A Project Report on

**JKG Fashion**

Submitted for the partial fulfillment of Project in

Bachelor Of Computer Application (Semester V)

**-: Submitted To:-**



**Department Of Computer Science,**

**Harivandana College, Rajkot.**

**-: Affiliated To :-**

**Saurashtra University, Rajkot**

**-: Submitted By :-**

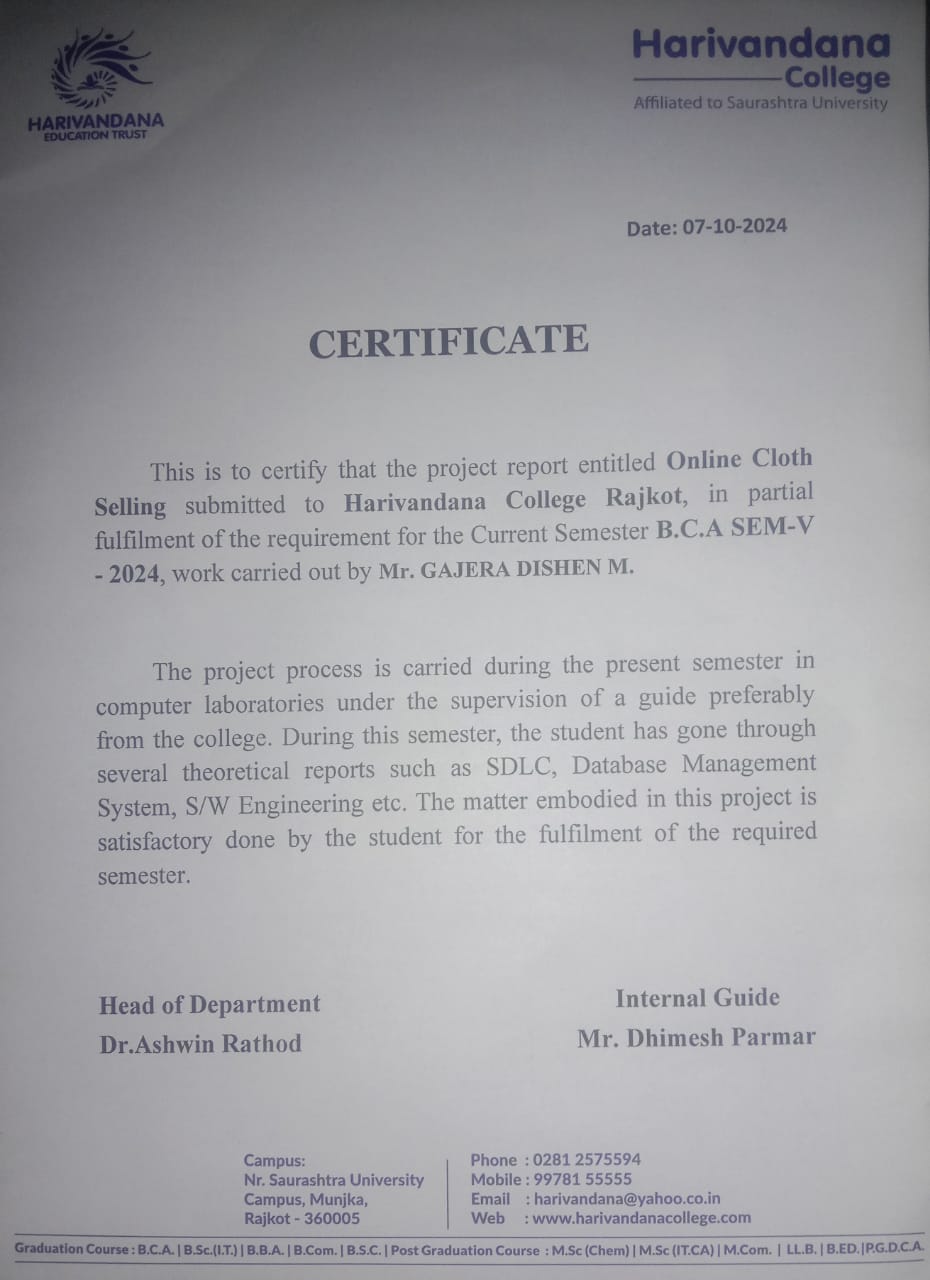
**Dishen M Gajera**

**(Enroll no: 003303221629)**

**-: Under the Guidance of :-**

**Dr. Ashwin Rathod (Professor & Head)**

**(Dr).Dhimesh Parmar (Project in charge)**



## **Acknowledgement**

* At first, I praise to “Harivandana College” who give me Opportunities, Capability and Energy to Complete this Project work.
* Provide us a lot of support and guidance from the Start to End of the project Development.
* We are very Thankful to the project coordinator of Dr. Ashwin Rathod and (Dr.)Dhimesh Parmar of Harivandana College.
* I would like to thanks to all of our friends and those who helped, inspired and gave us support at different stages in different moment in my project.
* Again, also thanks to the Almighty for helping in successfully ending this project work.

**Project Profile**

|  |  |
| --- | --- |
| **Developed At** | Harivandana College  Rajko**t.** |
| **Submitted To** | Harivandana College  Department of computer  Science |
| **Software Required** | Visual Studio, MySQLi |
| **Documentation Tool** | Microsoft Word, Power Point |
| **Main Activity** | Admin Panel,  Client Panel,  Registration Page,  Login Page,  Forgot Password Page  Category, Product  Detail Pages |
| **IDE** | Visual Studio Code |
| **Guided by** | Dr.Ashwin Rathod,Prof:  (Dr).Dhimesh Parmar |
| **Developed By** | Dishen Gajera, GR NO: 3482 |

## **Abstract**

* The Website is user friendly simple, fast and cost – effective, for personal use and makes the data processing very fast.

* It’s a full–featured website and Online Cloth shopping.

* Shopping Cart System is the Simple shopping solution. It’s a full–featured website and shopping cart system.

* The basic concept of the web application is to allow the Users to see virtually using the internet and allow users to see to the information of cloths and buy the Different type of cloths.
* Index

|  |  |  |
| --- | --- | --- |
| **NO** | **Subject Title** | **Page no.** |
| **1** | **Introduction**   * Project Summery with Details * Objectives/Aim/Vision/Scope * Scope * Study Of Existing System * Proposed System * Project Boundary * Duration | **9** |
| **2** | **Literature Review**   * Technology Used * Feature of MySQLi * Connectivity * Localization * Client and Tools * Why to Use MySQLi * phpMyAdmin * Hardware Requirement * Software Requirement | **11** |
| **3** | **System Analysis**   * Study of Current System * Requirement of new System * Overview of proposed System | **16** |
| **4** | **Project Management**   * Project Planning and scheduling * Phases * Project Planning | **20** |
|  | * Risk Management * Risk Identification |  |
| **5** | **User Case**  ➢ Administrator Function  ➢ User Function | **22** |
| **6** | **Data Dictionary**  **JKG Fashion**   * Admin * User * Cloth catalogue * Customer Feedback | **23** |
| **7** | **Data Flow Diagram**   * **Both Side**   + User Case   + Client-Side User Case   + Admin Side User Case * **DFD Diagram**   + DFD 0 Level   + DFD Level 1 Client Side   + DFD Level 1 Admin Side | **28** |
| **8** | **CFD (Control Flow Diagram)**   * CFD from Admin Side * CFD from User Side | **33** |
| **9** | **Implementation Planning and Details** | **35** |
| **10** | **Testing**   * Static Testing * Dynamic Testing * Black Box Testing * White Box Testing | **36** |
| **11** | **Screen Shot** | **41** |
| **12** | **Limitation and Future Enhancement** | **51** |

## **1.Introduction**

* The Project JKG Fashion has been developed on HTML 5, PHP, CSS 3, JavaScript, Bootstrap 5 and MySQLI.
* The Web Portal will have an online interface in the form of an Ecommerce Website that will allow users to see details about clothes and Buy it’s.
* There are two types of users available in the Project.
  1. Customer Side
  2. Admin Side
* First one is **Customer** and second one is **Admin**.
* Customers or users have limited access right to access the system while the admin users have control over the system.
* I have Used PHP for business logic.
* MySQLi as a Database,
* HTML 5 for Structure Designing,
* CSS 3 And Bootstrap 5 for Web Page formatting,
* JavaScript for Validation and Animation.
* I have also attached the project screenshots of this project look into it.

## **Scope of Project**

You can order any Clothes at Home.

* Go to Web Application add to cart any item and buy them.

* Add your item in cart.

* Then Checkout your order.

## **Purpose**

* Main Purpose is customer can easily buy And order any Clothes at Home.

* People can get Clothe related info.

* User can get easily Clothes for Category wise.
* It is customer friendly Website.

**2.Literature Serve**

**2.1) Technology Used:**

**Overview of the Technology:**

**Font-End: -** Html 5, CSS 3, Bootstrap 5 JavaScript.

* **HTML 5: -** HTML is use to create and save Web Document. E.g., Notepad/Notepad++, Visual Studio Code.

Html Stands for “Hypertext Mark-up Language”. It can be assisted by technologies such as Cascading Style Sheet (CSS) and Scripting language such as JavaScript.

* **CSS 3: -** (Cascading Style Sheets) Create attractive Layout.

Using CSS, you can control of the text, the Style of Fonts, the Spacing between Paragraphs, how Columns are sized and layout, what Background Images or colors are used**,** layout designs, Variations in display for different devices and Screen sizes as well as variety of other Effects.

* **JavaScript: -** It is a Programming Language, commonly use with Web browsers. JavaScriptis a dynamic Computer Programming Language. It is lightweight and most commonly used as part of web pages, whose implementations allow client-side Script to interact with the user and make dynamic Pages. It is an Interpreted Programming Language with object-oriented capabilities.
* **Bootstrap 5: -** Bootstrap is a free, open source front-end development framework for the creation of websites and web apps. Designed to enable responsive development of mobile-first websites, Bootstrap provides a collection of syntax for template designs.
* **Back-End: -** PHP, MySQLi.

* **PHP: -**Hypertext Pre-processor (PHP) is a technology that allows Software developers to create dynamically generated web page, In HTML, XML or other document type, as per Client request. PHP is Open-Source Software.

PHP is a web scripting language; this language is used for developed it project. Many files are created in PHP file and use with CSS and complete a design of home page and other page. PHP in a many classes and session are create and use for some output.

However, PHP is the best known for its database interfacing capabilities. With PHP you can establish a database connection to any of standard database servers. Update the content of a database, even manipulate a particular database schema. The results of queries are easily converted into a valid HTML that is sent back to the client.

* **MySQLi Server: -** MySQLi is an Open Source, SQL Relational Database Management System (RDBMS) that is Free for many uses. The MySQLi Extension(MySQL improved) is a relation database Driver used in the PHP Scripting Language to Provide an interface with MySQL databases. There are three main API options when considering Connecting to a MySQL database Server: PHP’s MySQL Extension. PHP Data Object (PDO).

* The data in MySQLi is stored in tables. A table is a Collection of related data, and it consists of columns and rows, Databases are useful when storing information categorically.
* There is for instance, a function library for Manipulating MySQLi Database, Informix Database and much more.

