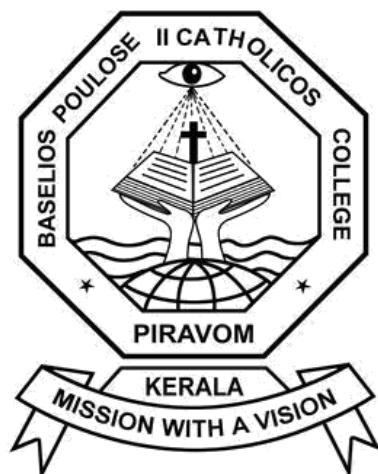


**BASELIOS POULOSE II CATHOLICOS COLLEGE
BASELIOS MOUNT, PIRAVOM**

**Re-accredited with 'A' Grade by NAAC
(Affiliated to Mahatma Gandhi University)**

DEPARTMENT OF COMPUTER APPLICATIONS



**2023-24
Project Report
On
ARTISANAL
(ONLINE HANDICRAFT STORE)**

**BASELIOS POULOSE II CATHOLICOS COLLEGE
BASELIOS MOUNT, PIRAVOM**

**Re-accredited with ‘A’ Grade by NAAC
(Affiliated to Mahatma Gandhi University)**

2023-24



Project Report

On

ARTISANAL

**Submitted in partial fulfillment of the requirements for the award
of the degree of**

BACHELOR OF COMPUTER APPLICATION

Guided by: Dr. DHANYA JOB

(Dept. of Computer Applications)

Submitted by:

Sinju Mathews

Reg No: 210021093796

BASELIOS POULOSE II CATHOLICOS COLLEGE

Re-accredited with ‘A’ Grade by NAAC

(Affiliated to Mahatma Gandhi University)

PIRAVOM

2023-24

DEPARTMENT OF COMPUTER APPLICATIONS



Certificate

This is to certify that the project entitled “ARTISANAL” submitted in partial fulfillment for the award of the degree of BACHELOR OF COMPUTER APPLICATION is a bonafide report of the project done by Sinju Mathews (Reg no: 210021093796) during the year 2023-24.

Internal Guide:

Dr. Dhanya Job

Head of the Department

Dr. Anu Paul

Examiner: 1

College Seal

Department Seal

DECLARATION

*I hereby declare that this project work entitled “ARTISANAL” is a record of original work done by me under the guidance of **Dr. Dhanya Job**, Assistant Professor, Department of Computer Applications and the work has not formed the basis for the award of any degree or diploma or similar title to any candidate of any university subject.*

Internal Guide

Dr. Dhanya Job

Signature of Student

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

At the outset, I thank God Almighty for making endeavor a success.

I express my gratitude to **Dr. Santhosh Potharay Kuruvilla**, Principal, Baselios Poulose II Catholicos College, for providing me with adequate facilities, ways and means by which I was able to complete the project work. I express my sincere thanks to my internal guide **Dr. Dhanya Job**, who guide me properly from the beginning to the end of my project and examining the draft of this project, suggestions and modifications. With immense pleasure I take this opportunity to record out sincere thanks to Head of the Department **Dr. Anu Paul**, Associate Professor, Department of Computer Applications for her motivation throughout this project.

Last but not the least, I also express my gratitude to all other members of the faculty and well-wishers who assisted me in various occasions during the project work.

- **Sinju Mathews**

ABSTRACT

ABSTRACT

Artisanal is an online handicraft store where people can buy handmade products from different Artisans/Sellers. And also interact with them by viewing the post they share, liking the posts, commenting on them and follow the sellers of your liking.

Artisanal has three users namely Admin, Seller and Customer. The Admin is the system administrator and is in charge of maintaining and supervising the website. The admin is responsible for verifying different sellers that sign up, managing the categories, and overviewing the orders dealings and other activities that happens in the website.

Next the sellers, the sellers/artisans are the ones that sell their handcrafted products in this website and share their moments and experiences as an artisan by posting them as feeds. They add different products they would like to sell, manage orders they received and can overview reports on their sales.

Customers can shop the products of their liking and follow their favorite seller in the platform customers can view post by different sellers, like, comment and report them if it does not seem right. Customers can see their orders details, rate and review the products they bought.

This website is specially designed for handicraftsmen to show there talent to the world and for people that are looking for real authentic and creative products.

