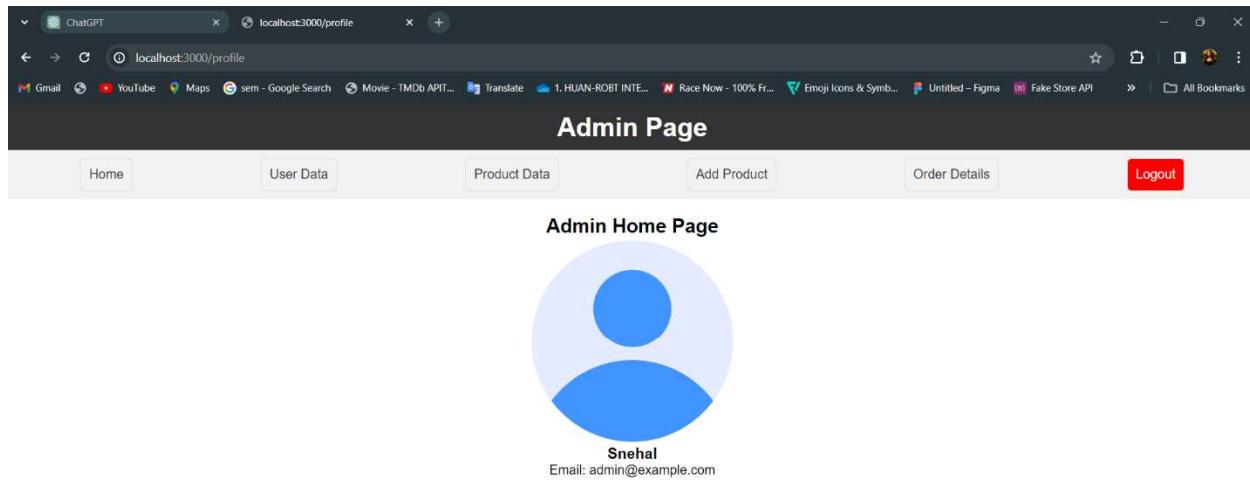
#### 6.2.1 Home Page



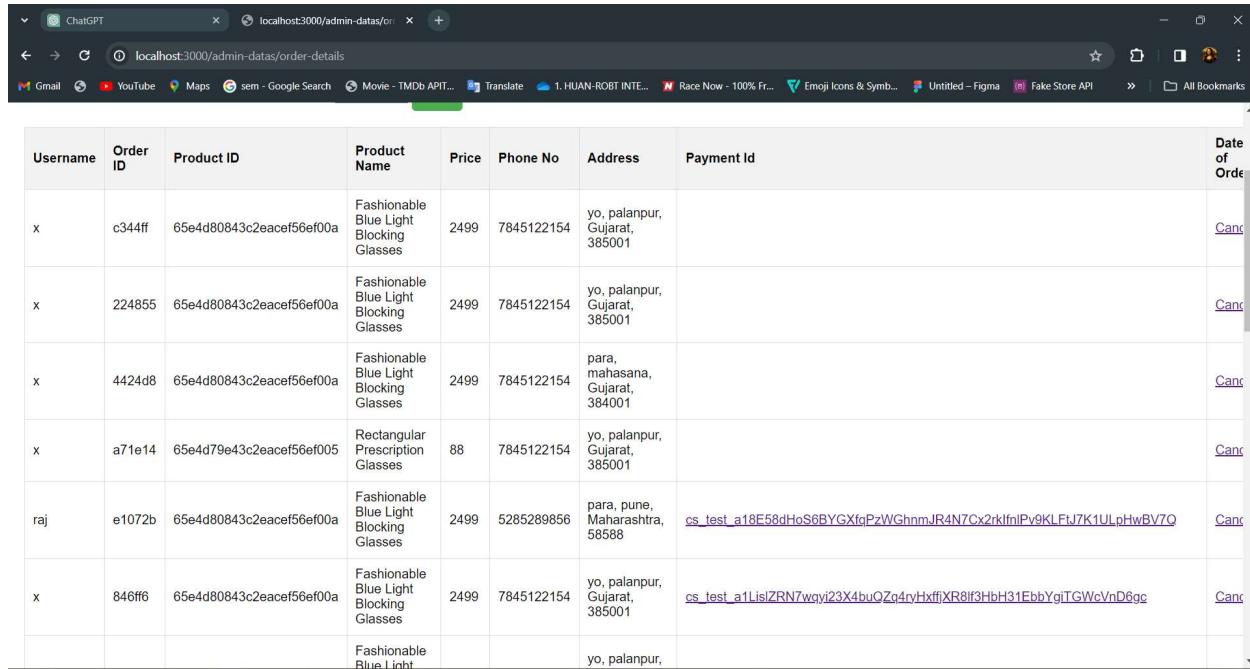
#### 6.1.2 Login page



### 6.1.3 About Page

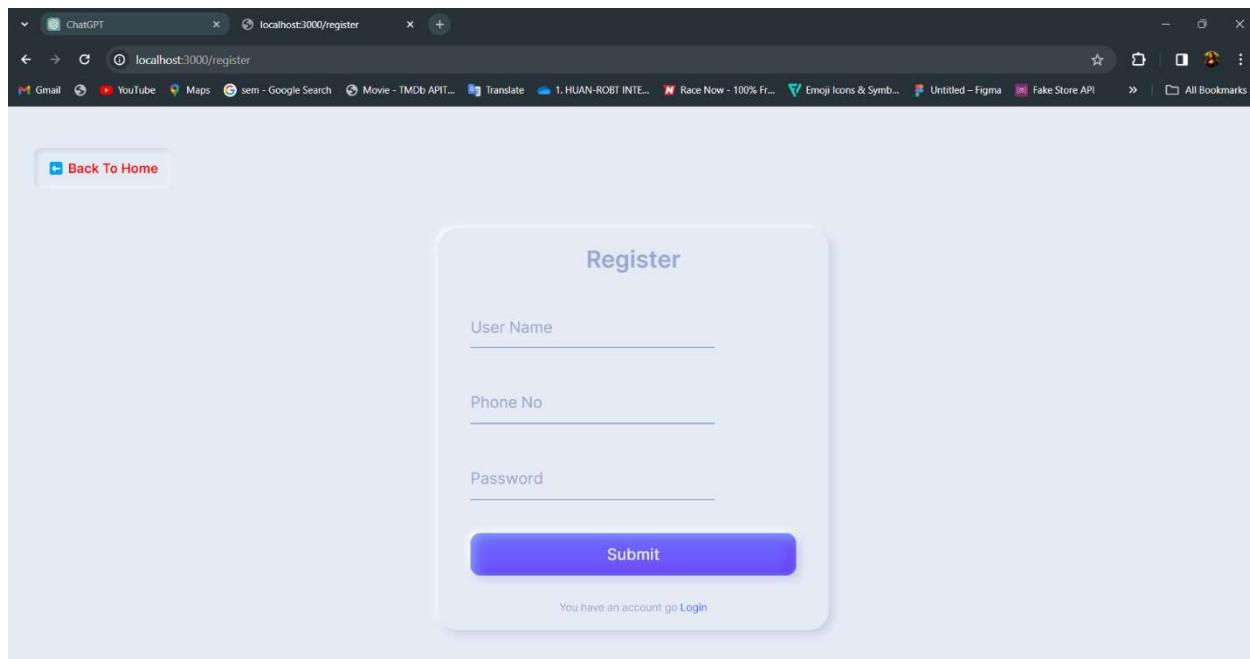


### 6.2.1 Admin Page



Username	Order ID	Product ID	Product Name	Price	Phone No	Address	Payment Id	Date of Order
x	c344ff	65e4d80843c2eacef56ef00a	Fashionable Blue Light Blocking Glasses	2499	7845122154	yo, palanpur, Gujarat, 385001		<a href="#">Cancel</a>
x	224855	65e4d80843c2eacef56ef00a	Fashionable Blue Light Blocking Glasses	2499	7845122154	yo, palanpur, Gujarat, 385001		<a href="#">Cancel</a>
x	4424d8	65e4d80843c2eacef56ef00a	Fashionable Blue Light Blocking Glasses	2499	7845122154	para, mahesana, Gujarat, 384001		<a href="#">Cancel</a>
x	a71e14	65e4d79e43c2eacef56ef005	Rectangular Prescription Glasses	88	7845122154	yo, palanpur, Gujarat, 385001		<a href="#">Cancel</a>
raj	e1072b	65e4d80843c2eacef56ef00a	Fashionable Blue Light Blocking Glasses	2499	5285289856	para, pune, Maharashtra, 58588	<a href="#">cs_test_a18E58dHoS6BYGXfgPzWGhnmJR4N7Cx2rklnlPv9KlFjJ7K1ULpHwBV7Q</a>	<a href="#">Cancel</a>
x	846ff6	65e4d80843c2eacef56ef00a	Fashionable Blue Light Blocking Glasses	2499	7845122154	yo, palanpur, Gujarat, 385001	<a href="#">cs_test_a1LislZRN7wqyi23X4buOZq4ryHxffiXR8lf3HbH31EbbYgiTGWcVnD6gc</a>	<a href="#">Cancel</a>
			Fashionable Blue Light			yo, palanpur,		

## 6.2.2 Order Page



The screenshot shows a registration form titled "Register". The form includes fields for "User Name", "Phone No", and "Password", each with a corresponding input field. Below the input fields is a large blue "Submit" button. At the bottom of the form, there is a link "You have an account go [Login](#)". In the top left corner of the page, there is a "Back To Home" button.