There are Several different Steps that execute Scripts:

* + - * **Hostname: -** The name of the host to which to establish the connection. In our case it is the “**localhost**” because both Servers. The Web Server and the MySQLi Server are running on the same machine.

* + - * **Username: -** The name of a user that has privileges to Manipulate the Test Database that we Created. Default is “**root**”.

* + - * **Password: -** The Valid Password of the User.

**2.2) Features of MySQLi:**

**Internals and portability:**

* + - Written in C and C++.
    - Tested with a board range of different compilers.
    - Works on many different platforms.
    - Uses multi-layered server design with independent modules Security .
    - A privileges and password System that is very flexible, secure and that enables host-based Verification.
    - Password Security by encryption of all Password traffic when you connect to a server.

**2.3) Connectivity:**

**Client can connect to MySQLi Server using Several Protocols:**

* + - Client can connect using TCP/IP sockets on any platform.
    - On Windows Systems in the NT family (NT, 2000, XP, 2003, or Vista), Client can connect using named pipes if the server is started with the enable-named-pipe option. In MySQL 4.1 and higher, Windows servers also support Shared-Memory connections if started with the shared-memory option. Client can connect through shared memory by using the protocol memory option.

**2.4) Localization:**

➢ The Server can Provide error message to Clients in many Languages.

**2.5) Client and Tools:**

* + - MySQLi includes several Client and utility Programs. These include both command-line Programs such as **MySQL dump** and **mysqladmin,** and graphical Programs such as MySQL Workbench.
    - MySQLi server has built-in support for SQL statements to check, optimize, and repair tables. These statements are available from the command line through the **MySQL check** Client.
    - MySQLi Programs can be invoked with the help of Option to obtain online assistance.

**2.6) Why to Use MySQLi:**

* + - * Leading Open-Source RDBMS
      * Ease of use – No frills
      * Fast
      * Robust
      * Security
      * Multiple OS support
      * Free
      * Technical Support
      * Support Large Database -up to 50 million rows, file size limits up to 8 million TB.

**2.7) phpMyAdmin:**

* + - * phpMyAdmin is a Free software tool written in PHP, intended to handle the administration of MySQLI over the Web. phpMyAdmin supports a wide range of operations on MySQLI and Maria DB.
      * Frequently used operations (managing, databases, tables, columns, relations, indexes, users, permissions, etc.) can be performed via the user interface, while you still could directly execute any SQL statement.

**2.8) Hardware Requirements:**

* + - * **Processor**: Pentium 3.0 GHz or higher
      * **RAM**: 1024 MB or more
      * **Hard Drive**: 10GB or more
      * **Minimum OS:** Windows XP, Vista, 7, 8,10,11 Browser Software.
      * Internet Services or Local host Server

**2.9) Software Requirement:**

* + - * Windows XP/7/8/10/11.
      * XAMPP or WAMPP
      * Notepad++ / Visual Studio Code

**3.System Analysis**

**3.1) Objective:**

A key objective of this system is that it saves lots of time of user. User no longer need to depend on opinions from other people they can see what other people think about movie and critics ratings they can decide what to watch next. user can also create account and do more cool things.

➢ To identify via Internet and Online.

* + - * User can see all the organic content which available in our site. And get the full detail information about the Clothes.
      * User has no need to go out of their home.
      * It is Save the time of the use.
      * User can also visit our website any further query related to education system.
      * The main objective of the project is to create a system that allows users to save their time.
      * The selected images would display our all-clothes information if you selected gallery images then gallery display otherwise you can see our faculty.

**3.2) Requirement of new System:**

➢ You need to identify relevant keywords and phrases related to your website's content and integrate them into your content.

**3.3) Overview of proposed System:**

* + - We are trying to take less time of user and provide them important stuff.

* + - User can get all information in about Clothes in one place.

**3.4) Hardware:**

➢ The Project of JKG Fashion Website will be compatible on every Browser. Operating System like Windows 7 and higher versions will Support it.

**3.5) Software:**

* + - **Font End Tools:** PHP
    - **Back End Tools:** MySQLI

• **Feasibility Study:**

A feasibility study is carried out to select the best system that meets performance requirements. A feasibility study is designed to provide an overview of the primary issues related to a business idea. Three tests of feasibility-all equally important-are studied: Operational, Technical and Financial.

* + - **Operational Feasibility:**

The proposed System will meet the operational Requirements like System Performance, accessibility of information, client acceptance and efficient solutions to the queries of the user.

If user has some basic knowledge of Internet, User can operate this Website easily. It provides easy user Interface.

Operational feasibility has been considered form the user’s point of View. This Website once deployed. It can run easily without any maintenance at this point to time. After the inclusion of Database in future, the Database might need some clean up after some period of time. If the database size becomes large, then it might need some changes in handling of the Website and require some Optimization so that Website runs Faster and retrieves data faster.

* + - **Behavioural Feasibility:**

Proposed System will behave according to the requirement made or not. Responses time of the System must be noticed because it is a web-based System whether it takes too much time to response or give quick response or response in the specific period of time this consideration is most important. Time consuming Processes are possible to run on this environment or not it is also important.

* + - **Technical Feasibility:**

This System can be easily Supported by the Hardware and Software requirements of any System.

For End user they just require Username and Password facility to manage their Project. And we are developing Website using most recent Apache web server (XAMPP), which uses MySQLi Developer as Database. It is capable to Store data and provide concurrent access to Information and Adequate responses accurately. Also, we will try our system to make it as Expandable as Possible. We also provide Reliability, ease of access, and data Security because this is not a Final Project.

* + - **Economic Feasibility:**

The Hardware requirement of the System is at least a PC for an Administrator to handle the site from Admin panel. The development cost of the Projects is not much higher.

The cost required for the Creation of our System will be less Compared to the benefits provided by our System.

This System is User friendly so anyone who have a basic knowledge of Computer and then he/she can use easily. And it is also low cost because no need to attach extra Hardware.

## 

**4 Project Management**

Project Management is an important part of project development. It deals with all the main areas of the project development like Feasibility, Requirement analysis, Project Schedule, Project plan etc. We have used the Project Management approach to deal with all these areas. It is achieved by proper selection of Software Life Cycle Model.

**4.1) Project Planning and Scheduling:**

Project planning is perhaps one of the most important works in developing works in developing any project. Before the project can begin estimate regarding work to be done, what recourse’s will be required and how much time will elapse from start to the finish of project. Planning helped us to prepare a framework that enabled to make us a reasonable estimate of all such things.

**4.2) Phases:**

Incremental development slices the System functionality into increments (portions). In each increment, a slice of functionality is delivered through cross-discipline work, from the requirements to the deployment. The unified process groups increments/iterations into phases:

* **Inception:** Inception identifies Project scope, Requirements (functional and non-functional) and risks at a high level but in enough detail that work can be estimated.
* **Elaboration:** Elaboration delivers a working architecture that mitigates the top risks and full fills the non-functional requirements.
* **Construction:** Construction incrementally fills-in the architecture with production-ready code produced from analysis, design, implementation, and testing of the functional requirements.
* **Transition:** Transition delivers the System into the Production Operating environment.

**5.User Case**

## Administrator Function

* The Administrator will be able do anything.
* The Administrator will have full authority over all the users.
* The Administrator will be able to see the entire data that saved.
* The Administrator will be able Delete users and product.
* The Administrator will confirm and cancel the order of customers.
* The Administrator can add, delete and update the product information.

## 

## User Function

* The user will be able to view product and buy the product.
* The user can give us feedback after login.
* If user can give order by mistake, then he/she can also functionality to cancel the order.
* The User can order clothe at any place.
* In Our site user can also get customer friendly support from our staff 24\*7.

## **6.Data Dictionary**

### JKG Fashion DATABASE TABLES

|  |  |  |
| --- | --- | --- |
| **No** | **Table Name** | **Table Use** |
| 1 | adminlogin | For Admin only |
| 2 | user | For user Login |
| 3 | add\_product | For store available product details |
| 4 | buyers | Store buyers information |
| 5 | orders | Store Pending Orders |
| 6 | confirm\_orders | Store Completed order |
| 7 | cancel\_orders | Store canceled orders |
| 8 | feedback | Store Customers Complain and Complement |

**Table’s Columns information**

**1.adminlogin**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
|  | \* | id | int | 20 | Primary key |
|  |  | name | varchar | 60 | Not Null |
|  |  | password | varchar | 20 | Not Null |

**2.user**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
|  | \* | id | int | 11 | Primary key |
|  |  | user\_name | varchar | 200 | Not Null |
|  |  | number | bigint | 10 | Not Null |
|  |  | password | varchar | 200 | Not Null |

**3.add\_product**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
|  | \* | Product\_id | int | 11 | Primary key |
|  |  | Product\_name | varchar | 300 | Not Null |
|  |  | price | bigint | 20 | Not Null |
|  |  | qut | bigint | 20 | Not Null |
|  |  | gender | varchar | 20 | Not Null |
|  |  | category | varchar | 20 | Not Null |
|  |  | size | varchar | 20 | Not Null |
|  |  | image | varchar | 300 | Not Null |

**4.buyers**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
|  | \* | Buyer\_id | int | 11 | Primary key |
|  |  | buyer\_name | varchar | 200 | Not Null |
|  |  | mobileno | bigint | 10 | Not Null |
|  |  | B\_address | varchar | 100 | Not Null |
|  |  | status | int | 11 | Not Null |

**5.orders**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
|  |  | Sr\_number | int | 11 | Not Null |
|  |  | Prod\_id | int | 11 | Not Null |
|  |  | Prod\_name | varchar | 200 | Not Null |
|  |  | Prod\_price | bigint | 20 | Not Null |
|  |  | gender | varchar | 20 | Not Null |
|  |  | size | varchar | 20 | Not Null |
|  |  | Prod\_qut | bigint | 20 | Not Null |
|  |  | total | bigint | 20 | Not Null |
|  |  | Pay\_mode | varchar | 40 | Not Null |
|  |  | date | date |  | current\_timestamp() |

**6.confirm\_orders**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
| \* |  | conf\_id | int | 11 | Primary key |
|  |  | sr\_number | int | 11 | Not Null |
|  |  | order\_id | int | 11 | Not Null |
|  |  | conf\_name | varchar | 200 | Not Null |
|  |  | conf\_price | bigint | 20 | Not Null |
|  |  | gender | varchar | 20 | Not Null |
|  |  | size | varchar | 20 | Not Null |
|  |  | qut | int | 11 | Not Null |
|  |  | total | bigint | 20 | Not Null |
|  |  | ay\_mode | varchar | 40 | Not Null |
|  |  | order\_time | date |  | Not Null |
|  |  | conf\_time | date |  | current\_timestamp() |

**7.cancel\_orders**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
| \* |  | canc\_id | int | 11 | Primary key |
|  |  | sr\_number | int | 11 | Not Null |
|  |  | order\_id | int | 11 | Not Null |
|  |  | canc\_name | varchar | 200 | Not Null |
|  |  | canc\_price | bigint | 20 | Not Null |
|  |  | gender | varchar | 20 | Not Null |
|  |  | size | varchar | 20 | Not Null |
|  |  | qut | int | 11 | Not Null |
|  |  | total | bigint | 20 | Not Null |
|  |  | pay\_mode | varchar | 200 | Not Null |
|  |  | order\_time | date |  | Not Null |
|  |  | conf\_time | date |  | current\_timestamp() |

