

“ONLINE PERFUME PURCHASE”

A PROJECT REPORT

Submitted By

ISHITA JAVIYA - 210310116505

In Partial Fulfillment for the award of the Degree of

BACHELOR OF ENGINEERING

In

INFORMATION TECHNOLOGY

LUKHDHIRJI ENGINEERING College, Morbi -363642

Affiliated To:-



Gujarat Technical University

Gujarat Technological University, Ahmedabad

Year :- 2023/24



Gujarat Technical University

LUKHDHIRJI ENGINNERING COLLEGE MORBI(3636342)

CERTIFICATE

This is to certify that the project entitled “**ONLINE PERFUME PURCHASE**” has been carried out by “**ISHITA JAVIYA (210310116505)**” under my guidance in fulfillment for the degree of Bachelor of Engineering in **Information Technology** – 8th Semester of Gujarat Technological University, Ahmedabad during the academic year 2023- 2024.

Internal Guide

Prof. Rakesh Parmar
Lecturer At IT Department

Head of Department

Chirag Patel
Head of Department

COMPANY CERTIFICATE



Project Internship Completion Letter

Date: 30 May 2024

This is to certify that Ishita Javiya , a student of Lakhdirji Engineering College has successfully completed her internship in the field of Web Development from **17/01/2024 to 28/04/2024 (total number of weeks :15 Weeks)** under the guidance of Mr. Brijesh Pandey.

Her internship activities including different technology/framework such as **HTML, CSS, JAVASCRIPT, REACTJS**. During the period of her internship program with us, she had been exposed to different processes and was found diligent, hardworking and inquisitive.

We wish her every success in her life and career.

Best Regards,
Payal Khandediya

Manager
9724004242

TOPS TECHNOLOGIES PRIVATE LIMITED



3rd Fl Aval Complex, University Road, Panchayat Nagar Chowk, Rajkot, Gujarat 360005 +91 97240 04242
www.tops-int.com

COMPLETION CERTIFICATE



GUJARAT TECHNOLOGICAL UNIVERSITY

CERTIFICATE FOR COMPLETION OF ALL ACTIVITIES AT ONLINE PROJECT PORTAL

B.E. SEMESTER VIII, ACADEMIC YEAR 2023-2024

Date of certificate generation : 01 May 2024 (15:07:34)

This is to certify that, **Javiya Ishitaben Sureshbhai** (Enrolment Number - 210310116505) working on project entitled with **Online perfume purchase** from **Information Technology** department of **LUKHDHIRJI ENGINEERING COLLEGE, MORBI** had submitted following details at online project portal.

| | |
|---------------------------|-----------|
| Internship Project Report | Completed |
|---------------------------|-----------|

Name of Student : Javiya Ishitaben
Sureshbhai

Name of Guide : Mr.keshwala rahul ranmalbhai

Signature of Student : _____

*Signature of Guide : _____

Disclaimer :

This is a computer generated copy and does not indicate that your data has been evaluated. This is the receipt that GTU has received a copy of the data that you have uploaded and submitted as your project work.

*Guide has to sign the certificate, Only if all above activities has been Completed.



Gujarat Technical University

DECLARATION

I hereby declare that the Project report submitted along with the Internship Project entitled "**ONLINE PERFUME PURCHASE**" submitted in partial fulfillment for the degree of Bachelor of Engineering in Information Technology to Gujarat Technological University, is a Bonafede record of original project work carried out by me , at Tops Technologies under the supervision of Mr.Brijesh Pandey and Under Guidelines of my Faculties .

Name of the Student

ISHITA JAVIYA

Sign of Student

ACKNOWLEDGEMENT

I am very thankful to the project Coordinator of **MR.BRIJESH PANDEY** for their suggestion and help. who has provided us a lot of support & guidance from the beginning to the end of the project development.

A work of this nature would not have been possible without the encouragement and meticulous attention received from them. The faculties has also played a vital role in building up my project website, under their guidance and training it became much easier to develop a project.

**ISHITA JAVIYA
(210310116505)
(Semester 8th)**

ABSTRACT

The upon project report the “**ONLINE PERFUME PURCHASE**” we will need to buy lots of products from a shop can interact with this website.

Online Perfume Purchase is an online website to buy fragrance ,Perfume, Attars ,make your own perfume kit. destination dedicated to simplifying the search for the perfect scent. Our platform, powered by React, offers a straightforward and enjoyable experience for users to explore and shop for their favorite fragrances.

At Perfume Haven, users are welcomed into a visually stunning interface that showcases our diverse range of perfumes. Through intuitive navigation and dynamic filtering options, visitors can effortlessly browse through various categories, from floral to woody, and everything in between.

Our React-powered website prioritizes user experience, ensuring fast loading times and responsive design across all devices. With interactive features such as scent quizzes and personalized recommendations, Perfume Haven tailors each visit to the unique preferences of every user.

INDEX

Acknowledgement

Abstract

1. Introduction

| | |
|--|-------|
| 1.1 Project Summary | 1 |
| 1.2 Purpose | 1 |
| 1.3 Scope | 2-3 |
| 1.4 Technologies and Literature Review | 3 - 6 |

2. System Requirements and Study

| | |
|---------------------------------------|-------|
| 2.1 SRS - S/W & H/W REQUIREMENT | 7 - 8 |
| 2.2 User Characteristics | 8 |
| 2.3 Constrain | 8 |

3. System Analysis

| | |
|--|-------|
| 3.1 Study of current System | 9 |
| 3.2 Problem and weakness of current system | 9 |
| 3.3 Requirements of new system | 10 |
| 3.4 Feasibility Study | 10-11 |
| 3.5 Requirements Validation | 11 |

4 Project Management

| | |
|--|--------|
| 4.1 Project Planning | 12 -13 |
| 4.2 Project Development Approach and Justification | 14 |
| 4.3 Project Scheduling | 14-17 |
| 4.4 Risk Management | 17-20 |

| | | |
|--|---------|-------|
| 4.5 Estimation: | | |
| 4.5.1 Effort Estimation | 21 | |
| 4.5.2 Cost Analysis | 22-23 | |
| 5. Functional and Behavioral Design of System and Diagrams | | |
| 5.1 Data Modeling diagrams | | |
| 5.1.1 Data Flow Diagram | 24-28 | |
| 5.1.2 Use Case Diagram | 29 - 30 | |
| 5.1.3 E-R Diagram | 31 - 34 | |
| 5.1.4 Activity Diagram | 35 - 38 | |
| 5.1.5 Component Diagram | 39 | |
| 5.1.6 Class Diagram | 40 | |
| 5.2 Data Dictionary | 41 - 44 | |
| 6. Implementation Planning and Details | | |
| 6.1 Implementation Environment | 45 - 47 | |
| 7. Maintenance | | 48 |
| 7.1 Testing | | |
| 7.1.1 White -Box Testing | 48 - 50 | |
| 7.1.1 Black -Box Testing | 50 - 51 | |
| 8. Screen Shots Of websites | | |
| 8.1.1 Admin Screen Shots | 52 - 58 | |
| 8.1.2 Client or User Screen Shots | 59 - 64 | |
| 9. Limitation and Future Enhancement | | 65-66 |
| 10. CONCLUSION | | 67 |
| 11. BIBLIOGRAPHY | | 68 |

1. INTRODUCTION

1.1 PROJECT OVERVIEW:

In day to day life we will need to buy lots of products from a shop. And Fragrance, perfume is a day to day used product for Professional and personal Purpose. Now a days it is really hard to get some time to go out and get them by ourselves due to busy life style or lots of works.

In order to overcome these, we have one solution that is e-commerce site. Where we can get all required products online. The proposed system helps in building a website to buy products online.

React-powered website prioritizes user experience, ensuring fast loading times and responsive design across all devices. With interactive features such as scent quizzes and personalized recommendations, Perfume Haven tailors each visit to the unique preferences of every user.

User can get all information about My website .User can also get ONLINE PERFUME PURCHASE information. for these, user have to visit my web-site.

1.2 PURPOSE: GOALS & OBJECTIVES

The purpose of this project is to develop the user friendly and interactive website which will allow the user to get the information and pictures AND Deep knowledge of product making at our website ONLINE PERFUME PURCHASE preparation. User will be able to see the site-maps. User can get information about verity of products in My website.

The main purpose of this project is about giving flexibility to the Products as this website plays main role at admin side. With a clean and user-friendly interface, Online perfume Purchase makes browsing and purchasing fragrances Perfumes Attars a breeze. Our website features a carefully curated selection of perfumes, categorized by scent families and popular brands, ensuring that every visitor can easily find what they're looking for.

1.3 SCOPE

The aim of the project is to build a simple, effective and which gives the easily information of any types of question in my website.

Focus on creating a seamless and visually appealing user experience to attract and retain customers. This includes responsive design, smooth navigation, and intuitive UI/UX elements.

Search and Filters: Implement robust search functionality and filtering options to help users find their desired perfumes quickly and efficiently.

Reviews and Ratings: Allow users to leave reviews and ratings for products, fostering trust and credibility among potential buyers.

Integration with APIs: Integrate with APIs to fetch product data, manage inventory, and handle payment processing securely.

SEO Optimization: Ensure that your website is optimized for search engines to increase visibility and attract organic traffic.

Users:

- Log in
- Access their own account
- Put request for their require information
- Give Feedback
- Search Product View Product
- Add to cart And Like or Wishlist Product
- Register personal information
 - Phone number
 - Address
 - Password
 - Email-Id

Admin:

- Add or Remove users
- Add, Update Products and Categories
- Check out media
- Count the no of visitors
- View all user information
- Manage User
- Manage Feedback

1.4 Technology & Literature review

ABBREVIATION

| | |
|-------|-----------------------------------|
| HTML5 | Hyper Text Markup Language |
| CSS3 | Cascading Style Sheet |
| JS | Java Script |
| REACT | Java Script Library |
| JSON | Java Script Object Notation |
| XML | Extensible Markup Language |
| REDUX | Java Script Library |
| API | Application programming Interface |

About REACT

React.js, often referred to simply as React, is a powerful JavaScript library for building user interfaces, particularly for single-page applications and interactive web experiences. Developed an

d maintained by Facebook, React has gained widespread adoption in the web development community due to its efficiency, flexibility, and declarative approach to building UI components.

At the core of React is philosophy is the concept of reusable components. Developers can create encapsulated components that manage their state, making it easier to build complex user interfaces while keeping the codebase organized and maintainable. These components are composed together to form the UI of an application, enabling developers to efficiently update and render specific parts of the interface in response to changes in data or user interactions.

One of the key features of React is its use of a virtual DOM (Document Object Model), which is an in-memory representation of the actual DOM. React compares the virtual DOM with the real DOM and applies only the necessary updates, resulting in improved performance and faster rendering times, especially for applications with dynamic content.

React also promotes a unidirectional data flow, where data flows from parent components to child components via props (properties), and communication between components is facilitated through callbacks and event handling. This architecture enhances predictability and makes it easier to debug and reason about the application's behavior.

Furthermore, React ecosystem is enriched with tools and libraries that complement its functionality, such as React Router for client-side routing, Redux for managing application state, and Material-UI for implementing customizable UI components following Google's Material Design guidelines.

React.js empowers developers to build dynamic and interactive user interfaces with ease, leveraging its component-based architecture, virtual DOM rendering, and unidirectional data flow. Whether creating simple UI elements or complex web applications, React provides a robust foundation for delivering high-performance and engaging user experiences on the web.

Advantages of REACT

Component-Based

Virtual DOM

Declarative Syntax

Efficiency

Rich Ecosystem

SEO-Friendly

Cross-Platform Compatibility

Community Support

❖ CSS – Cascading Style Sheet

Styles Solve a Common Problem

HTML tags were originally designed to define the content of a document. They were supposed to say "This is a header", "This is a paragraph", "This is a table", by using tags like `<h1>`, `<p>`, `<table>`, and so on. The layout of the document was supposed to be taken care of by the browser, without using any formatting tags.

As the two major browsers - Netscape and Internet Explorer - continued to add new HTML tags and attributes (like the `` tag and the `color` attribute) to the original HTML specification, it became more and more difficult to create Web sites where the content of HTML documents was clearly separated from the document's presentation layout.

To solve this problem, the World Wide Web Consortium (W3C) - the non profit, standard setting consortium, responsible for standardizing HTML - created STYLES in addition to HTML 4.0. All major browsers support Cascading Style Sheets.

Style Sheets Can Save a Lot of Work

Styles sheets define HOW HTML elements are to be displayed, just like the `font` tag and the `color` attribute in HTML 3.2. Styles are normally saved in external .css files. External style sheets enable you to change the appearance and layout of all the pages in your Web, just by editing one single CSS document.

CSS is a breakthrough in Web design because it allows developers to control the style and layout of multiple Web pages all at once. As a Web developer you can define a style for each HTML element and apply it to as many Web pages as you want. To make a global change, simply change the style, and all elements in the Web are updated automatically.

Multiple Styles Will Cascade Into One Style sheets allow style information to be specified in many ways. Styles can be specified inside a single HTML element, inside the `<head>` element of an HTML page, or in an external CSS file. Even multiple external style sheets can be referenced inside a single HTML document.

❖ REDUX - JAVA SCRIPT LIBRARY

Redux is a predictable state container for JavaScript applications, designed to manage the application state in a centralized and predictable manner. It works seamlessly with libraries like React but can be used with any framework or library for building web applications.

At its core, Redux follows the principles of Flux architecture, providing a single source of truth for the entire application state. This centralized state management simplifies the process of handling and updating application data, making it easier to maintain and reason about complex applications.

The key concept in Redux is the store, which holds the entire application state. The state within the store is immutable, meaning it cannot be directly modified. Instead, state changes are achieved through actions, which are plain JavaScript objects describing the type of change to be made. These actions are dispatched to reducers, which are pure functions responsible for updating the state based on the given action.

Reducers take the current state and an action as input and return a new state, ensuring that state transitions are predictable and deterministic. This unidirectional data flow simplifies debugging and tracking state changes, as each modification is traceable through the sequence of dispatched actions.