LIST OF TABLES

Sl.No.	Table Name	Page Number
1	tbl_admin	25
2	tbl_district	26
3	tbl_city	26
4	tbl_category	26
5	tbl_subcategory	27
6	tbl_seller	27
7	tbl_type	28
8	tbl_product	28
9	tbl_product_gallery	28
10	tbl_post	29
11	tbl_likes	29
12	tbl_comments	29
13	tbl_follow	30
14	tbl_seller_bio	30
15	tbl_stock	30
16	tbl_post_report	31
17	tbl_notification	31
18	tbl_customer	32
19	tbl_wishlist	32
20	tbl_cart	33
21	tbl_order	33
22	tbl_payment	33
23	tbl_complaint	34
24	tbl_feedback	34
25	tbl_rating	34
26	Data Flow Diagram Symbols	36
27	Test Cases	71

LIST OF FIGURES

Sl.No.	figure Name	Page Number
1	Zeroth level DFD for Artisanal	37
2	1 st level DFD for Artisanal admin side	38
3	1 st level DFD for Artisanal seller side	39
4	1 st level DFD for Artisanal customer side	40
5	2 nd level DFD for Artisanal admin side Manage Categories	41
6	2 nd level DFD for Artisanal admin side Manage Complaints	41
7	2 nd level DFD for Artisanal seller side Manage Account	42
8	2 nd level DFD for Artisanal seller side Manage Products	42
9	2 nd level DFD for Artisanal seller side Manage Orders	43
10	2 nd level DFD for Artisanal customer side Manage Account	43
11	2 nd level DFD for Artisanal customer side Shop Products	44
12	2 nd level DFD for Artisanal customer side Make Orders	44
13	2 nd level DFD for Artisanal customer side View Feeds	45
14	Customer registration form	47
15	Product Adding form	48
16	Report Sales by Category - Bar Chart	50
17	Unit testing 1	62
18	Unit testing 2	63
19	Integration testing	65
20	Integration testing result 1	65
21	Integration testing result 2	66
22	User name validation	68
23	Password validation	68
24	Already Existing validation testing	69
25	Guest Page	75
26	Login Page	76
27	Admin dashboard	76
28	Seller verification	77

29	Adding Districts	77
30	Adding Cities	78
31	Adding Types	78
32	Adding Categories	79
33	Adding Subcategories	79
34	Reported Posts	80
35	View Feedbacks	80
36	View Complaints	81
37	Reply to Complaints	81
38	Sales report by category	82
39	Sales Report by Sellers	83
40	Bar chart on Followers of sellers	84
41	General Report on Orders received	84
42	Admin notifications	85
43	Seller Registration	86
44	Customer Registration	87
45	Forgot Password	88
46	Password Recovery OTP	88
47	Change Password	88
48	Seller Dashboard	89
49	Seller MyAccount	89
50	Seller Edit Account	90
51	Seller change Password	90
52	Seller Add Product	91
53	Seller My Products	91
54	Add product Images	92
55	Add product Stock	92
56	Edit Product	93
57	Seller Set Profile	94
58	Seller My Profile	94

59	Seller Add post	95
60	Seller My posts	95
61	Seller Sales Per Product	96
62	View Orders	97
63	Seller notification	98
64	Seller sent Feedback	98
65	Seller sent Complaint	99
66	Customer Home Page	100
67	Customer My Account	101
68	Customer Edit Account	101
69	Search Product	102
70	View Product	103
71	Wishlist	104
72	My Cart	104
73	Order Summary	105
74	Payment Gateway	105
75	View Sellers	106
76	Each Seller	106
77	Feeds	107
78	My Orders	108
79	Invoice Bill	108
80	Each Order	109
81	Customer Notification	109
82	Customer feedback	110
83	Customer Complaint	110

TABLE OF CONTENTS

Acknowledgment	iv
Abstract	vi
List of Tables	viii
List of Figures	ix
1. Introduction	1
1.1. Background and motivation	1
1.2. The proposed system	2
1.3. Project Scope	2
2. System Analysis	4
2.1. Introduction	4
2.2. Stake holders of this project	5
2.2.1. Admin	5
2.2.2. Seller	5
2.2.3. Customer	6
2.3. Software requirement specifications	7
2.3.1. System Features	7
2.3.1.1. Admin	7
2.3.1.2. Guest	7
2.3.1.3. Seller	8
2.3.1.4. Customer	8
2.3.2. Non-functional Requirements	9

2.4. Feasibility study	10
2.5. Software Development lifecycle model	12
2.6. Hardware and software requirements	13
2.6.1. Software specifications	13
2.6.1.1. HTML	14
2.6.1.2. CSS	14
2.6.1.3. AJAX	15
2.6.1.4. JQuery	15
2.6.1.5. Bootstrap	15
2.6.1.6. Javascript	15
2.6.1.7. PHP	16
2.6.1.8. MySQL	16
2.6.1.9. XAMPP Server	17
2.6.1.10. Windows 11	18
2.6.2. Hardware requirements	19
3. System Design	20
3.1. System Architecture	20
3.2. Module Design	21
3.3. Database Design	22
3.3.1. Normalization	23
3.3.2. Table Structure	24
3.3.3. Data Flow Diagram	35
3.3.3.1. Introduction to data flow diagrams	35
3.3.3.2. Data flow diagram	37
3.4. Interface Design	46
3.4.1. User interface screen design	46
3.4.2. Output design	49
4. Implementation	51
4.1. Coding Standard	51
4.2. Sample Code	52