**8.feedback**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** |  | **Field Name** | **Type** | **Size** | **Constraints** |
|  | \* | id | int | 11 | Primary key |
|  |  | name | varchar | 30 | Not Null |
|  |  | number | bigint | 10 | Not Null |
|  |  | feedback | varchar | 200 | Not Null |

### **7) Data Flow Diagram**

❖ **Data Flow Diagram from both Side**

**View orders oo]]orOrders**

**Cart**

**Login**

❖

**Client**

**JKG Fashion**

**Admin**

**Home**

**feedback**

**Contact**

**Men**

**Login**

**View Product**

**View feedback**

**Add Product**

**View/manage**

**Order**

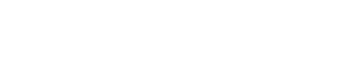
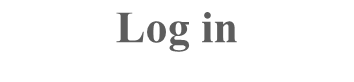
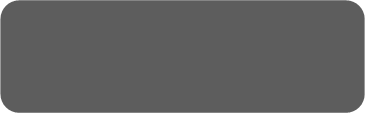
**Women**

**About Us**

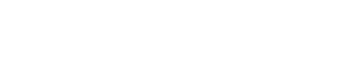
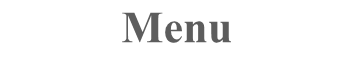
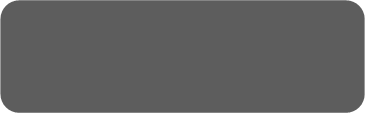
**Manage Product**

**Client Side Use Case**

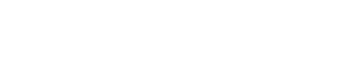
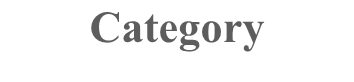
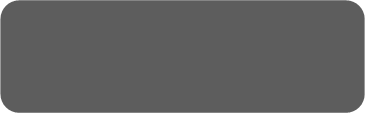
USER



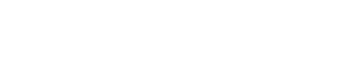
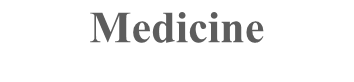
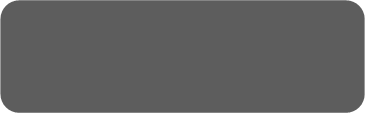
**Log in**



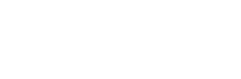
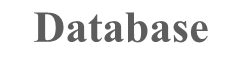
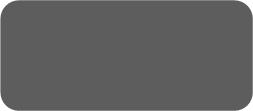
**Menu**



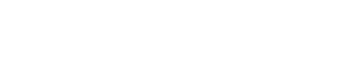
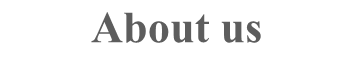
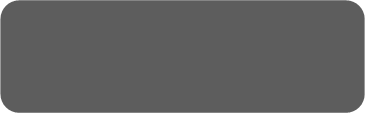
**Men/Women**



**Contact/aboutus**



**Database**



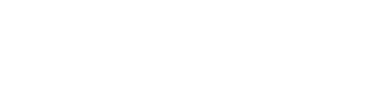
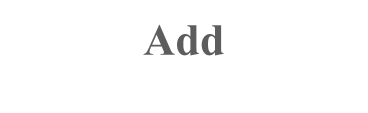
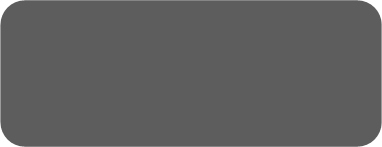
**feedback**

**Admin Side Use Case**

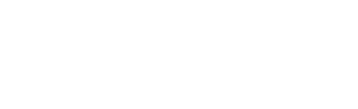
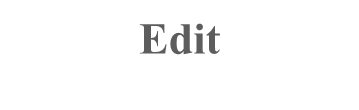
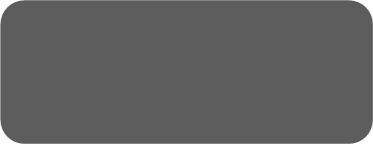
* **Here Admin can view, add, delete, update food images, gallery, contact detail etc.**



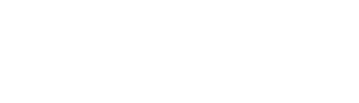
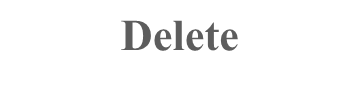
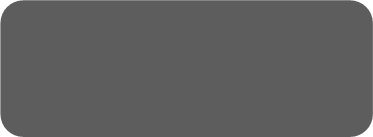
**Manage Feedback**



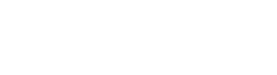
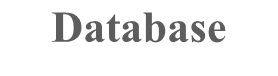
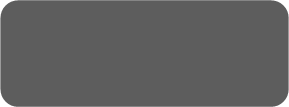
**Add product**



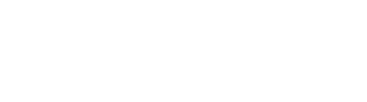
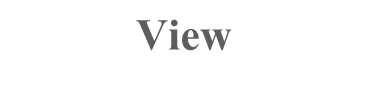
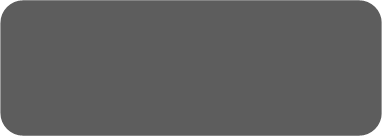
**Manage Order**



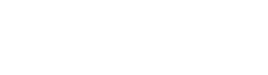
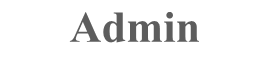
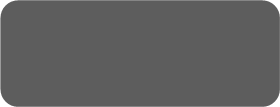
**Manage User**



**Database**



**Manage Products**



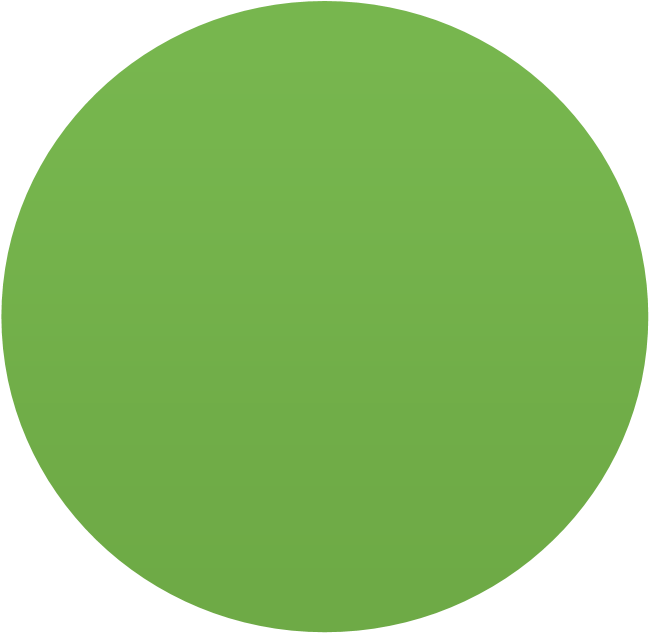
**Admin**

**DFD Diagram (Data flow Diagram)**

**DFD 0 Level**



**Order Clothes**



**Admin**



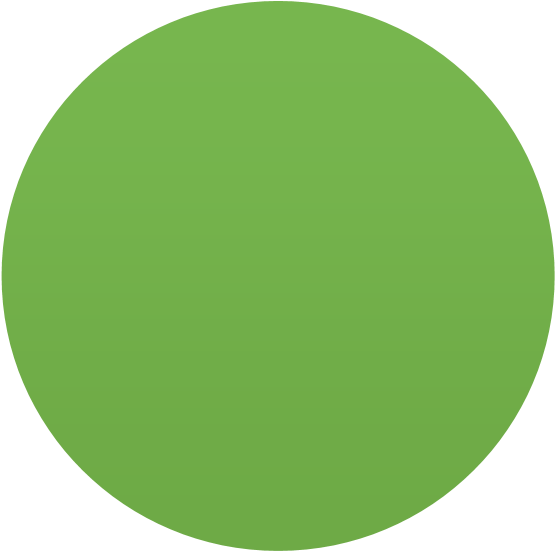
**Log in**

* **DFD Level 1 (Diagram)**

### **Client Side**



**Registration**



**Process**

**To Store**

**Data**



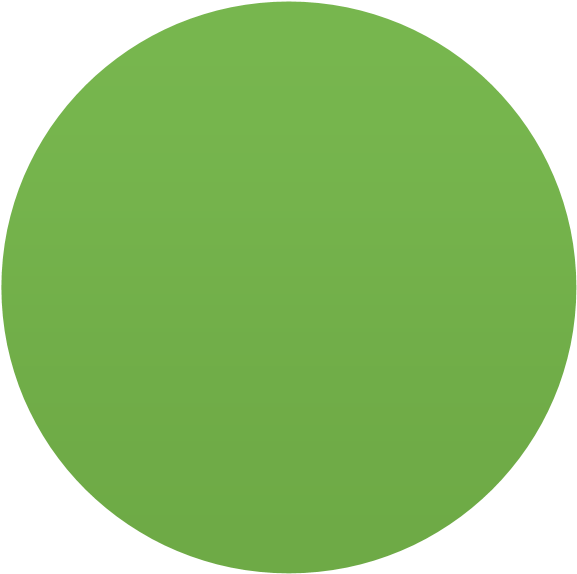
**Login**



**Register**



**Login**



**Process**

**To Store**

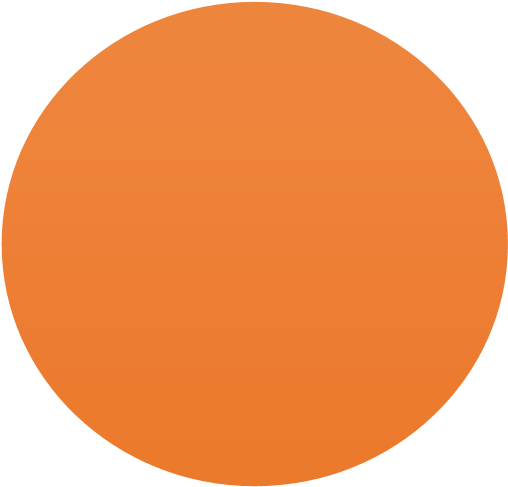
**Data**



**Order Clothes**



**View Orders**



**Project**

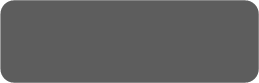
**Database**



**User**



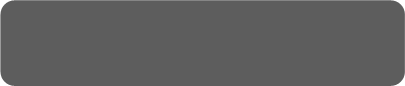
**Database**



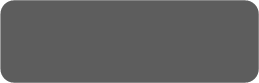
**Logout**

➢ **DFD Level 1 (Diagram)**

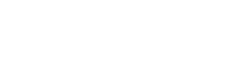
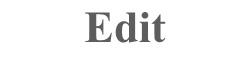
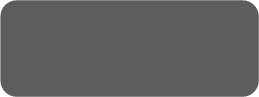
### Admin Side