Redux also provides middleware, which allows developers to extend the functionality of Redux with features such as logging, asynchronous actions, and routing. Middleware intercepts dispatched actions before they reach the reducers, enabling additional logic to be applied, such as making asynchronous API calls or logging state changes.

2. SYSTEM REQUIREMENT STUDY

2.1.SRS - SOFTWARE & HARDWARE REQUIREMENT

HARDWARE DESCRIPTION

The selection of hardware is very important in the existence and proper working of any software. When selecting hardware, the size and requirements are also important.

Minimum Requirements:

Processor :- AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz
RAM :- 8.00 GB

The proposed System is developed on:

Processor :- AMD Ryzen 5 5500U with Radeon Graphics 2.10 GHz
RAM :- 8.00 GB
Hard Disk Drive :- 200 GB
Key Board : Standard 101/102 or Digi Sync Family
Monitor : Display Panel (1024 X 764)
Display Adapter : AMD Radeon (TM) Graphics

List of Browsers Supported :-

- Chrome
- Safari
- UC
- Firefox
- Edge
- Samsung Internet
- Opera
- Epic
- Brave
- Vivaldi

SOFTWARE DESCRIPTION

Operating System : Windows 11

Front- End : React js

Back- End : Database FireBase Or React API
Tools : Visual Studio

2.2 USER CHARACTERISTICS

When the topic comes to the types of user who interact with this application, mainly two categories come into picture:

1. Administrator.
2. Common Person (User).
3. Common Admin (User).

- An administrator is kept mainly to maintain the database of shape files and their related data. The administrator can upload the purchases of the media. He is also responsible for the web application maintenance and queries or requests by the user to him.
- The application deals with the collaborated efforts from the user too. The task of the user here is a major one regarding the application he opts to use. This application enables a user to get his own services which can be used by him or others about the particular area he wants. The user is responsible for the daily transactions of media done by administrator.

2.3 CONSTRAINTS

My project includes login module so that unauthorized members are not accessible to the project. There are only a few entry and exit points in each module so there is no possibility of change to the data by an outside function. This application is critical under some circumstances which includes low memory space of the system. So it is pre-assumed that memory space of the system where this project is going to execute is of higher speed.

There is not use of more size data so high range of RAM is not required on this application. To get result very fast and appropriate not needs to have high speed of RAM. Login and password is used for identification of user and there is no facility for guest.

3. SYSTEM ANALYSIS

In this phase, the analysis of the whole project is done i.e. how the project will be developed, what the whole project will contain and how all the functions that are included in this project will work.

All these things will be visualized, conceptualized and put on the paper work which will take the body of the actual project work. The working of the whole project work will be described by Data Flow Diagrams, Class Diagrams, Use Case Diagrams, etc. so that the prior idea of the actual functionality if the project, the functions of different modules and classes and the user interface of the project can be visualized.

The current scenario of the company was based on the desktop and web application made up for shape file handling and manipulation. All the different kinds of projects that were going on into the company were either the desktop applications or the web applications and were made using the tools like Visual WORDPRESS , C++, VB .NET, C# .NET, Asp. Net, Php etc. This software were providing the functionalities to view the shape files, to create shape files, to perform various operations upon the shape files like zooming, panning, clipping etc., to have layers of more than one shape files at a time etc. As such the current projects dealt with the shape file handling on the web basis.

3.1 STUDY OF CURRENT SYSTEM

In Current System User can also get Shopping information. The Online Perfume Purchase provides totally products less work so there is no wasting of time and saving of manpower and work becomes faster with react and current system not providing option of some questions to help choose best fit fragrance according to your event and moods. We provide quiz facility that help us to you direct choose fragrance The basic functionality of the system user can get all information in my web site.

3.2 PROBLEM AND WEAKNESS OF CURRENT SYSTEM

The major applications that were going on into the organization were not being made by the free tools . So, the organization could not generate the free cost solution for the same structure. The application is also targeted at the mass which can use it to the fullest.

They were not able to get any information about any preparation. So, they used paper maps and other recourses. So that process was time consuming.

Most of websites are built in WORDPRESS so the page loading and refreshment is slowing and take time.

3.3 REQUIREMENT OF A NEW SYSTEM

The new system that was to be built in the organization need to have some creative concept that can help the user in the real manner and the next important thing is that it should give the cost effective solution to the user. Due to the collaborative nature of the application the user can really be an important part of it rather than just using it blindly.

Most of perfume website developed using WORDPRESS so the Page refreshment and page loading is slower and in react it is easy to find because it is component based approach and in react without Page refreshment and page loading data can fast loading and updating. And in my website there is quiz functionality so user easily get perfumes according to event.

My Online Perfume Purchase provides totally paper less work so there is no wasting of time and saving of manpower and work becomes faster. My project will solve all this problems. Unknown user can get all type of information.

3.4 FEASIBILITY STUDY

Market Analysis:

Research the perfume industry, including current trends, market size, and growth projections.

Identify the target audience, their preferences, and purchasing behaviors.

Analyze competitors, their products, pricing strategies, and online presence.

Technical Feasibility:

Assess the feasibility of building a perfume project using React. React is a popular JavaScript library for building user interfaces, but ensure it meets the project's requirements. Evaluate the availability of skilled React developers or the feasibility of training existing team members.

Consider the compatibility of React with other technologies required for the project, such as backend systems, APIs, and databases.

Financial Feasibility:

Estimate the development costs, including salaries of developers, designers, and testers, as well as the cost of software tools and licenses.

Project infrastructure costs, such as hosting, domain registration, and database services.

Marketing and promotion expenses, including advertising, SEO, and social media campaigns.

Maintenance and support costs, such as bug fixes, updates, and ongoing server maintenance. Calculate the total investment required and compare it with the projected revenue to determine financial viability.

Operational Feasibility:

Evaluate the operational aspects of running the perfume project, such as day-to-day management, customer support, and fulfillment processes.

Determine the scalability of the project to accommodate growth in traffic, sales, and product offerings.

Assess the availability of resources and infrastructure required to operate the project efficiently.

Legal and Regulatory Feasibility:

Identify legal and regulatory requirements relevant to the perfume industry, such as product labeling, ingredient regulations, and online sales regulations.

Ensure compliance with data protection laws, such as GDPR or CCPA, if the project collects and processes user data.

Evaluate any intellectual property considerations, such as trademarks or patents related to the perfume brand or product formulations.

3.5 Requirements Validation

Requirements validation is concerned with showing that the requirements actually define the system, which the customer wants. It has much in common with analysis as it is concerned with finding problems with the requirements. Requirements validation is important because errors in a requirements document can lead to extensive rework costs when they are subsequently discovered during development or after the system is in service.

The cost of making a system change resulting from a requirements problem is much

greater than repairing design or coding errors. There are number of requirements validation techniques which can be used in conjunction or individually. Requirement .Validation examines the specification to ensure that all system requirements have been stated unambiguously; that errors have been detected and corrected. Primary requirements validation mechanism is Formal Technical Review.

4. PROJECT MANAGEMENT

- ❖ **Project Planning and Scheduling**
 - Project Development Approach
 - Project Plan
 - Schedule Representation
- ❖ **Risk Management**
 - Risk Identification
 - Risk Analysis
 - Risk Planning
- ❖ **Estimation**
 - Effort Estimation
 - Cost Analysis

4.1 PROJECT PLANING:

Project planning is concerned with identifying and measuring activities, milestone and deliverables produced by project. The effectiveness of the subsequent planning activities is based on the accuracy of this estimation.

- ❖ Scheduling man power and other resources
- ❖ Staff organization and staffing plan
- ❖ Risk identification. Analysis and abatement planning
- ❖ Miscellaneous plans such as quality assurance plan, configuration management plan, etc.

The project plan sets out the resources available to the project, the work breakdown and schedule for carry out the work.

MILESTONES AND DELIVERABLES

Management needs information. As Software is intangible, this information can only be provided as documents that describe the state of the software being developed. Without this information, it is impossible to judge the progress and cost estimates and scheduling cannot be update. When planning a project series of milestones are established.

MILESTONE:

Milestone is an end-point of the software process activity.

- ❖ At each milestone there should be formal output, such as report, that can be represented to the management.
- ❖ Milestone report need not be large document; they are the short report of achievements in software project activity.
- ❖ Milestone represents the end of the distinct, logical state in the project.

DELIVERABLE:

- ❖ Deliverables is a project report that is delivered to the user.
- ❖ Deliverables are delivered to the user at the end of some major project Phase such as specification, design, etc.

ROLES AND RESPONSIBILITIES

Division of Responsibility

| | |
|-----------------|--------|
| Analysis | Ishita |
| Analysis Review | Ishita |
| Design | Ishita |
| Design Review | Ishita |
| Coding | Ishita |
| Testing | Ishita |
| Documentation | Ishita |
| Implementation | Ishita |

Project management involves planning, monitoring and control of people, process and the events that occurs as software evolves from a preliminary concept to an operational implementation. Cost estimation is a related activity that is concerned with estimating the resources to accomplish the project plan. Software project management is an umbrella activity within software engineering. It begins any technical activity is initiated and continues throughout the definition, development and support of computer software.

Project must be organized into effective teams, motivated to do high quality software work and coordinated to achieve effective communication. The product requirement must be communicated customer to develop, partitioned into their constituted parts and position for

work by the software team. The process framework is selected and appropriate software engineering paradigm is applied and set of work, task is chosen to get the work done.

The project must be organized in a manner that enables the software team to succeed. A project management activity encompasses measurement and matrix, estimation, risk analysis, schedules, tracking, and control.

4.2 Project Development Approach and Justification

Software process model is an abstract representation of a software process. Each process model represents a process from a particular perspective so only provides partial information about that process. These generic models are not definitive descriptions of software process. Rather, they are useful abstractions, which can be used to explain different approaches to software developments. For many large systems, of course, there is no single software process that is used. Different processes are used to develop different parts of the system.

Development approach of any project depends on many factors. They are like resources, clear-cut information available in the form of requirements, process maturity, tool maturity, manpower with skills, time duration to develop, complexity and criticality of the project. In this project, spiral model seems to be more suitable for development approach. There are other factor that clearly directs towards Spiral Model.

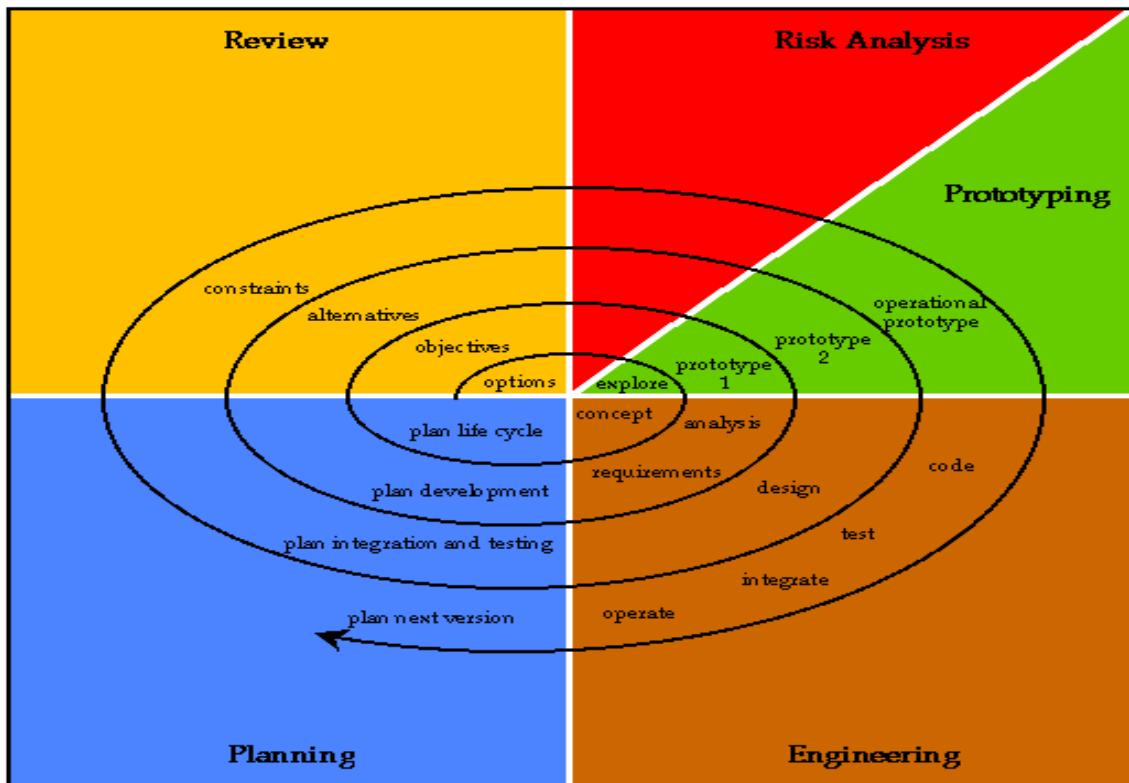


Figure 4.1 Spiral Model

Basically Spiral Model is split into four sectors or quarters as shown.

The first quadrant identifies the objective of the phase and alternatives solutions possible for the phase under consideration. In second phase, alternative solutions are evaluated to select the best solution possible, and for the chosen solution, the potential risks are identified and dealt with by developing an appropriate completion of a software project. Activities during third quadrant consist of developing and verifying the next level of the product. Activities during the fourth quadrant concerns reviewing the results of the stages traversed so far with customer and planning the next iteration around the spiral. With each iteration around the spiral (beginning at the centre and moving outwards), progressively a more complete version of the software gets built. And all the risks are resolved and the software is ready for development.

As we have already mentioned that ONLINE PERFUME PURCHASE is following the spiral approach to develop the whole project and spiral model can be viewed as a Meta Model because it subsumes all the models i.e. waterfall model, iterative model, prototype model, evolutionary model. So at each stage of the project development life cycle it follows

the model which is more appropriate based on the condition met as well as it is also having pre-determined potential risks which is generally resolved as the spiral move outwards from stage of development life cycle into next subsequent stage.