5. Testing	61
5.1. Test Cases	61
5.1.1. Unit testing	62
5.1.2. Integration testing	64
5.1.3. Black box testing	66
5.1.4. White box testing	67
5.1.5. Validation Testing	67
5.1.6. User acceptance testing	70
5.2. Test case documents	70
6. Conclusion	73
6.1. Future Enhancements	73
7. References	74
8. Appendix	75
8.1. Screenshots	75

1. INTRODUCTION

1.1 BACKGROUND AND MOTIVATION

The name of the website is **Artisanal** and this system is an online handicraft store. This web store just as its name suggest only sells hand made products of small artisans (handicraftsmen).

We know most of the online webstores sells all types of products as whole without giving any priority and recognition to the people who sell these. And it is often found that same products are sold by different sellers in different prices. Which makes all these web store nothing more than a business platform. Artisanal is solely dedicated for Artisans who make their own products with their own creativity. The main goal of Artisanal is to promote this small Artisans/Sellers. Let the world know more about these artisans and give them the recognition and hype they deserve.

We know lots of artisans are gone unrecognized and are not appreciated as much as they deserve. some artisans wish to show their talent to the world and some people even keeps hand craft as a hobby it is always great to have a way to show your interest to the world and making a small income from that. Artisanal is a platform where these small craftsmen can sell their unique hand made products and get the recognition they deserve. Any artisans who would like to sell their work or want to make their interest into a small business can register in our website and sell their products. Sellers can make their own profiles and upload posts related to the products or works they have been busy with, share their journey as an Artisan through this. Which will help them to let the people more about them as a Artisan than a mere Seller.

There are also people who love or are interested in unique hand crafts products but are not able to get them because there is no common platform where you can find the art works of different artisans. Artisanal would be a good option for them. They can browse around, view different products and can buy the products they like online. And also support the Artisan/seller they like the most by following them in the platform, liking their post and adding supporting comments.

1.2 PROPOSED SYSTEM

The Objective of the website is to support small handicraft business and make a website that is fully dedicated to hand made products which attracts customer who like or prefer unique hand made products over machine-made mass-produced products. Artisanal sells a wide range of handmade products like paintings, bags, wallets, hats, key chains, cups, home decors etc. And simple cloths like scarves, mufflers, sweaters, Beanies etc. Artisanal wants to give an amazing online shopping experience to everyone who visits the site. The scope encompasses designing a comprehensive platform that caters to both customers and artisans/sellers, with features ranging from an intuitive interface to robust inventory management.

The system is menu driven and it has higher user friendliness which makes the user to handle the website more conveniently. So that the user can enter error free data. There are provisions for data entry and report generation. Admin monitors the data provided by all the users and can generate various reports

1.3 PROJECT SCOPE

Limitations of existing system

- Product Listings:
 - Handmade products are listed alongside other types of products, making it challenging for customers to specifically find and support artisan-made items.
- Product Categories:
 - Limited categorization for handmade products, making it difficult for customers to navigate and explore artisan offerings.
- Artisan Identification:
 - Artisan's identities and stories might not be prominently featured, reducing the personal connection customers can establish with the creators.
- Product Descriptions:
 - Product descriptions may not focus on the unique craftsmanship and the story behind each handmade item.

- Community Engagement:
 - Lack of features that enable artisans to interact with customers directly, such as sharing their creative process and updates.
- Product similarity:
 - Same products are sold by different seller in different prices.

Advantages of proposed system

- Artisan Profiles:
 - Create dedicated profiles for each artisan, showcasing their background, inspiration, and the process behind their creations.
- Handmade Category:
 - Introduce a prominent "Handmade" or "Artisan" category that highlights artisan-made products.
- Product Stories:
 - Craft detailed product descriptions that emphasize the craftsmanship, materials used, and the story of each handmade item.
- Featured Artisans:
 - Rotate featured artisans on the homepage, offering exposure to various unknown artisans and their work.
- Artisan Interaction:
 - Enable artisans to share updates, progress photos, and engage with customers through feed posts.
- Customer Reviews:
 - Allow customers to provide feedback on artisan products through comments sections and reviews, encouraging support for lesser-known creators.
- Product Uniqueness:
 - Each product are unique works of the artisans. So, each product would be different and special.