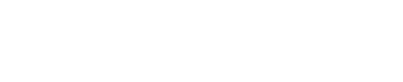
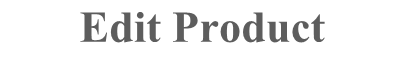
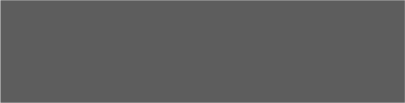
**Log in**



**Home**

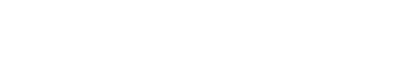
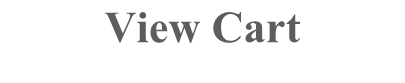
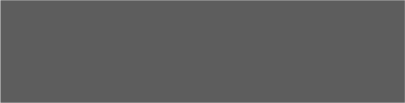


**Edit**



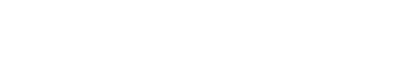
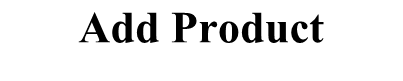
**Manage**

**Product**



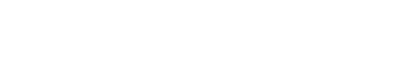
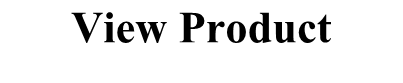
**View**

**Feedback**



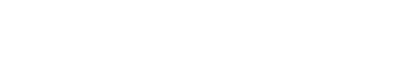
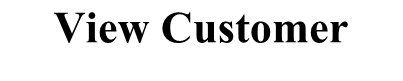
**Add**

**Product**



**Manage**

**Product**



**View Customer**

**8. CFD (Control Flow Diagram)**

•

**Control Flow Diagram from Admin Side**



**Administrator**



**Admin**

**Login**



**Home**



**Login**

**Page**

#### **➢** **Control Flow Diagram from User side**



**All Users**



**Non**

**-**

**Register User**



**User must Register first**

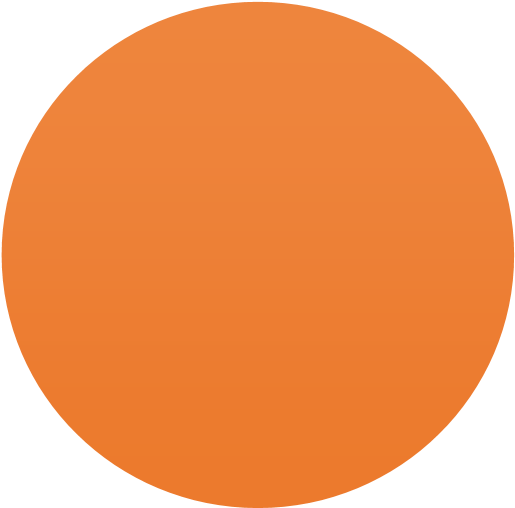
**then, he/she can Login**



**Register User**



**Home**



**User**

**Login**



**Log**

**-**

**in**

#### **9. Implementation Planning and Details**

• **Implementation Environment**

* This application is made by keeping in mind the basic Concept of Website Development, the Website can be used by each of the Person at a time on their respective Systems.

* This Website is also made in such a way as to provide user with the most Effective Graphical Interface. So that the user can easily search out for Food and Navigate through out the Website with the ease of using this Website.

* So, as the conclusion of above two points, this Website is having the multi-user and Effective GUI environment for the users.

#### **10. Testing**

* This includes the methods that were used for testing, Validating, and evaluating the System. The Conclusion and the Future Work for the Software are also given.

* Start with a base Function that you want to implement.

* Create a document with the detailed requirement definition, an activity diagram with a description of the Flow, Database tables to be used, a component diagram, and a Description of each component with the precondition and tables that would be affected by the component.

* Give the document to the tester, and work with the tester while he or she writes the code to check if the steps in the document can be implemented and if the result of each use case can be achieved.

* Ask the tester to log on all the errors and difficulties he or she encountered while working on the prototype implementation.

* When the testing approach was implemented, the following pros and cons regarding the testing approach were realized.
  + **Pros of using the methodology** 
    - Helps give a better understanding about the requirements.
    - Better design at the end of the cycle.
    - Reduced testing to be performed at the end of the cycle.
    - Documents produced would be of higher quality.

* + **Cons of using the methodology**

* + - The person working on the document should be experienced.
    - There are increased time and money involved with testing.
    - Different viewpoints for the same problem can lead to varying results.

* **Testing Method:**

There are mainly Four Strategies are there.

* + Static Testing
  + Dynamic Testing
  + Black Box Testing
  + White Box Testing

* **Static Testing:**

The Verification activities fall into the Category of Static Testing.

During Static testing, you have a Checklist to check whether the work you are doing is going as per the set standards of the organization.

These Standards can be for Coding, Integrating and Deployment. Reviews, Inspection’s and Walkthrough’s are Static Testing methodologies.

* + **Dynamic Testing:**

Dynamic Testing involves working with the Software, giving input Values and Checking if the output is as expected.

These are the Validation activities. Unit Test, Integration Tests, System Tests and Acceptance Tests are few of the Dynamic Testing methodologies.

As we go further, let us understands the various test of Life Cycle’s and get to know the Testing Terminologies. To understands more of Software testing, various methodologies, tools and Techniques.

* + **Black Box Testing:**

Black Box Testing is testing without knowledge of the internal workings of the Item being tested. For example, when black box testing is applied to Software Engineering, the tester would only know the legal Inputs and what the expected Outputs should be, but not how the Program arrives at those outputs.

It is because of this that black box testing can be considered testing with respect the specifications, on another knowledge of the program is necessary.

For this reason, the tester and the programmer can be independent of one another, avoiding programmer bias toward his own work. For this Testing, test groups are often used.

This Strategy has some advantage like it is more effective on larger units of code than glass box testing, tester needs no knowledge of implementation, including specific programming.

• **White Box Testing:**

White box testing strategy deals with the internal logic and structure of the code. White box testing is also called as glass, structural, open box or clears box testing.

The tests written based on the white box testing strategy incorporate coverage of the code written, branches, paths, statements and internal logic of the code etc.

To implement white box testing, the tester must deal with the code and hence is needed to possess knowledge of coding and logic i.e., internal working of the code.

White box test also needs the tester to investigate the code and find out which unit/statement/chunk of the code is malfunctioning.

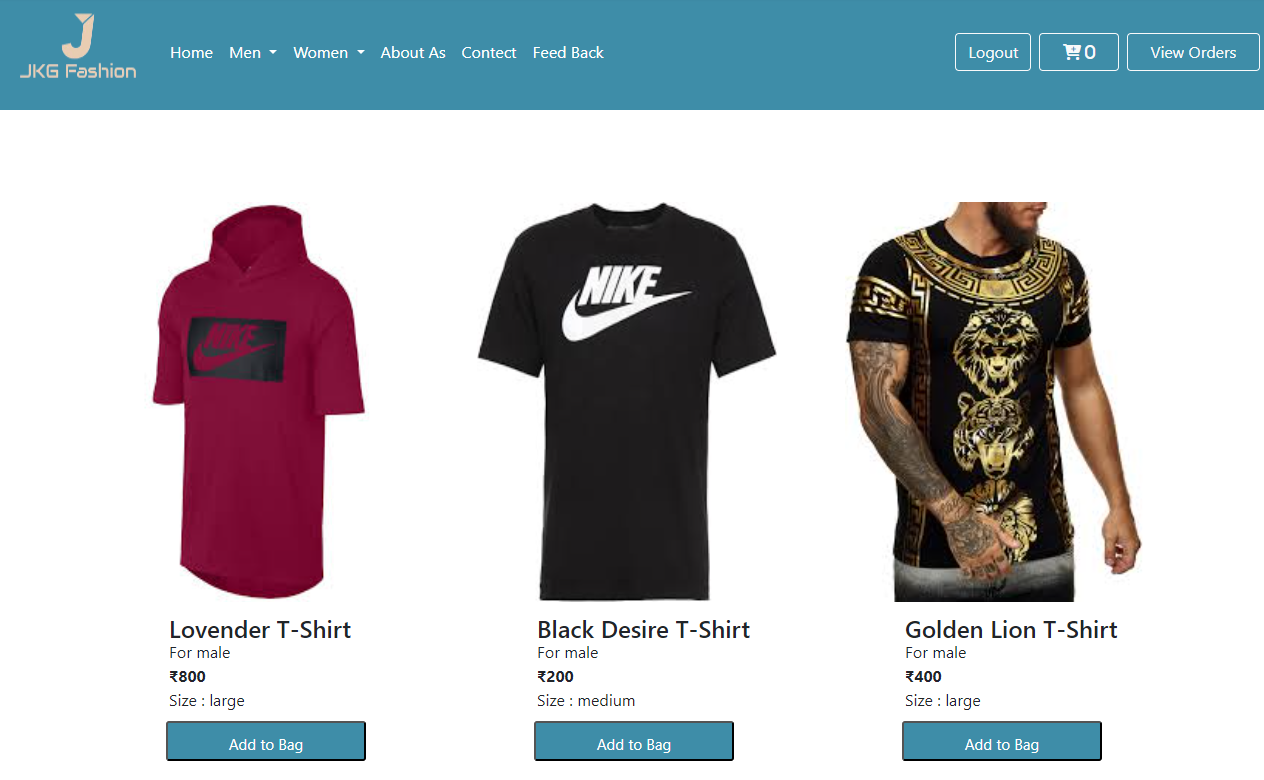
The White Box Testing has also some advantages like as the Knowledge of internal coding structure is prerequisite. It becomes very easy to find out which type of input/data can help in testing the website effectively.

It helps in optimizing the code it helps in removing the extra lines of code, which can bring in hidden defects.