Advantages of the Spiral Model:

- ❖ The spiral model is a realistic approach to the development of large-scale software products because the software evolves as the process progresses. In addition, the developer and the client better understand and react to risks at each evolutionary level.
- ❖ It avoids the pitfalls of existing software models through a risk driven approach.
- ❖ It tries to eliminate errors in the early phases.
- ❖ It can be extended to software maintenance also.
- ❖ It provides mechanism for software quality assurance.
- ❖ It works well for complex, dynamic, and innovative projects.
- ❖ It allows re-evaluation after each phase, which allows changes in user .

Perspectives, technology advance, and financial perspectives. System architecture, report layout design, actual report generation design of ONLINE PERFUME PURCHASE come under this phase and risks associated with it in the form of rejection of ideas or work carried out are also resolved during this phase.

❖ Implementation Phase

Implementation phase includes development of UI components, creating prototypes of each module involved with the proper functioning of all the stated tasks in the design phase as well as actual database design with entry of records are also carried out to check whether any inconsistency or redundancy occurred in the database or fulfilled all the constraints mentioned in the design phase.

❖ Testing Phase

The testing phase ensures whether the produced system meets the specification decided in analysis phase. The uncertainty is removed in the testing. The methods of testing are explained in testing phase. This phase mainly deals with checking the standards of the

system whether they fulfil or not.

4.3 Project Scheduling

Scheduling the project task is an important planning activity. It involves deciding which task would be taken when. Project Guide of Mr.Brijesh Pandey has done these following tasks:

- ❖ Identifying the entire task related to project.
- ❖ Determine the dependency among different activities.
- ❖ Establish the most likely estimates for the time durations necessary to complete the activities.
- ❖ Allocate resources to activities.
- ❖ Plan the starting and ending dates for various activities.
- ❖ Determine the critical path. A critical path is the chain of activities that determines the duration of the project.

4.4 RISK MANAGEMENT:

1) Technology Risks:

- ReactJS Version Compatibility: Ensure that the chosen version of ReactJS is compatible with other libraries, frameworks, and tools used in the project. Upgrading ReactJS later in the project could introduce compatibility issues and require significant refactoring.
- Third-Party Dependencies: Evaluate the stability and reliability of third-party libraries and packages used in the project. Dependence on unstable or poorly maintained packages can lead to security vulnerabilities, compatibility issues, or sudden breaking changes.

2) Resource Risks:

- Skills and Expertise: Assess the team's proficiency in ReactJS development. Inadequate knowledge or experience with ReactJS could lead to inefficient development practices, code quality issues, and increased risk of bugs and errors.

- Team Availability: Consider the availability of team members throughout the project timeline. Unexpected changes in team composition or availability could impact project progress and delivery deadlines.

3) Security Risks:

- Cross-Site Scripting (XSS) and Injection Attacks: ReactJS applications are vulnerable to XSS and injection attacks if proper security measures are not implemented. Failure to sanitize inputs, manage state securely, or protect against malicious scripts can compromise the application's integrity and user data.
- Authentication and Authorization: Implementing secure authentication and authorization mechanisms is crucial to prevent unauthorized access to sensitive data and functionalities within the ReactJS application.

4) Performance Risks:

- Rendering Performance: ReactJS applications may suffer from performance issues if not optimized properly. Inefficient rendering, excessive re-renders, or large component trees can result in sluggish user interfaces and poor user experience.
- Network Performance: Evaluate the impact of network latency and bandwidth constraints on the application's performance, especially when fetching data from external APIs or serving large files.

5) Integration Risks:

- API Integration: Assess the complexity and reliability of integrating with external APIs or backend services. Inadequate error handling, lack of fallback mechanisms, or inconsistent API responses can disrupt the application's functionality.
- Compatibility with Backend Technologies: Ensure compatibility between the ReactJS frontend and the backend technologies (e.g., databases, server frameworks). Incompatibility issues can arise in data transmission formats, authentication methods, or communication protocols.

6) Change Management Risks:

- Scope Creep: Monitor and manage changes to project requirements and scope throughout the development lifecycle. Uncontrolled scope creep can lead to timeline delays, resource overutilization, and increased project costs.
- Version Control and Dependency Management: Implement robust version control practices and dependency management strategies to track changes effectively and mitigate the risk of introducing bugs or regressions.

7) External Risks:

- Regulatory Compliance: Consider regulatory requirements and industry standards applicable to the project, especially regarding data privacy, security, and accessibility. Non-compliance with regulations can result in legal consequences and damage to the project's reputation.
- Economic and Market Risks: Assess external factors such as economic downturns, market fluctuations, or changes in technology trends that could impact project funding, resource availability, or user demand.

8) Testing and Quality Assurance Risks:

- Test Coverage: Ensure comprehensive test coverage for ReactJS components, functionality, and user interactions. Inadequate testing may result in undetected bugs, regressions, or usability issues, leading to post-release defects and user dissatisfaction.
- Performance Testing: Conduct performance testing to identify and address bottlenecks, scalability limitations, and resource usage inefficiencies in the ReactJS application under various load conditions.

9) Deployment and Maintenance Risks:

- Continuous Integration/Continuous Deployment (CI/CD): Implement automated CI/CD pipelines to streamline deployment processes and minimize human error. Manual deployment procedures increase the risk of deployment failures, configuration drift, and downtime.

- Monitoring and Maintenance: Establish monitoring mechanisms to track application performance, detect anomalies, and troubleshoot issues proactively. Regular maintenance and updates are essential to address security vulnerabilities, software bugs, and performance degradation over time.

| Sr No. | Risk Type | Possible Risk |
|--------|----------------|--|
| 1. | Technology | The database used in the system cannot process as many transactions per second as expected. Software components which should be reused contain defects which limit their functionality. |
| 2. | People | It is impossible to recruit staff with the skills required. Key staff is ill and unavailable at critical times. Required training for staff is not available. |
| 3. | Organizational | The organization is restructured so that different management are responsible for the project. Organizational financial problems force reductions in the project Budget. |
| 4. | Estimation | The time required to develop the software is underestimated. The rate of defect repair is underestimated. The size of the software is underestimated. |
| 5. | Tools | The code generated by CASE tools is inefficient. CASE tools cannot be integrated. |
| 6. | Requirement | Changes to requirement which require major design rework are Proposed. Customers Fail to understand the impact of requirement changes. |

4.5 Estimation:

4.5.1 Effort Estimation:

- ❖ “Software project scheduling is an activity that distributes estimated efforts across the planned duration by allocating the effort to specific software engineering tasks.”
 - ❖ Program Evaluation and Review Technique (PERT) and Critical Path Method (CPM) are two most widely used techniques in project management. Historically speaking, PERT and CPM developed independently out of research studies conducted by U.S. Navy and DuPont Company. The PERT was applied to Research and Development tools, while CPM was used to construct the projects.
 - ❖ These two project management & scheduling method that can be applied to software development. Both techniques are driven by information already developed in earlier project planning activities:
 - Estimation of effort.
 - A decomposition of the product function.
 - The selection of appropriate process model and task set.
 - Decomposition of tasks.
 - ❖ There are some differences between PERT and CPM for selecting it as project management technique like.
 - ❖ In PERT total project duration is regarded as a random variable and therefore associated probabilities are calculated to characterize it. PERT is normally used for project involving activities of non-repetitive in nature which time estimates are uncertain. PERT helps in pin pointing critical areas in a project so that necessary adjustment can be made to meet the scheduled completion date of the project. While In CPM duration was known with certainty, therefore it is deterministic approach. It involves repetitive activities.

4.5.2 Cost Analysis:

1. Development Costs:

Development Team: Include the salaries or hourly rates of developers, designers, and testers involved in the project.

Software Tools and Licenses: Cost of any development tools, IDEs, design software, version control systems, etc.

Training: If any team members required training or upskilling in React or related technologies.

2. Infrastructure Costs:

Hosting: Cost of hosting the React application. This could include services like AWS, Azure, Google Cloud, or hosting platforms like Netlify or Vercel.

Database: If the project involves database usage, include the cost of database services like AWS RDS, MongoDB Atlas, or Firebase.

Domain Name: Cost of purchasing and renewing the domain name for the project.

3. Third-Party Services:

APIs: If your project utilizes third-party APIs for features like payment processing, geolocation, etc., include the associated costs.

Analytics and Monitoring Tools: Cost of using tools like Google Analytics, Mixpanel, or Sentry for tracking and monitoring.

4. Marketing and Promotion:

Advertising: Budget allocated for marketing campaigns, social media promotion, influencer collaborations, etc.

SEO: Cost of SEO tools or services to improve the visibility of the project.

5.Maintenance and Support:

Bug Fixes and Updates: Estimate the ongoing cost of fixing bugs and releasing updates.

Customer Support: If the project requires ongoing customer support, include the associated costs.

Server Maintenance: Cost of server maintenance, monitoring, and security updates.

6.Contingency:

Buffer: It's wise to allocate a certain percentage of the total budget as a contingency for unexpected expenses or scope changes.

Total Cost Estimate:

Sum up all the above costs to get the total estimated cost of the React perfume project.

7.ROI Analysis:

Optionally, you may include a section on the expected return on investment (ROI) from the project. This could include revenue projections, cost savings, or other benefits that the project is expected to bring.

5. Diagrams or Data Modelling

5.1.1 Data Flow Diagram

- way to model a real world situation they are the interface between the real world activities and an understanding of how this can be converted into a computer system.
- It is a way of taking the physical view and converting it into a logical view.
- Separating each process into sub processes uncovers more and more detail balancing Process at lower level should have identical data flows if they flow out of a process Modeling Data Stores only use DATA STORES used within this process on the diagram Labels Should carry as much meaning as possible.

The four main elements of DFDs notation:

1. Data Flows, with a label to indicate what data is flowing
2. Processes that handle the data
3. Data stores, within the system (diary, filing cabinet or computer file)
4. Outside entities, outside sources of data

Data Flow is shown by a line with an arrowhead, indicating the direction of the flow of data. Each data flow should be named to indicate what data is being passed. Nouns or adjectives only no verbs are permitted.

- Context - Overview - contains only one process
- Level 1 - Utilizes all four elements
- Level 2 - A breakdown of a level 1 process

- Level 3 - A breakdown of a level 2 process
- There is no rule as to how many levels of DFD that can be used.
- Sequence not important - getting the Process is correct.

- Context or Level 0 - Identifies the system/ boundary/ExternalLinks
- Level 1 - Overview of function
- Level 2 - Breakdown to Understand

The Procedure for Constructing DFD's:

- Draw a document flow diagram of the current situation.
- Draw a systems boundary around the agencies that are part of the system.
- Draw a Context Diagram.
- Identify processes in the system.
- Complete the level 1 Current Physical DFD

➤ DFd Level-0:

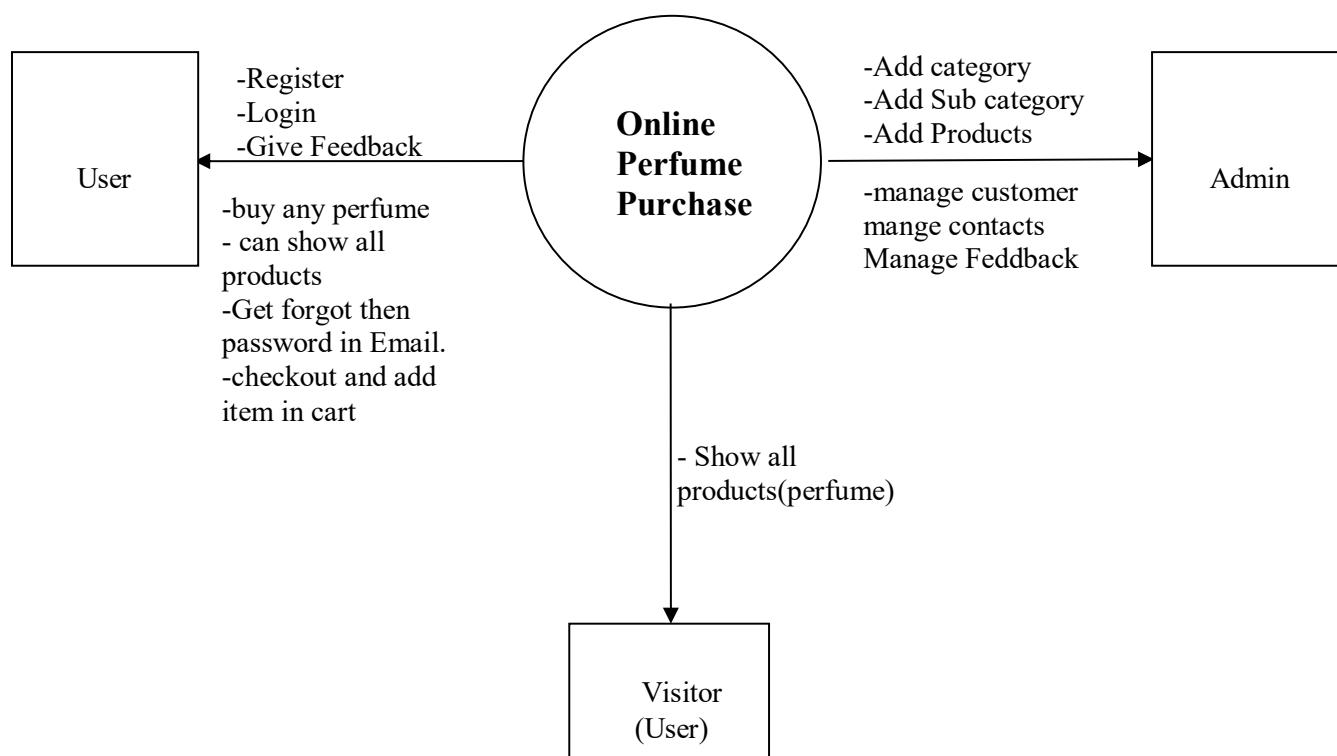
❖ User Characteristics :-

- **Register**
- **Login**
- **Give Feedback**
- **Buy And View Any product**
- **Forgot password**

- **Checkout and Add item in cart**

❖ **Admin Characteristics :-**

- **Add product**
- **Add category**
- **Add subcategory**
- **Manage Customer**
- **Mange Contacts**
- **Mange Feedback**