## 6.2.2 Registration Page

Category:

Optical  Sunglasses  Watches  Belts  Wallets

Gender:  Male  Female  Kids  For All

Price:

Stock Quantity:

Images (comma-separated URLs):  
 No file chosen

Availability:

Manufacturer:

### 6.2.3 Add New Product

Fashionable Blue Light Blocking Glasses  
₹2,499.00

Pay with card

Email

Card information  
 1234 1234 1234 1234   
 MM / YY  CVC

Cardholder name

Billing address  
 India   
  
Enter address manually.

Powered by | [Terms](#) | [Privacy](#)

### 6.2.4 Payment Page

## 6.3 Coding Convention

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>

<body>
  {{> nav}}
  </head>
  <header>
    <div class="home-section">
      <div class="home-section-main">
        <div class="btn-login-singus"> <a href="#" class="login-btn"> login </a> <a href="#" class="register"> sign up </a>
      </div>
      <div class="home-text">
        <h2>OMG</h2>
        <h4>optical & gift articles store</h4>
        <a class="home-text-a" href="#"> read about more </a>
      </div>
    </div>
  </div>
  </header>
  <div class="brand">
    <section class="brand-name">
      <a href="#"> </a>
      <a href="#"> </a>
      <a href="#"> </a>
      <a href="#"> </a>
    </section>
  </div>
  <div class="title">
    <h1>Optical</h1>
  </div>
  <div class="optical">
```

### 6.3.1 Index Page

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link rel="stylesheet" href="../../stylesheets/register.css">
</head>
<body>
    <div class="main vh">

        <div class="back"> <a class="back-btn" href="/"> ← Back To Home </a> </div>
        <div class="container">
            <h1>Login</h1>

            {{#if error.length}}
                <p class="error">{{error}} </p>
            {{else}}
            {{/if}}

            <form action="/login" method="post">
                <div class="form-control">
                    <input type="text" required name="username">
                    <label>username</label>
                </div>
                <input type="hidden" name="role" value="user">
                <div class="form-control">
                    <input type="password" required name="password">
                    <label>Password</label>
                </div>
                <input type="submit" class="btn" value="Login">
                {{!-- <p class="text"> <a href="#">Forgot Password</a></p> --}}
                <p class="text">Don't have an account? <a href="#">Register</a></p>
            </form>
        </div>
    </div>
...

```

### 6.3.2 login Page

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link rel="stylesheet" href="../../stylesheets/register.css">
    <style>
        .home-text {
            position: relative;
            top: -20rem;
        }
    </style>
</head>

<body>

{{!-- {{> nav}} --}}
<div class="main">
    <div class="back register-margin"> <a class="back-btn" href="/"> ← Back To Home </a> </div>
    <div class="container">
        <h1>Register</h1>

        {{!-- {{#if messages.error}} --}}
        <div class="error-message">
            {{messages.error}}
        </div>
        {{/if}} --}}

        {{#if errorMessages}}
        <div class="error-message">
            {{#each errorMessages}}
            <p class="error">{{this}}</p>
            {{/each}}
        </div>
    
```

### 6.3.3 Register Page

```
<html lang="en">
<head>
    <style>
        </style>
</head>

<body>
    {{> profileNav}}<!-- Assuming you have a navigation partial, replace with your own navigation code --&gt;

    &lt;div id="dynamicCards" class="img"&gt;
        {{#each product_data}}
            &lt;div class="main"&gt;

                &lt;img src="../../images/uploads/{{images}}" alt="{{imgAlt}}" class="img-e"&gt;
                &lt;div class="contant"&gt;
                    &lt;h1&gt;{{productname}}&lt;/h1&gt;
                    &lt;p class="Price"&gt;{{price}}&lt;/p&gt;
                    &lt;!-- Add other fields as needed --&gt;
                    &lt;a href="/userAddress/{{_id}}" class="Button"&gt;Buy&lt;/a&gt;
                    &lt;a href="/likeProduct/{{_id}}" class="Button"&gt;Add Like&lt;/a&gt;
                &lt;/div&gt;
            &lt;/div&gt;
        {{/each}}
    &lt;/div&gt;