2. SYSTEM ANALYSIS

2.1. INTRODUCTION

Software Engineering is the analysis, design, construction, verification and management of technical or social entities. To engineer software accurately, a software engineering process must be defined. System analysis is a detailed study of the various operations performed by the system and their relationship within and module of the system. It is a structured method for solving the problems related to the development of a new system. The detailed investigation of the present system is the focal point of system analysis. This phase involves the study of parent system and identification of system objectives. Information has to be collected from all people who are affected by or who use the system. During analysis, data are collected on the variable files, decision point and transactions handled by the present system. The main aim of system is to provide the efficient and user friendly automation. So the system analysis process should be performed with extreme precision, so that an accurate picture of existing system, its disadvantages and the requirements of the new system can be obtained.

System analysis involves gathering the necessary information and using the structured tool for analysis. This includes the studying existing system and its drawback, designing a new system and conducting cost benefit analysis. System analysis is a problem-solving activity that requires intensive communication between the system users and system developers. The system is studied to the minute detail and analyzed. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through various phases of processing of inputs.

There are a number of different approaches to system analysis. When a computer based information system is developed, systems analysis (according to the Waterfall model) would constitute the following steps:

- The development of a feasibility study, involving determining whether a project is economically, technologically and operationally feasible.
- Conducting fact-finding measures, designed to ascertain the requirements of the system's end-users. These typically span interviews, questionnaires, or visual observations of work on the existing system.
- Gauging how the end-users would operate the system (in terms of general experience in using computer hardware or software), what the system would be used for and so on.

Techniques such as interviews, questionnaires etc. can be used for the detailed study of these processes. The data collected by these sources must be scrutinized to arrive at a conclusion.

The conclusion is an understanding of how the system functions. This system is called the Existing System. The Existing system is then subjected to close observation and the problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as a proposal which is the Proposed System. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is then presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

2.2. STAKE HOLDERS OF THIS PROJECT

2.2.1. ADMIN

The admin is a privileged user responsible for managing and overseeing the entire online store platform. The site Administrator is the person who manages the software. He is the person who focuses on the data and reports of the software. Information is the greatest power and it is he who is having the highest information. They have access to a comprehensive set of tools and functionalities to ensure the smooth operation of the store. Some of their key responsibilities include:

The admin has the authority to deactivate user accounts of sellers. Admins can manage categories. Admins have access to sales reports, traffic analytics, and other key performance indicators to make informed decisions. Admins are responsible for ensuring the security of the platform, including managing permissions and safeguarding customer data.

2.2.2. SELLER

Sellers are users who offer products for sale on the Artisanal. Each seller operates as an independent entity within the platform. Sellers can create and manage their own product listings, complete with images, descriptions, and pricing. They can track stock levels, update quantities, and manage product variants. They can view and process orders,

manage order statuses, and handle customer inquiries related to orders. Sellers receive notifications for new orders, and they are responsible for packaging and shipping the products to customers.

Sellers can interact with customers through the platform, addressing inquiries and providing support. Sellers have access to sales data and performance metrics for their products. Sellers can promote themselves by posting images /videos related to their products. Sellers can manage their seller bio. They can manage their account profile as well. They can send complaints and feedback to the admin regarding the system working. The system helps the vendor to analyze different data in the form of pie charts, bar charts, and excels reports.

2.2.3. CUSTOMER

Customers are the end-users of Artisanal, browsing products and making purchases. Their interactions with the system include these, Customers can search for products, filter results, and view detailed product pages with images and descriptions. Customers can add products to their Wishlist if they are not going to buy them soon but someday later. They can add products to their shopping cart, review cart contents, and adjust quantities before proceeding to checkout. Customers can place orders, provide shipping information, and select payment options. After placing an order, customers can view the status, and receive shipping notifications. Customers track order history, and save favorite products. Customers can leave reviews and ratings for products they've purchased, helping others make informed decisions. The customers can view posts by the seller like the posts comment on them to show their support and can also report on the posts if the there is something inapporiate about the post. Customer can report complaints about the services and working of the store if they have any also add feedback about the website helping the website improve.

2.3. SOFTWARE REQUIREMENTS SPECIFICATIONS

2.3.1. SYSTEM FEATURES

2.3.1.1. ADMIN

- The admin has the supreme power of the system.
- Artisanal should allow the admin to login to the system.
- The admin enters valid username and password then go to the admin home page otherwise display the error message like invalid username and password.
- The admin should verify and approve or reject seller that register.
- The admin shall able to view the customers and sellers details.
- Admin is responsible for maintaining and updating the whole system.
- Admin has the responsibility to notify the seller for any order of product made by the customer. And should also notify the seller when one product is left or when it is out of stock.
- Admin has to notify the customer regarding updates of the order they placed.
- Admin shall able to view reports on different customers and sellers registered.
- Admin can see reports regarding, which are most sold categories, which seller's products are most sold, which sellers have most followers and general report on customers orders.
- Admin can check feedbacks and reviews from the customers about the product and services. And should find solutions for the complaints of the customers
- Admin may receive requests and complaints from Sellers.
- The admin can manage about us pages.