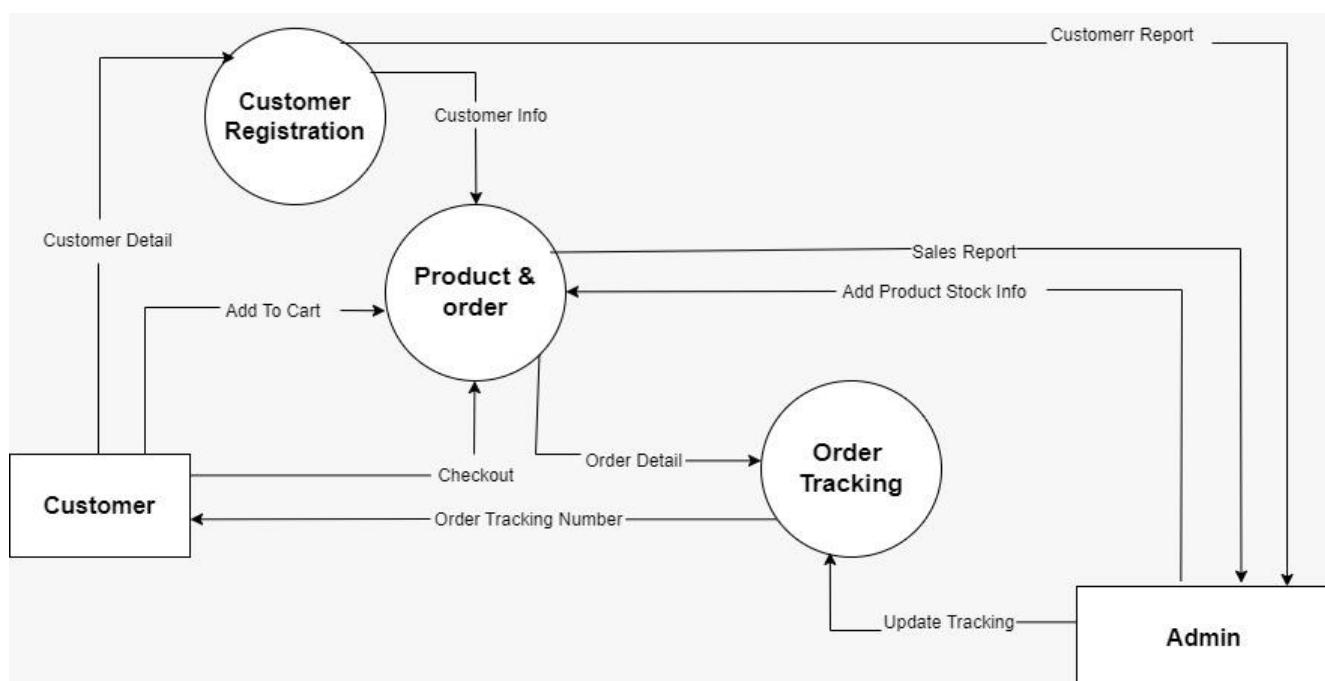
➤ DFD Level – 1 :-

Customer :-

- First Customer Register
- View product
- Search product
- Add To cart
- Checkout
- Order Tracking

➤ Admin :-

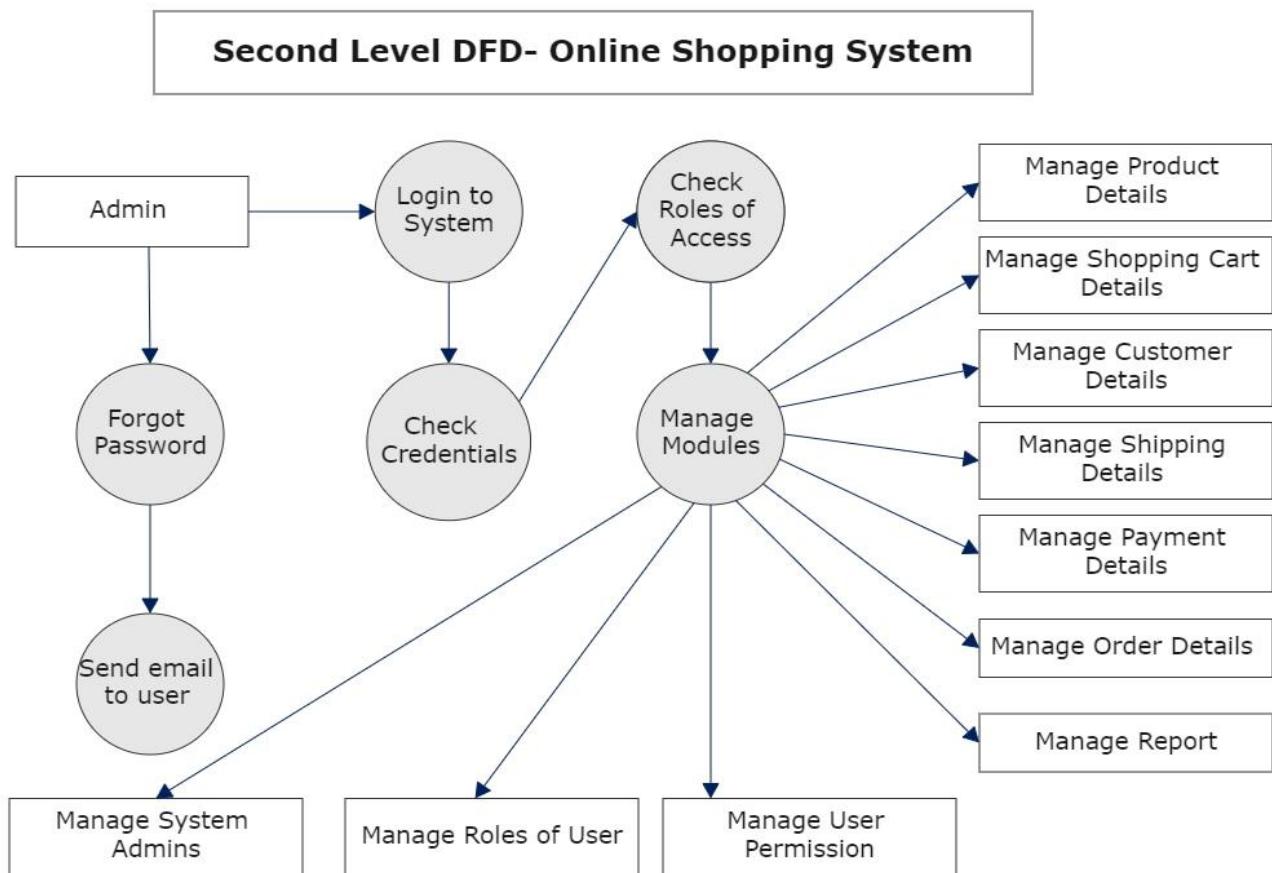
- Customer Info Maintain
- Sales report
- Add product Stock info
- Update tracking
- Maintain Checkout



➤ DFD Level – 2

Admin :-

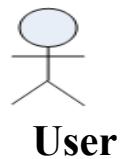
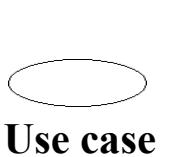
- Login To System
- Maintain user info
- Send email to user on forgetting Password
- Manage Product Details
- Manage Shipping Details
- Manage shopping Cart Details
- Manage Order and Feedback
- Manage User Roles



5.1.2 Use Case Diagram

User And visitor Use case Diagram:-

Symbols used for Use Case Diagram:



➤ **Visitor:-**

➤ Show All Products

➤ Search Product

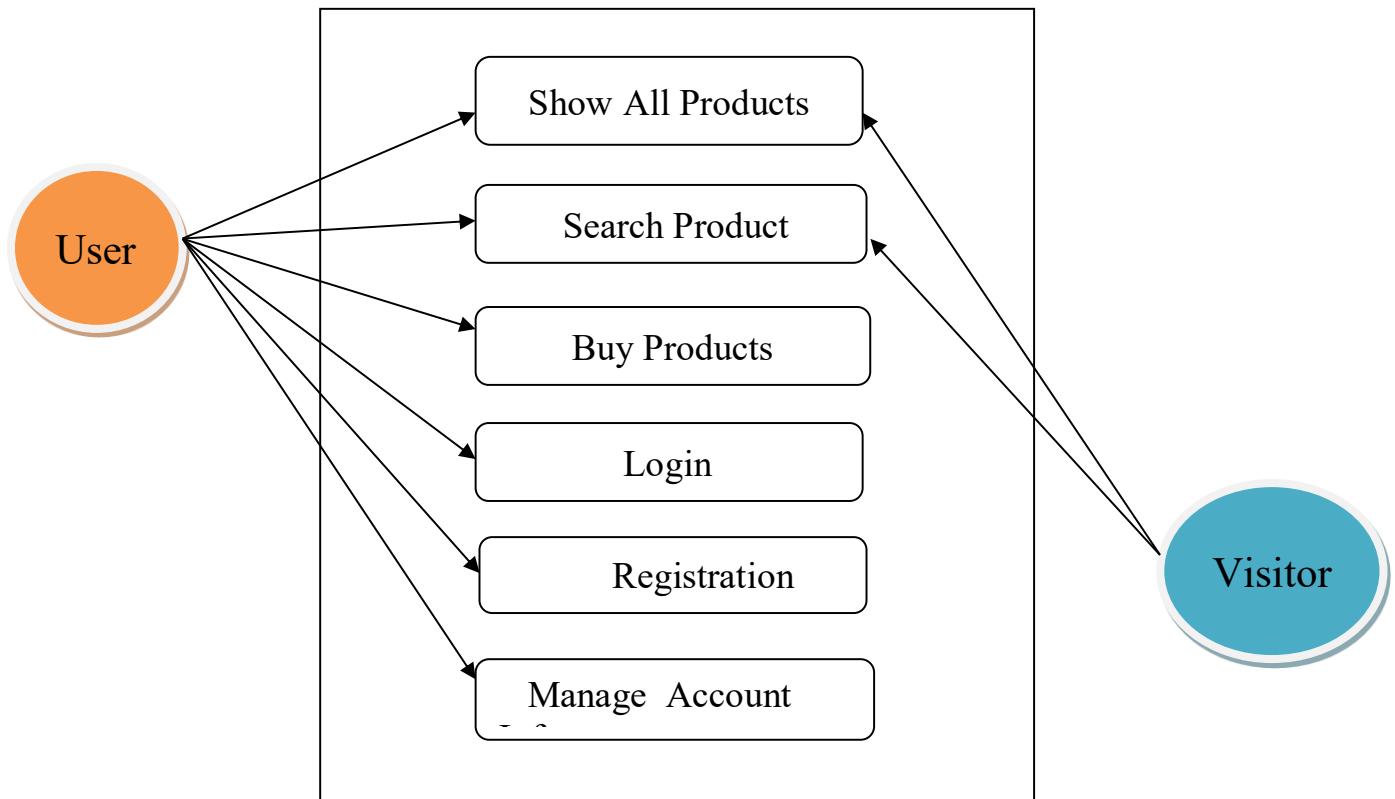
➤ **User :-**

- Login

- Registration

- Show And Search product

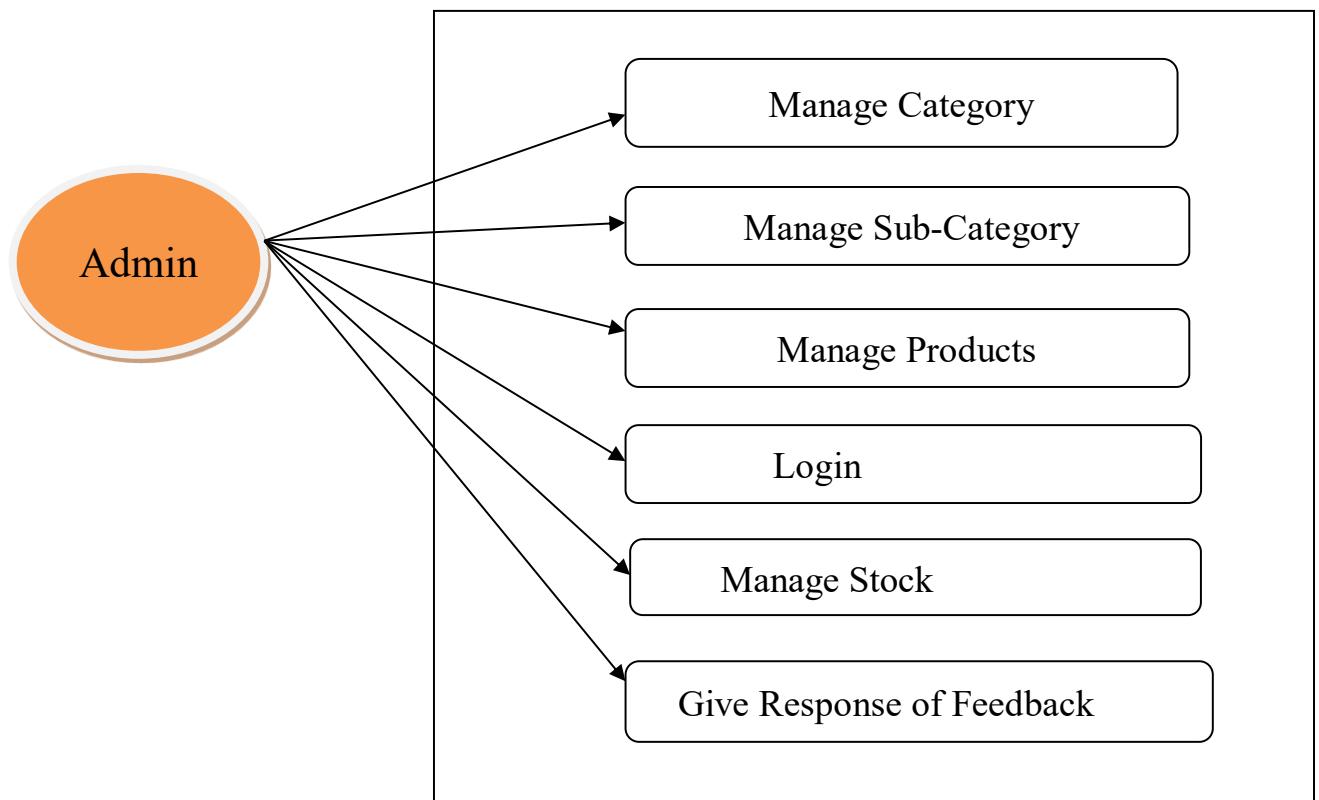
- User Manage their Account



Admin use-case diagram:-

Admin:-

- Manage Sub-Category
- Manage Category
- Manage Product
- Login
- Manage Products
- Manage Stock
- Give Response of Feedback



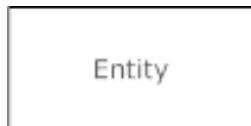
5.1.3 E - R Diagram

➤ Entity Relationship Diagram Notations

❖ Entity

An entity is an object or concept about which you want to store information.