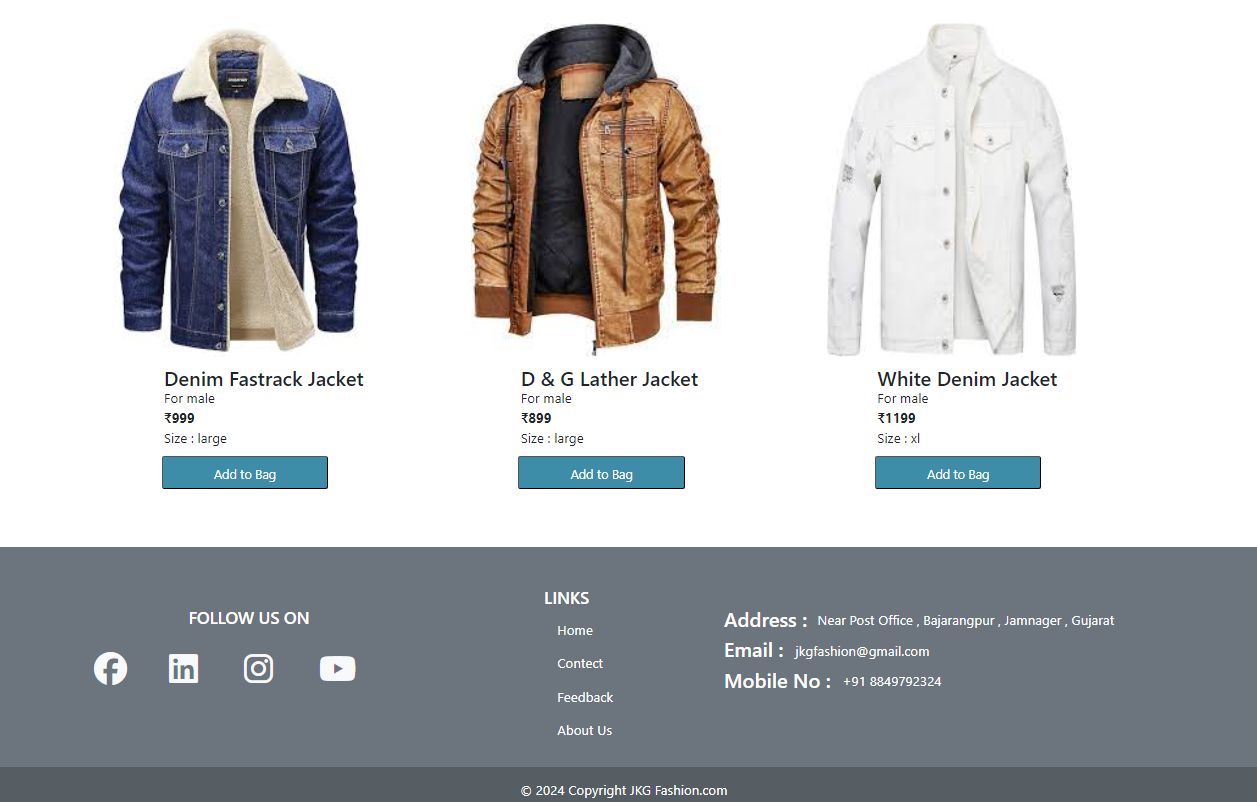
### **11. Screenshots**

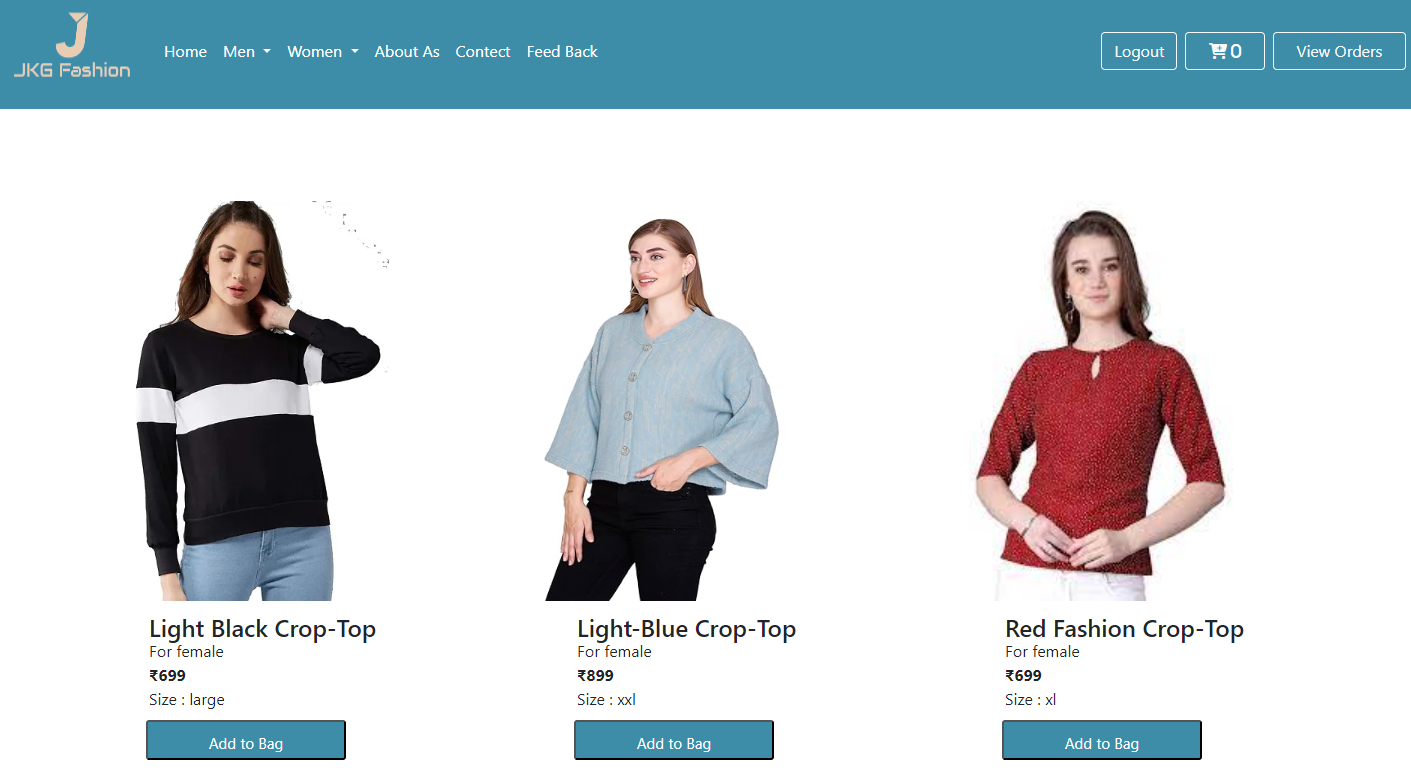
* **Client-side home page: -**

**1.Index Page**

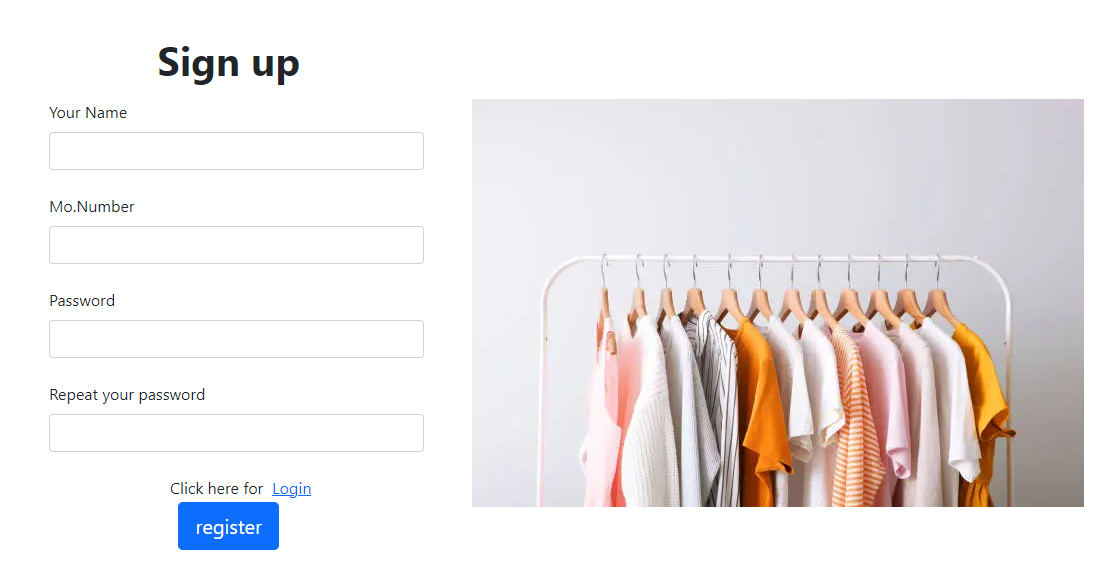


**2.Perticular Product Page**

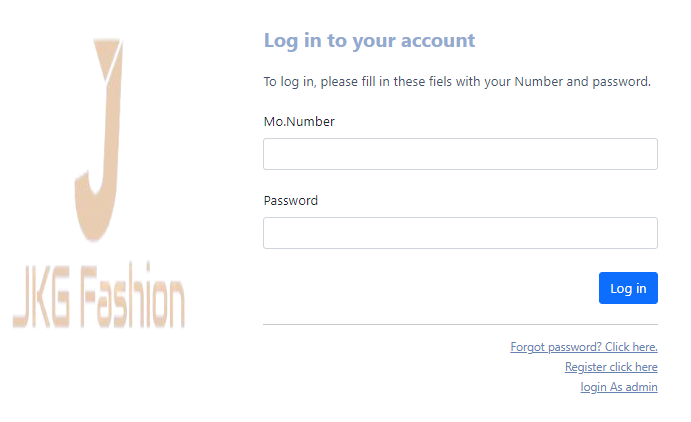
****

****

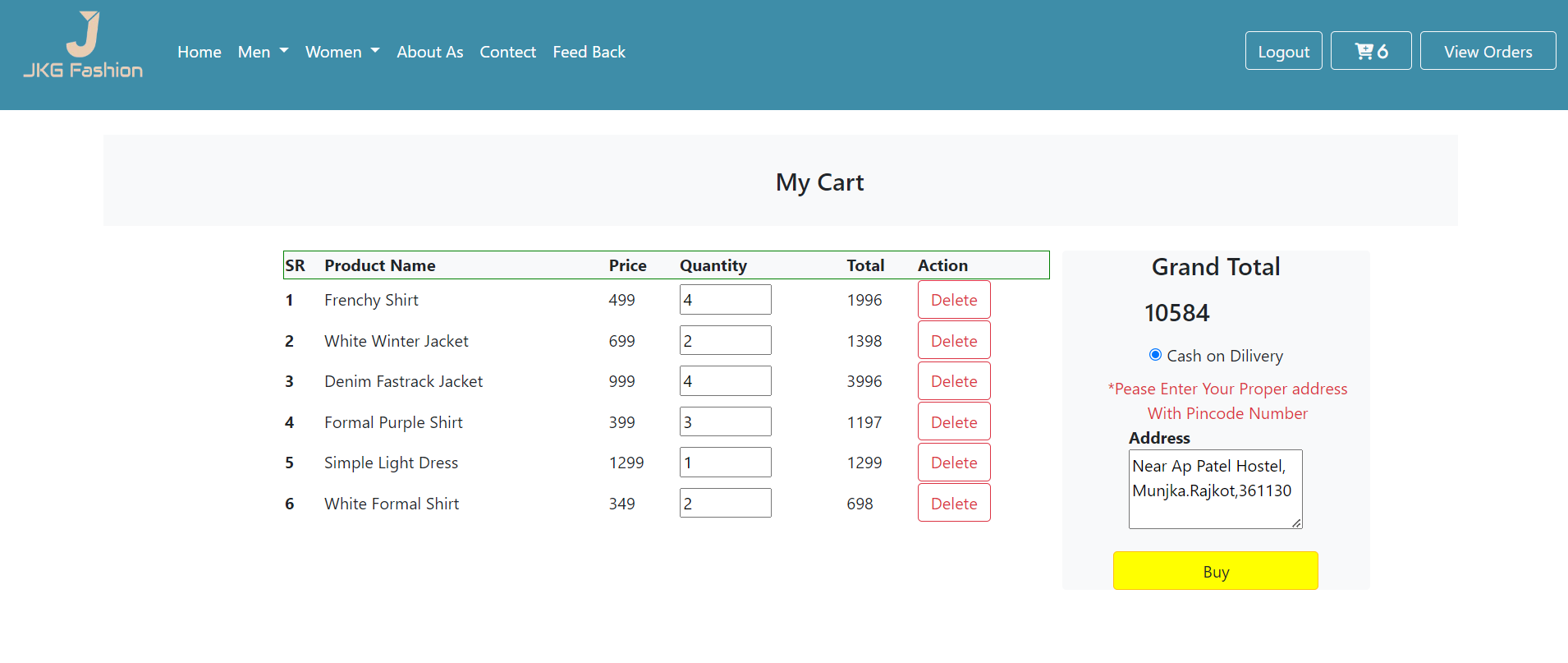
**3.Register Page**

****

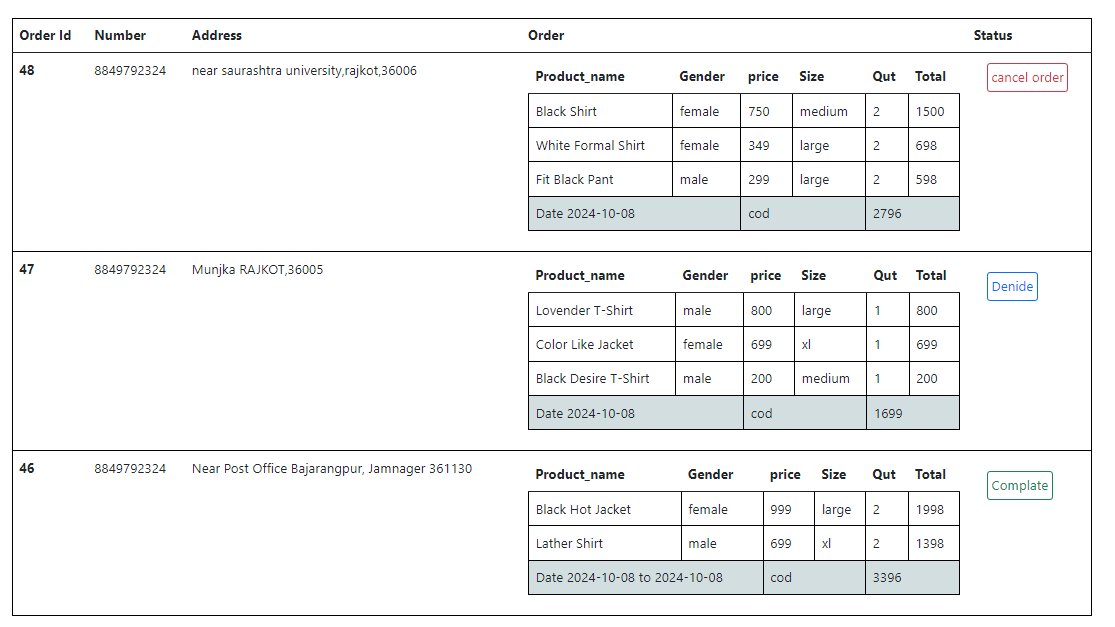
**4.Login Page**

****

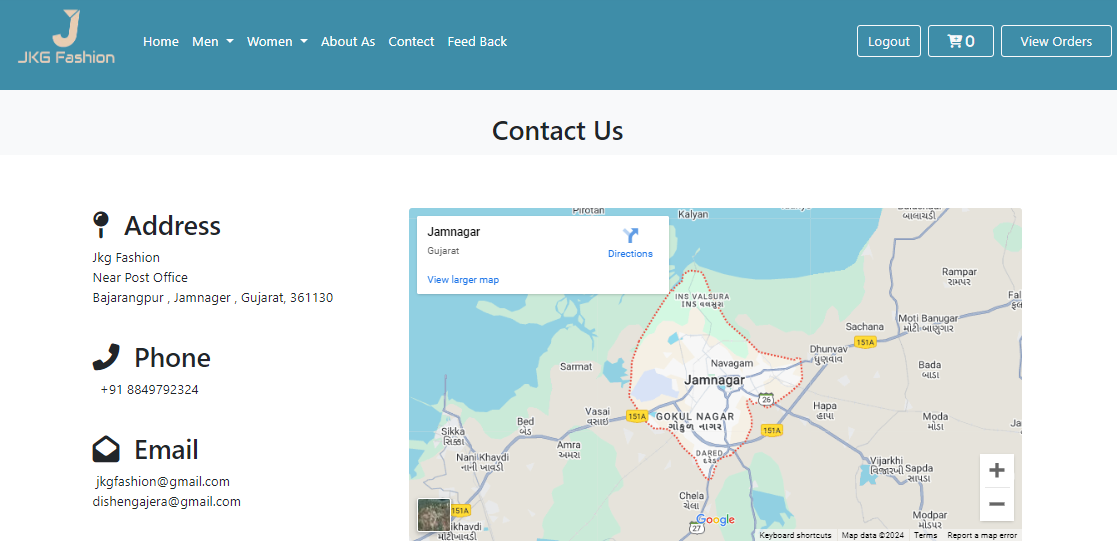
**5.Cart**

****

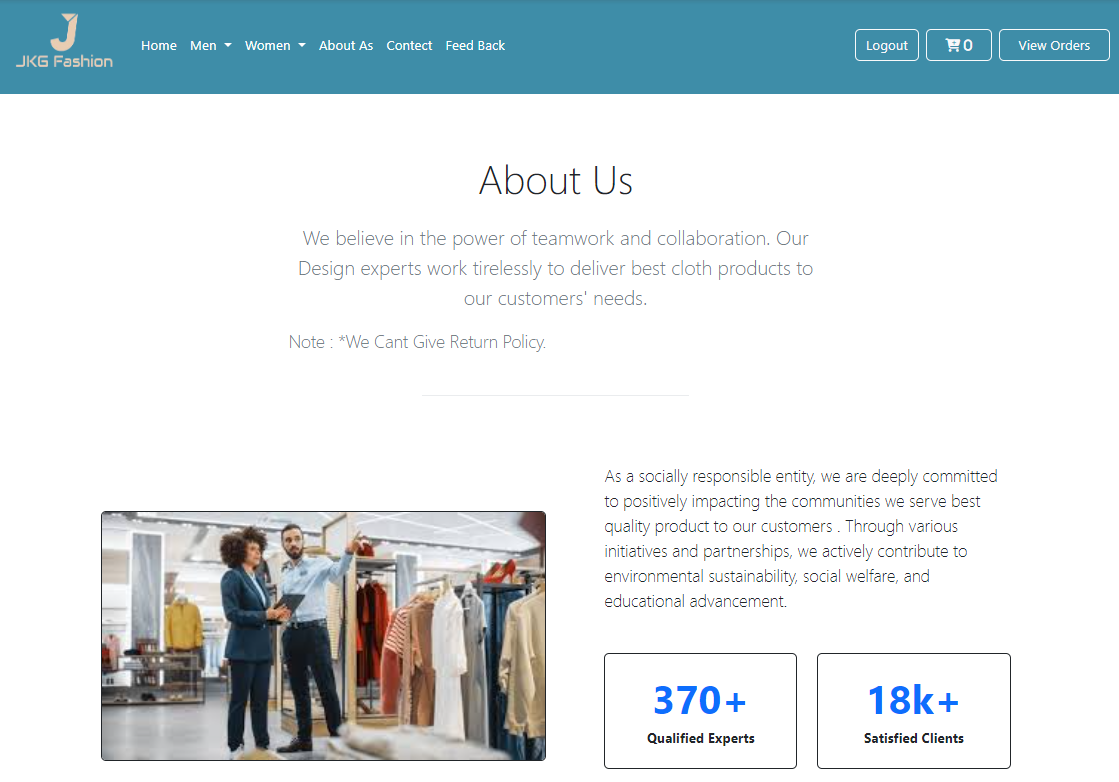
**6. Order Status Page**

****

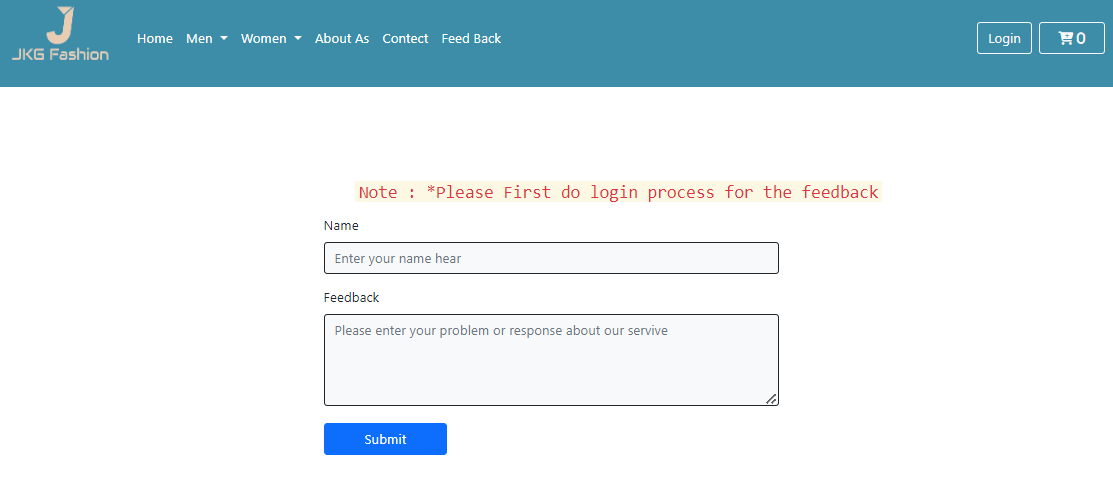