2.3.1.2. GUESTS

- Anyone who visits the website is a guest.
- Artisanal allows the guest to browse through the website and view products.
- The Guest can sign up if they does not have an account or can login to do further activities.
- The Guest would be either a Seller or a Customer after login.

2.3.1.3. SELLERS

- The Sellers initially has to sign up and wait for the approval of admin.
- Once approved Artisanal shall allow the Seller to login to the system.
- The Seller enters valid username and password then goes to the seller home page otherwise display the error message like invalid username and password.
- The Seller should post their product details, select the category and sub categories which the product belongs to and upload images of their product.
- The seller can include posts of their product making and procedures in the feeds if they like to.
- The Seller can add different multiple products.
- The Seller can request to the admin for adding categories and subcategories if their product does not belong to the provided categories.
- The system should allow sellers to add, edit, and remove products from their inventory.
- The Seller should able to view the details of the customer who made an order.
- Seller can see details of orders received for each product, orders shipped, orders delivered and orders cancelled.
- Sellers can view report on their most sold products and the categories with most sales.
- The sellers should be notified if the products are out of stock.
- The sellers should update if new stocks are available.
- Sellers will be notified regarding new orders and stock updates.
- Seller can view products that are sold in particular periods.
- Seller is able to edit/update their personal details whenever they need.
- Sellers can create a profile if they like to which will be viewed and followed by the customers
- Sellers can add posts on feeds which are viewed and reacted by the customers

2.3.1.4. CUSTOMERS

- The customers can browse around and view products as they like.
- To purchase a product the customer should sign up.
- Artisanal shall allow the Customer to login to the system.

- The customer enters valid username and password then goes to the customer homepage. Otherwise display the error message like invalid username and password.
- The customer can add products to their cart, place order and make payment in Artisanal.
- They customers can specify the order address if the address is different from the default address (home address).
- The Customer can see a comprehensive product catalog with detailed information (e.g., name, description, price, ratings, images) for various products they select.
- The system should generate order confirmations with details such as order date, and total price.
- The customer can see updates and details on orders they made.
- The customers can add reviews and ratings about the products they purchased.
- Customer can add the products into wishlist if they are not planning to buy them soon.
- Customer can see the latest products added to the store.
- Customers should be able to search and browse products based on different criteria (e.g., category, brand, price range).
- The customers find the products by each sellers and can view posts by sellers in Feeds and can like, add comments and report the posts if something is not right about the post.
- The customer can follow different sellers if they like.
- The customer can update their personal details when they want.

2.3.2. NON-FUNCTIONAL REQUIREMENTS

Usability: The user interface of Artisanal should be intuitive, responsive, and provide a seamless shopping experience.

Performance: The system should handle a large number of concurrent users and provide fast response times for browsing, searching, and purchasing products.

Security: The system should implement robust security measures to protect user data, including secure payment processing and data encryption.

Reliability: Artisanal should be highly available and reliable, with minimal downtime and effective error handling mechanisms.

Scalability: The system should be designed to handle high traffic volumes, supporting a large number of products, sellers, and customers.

Compatibility: Artisanal should be compatible with major web browsers and mobile devices to ensure broad accessibility.

2.4. FEASIBILITY STUDY

Feasibility is defined as the practical extent to which a project can be performed successfully. To evaluate feasibility, a feasibility study is performed, which determines whether the solution considered to accomplish the requirements is practical and workable in the software. Information such as resource availability, cost estimation for software development, benefits of the software to the organization after it is developed and cost to be incurred on its maintenance are considered during the feasibility study. The objective of the feasibility study is to establish the reasons for developing the software that is acceptable to users, adaptable to change and conformable to established standards. Various other objectives of feasibility study are listed below.

- To analyse whether the software will meet organizational requirements.
- To determine whether the software can be implemented using the current technology and within the specified budget and schedule.
- To determine whether the software can be integrated with other existing software.

Various types of feasibility that we checked include technical feasibility, operational feasibility, and economic feasibility.

Technical Feasibility

Technical feasibility assesses the current resources (such as hardware and software) and technology, which are required to accomplish user requirements in the software within the allocated time and budget. For this, the software development team ascertains whether the current resources and technology can be upgraded or added in the software to accomplish specified user requirements. Technical feasibility also performs the following tasks.