[Learn how to edit text on an entity.](#)



❖ Entity

A weak entity is an entity that must be defined by a foreign key relationship with another entity as it cannot be uniquely identified by its own attributes alone.

[Learn how to edit text on this object.](#)



❖ Key attribute

A key attribute is the unique, distinguishing characteristic of the entity. For example, an employee's social security number might be the employee's key attribute.



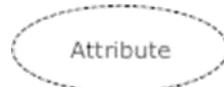
❖ Multivalued attribute

A multivalued attribute can have more than one value. For example, an employee entity can have multiple skill values.



❖ Derived attribute

A derived attribute is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary.



❖ Relationships

Relationships illustrate how two entities share information in the database structure.



❖ Cardinality

Cardinality specifies how many instances of an entity relate to one instance of another entity.

Ordinarily is also closely linked to cardinality. While cardinality specifies the occurrences of a relationship, ordinarily describes the relationship as either mandatory or optional. In other words, cardinality specifies the maximum number of relationships and ordinarily specifies the absolute minimum number of relationships.

| Symbol | Notation | Represents..... |
|--------|-----------|--------------------|
| | Rectangle | An Entity Set |
| | Ellipse | An attribute |
| | Diamond | A relationship set |

| | | |
|---|------|--|
|  | Line | Link between an entity set & attribute, and Link between an entity set & relationship set |
|---|------|--|

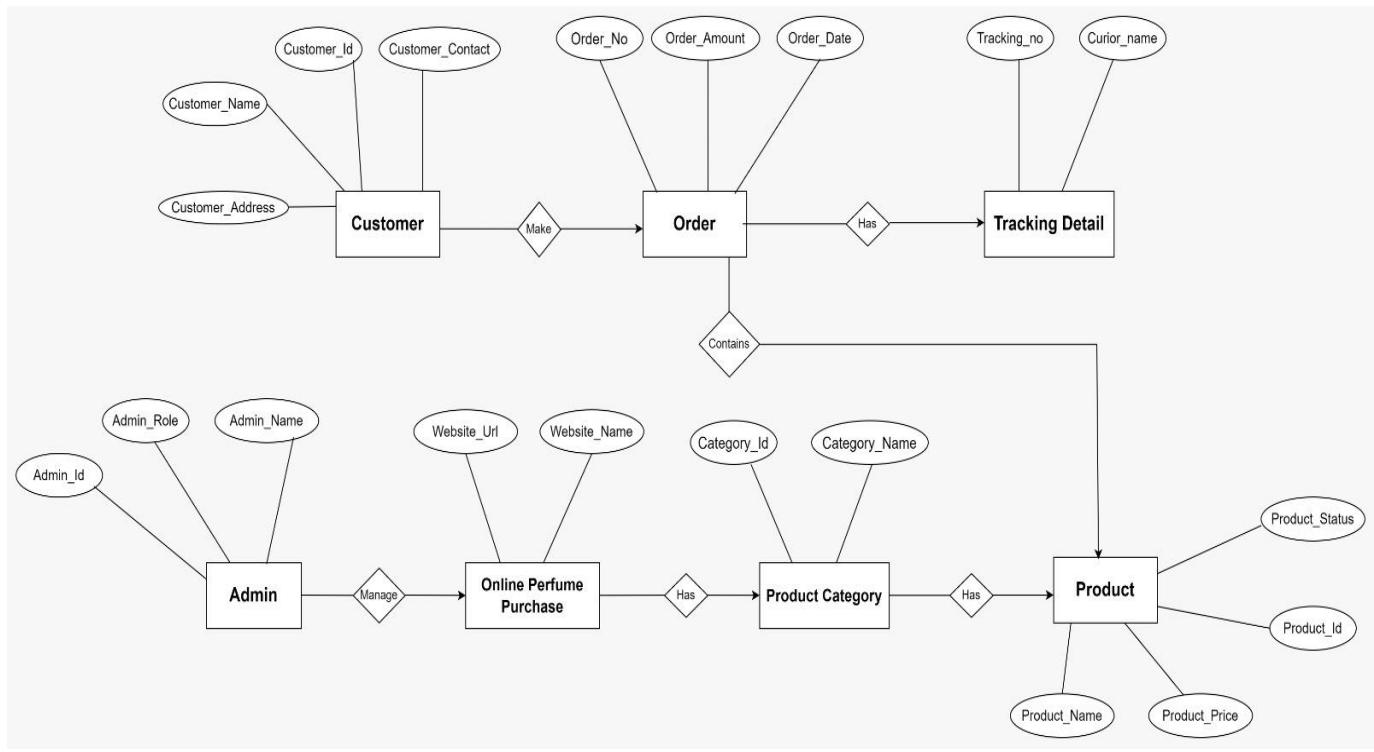
Admin And User :-

- Admin:-
 - Admin_id
 - Admin role
 - Admin_name
- Website url
- Website name

- Product category :-
 - C-Name
 - C-id
- Product :-
 - p-id
 - p-name
 - p-status
 - p-price

- Tracking Detail
 - T-no
 - T-name

- Order
- O-no
- O-date
- o-amount
- Customer
- C-name
- c-id
- c-Contact number
- C-adress



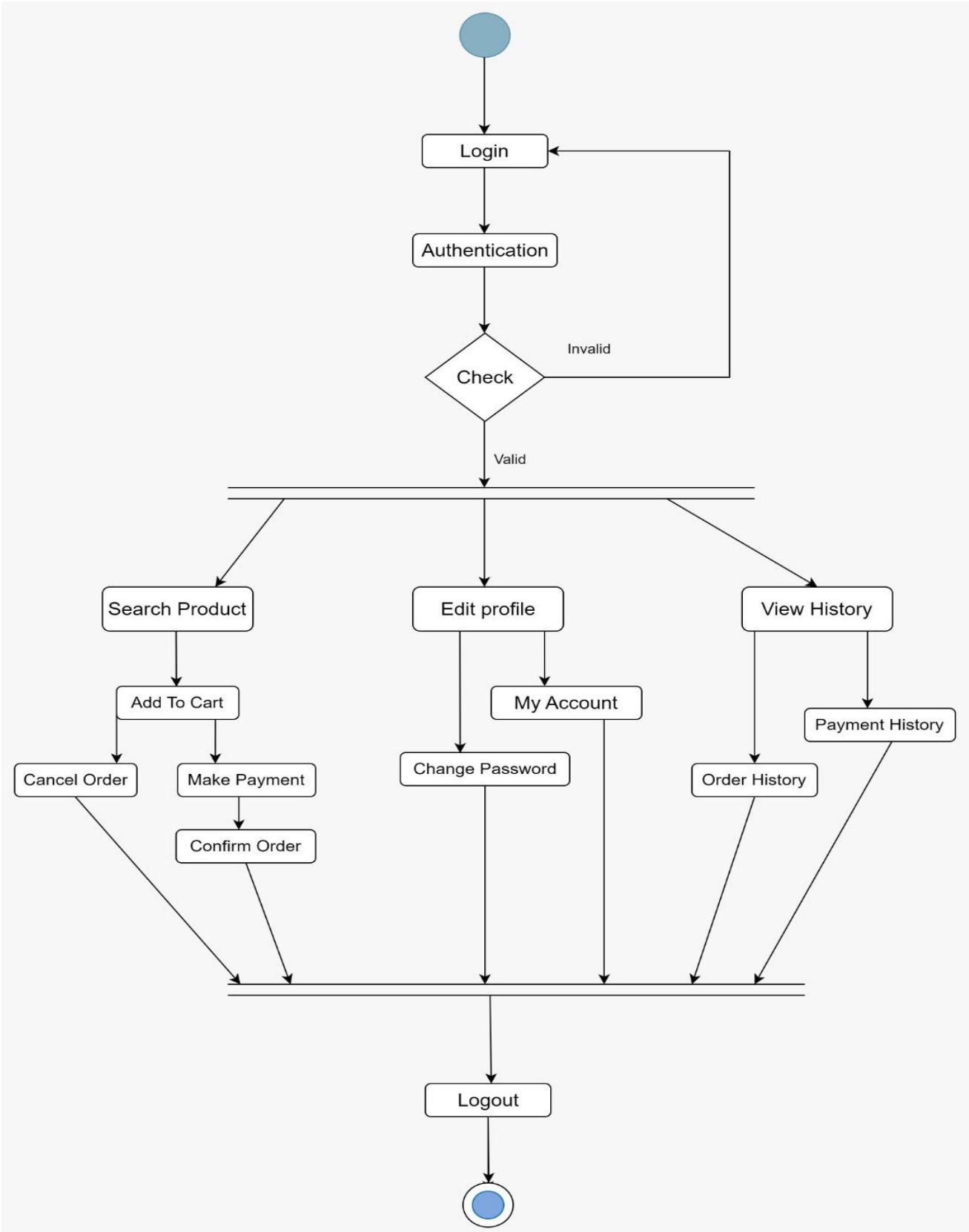
5.1.4 Activity Diagram

Activity user Diagram:-

User :-

- Login
- Authentication check
- Search Product
- Add to Cart
- Cancel order
- Make Payment
- Confirm Order
- Edit Profile
- My Account
- Change Password
- View History
- Payment History
- Order History
- Logout