**7.Contect Us Page**

****

**8.About as Page**

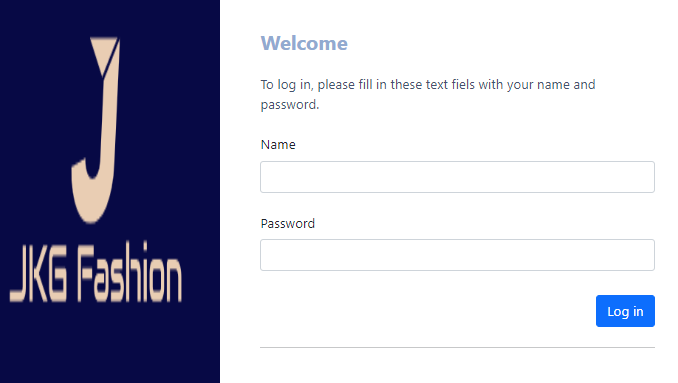
****

**9.FeedBack Page**

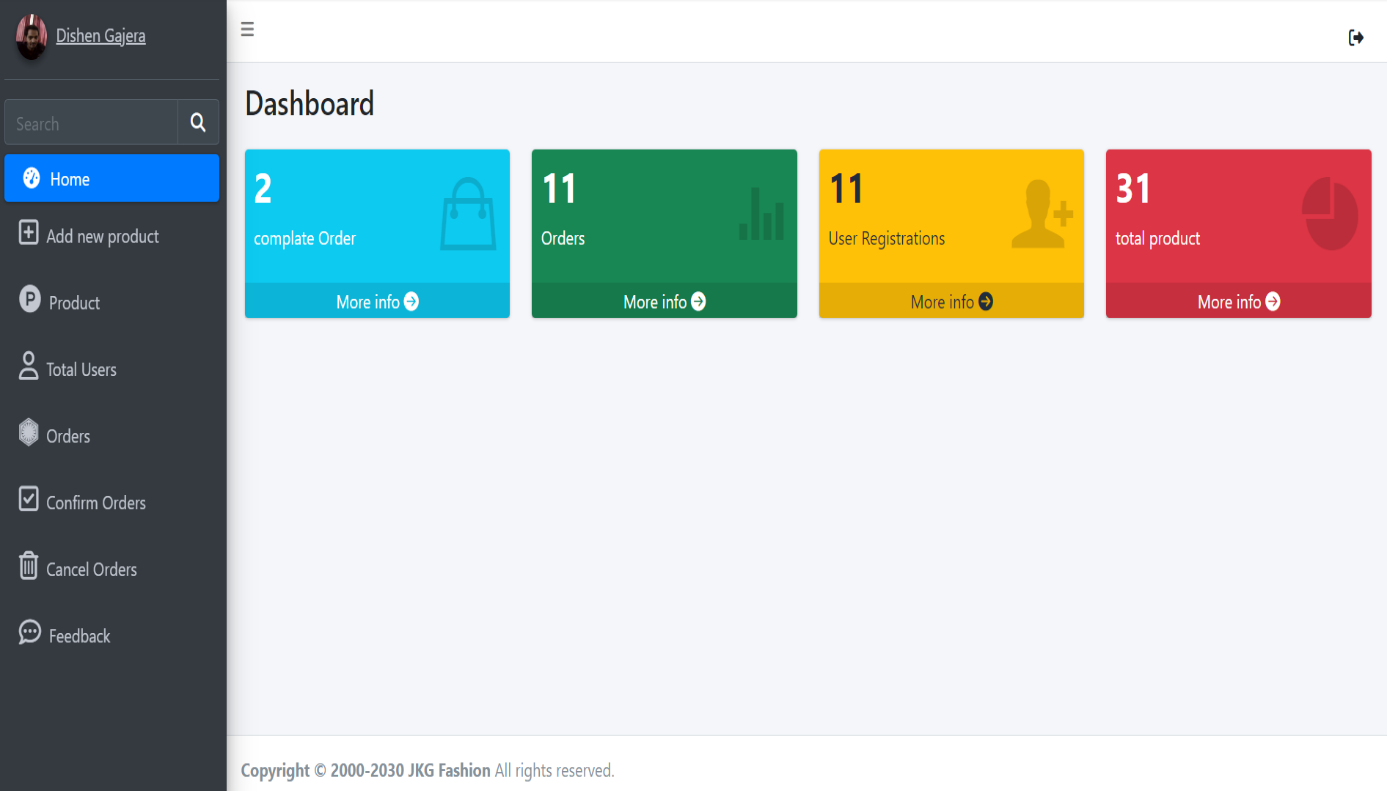
****

* **Admin-side home page: -**

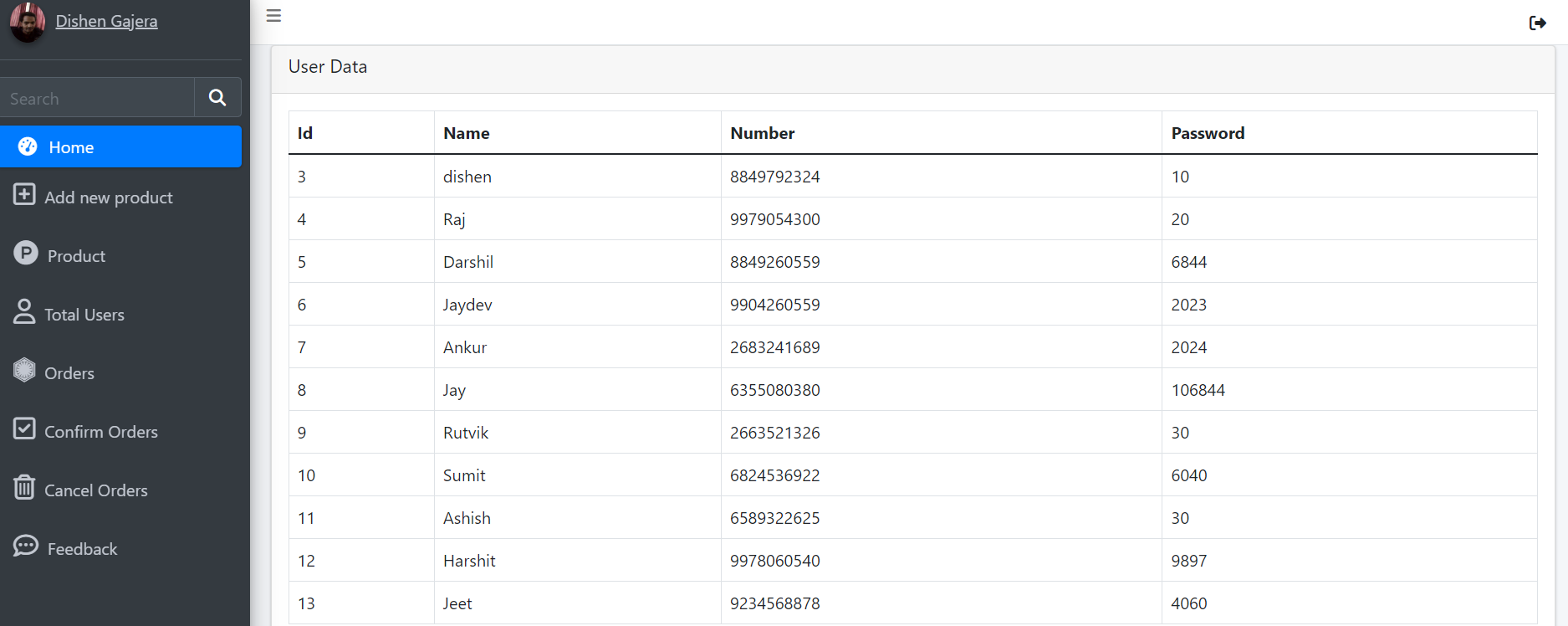
**1.Admin Login Page**

****

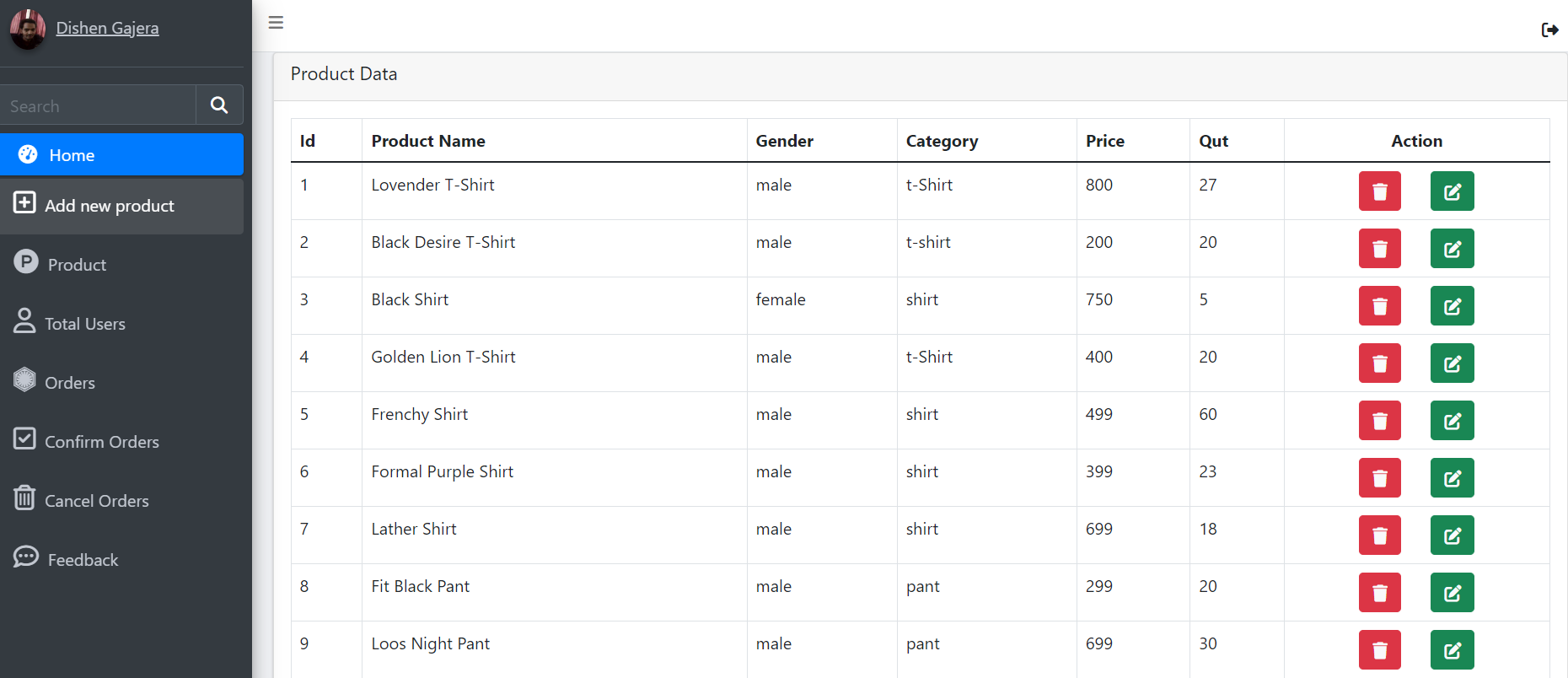
**2.Admin Home Page**

****

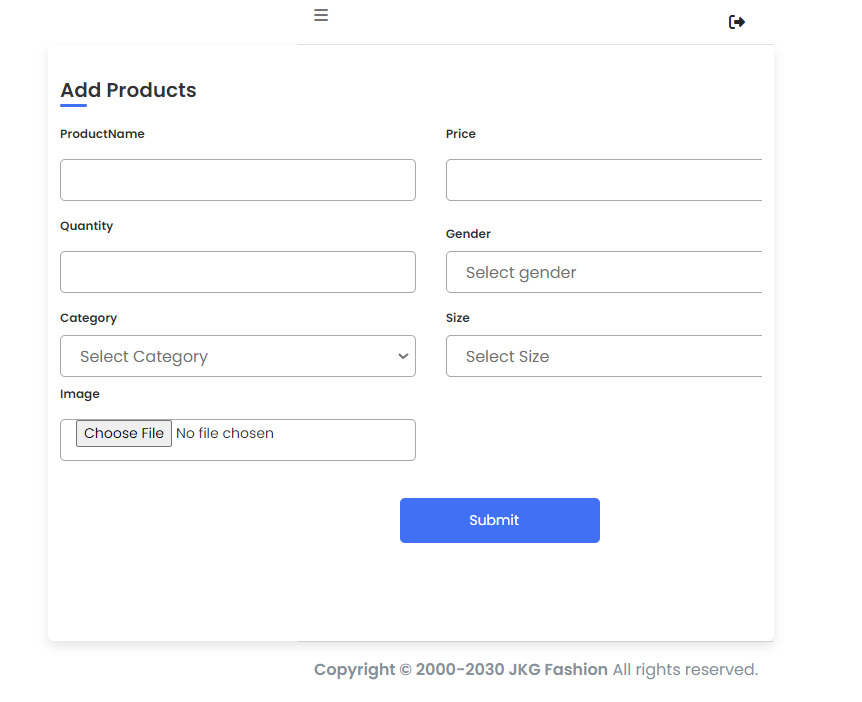
**3.User Data Page**

****

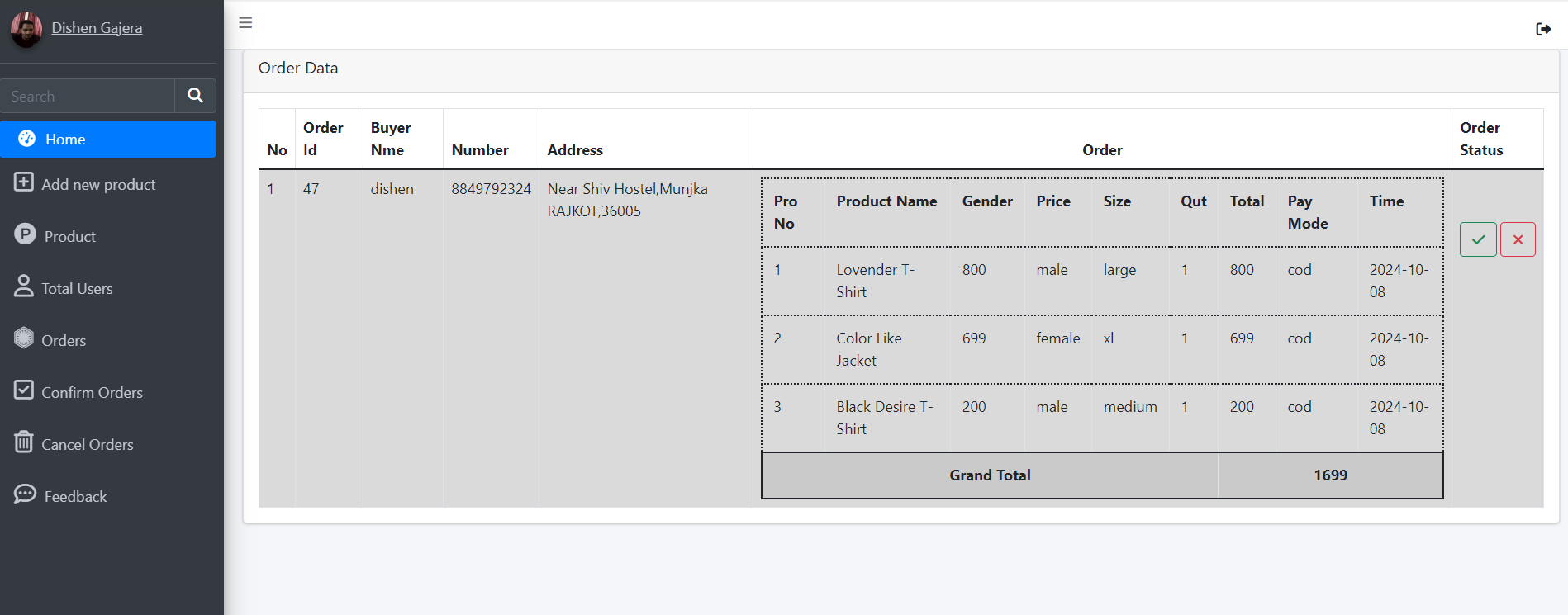
**3.Product Manage Page**

****

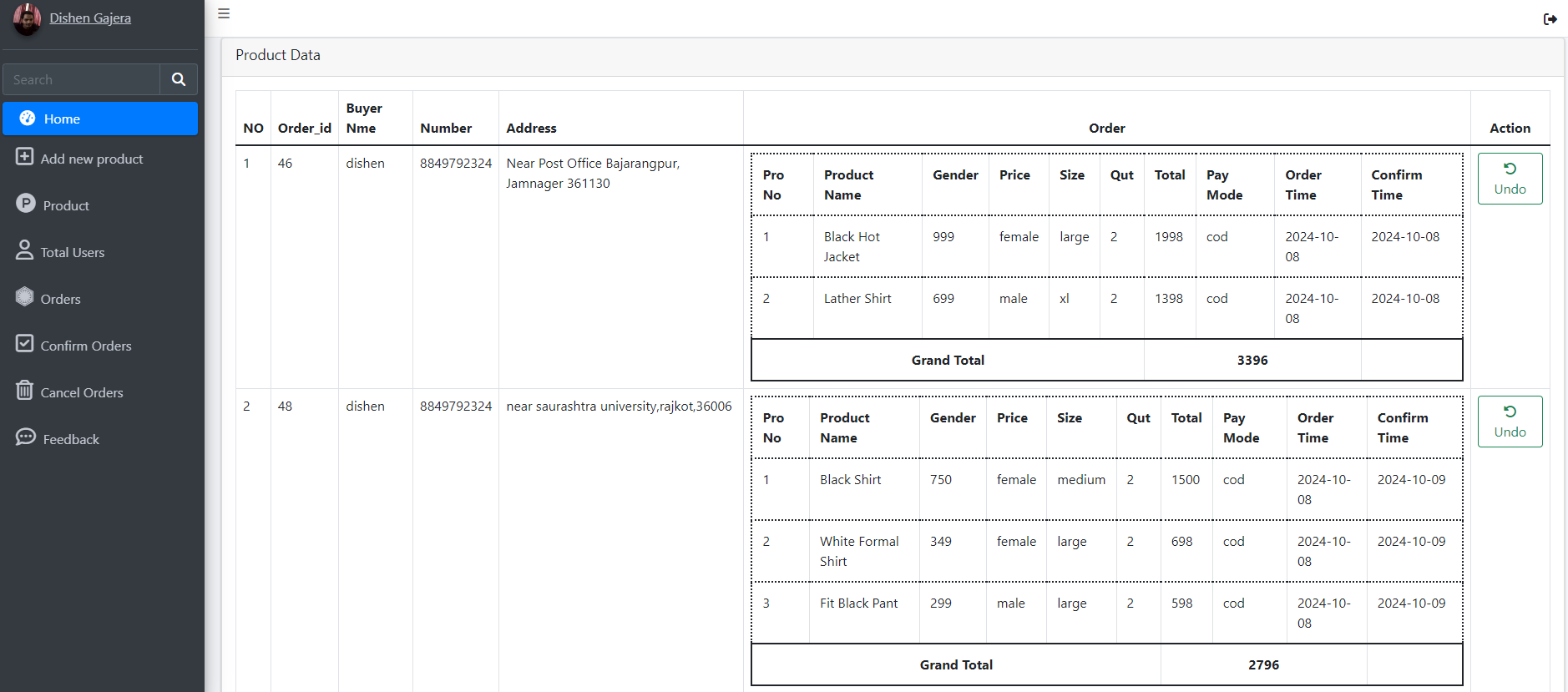
**4.Add New Product Page**

****

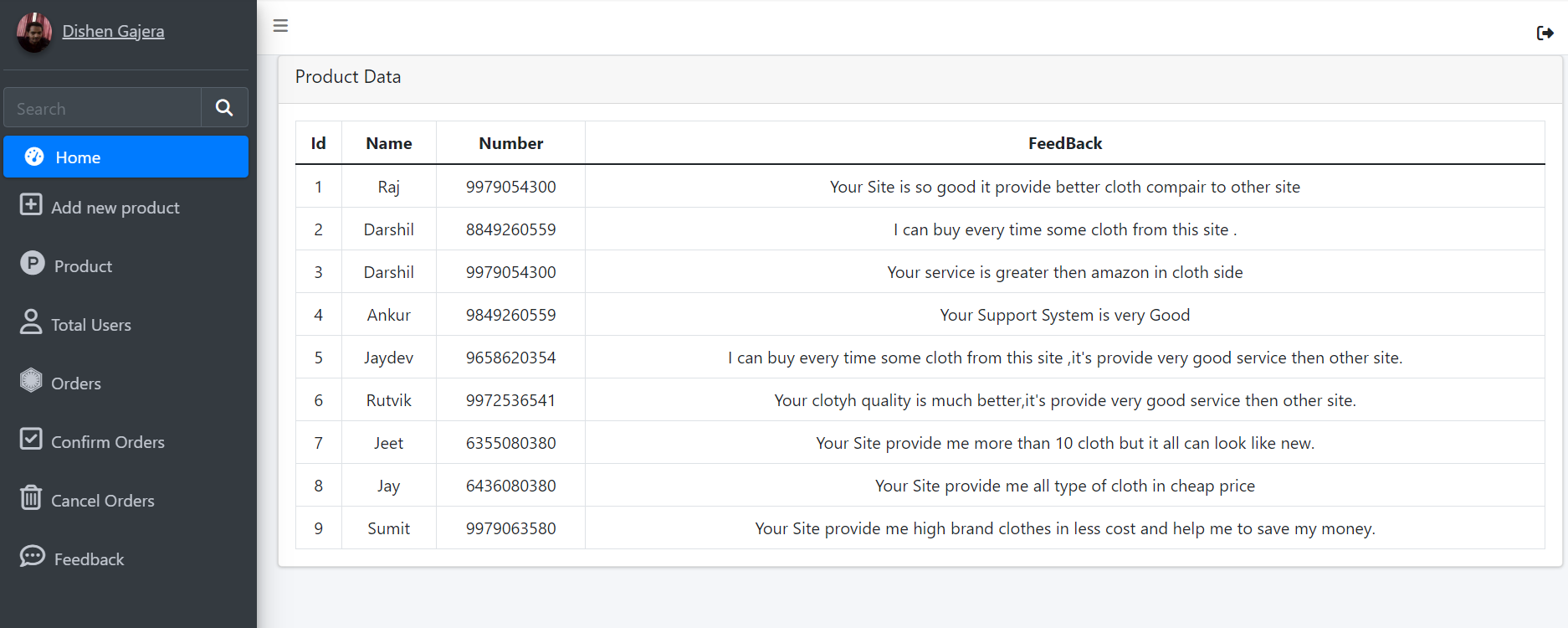
**5.New Order Page**

****

**6.Confirm Order Page**

****

**7.Feedback Manage Page**

****

**12. Limitation and Future Enhancement**

* + **Limitation**
  + **Future Enhancement**
  + **Conclusion**
  + **References**

* **Limitation** 
  + Our website not Accept Online Money Like Google Pay Or Any UPI.
  + We Can not provide return policy.
* **Future Enhancement** 
  + We want to expand our website users count.
  + We also add more Category and Product in our website.
  + We Provide online payment facility to user.
  + We Also like to provid to return policy to user.

* **Conclusion** 
  + The Project report entitled “JKG Fashion” has come to its conclusion.

* + The new System has been developed with so much care that it is free of errors and at the same time efficient and less time consuming.

* + System is robust. Also, Provision is provided for future Developments in the System.

* **References:**

• **Websites**

* + [www.google.com](http://www.google.com/)
  + [www.wikipedia.com](http://www.wikipedia.com/)
  + http://www.amazon.in
  + [www.netmeds.com](http://www.netmeds.com/)
  + [www.youtube.com](http://www.youtube.com/)

• **Professor**

* + Dr. Ashwin Rathod
  + (Dr).Dhimesh Parmar

Thank You