- Analyses the technical skills and capabilities of the software development team members.
- Determines whether the relevant technology is stable and established.
- Ascertains that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.

From our perspective there are two languages PHP, HTML and database MySQL which are used to develop this web based applications. PHP is used in the front end and MySQL is used in the back end. The Word to the Wise is web based and thus can be accessed through any browsers. As we are using these latest technologies which are currently trending and used by a number of developers across the globe, we can say that our project is technically feasible.

Operational Feasibility

Operational feasibility assesses the extent to which the required software performs a series of steps to solve business problems and user requirements. This feasibility is dependent on human resources (software development team) and involves visualizing whether the software will operate after it is developed and be operative once it is installed. Operational feasibility also performs the following tasks.

- Determines whether the problems anticipated in user requirements are of high priority.
- Determines whether the solution suggested by the software development team is acceptable.
- Analyses whether users will adapt to a new software.
- Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

We found that our project will be satisfied for the client since we were discussing every detail about the software with the client at every step. The most important part of operational feasibility study is the input from client. So the software is built completely according to the requirements of the client. We have used the current industry standards for the software. Hence we can say that this software is operationally feasible.

Economic Feasibility

Economic feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software, cost of performing feasibility study, and so on. For this, it is essential to consider expenses made on purchases (such as hardware purchase) and activities required to carry out software development. In addition, it is necessary to consider the benefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below.

- Cost incurred on software development to produce long-term gains for an organization.
- Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis).
- Cost of hardware, software, development team, and training.

It is estimated that our project is economically feasible as development cost is very minimal since the tools and technologies used are available online. It's a group student project so there are no personnel costs. Development time is well planned and will not affect other operations and activities of the individuals. Once the system has been developed, the companies purchasing the system will be providing with a manual for training purposes. There is no need to purchase new hardware since the existing computers can still be used to implement the new system.

2.5. SOFTWARE DEVELOPMENT LIFECYCLE MODEL

One of the basic notions of the software development process is SDLC models which stand for Software Development Life Cycle models. SDLC – is a continuous process, which starts from the moment, when it's made a decision to launch the project, and it ends at the moment of its full remove from the exploitation. Software development lifecycle (SDLC) is a framework that defines the steps involved in the development of software. It covers the detailed plan for building, deploying and maintaining the software. SDLC defines the complete cycle of development i.e. all the tasks involved in gathering a requirement for the maintenance of a Product.

Some of the common SDLC models are Waterfall Model, V-Shaped Model, Prototype Model, Spiral Model, Iterative Incremental Model, Big Bang Model, Agile Model. We used Agile Model for our Project.

Agile Model

Agile Model is a combination of the Iterative and incremental model. This model focuses more on flexibility while developing a product rather than on the requirement. In the agile methodology after every development iteration, the client is able to see the result and understand if he is satisfied with it or he is not. Extreme programming is one of the practical use of the agile model.

The basis of this model consists of short meetings where we can review our project. In Agile, a product is broken into small incremental builds. It is not developed as a complete product in one go. At the end of each sprint, the project guide verifies the product and

after his approval, it is finalized. Client feedback is taken for improvement and his suggestions and enhancement are worked on in the next sprint. Testing is done in each sprint to minimize the risk of any failures.

Advantages of Agile Model:

- It allows more flexibility to adapt to the changes.
- The new feature can be added easily.
- Customer satisfaction as the feedback and suggestions are taken at every stage.
- Risks are minimized thanks to the flexible change process

Disadvantages:

- Lack of documentation.
- If a customer is not clear about how exactly they want the product to be, then the project would fail.
- With all the corrections and changes there is possibility that the project will exceed expected time

2.6. HARDWARE AND SOFTWARE REQUIREMENTS.

2.6.1. SOFTWARE SPECIFICATION

This project is built upon the latest technology software.

Front end	:	PHP
Development tool	:	PHP, HTML, JavaScript, CSS, Ajax, JQuery, Bootstrap
Database	:	MySQL
Web server	:	XAMPP
Operating System	:	Windows 11

2.6.1.1. HTML

HTML is a computer language devised to allow Website creation. These Websites can then be viewed by anyone else connected to the Internet. It is relatively easy to learn, with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create. It is constantly undergoing revision and evolution to meet the demands and requirements of the growing Internet audience under the direction of the W3C, the organization charged with designing and maintaining the language.

HTML consists of a series of short codes typed into a text-file by the site author - these are the tags. The text is then saved as a HTML file, and viewed through a browser, like Internet Explorer. This browser reads the file and translates the text into a visible form, hopefully rendering the page as the author had intended. Writing your own HTML entails using tags correctly to create your vision. You can use anything from a rudimentary text-editor to a powerful graphical editor to create HTML pages.