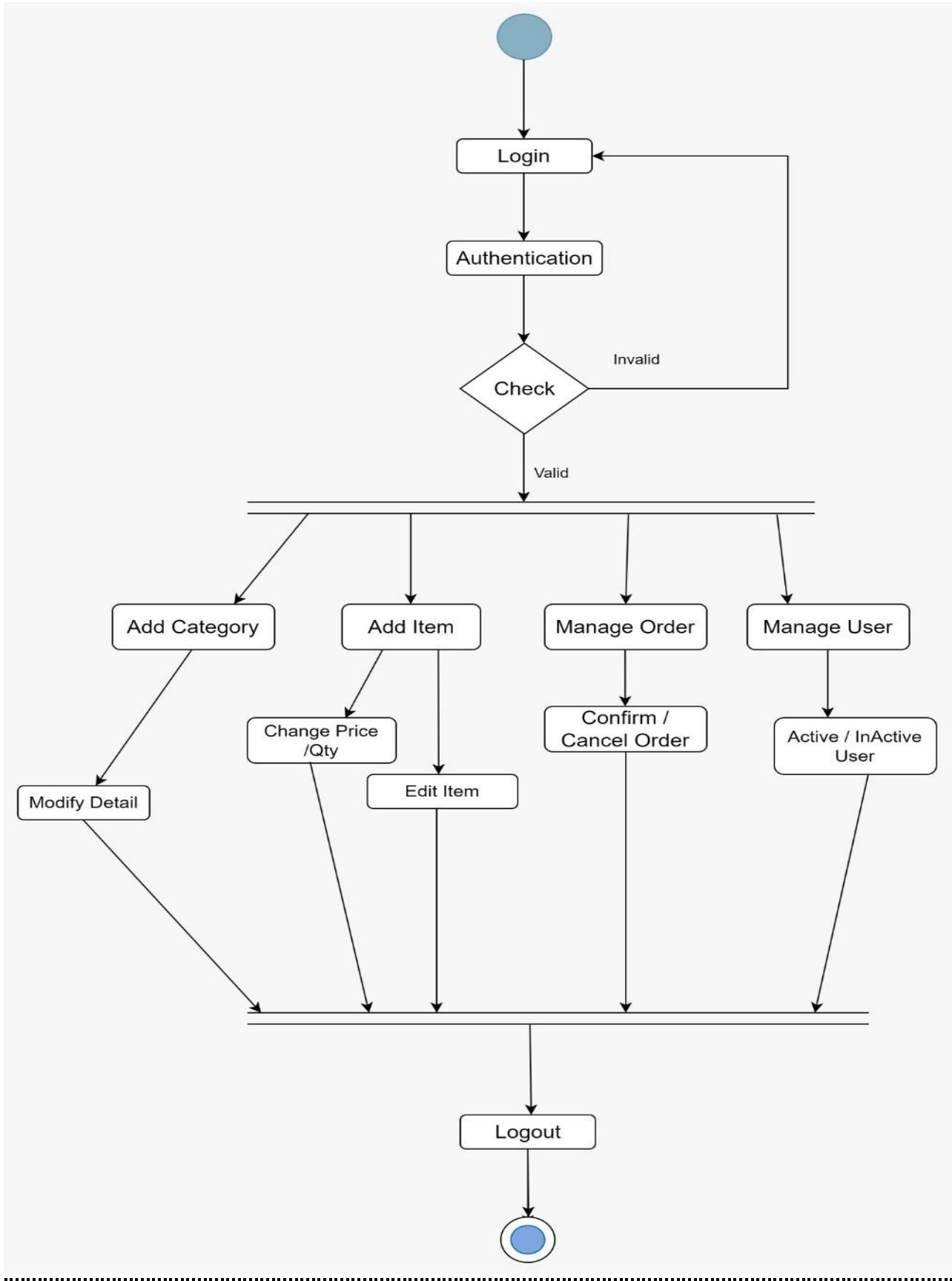
Activity User :-



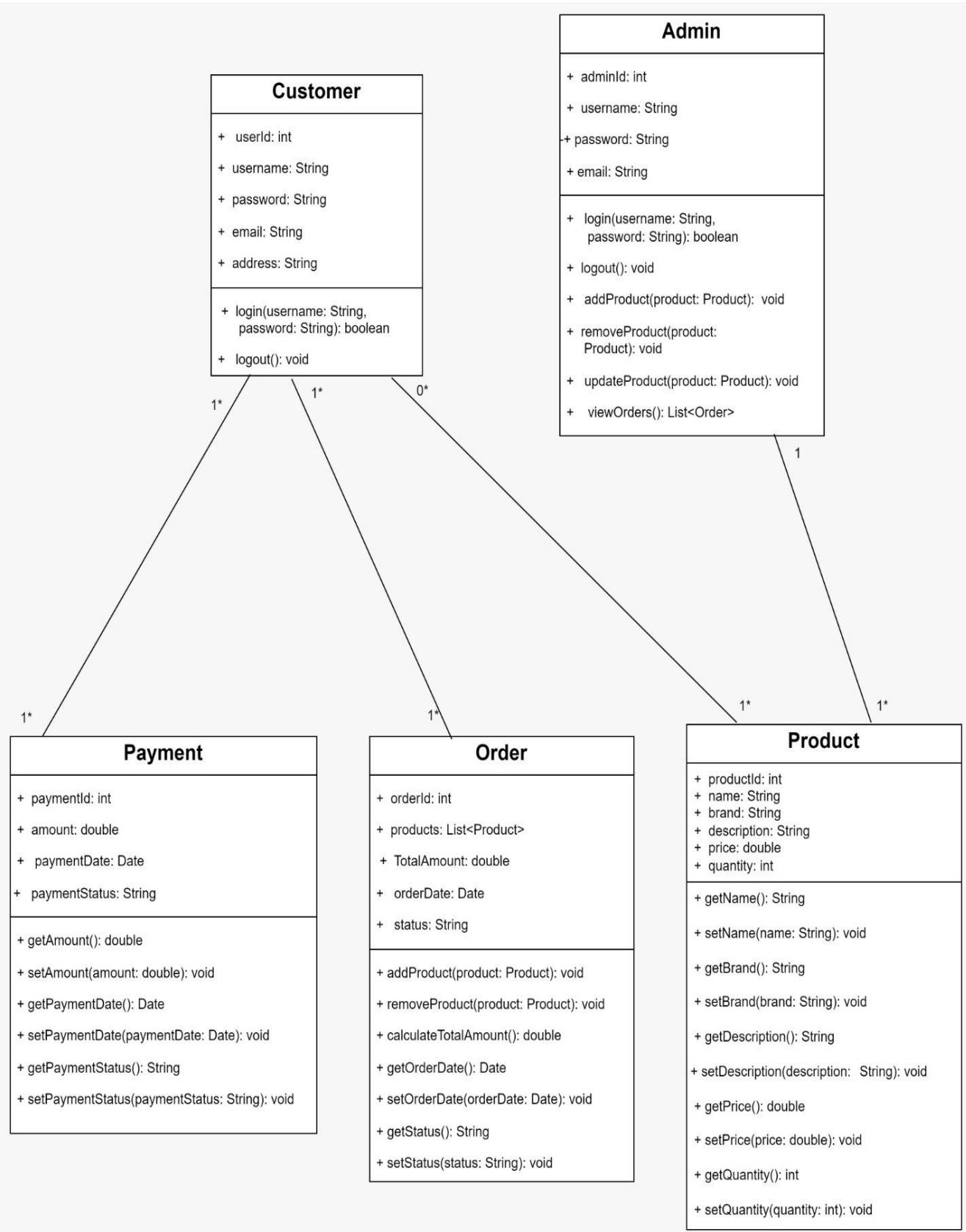
Admin Activity Diagram :-

Admin :-

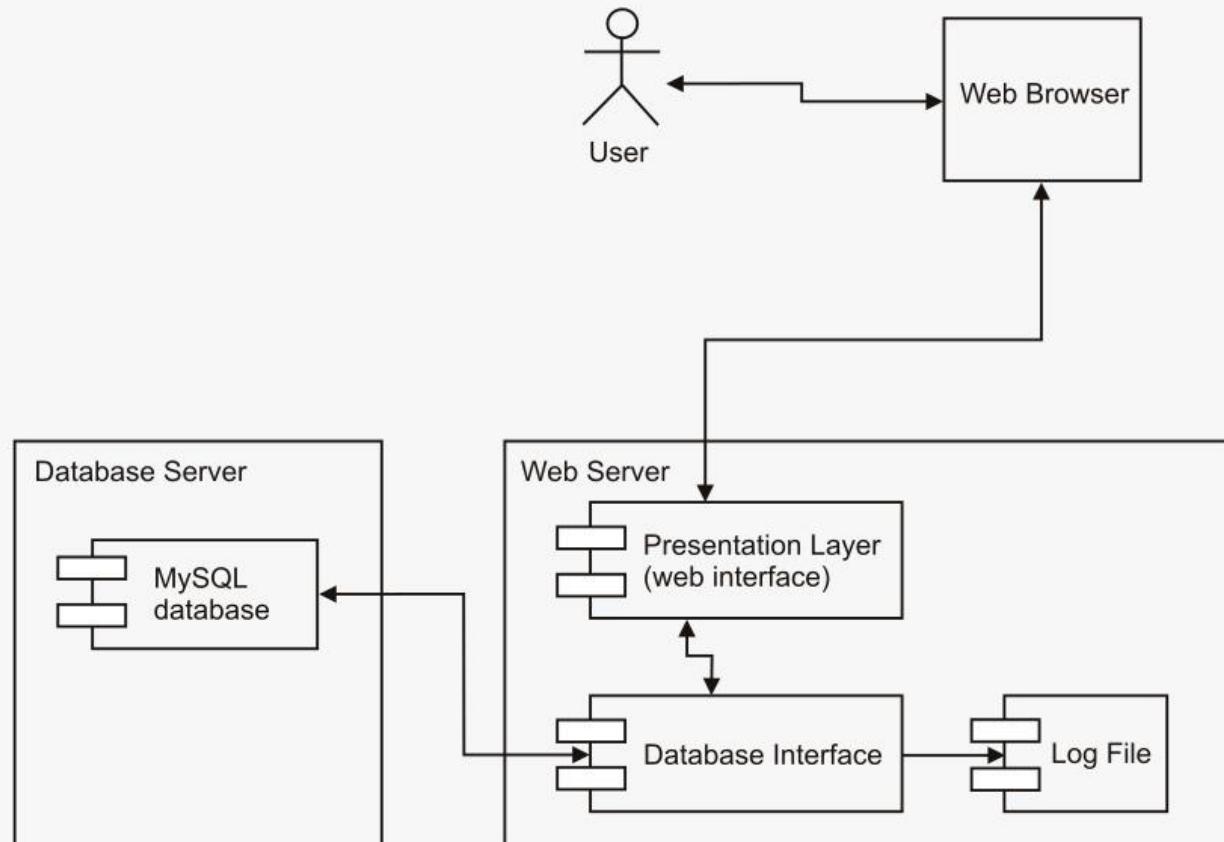
- Login
- Authentication check
- Add Category
 - Modify Detail
- Add item
 - Change price / Qty
 - Edit Item
- Manage Order
 - Confirm/Cancel order
- Manage User
 - Active/Inactive User
- Logout



5.1.5 Class Diagram



5.1.6 Component Diagram



5.2 Data Dictionary:-

1) Register:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action | | | | | |
|--|------------|--------------|-------------------|------------|------|---------|----------------|--------|--|---|--|--|--|
| | rid | int(11) | | | No | None | auto_increment | | | X | | | |
| | image | varchar(255) | latin1_swedish_ci | | No | None | | | | X | | | |
| | firstname | varchar(255) | latin1_swedish_ci | | No | None | | | | X | | | |
| | middlename | varchar(25) | latin1_swedish_ci | | No | None | | | | X | | | |
| | lastname | varchar(255) | latin1_swedish_ci | | No | None | | | | X | | | |
| | email | varchar(255) | latin1_swedish_ci | | No | None | | | | X | | | |
| | password | varchar(255) | latin1_swedish_ci | | No | None | | | | X | | | |
| | repassword | varchar(255) | latin1_swedish_ci | | No | None | | | | X | | | |
| | mobileno | bigint(20) | | | No | None | | | | X | | | |
| | pincode | int(11) | | | No | None | | | | X | | | |
| | status | varchar(255) | latin1_swedish_ci | | No | enable | | | | X | | | |

2) Category:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action |
|--|--------------|--------------|-------------------|------------|------|---------|----------------|---|
| | <u>catid</u> | int(11) | | | No | None | auto_increment |       |
| | catname | varchar(255) | latin1_swedish_ci | | No | None | |       |
| | description | text | latin1_swedish_ci | | No | None | |       |

3) SubCategory:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action |
|--|--------------|--------------|-------------------|------------|------|---------|----------------|---|
| | <u>subid</u> | int(11) | | | No | None | auto_increment |       |
| | catid | int(11) | | | No | None | |       |
| | subcatname | varchar(255) | latin1_swedish_ci | | No | None | |       |
| | description | text | latin1_swedish_ci | | No | None | |       |

4) Contact us:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action | | | | | |
|--|-----------|--------------|-------------------|------------|------|---------|----------------|--------|--|--|--|--|--|
| | cid | int(11) | | | No | None | auto_increment | | | | | | |
| | name | varchar(255) | latin1_swedish_ci | | No | None | | | | | | | |
| | mobilenno | bigint(20) | | | No | None | | | | | | | |
| | email | varchar(255) | latin1_swedish_ci | | No | None | | | | | | | |
| | message | text | latin1_swedish_ci | | No | None | | | | | | | |

5) Feed Back:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action | | | | | |
|--|-----------|--------------|-------------------|------------|------|---------|----------------|--------|--|--|--|--|--|
| | fid | int(11) | | | No | None | auto_increment | | | | | | |
| | name | varchar(255) | latin1_swedish_ci | | No | None | | | | | | | |
| | email | varchar(255) | latin1_swedish_ci | | No | None | | | | | | | |
| | mobilenno | bigint(20) | | | No | None | | | | | | | |
| | message | text | latin1_swedish_ci | | No | None | | | | | | | |

6) Cart:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action |
|--------------------------|----------------|--|-------------------|------------|------|---------|----------------|--------|
| <input type="checkbox"/> | cid | int(11) | | | No | None | auto_increment | |
| <input type="checkbox"/> | uid | int(11) | | | No | None | | |
| <input type="checkbox"/> | pid | int(11) | | | Yes | NULL | | |
| <input type="checkbox"/> | pname | varchar(55) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | color | varchar(55) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | size | varchar(55) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | price | int(11) | | | Yes | NULL | | |
| <input type="checkbox"/> | qty | int(11) | | | Yes | NULL | | |
| <input type="checkbox"/> | subtotal | bigint(11) | | | Yes | NULL | | |
| <input type="checkbox"/> | o_code | varchar(55) | latin1_swedish_ci | | Yes | NULL | | |
| <input type="checkbox"/> | payment_status | enum('pending','paid') | latin1_swedish_ci | | Yes | NULL | | |
| <input type="checkbox"/> | delevery_st | enum('pending','dispatched','delivered') | latin1_swedish_ci | | No | None | | |

7) Admin Login:-

| | Field | Type | Collation | Attributes | Null | Default | Extra | Action |
|--------------------------|-----------|--------------|-------------------|------------|------|---------|----------------|--------|
| <input type="checkbox"/> | adid | int(11) | | | No | None | auto_increment | |
| <input type="checkbox"/> | image | varchar(255) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | firstname | varchar(255) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | lastname | varchar(255) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | email | varchar(255) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | gender | varchar(255) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | password | varchar(255) | latin1_swedish_ci | | No | None | | |
| <input type="checkbox"/> | cpassword | varchar(255) | latin1_swedish_ci | | No | None | | |

6. IMPLEMENTATION PLANNING AND DETAILS

6.1 IMPLEMENTATION ENVIRONMENT :

1. Development Environment Setup:

Node.js and npm or Yarn:

Install Node.js, which includes npm (Node Package Manager) by default, or you can use Yarn.

Node.js allows you to run JavaScript on the server-side and npm or Yarn helps manage project dependencies.

Text Editor or IDE:

Choose a text editor or integrated development environment (IDE) such as Visual Studio Code, Sublime Text, or Atom for writing code.

Git:

Install Git for version control management. Git allows you to track changes, collaborate with others, and deploy your code.

2. Project Structure:

Create React App:

Use Create React App to set up a new React project quickly. It provides a pre-configured build setup with no need for complex configuration.

Run `npx create-react-app my-app` to create a new React project named `my-app`.

Folder Structure:

After creating the app, you'll have a basic folder structure with directories like `src` (for source code), `public` (for static assets), and `node_modules` (for installed dependencies).

3. Dependencies:

React and ReactDOM:

These are the core dependencies for building React applications. They enable you to create and render React components.

Additional Libraries:

Depending on your project requirements, you may need additional libraries like React Router for routing, Axios for HTTP requests, Redux for state management, etc.

Install dependencies using npm or Yarn. For example, `npm install react-router-dom`.

4. Coding:

Components:

Create React components to structure your UI. Components can be functional (using hooks) or class-based.

Organize components based on their functionality and reuse them across your application.

State Management:

Manage state using React's built-in state or use external libraries like Redux or Context API for more complex state management.

Styling:

Style your components using CSS, CSS-in-JS libraries like styled-components, or preprocessors like Sass.

Routing:

Implement routing using React Router to navigate between different views/pages within your application.

5. Testing:

Unit Testing:

Write unit tests using tools like Jest and React Testing Library to ensure individual components work as expected.

Integration Testing:

Test interactions between components and external dependencies.

6. Deployment:

Build:

Before deploying, build your React app by running npm run build. This creates an optimized production build in the build folder.

Hosting:

Choose a hosting provider like Netlify, Vercel, or Firebase Hosting.

Upload your build files to the hosting provider using CLI tools or through their web interface.

Configure domain settings if you have a custom domain.

Continuous Integration/Continuous Deployment (CI/CD):

Set up CI/CD pipelines using services like GitHub Actions or Travis CI for automated testing and deployment.