    &lt;!-- Your script block here (if needed) --&gt;

&lt;/body&gt;

&lt;/html&gt;</pre>
```

### 6.3.4 Cart Page

## 7. Testing

### 7.1 Test Strategy

The process or method of finding error/s in a software application or program so that the application functions according to the end user's requirement is called software testing. Software testing is the process of verifying a system with the purpose of identifying any errors, gaps or missing requirement versus the actual requirement. Software testing is broadly categorised into two types - functional testing and non-functional testing.

### 7.2 Test Cases

#### White Box Testing:-

- White –box testing, sometimes called glass-box testing, is a test case design method that uses the control structure of the procedural design. Using white box testing methods, we had exercised all loops at their boundaries and within their operational bounds, and exercise internal data structures to ensure their validity.
- Some type of errors like typographical errors occurs when a program is translated into programming language source code. Many will be uncovered by syntax and type checking mechanisms, but others may go undetected until testing begins. So, we had tried to remove all type of source code errors.

#### ➤ Black Box Testing:-

- Black-box testing, also called behavioural testing, focuses on the functional requirements of the software. That is, black-box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black-box testing is not an alternative to White box techniques. Rather, it is a complementary approach that is likely to uncover a different class of errors than white-box methods.
- Using black-box testing we had found some errors like incorrect or missing functions, interface errors, errors in data structures or external database access, behaviour or performance errors, initialization and termination errors.

## 8. Future Enhancement

OMGOpticals & Gift Articles is an innovative online shopping platform dedicated to offering a diverse range of optical products and accessories, including watches, wallets, belts, and stylish eyewear. Our mission is to provide customers with a seamless and enjoyable shopping experience, focusing on quality, user-friendly navigation, and personalized choices.

Well, I and my team members have worked hard in order to present an improved website better than the existing one's regarding the information about the various activities. Still, we found out that the project can be done in a better way.

Primarily, our platform currently allows users to explore our collection of products, browse through various categories, and make purchases. However, we recognize the importance of providing quick and easy access to specific items or brands that our customers may be searching for.

To address this, we plan to introduce a sophisticated search functionality that enables users to directly search for particular products, brands, or attributes. By implementing this enhancement, customers will be able to quickly find the exact items they desire, streamlining the shopping process and enhancing overall satisfaction.

Our goal is to seamlessly integrate this search feature into our website, ensuring that it aligns with our commitment to user-friendly navigation and personalized choices. Through continuous refinement and optimization, we aim to solidify OMGOpticals & Gift Articles as the premier destination for individuals seeking high-quality eyewear and fashion accessories online.

## 9.Bibliography

### 1. Books:

- Krug, S. (2014). "Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability." New Riders.
- Chaffey, D., & Ellis-Chadwick, F. (2019). "Digital Marketing: Strategy, Implementation and Practice." Pearson Education Limited.

### 2. Websites and Online Resources:

- W3Schools. (n.d.). Retrieved from <https://www.w3schools.com/>
- MDN Web Docs. (n.d.). Retrieved from <https://developer.mozilla.org/en-US/>

### 3. \*\*Frameworks and Libraries\*\*:

Node.js . <https://nodejs.org/en>

Express.js. <https://expressjs.com/>

Mongoose. <https://mongoosejs.com/>

### 4. Programming Languages:

- JavaScript. (n.d.). Retrieved from <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- HTML. (n.d.). Retrieved from <https://developer.mozilla.org/en-US/docs/Web/HTML>
- CSS. (n.d.). Retrieved from <https://developer.mozilla.org/en-US/docs/Web/CSS>

### 5. Software and Tools:

- Visual Studio Code. (n.d.). Retrieved from <https://code.visualstudio.com/>
- Adobe Photoshop. (n.d.). Retrieved from <https://www.adobe.com/products/photoshop.html>

**6. Version Control:**

- GitHub. (n.d.). Retrieved from <https://github.com/harshad00/omg.git>

**7. Content Management System (CMS):**

- WooCommerce. (n.d.). Retrieved from <https://woocommerce.com/>

**8. Payment Gateways:**

- PayPal. (n.d.). Retrieved from <https://www.paypal.com/>

- Stripe. (n.d.). Retrieved from <https://stripe.com/>

**9. Analytics and Tracking:**

- Google Analytics. (n.d.). Retrieved from <https://analytics.google.com/analytics/web/>

**10. Security:**

- SSL.com. (n.d.). Retrieved from <https://www.ssl.com/>

**13. Documentation:**

- API Documentation for integrated services and libraries.