2.6.1.2. CSS

Stands for "Cascading Style Sheet." Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML. CSS helps Web developers create a uniform look across several pages of a Web site. Instead of defining the style of each table and each block of text within a page's HTML, commonly used styles need to be defined only once in a CSS document. Once the style is defined in cascading style sheet, it can be used by any page that references the CSS file. Plus, CSS makes it easy to change styles across several pages at once. For example, a Web developer may want to increase the default text size from 10pt to 12pt for fifty pages of a Web site. If the pages all reference the same style sheet, the text size only needs to be changed on the style sheet and all the pages will show the larger text.

While CSS is great for creating text styles, it is helpful for formatting other aspects of Web page layout as well. For example, CSS can be used to define the cell padding of table cells, the style, thickness, and color of a table's border, and the padding around images or other objects. CSS gives Web developers more exact control over how Web pages will look than HTML does. This is why most Web pages today incorporate cascading style sheets.

2.6.1.3. Ajax

Ajax is a set of web development techniques using many web technologies on the client side to create asynchronous web applications. With Ajax, web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behavior of the existing page. By decoupling the data interchange layer from the presentation layer, Ajax allows web pages and, by extension, web applications, to change content dynamically without the need to reload the entire page. In practice, modern implementations commonly utilize JSON instead of XML.

2.6.1.4. jQuery

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax. It is free, open source software using the permissive MIT License. As of May 2019, jQuery is used by 73% of the 10 million most popular websites. Web analysis indicates that it is the most widely deployed JavaScript library by a large margin, having 3 to 4 times more usage than any other JavaScript library.

2.6.1.5. Bootstrap

Bootstrap is a free and open-source CSS framework directed at responsive, mobile first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation and other interface components.

2.6.1.6. JavaScript

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a 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.

JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose

core of the language has been embedded in Netscape, Internet Explorer, and other Web browsers.

2.6.1.7. PHP

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

PHP is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain. The key advantages of learning PHP are:

PHP is a recursive acronym for "PHP: Hypertext Preprocessor". PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites. It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server. PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time. PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time. PHP is forgiving: PHP language tries to be as forgiving as possible. PHP Syntax is C-Like.

2.6.1.8. MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius's daughter and "SQL", the abbreviation for Structured Query Language. MySQL is free and open-source software under the terms of the GNU General Public License, and is also available under a variety of proprietary licenses. MySQL was owned and sponsored by the Swedish company MySQL AB, which was bought by Sun Microsystems (now Oracle Corporation). In 2010, when Oracle acquired Sun, Widenius forked the open-source MySQL project to create MariaDB.

MySQL is a component of the LAMP web application software stack (and others), which is an acronym for Linux, Apache, MySQL, Perl/PHP/Python. MySQL is used by many database-driven web applications, including Drupal, Joomla, phpBB, and WordPress. MySQL is also used by many popular websites, including Facebook, Flickr, MediaWiki, Twitter, and YouTube

2.6.1.9. XAMPP Server

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible.

XAMPP's ease of deployment means a WAMP or LAMP stack can be installed quickly and simply on an operating system by a developer. With the advantage of common add-in applications such as WordPress and Joomla! can also be installed with similar ease using Bitnami.

A database system is an overall collection of different database software components and database containing the part viz. Database application programs, front-end components, Database management systems and Database.

A database system must provide the following features:

- A variety of user interfaces
- Physical data independence
- Logical data independence
- Query optimization
- Data integrity
- Concurrency control
- Backup and recovery
- Security and authentication

When creating a database, the main concept is to know how the database is structured in SQL. SQL stands for Structured Query Language. It is a language that enables us to create

and operate on relational database, which are sets of related information stored in tables. Because of its elegance and independence.

2.6.1.10. Windows 11

Windows 11 is the latest major release of Microsoft's Windows NT operating system, released on October 5, 2021. It succeeded Windows 10 (2015) and is available for free for any Windows 10 devices that meet the new Windows 11 system requirements.

Windows 11 features major changes to the Windows shell influenced by the canceled Windows 10X, including a redesigned Start menu, the replacement of its "live tiles" with a separate "Widgets" panel on the taskbar, the ability to create tiled sets of windows that can be minimized and restored from the taskbar as a group, and new gaming technologies inherited from Xbox Series X and Series S such as Auto HDR and Direct Storage on compatible hardware. Internet Explorer (IE) has been replaced by the Chromium-based Microsoft Edge as the default web browser, like its predecessor, Windows 10, and Microsoft Teams is integrated into the Windows shell. Microsoft also announced plans to allow more flexibility in software that can be distributed via the Microsoft Store and to support Android apps on Windows 11 (including a partnership with Amazon to make its app store available for the function).