7. Maintenance

Dependency Management:

Regularly update dependencies to keep your project secure and up-to-date.

Use tools like npm audit to identify and fix vulnerabilities.

Performance Optimization:

Monitor and optimize the performance of your React app using tools like Lighthouse or Google PageSpeed Insights.

Implement performance best practices such as code splitting, lazy loading, and image optimization.

Bug Fixes and Updates:

Address bugs reported by users and release updates with new features or improvements regularly.

By following these steps, we can maintain React website in our environment.

7.1 TESTING

❖ Testing Methods

- **Black – Box Testing**
- **White -Box Testing**

TESTING METHODS:

Software Testing involves executing an implementation of the software with test data and examining the outputs of the software and its operational behavior to check that it is performing as required.

7.1.1 Black-Box Testing:

In Black-Box Testing also called as Functional Testing, Developers are concerned about the output of the module and software, i.e. whether the software gives proper output as per the requirements or not.

The program just gets a certain input and its functionality is examined by observing the output.

In my project I have done the testing as follows:

Homepage Navigation:

- navigation links (e.g., Home, About Us, Products, Contact) direct to the Layout Homepage navbar .
- the logo redirects to the homepage.
- Media Query that responsiveness across different devices.

Product Listing:

- Confirm all listed products are displayed correctly with images, names, and prices.
- clicking on a product takes the user to the correct product detail page.
- sorting and filtering functionalities available like range between this to this and the category selected .

Product Detail Page:

- all details (name, price, description, images) are displayed accurately.
- Checkout Process:
- Test the user registration/login process (if applicable).
- Verify that the checkout process is smooth and intuitive.
- Test different payment methods (PayPal, etc.) if available.

Login Form:

- the contact form is functional and submits data correctly on admin side mange contacts.
- Validation for required fields and proper error messages like please enter name enter valid email address. Email validations used and in admin-side I used email-js api fetching that when user create account it show message on user email using email-js in React.
- users receive a confirmation message after submitting the form and then it navigates user to the homepage.

Contact us Form:

User can enter there username , email and message and submit and this data store and shown in admin side dashboard. And if user can submit without submitting data it can show validation errors likewise please enter valid email, name etc.

User Account:

- users can update their profile information and password.

Admin side testing :-

- admin can manage contact us page that user can add message for inquiry and delete that message and reply to that customer.

- Admin can manage the login form details of each customer through api calling and delete also.
- Admin can add dynamically add product manage product delete and update product and display in user-panel dynamically. Same for category of each product manage by admin dynamically.

7.1.2 White-Box Testing:

White Box Testing is also called ‘Glass Box’ or ‘Structural’ testing. The intention in white box testing is to ensure that all possible feasible flows of control paths through a subprogram are traversed while the software is under test.

❖ Component Testing:

each React component individually render correctly and behave as expected. And navigates properly..
props are passed correctly and that components update appropriately in response to state changes.

❖ State Management Testing:

Test state management libraries (e.g., Redux) to ensure that state updates are handled correctly.

Check that actions and reducers work as intended.

❖ Integration Testing:

Test the interactions between different components to ensure they work together seamlessly.

Verify that data flows correctly between parent and child components.

❖ API Testing:

API endpoints used by the application are correctly fetch data from the server and store in admin side.

requests are sent correctly and responses are parsed and handled appropriately.

❖ Form Validation Testing:

form components to ensure that input validation is working correctly.
that error messages I used sweet-alert library for react message error message box are

displayed when expected and that form submission behaves as expected.

Performance Testing:

Performance testing is designed to test the runtime performance of the system within the context of the system. These tests were performed module level as well as system level. Individual modules were tested for required performance.

- ❖ In performance testing we counted the processing time and response of operation.
- ❖ We also checked out the total execution time for intersection file creation.

8. SCREEN SHOTS:-

8.1.1 ADMIN SIDE SCREEN SHOTS:

1) Admin Dashboard :-

The screenshot shows the Ishita Dashboard Admin Setting. On the left sidebar, there are links for Manage Customers, Add Category, Add Products, Manage Feedbacks, Manage Contacts, Manage reviews, and Manage Orders. A red 'Logout ?' button is at the bottom. The main area has a header 'Dashboard Admin Setting' with three notification icons. Below it are six colored boxes showing counts: Total Customers (5), Total Reviews (2), Total Profile (25), Total Menus (5), Total Contacts (5), and Total Customers (5). There's a section 'We are on' with links for Facebook, Twitter, and Google. A pie chart titled 'My Daily Activities' shows percentages for different categories. The footer contains copyright information and three notification icons.

2) Manage Customer :-

The screenshot shows the 'Manage All Customers' page. The left sidebar is identical to the Admin Dashboard. The main area has a header 'Manage All Customers' and a table with columns: #, Email, Address, City, ZipCode, and Action. The table data is as follows:

| # | Email | Address | City | ZipCode | Action |
|------|----------------------------|------------------|-----------|---------|--------|
| a96c | ishitaspatel2802@gmail.com | Akashar Parisar | vadodra | 310005 | |
| 8ca0 | ishitajaviya2001@gmail.com | prahlad nagar | ahmedabad | 300006 | |
| 9c49 | ishitajaviya2001@gmail.com | isckon temple | vadodra | 310006 | |
| 0d81 | ishitaspatel2802@gmail.com | 150 ring road hi | rajkot | 360005 | |

The footer contains copyright information and three notification icons.

3) Manage Contacts :-

| # | Name | Email | Message | Action |
|------|---------------|----------------------------|--|--------|
| 1 | ishita javiya | ishitaspatel2802@gmail.com | hi i want information regarding perfumes | |
| 8ec4 | maya savseta | mayasavseta@gmai.com | hi i want information regarding first rain perfume | |
| 7c59 | Ankita Patel | ankitapatel@gmail.com | Hi i want Shipping info acrosss india do u shipped or not ?? | |
| bc86 | gargee Adroja | gargeeadroja@gmail.com | Can I get my order gift wrapped? | |
| 874f | Diya patel | diyapatel1234@gmail.com | How long will it take to get my orders? | |

4) Add Category :-

5) Manage Category :-

Ishita Dashboard

Dashboard Admin Setting

Manage All Category

| # | categoryname | addedate |
|------|---------------------|------------|
| 0c13 | above beyond attar | 2024-04-26 |
| 99de | first rain perfume | 2024-04-18 |
| 79eb | Forest Rain Perfume | 2024-04-18 |
| 0aee | Perfume | 2024-04-01 |
| f5ad | Fragrance | 2024-04-01 |

Export in CSV

Logout ?

6) Delete Category :-

Ishita Dashboard

Dashboard Admin Setting

Manage All Category

| # |
|------|
| 0c13 |
| 79eb |
| 0aee |
| f5ad |

Export in CSV

Logout ?

7) Update Category :-

Ishita Dashboard Admin Setting

Update Your Category

Add Category Name

Add Date

Update product **Reset**

8) Add Product :-

Ishita Dashboard Admin Setting

Add Product

Select Product Category *

Photo Forest Rain.webp

Descriptions

Forest Rain Perfume

Old price *

New price *

Add product **Reset**

9) Manage Product :-

Ishita Dashboard Admin Setting

Manage All Products

| # | productcategory | photo | description | old price | new price |
|------|------------------|--|---------------------|-----------|-----------|
| 975a | firstrainperfume | C:\fakepath\WhatsApp Image 2024-03-23 at 1:46:47 AM.jpeg | sdfas | 789 | 567 |
| 3484 | firstrainperfume | C:\fakepath\Forest Rain.webp | Forest Rain Perfume | 3214 | 3000 |

Logout ?

10) Update Product :-

Ishita Dashboard Admin Setting

Manage All Products

| # | productcategory | photo | old price | new price |
|------|------------------|--|-----------|-----------|
| 975a | firstrainperfume | C:\fakepath\WhatsApp Image 2024-03-23 at 1:46:47 AM.jpeg | 3200 | 2800 |
| 3484 | firstrainperfume | C:\fakepath\Forest Rain.webp | 3214 | 2800 |

Good job!

Thanks for Update New product data!

OK

Logout ?

11) Delete Product :-

The screenshot shows the Ishita Dashboard interface. On the left sidebar, there is a navigation menu with options like Manage Customers, Add Category, Add Products, etc. The main content area is titled "Manage All Products". A modal window is open in the center, displaying a large red "X" icon and the message "Oh! Product data deleted successfully!". Below the message is an "OK" button. In the background, a table lists products with columns for #, product name, old price, and new price. One row shows a product with ID 3484, name "firstrain", old price 3214, and new price 2800.

12) Manage Feedback :-

The screenshot shows the Ishita Dashboard interface. The left sidebar has a similar navigation menu. The main content area is titled "Add Clients Reviews". It contains several input fields: "Name" (with a placeholder "Name"), "Photo" (with a "Choose File" button and a message "No file chosen"), "Email" (with a placeholder "name@example.com"), "Mobile" (with a placeholder "Phone *"), and a "Add Message" text area. At the bottom are two buttons: "Add Reviews" (in a dark blue box) and "Reset".

13) Manage Contacts :-

The screenshot shows a web application interface for managing contacts. On the left is a sidebar with a logo and links: 'Ishita Dashboard', 'Manage Customers', 'Add Category', 'Add Products', 'Manage Feedbacks', 'Manage Contacts', 'Manage reviews', and 'Manage Orders'. The main content area has a header 'Dashboard Admin Setting' with notification icons (99+). Below is a section titled 'Manage Contacts' with a table:

| # | Name | Email | Message | Action |
|------|---------------|----------------------------|--|--------|
| 1 | ishita javiya | ishitaspatel2802@gmail.com | hi i want information regarding perfumes | |
| 8ec4 | maya savseta | mayasavseta@gmail.com | hi i want information regarding first rain perfume | |
| 7c59 | Ankita Patel | ankitapatel@gmail.com | Hi i want Shipping info acrosss india do u shipped or not ?? | |
| bc86 | gargee Adroja | gargeeadroja@gmail.com | Can I get my order gift wrapped? | |
| 874f | Diya patel | diyapatel1234@gmail.com | How long will it take to get my orders? | |

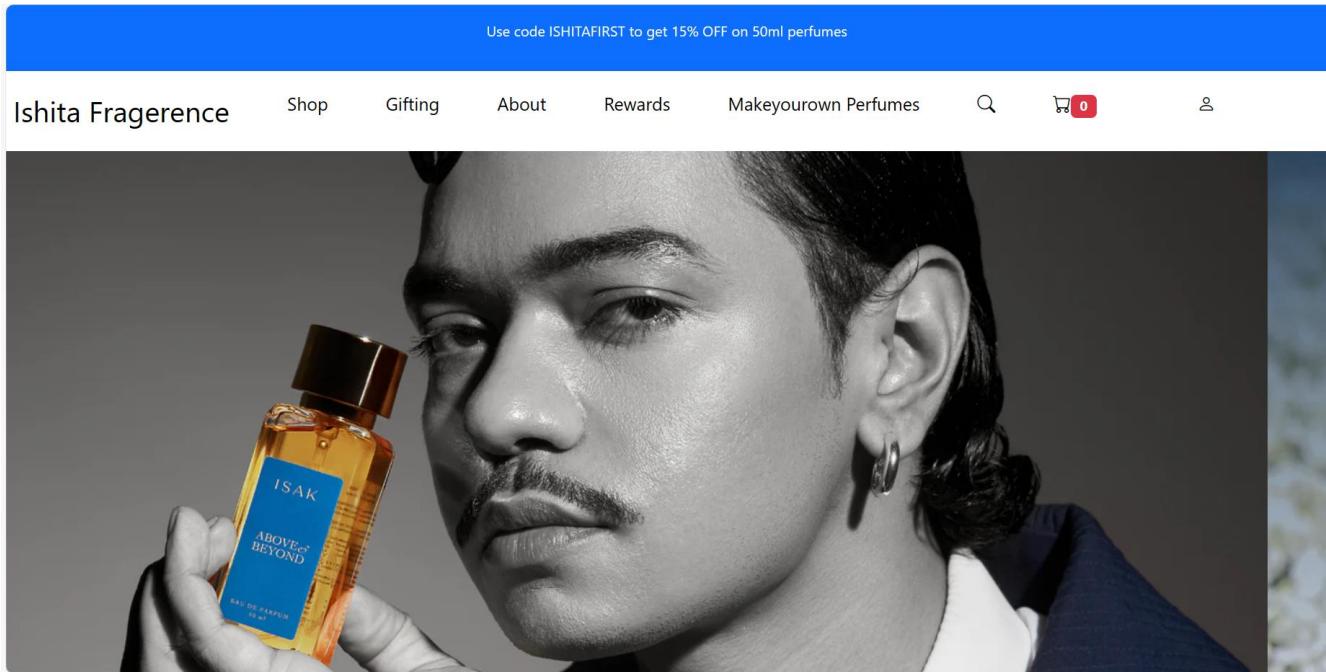
Below the table is a section titled 'Customer Traffic'.

14) Delete Contacts :-

The screenshot shows the same web application interface. A modal dialog box is centered over the 'Manage Contacts' table. The dialog contains a large red 'X' icon, the word 'Oh!', and the message 'Contact data deleted successfully!'. At the bottom right is a blue 'OK' button. The background table remains visible.

8.1.2 Client or User side screen shots:

1.) Home Page :-



2) Spray Product :-

AMBIENT SPRAYS

NEW LAUNCH

| | | | |
|--|---|---|--|
|  <p>FRESH WAVES AIR FRESHENER RS. 390.00</p> <p>VIEW PRODUCTS</p> |  <p>CITRUS BURST AIR FRESHENER RS. 390.00</p> <p>VIEW PRODUCTS</p> |  <p>GREEN WOODS AIR FRESHENER RS. 390.00</p> <p>VIEW PRODUCTS</p> |  <p>ROSE WONDER AIR FRESHENER RS. 390.00</p> <p>VIEW PRODUCTS</p> |
|--|---|---|--|

3) Shop By Collection :-

SHOP BY COLLECTION



FIRST RAIN COLLECTION
Rs. 2900.00

[VIEW PRODUCTS](#)



THE SIGNATURE COLLECTION
Rs. 2900.00

[VIEW PRODUCTS](#)



ROYAL AWADH COLLECTION
Rs. 2900.00

[VIEW PRODUCTS](#)



LEGACY X BACARDI
Rs. 390.00

[VIEW PRODUCTS](#)

4) Footer :-

About Isak

We are a fragrance house dedicated to the creation of unrivaled niche Indian artisanal fragrances of rare notes and blends, backed with the family experience of over a century and a half.

Less Fake, More Real, Less Frill, More Essence. Less Everybody, More You.

Information

[Shipping Policy](#)
[Refund Policy](#)
[Privacy Policy](#)
[Terms of Service](#)
[Contact Us](#)
[Careers](#)

About Us

[Our Story](#)
[Our Stockist](#)
[Isak Rewards](#)
[FAQ](#)
[Offers](#)

Newsletter

Subscribe to receive updates, access to exclusive deals, and more.

[SUBSCRIBE](#)

5) Our Story :-

Use code ISHITAFIRST to get 15% OFF on 50ml perfumes

Ishita Fragerence
Shop
Gifting
About
Rewards
Makeyourown Perfumes
🔍

👤

OUR STORY

The Founder



The Founder & CEO of ISHITA Fragrances - Ishita Javvi as a perfumer & serial entrepreneur to understand the new world as well as extensive interest and research in aroma customization started from Switzerland, France, and India.

"Having grown up around fragrances I have always been drawn to them and as I ventured towards exploring it more, I came to realize that Indian perfumery has its own unique charm. There are a lot of people who appreciate it but know little about the art. Thus, through ISHITA my intention is to talk more about our roots, heritage and the charm in Indian perfumery and introduce it to the rest of the world to be mesmerized with." - Ishita Javvi.

6) FAQ :-

Ishita Fragerence

Shop

Gifting

About

Rewards

Makeyourown Perfumes



NEED HELP?

FREQUENTLY ASKED QUESTIONS

I want a long lasting perfume. Please suggest.

All our perfumes are long lasting. We suggest you select the fragrance based on the perfume family preference.

How long do you take to deliver.

It depends on your pincode/ delivery location. Orders to be delivered in India will take 2- 5 business days to arrive. Overseas deliveries can take anywhere from 7-16 days. Delivery details will be provided in your confirmation email.

what are your delivery charges.

We charge INR 60 for domestic orders below INR 1000. For order above INR 1000 we do not charge any shipping charges domestically. For international order, the shipping charges are based on the order and the country.

What is perfume ?

7) Perfumes Product :-

localhost:3000/collection/perfumes

AVAILABILITY

PRICE

PRODUCT TYPE

MORE FILTERS

Above&Beyond

Crimson Oudh

Cosmic Dance

Add to Cart

Add to Cart

Add to Cart

ISAK Edge of Heaven

ISAK Sunshine of All

ISAK Top of the World

8) Gifting :-

The screenshot shows a web application for gifting perfumes. On the left, there are four filter dropdowns: Availability, Price, Product Type, and More Filters. Below these are three main product cards:

- Discovery's pack of perfumes**: An image of a white box containing several small perfume bottles. **Add to Cart** button.
- Crimson Oudh**: An image of a rectangular bottle of perfume with a gold cap. **Add to Cart** button.
- Cosmic Dance**: An image of a rectangular bottle of perfume with a gold cap, surrounded by orange slices and green moss. **Add to Cart** button.

Below these are three smaller, partially visible images of perfumes in various settings.

9) Rewards :-

The screenshot shows a rewards program landing page with a dark blue background and decorative floral patterns.

Join and Earn Rewards

Win Coins for every spend to redeem exclusive Rewards and Discounts

[Sign Up and Get Free Coins](#) [Sign In](#)

How it Works?

Win Coins for every spend to redeem exclusive Rewards and Discounts.

1 Sign Up
Create an account on our store

2 Earn Coins
Earn coins for shopping

3 Redeem
Redeem coins for discounts

10) Make Your Own Perfume Product :-



MAKE YOUR OWN PERFUME KIT (LAB IN A BOX)

★★★★★

ISAK labs bring you a unique box to stimulate your olfactory senses and 'Make your own Perfume'. This 'lab in a box' is a wonderful way to understand the science and art behind perfumery through a direct hands-on experience with perfumery ingredients and tools. This box will help you explore the basics in perfumery in a fun and simple way. All of this, from the comfort of your homes.

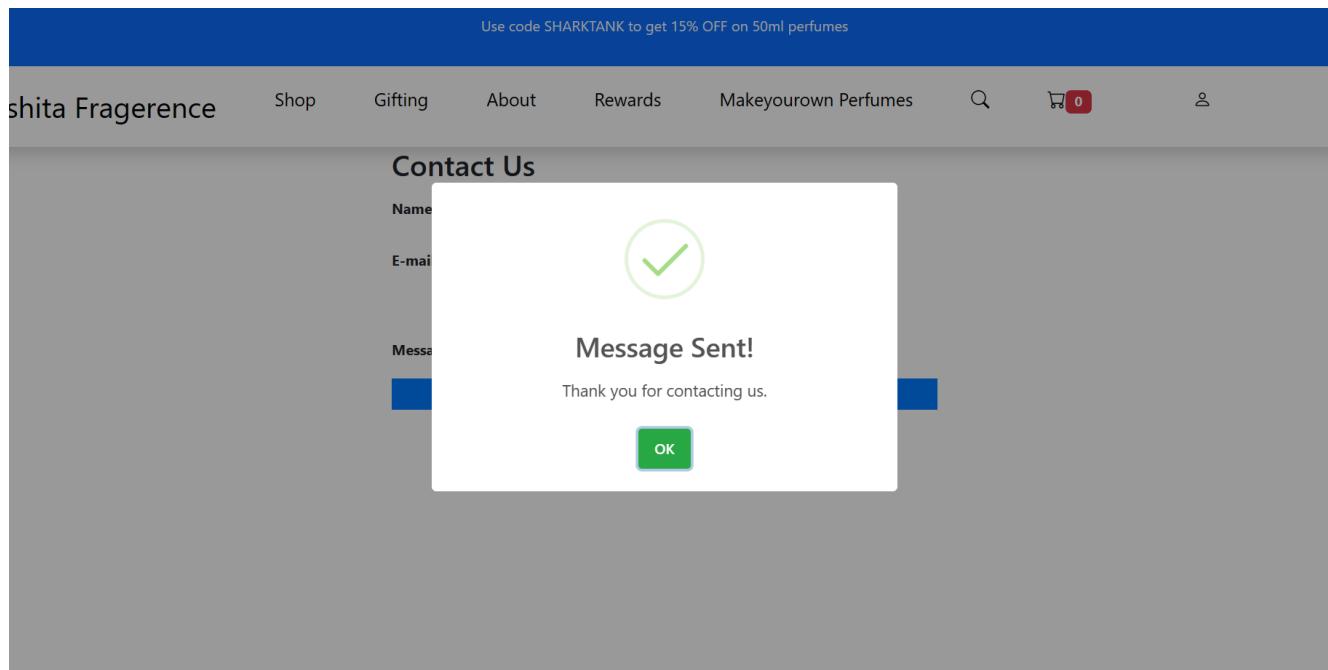
The ISAK Perfumery kit consists of everything you need. You can make upto 20 perfumes.

There are 12 ingredients/bases ready for perfumery along with an ISAK Lab Workbook with notes, formula sheets and space to take down your personal notes. The detailed instructions and steps will guide you through the process helping you explore your creativity while mesmerising you through the creation of your unique bespoke perfume.

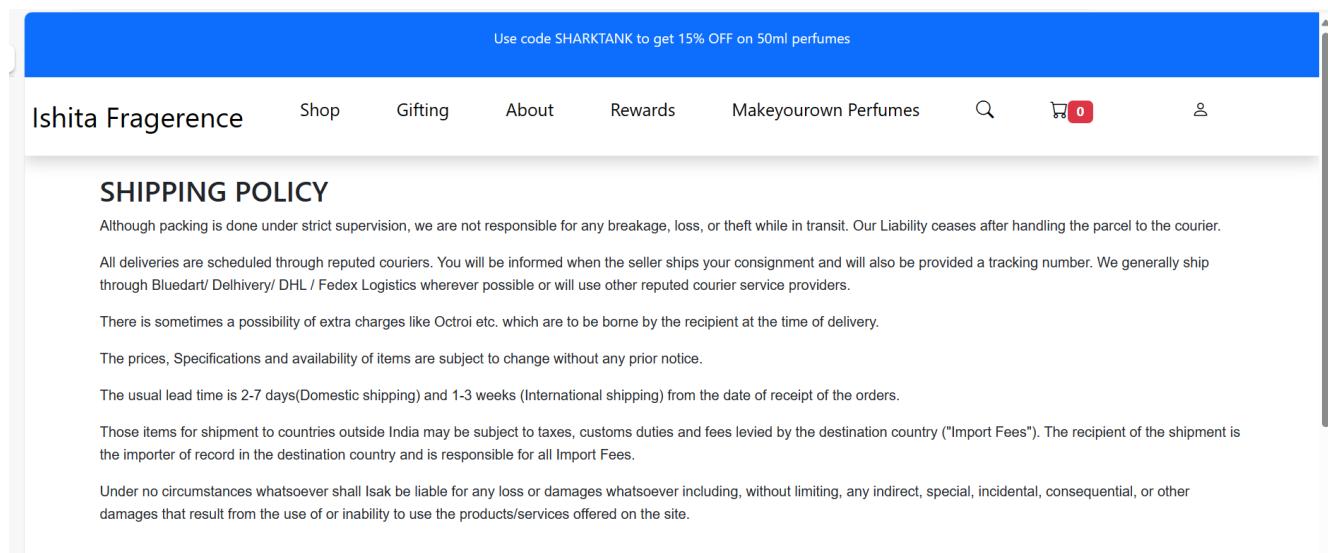
The kit is perfect for perfumery enthusiasts, experimenters, rule-

11) User Login :-

12) Contact Us Form :-



13) shipping Policy Instructions:-



9 LIMITATIONS AND FUTURE ENHANCEMENT

Limitations:

ONLINE PERFUME PURCHASE is a web-based system. At first time, there are some limitations of each and every system which can be removed in future. The few limitations of our project are as follows:

- ❖ The presentation tier and business logic tier are required to be deployed within the same server computer. For better scalability it is required that these two tiers run on different server computers.
- ❖ SEO Challenges:
- ❖ Single-page applications (SPAs) built with React might face difficulties with search engine optimization (SEO) since search engine crawlers may not execute JavaScript, leading to poor search engine rankings.
- ❖ Implementing server-side rendering (SSR) or prerendering techniques can mitigate this issue but adds complexity to the development process.
- ❖ There is no provision to inform the administrator automatically about the error.
- ❖ Power failure breaks the continuity of the application.
- ❖ Should have knowledge about computers and internet.

Regulatory Compliance: Selling perfumes online may require compliance with various regulations, such as labeling laws, shipping restrictions (due to the flammable nature of some perfumes), and fragrance ingredient disclosure requirements. Meeting these regulations can add complexity and potentially limit the types of products you can offer.

Product Representation: Communicating the scent of perfumes through online descriptions and images can be challenging. While you can provide detailed descriptions and high-quality images, they may not fully capture the nuances of a fragrance. This limitation can lead to mismatches between customer expectations and the actual product received.

Returns and Exchanges: Perfume preferences are highly subjective, and what smells good to one person may not appeal to another. As a result, online perfume retailers often face higher rates of returns and exchanges compared to other products. Managing returns and exchanges can be costly and time-consuming.

Shipping Restrictions: Shipping perfumes internationally or to certain locations may be restricted due to regulations or carrier policies. This limitation can impact your ability to reach a global audience or fulfill orders in specific regions.

Brand Reputation: Perfume is a highly competitive industry with many established brands. Building a reputable online perfume website and gaining customer trust can be challenging, especially if you're competing with well-known brands or retailers.

Cost of Marketing: Promoting an online perfume website and driving traffic to your store can be costly, especially considering the competitive landscape and the need to stand out among established brands. Investing in effective marketing strategies, such as social media advertising, influencer partnerships, and search engine optimization, is essential but can strain your budget.

Technology Limitations: Building and maintaining an e-commerce website capable of handling online perfume sales requires robust technology infrastructure, including secure payment processing, inventory management, and customer support systems. Ensuring your website can handle high traffic volumes and provide a seamless shopping experience is crucial but may require significant investment in technology and resources.

Future Enhancements:

Some Of The Features Like Quiz Or Question Answer Is Not Done. I Am Working On It. Here is always a scope for enhancement in any system, especially in the ever changing world of computers. The **ONLINE PERFUME PURCHASE** can be modified according to the future requirements and the advancement of the technology. Below mentioned are some of the changes that are possible in the future, to increase the efficiency and adaptability of the system:

- ❖ In future, I have to develop proper payment gateway.
- ❖ Existing system has all basic functionalities. Quiz functionality is pending through which user easily answer to some questions and then according to their answer we show similar products.
- ❖ In future I build this website proper mern stack fullstack web – based
- ❖ Try to make as more user friendly as possible.

10.CONCLUSION & DISCUSSION

CONCLUSION AND DISCUSSION:

- ❖ As we worked in a team we learnt good team management skills and importance Of Team Work.
- ❖ Through this project we also learnt how to manage Time and to get things done within time.
- ❖ We learned many Technology and Tools while working on the project.
- ❖ Working under a corporate environment was also a learning experience where we tried to follow the discipline and rules laid by the organization. Overall this project is the foundation for all our future endeavors and we feel satisfied with the efforts that we have put in.
- ❖ As I only Learn Front-end But in future in would like to know Backend Node.js, Express.js.
- ❖ Overall this project is the foundation for all our future endeavors and we feel Satisfied with the efforts that we have put in.. So , this project is useful for any company. It is based on Organization structure.

11. BIBLIOGRAPHY

- <https://react.dev>
- <https://www.youtube.com>
- <https://www.google.com>
- <https://www.w3schools.com>