Citing security considerations, the system requirements for Windows 11 were increased over Windows 10. Microsoft only officially supports the operating system on devices using an eighth-generation Intel Core CPU or newer (with some minor exceptions), a second-generation AMD Ryzen CPU or newer, or a Qualcomm Snapdragon 850 ARM system-on-chip or newer, with UEFI and Trusted Platform Module (TPM) 2.0 supported and enabled (although Microsoft may provide exceptions to the TPM 2.0 requirement for OEMs). While the OS can be installed on unsupported processors, Microsoft does not guarantee the availability of updates. Windows 11 removed support for 32-bit x86 and 32-bit ARM CPUs and devices that use BIOS firmware.

Windows 11 has received a mostly positive reception. Pre-release coverage of the operating system focused on its stricter hardware requirements, with discussions over whether they were legitimately intended to improve the security of Windows or as a ploy to upsell customers to newer devices and over the e-waste associated with the changes. Upon release, it was praised for its improved visual design, window management, and stronger focus on security, but was criticized for various modifications to aspects of its user interface

that were seen as worse than its predecessor; some were seen as an attempt to dissuade users from switching to competing applications.

As of October 2023, Windows 11, at 24.42% worldwide, is the second most popular Windows version in use, with its predecessor Windows 10 at three times the market share. Windows 11 has an estimated 16.62% share of all PCs (the rest being other Windows editions and other operating systems such as macOS and Linux), and an estimated 7.54% share of all devices (including mobile, tablet and console) are running Windows 11.

2.6.2. Hardware requirements

The selection of hardware configuring is a very task related to the software development, particularly inefficient RAM may affect adversely on the speed and corresponding on the efficiency of the entire system. The processor should be powerful to handle all the operations. The hard disk should have the sufficient to solve the database and the application.

Hardware used for development:

CPU : Intel i5

Processor Memory : 16 GB

Cache : 12 MB

SSD : 512 TB

Monitor : 14.6”

Monitor Keyboard : Standard 108 keys Enhanced Keyboard

Mouse : Optical Mouse

Minimum Hardware Required For Implementation:

CPU : Pentium IV Processor

Memory : 256MB Above

Cache : 512 KB Above

Hard Disk : 20 GB Above

Monitor : Any

Keyboard : Any

Mouse : Any

3. SYSTEM DESIGN

3.1. SYSTEM ARCHITECTURE

A system architecture or system's architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures of the system,

System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behavior) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture; collectively these are called architecture description languages (ADLs).

The system architecture can best be thought of as a set of representations of an existing (or to be created) system. It is used to convey the informational content of the elements comprising a system, the relationships among those elements, and the rules governing those relationships. The architectural components and set of relationships between these components that architecture describes may consist of hardware, software, documentation, facilities, manual procedures, or roles played by organizations or people.

System architecture is primarily concerned with the internal interfaces among the system's components or subsystems, and the interface between the system and its external environment, especially the user.

The structural design reduces complexity, facilitates change and result in easier implementation by encouraging parallel development of different parts of the system. The procedural design transforms structural elements of program architecture into a procedural description of software components. The architectural design considers architecture as the most important functional requirement. The system is based on the three-tier architecture.

The first level is the user interface (presentation logic), which displays controls, receives and validates user input. The second level is the business layer (business logic) where the application specific logic takes place. The third level is the data layer where the application information is stored in files or database. It contains logic about to retrieve and update data. The important feature about the three-tier design is that information only travels from one level to an adjacent level.

3.2. MODULE DESIGN

Modular programming is a software design technique that emphasizes separating the functionality of a program into independent, interchangeable modules, such that each contains everything necessary to execute only one aspect of the desired functionality. Conceptually, modules represent a separation of concerns, and improve maintainability by enforcing logical boundaries between components. Different modules of this project include.

1. User Authentication

This module allows the stakeholders to authenticate themselves to the system. This is mainly for authentication purposes. This improves the security of the system. This module allow admin to log in to the system. Admin can log into the system by using their corresponding username and password. Admin can perform all the actions in the system only after login. This module also includes sellers login. sellers log into the system using the username and password specified by the admin during registration. After when seller is login it's able to access the details and connect to the system. seller can generate reports and find details only after login. This module also allows Customer login. Customer logs into the system using username and password created during registration. so he can view the information and connect with the seller.

2. Registration

This module contains the all registration process in the system. There is much registration in the system. All registration specified in the system is included for the smooth running of the system. This module includes the registrations that can be performed by all stakeholders. Admin can register the details like name, email. The seller can register in the system. They should provide details like name, email, contact, address, profile photo, proof. This registered information is helpful to provide meaning full knowledge to other stakeholders. The customer also should register to use the site the customer should provide details like name, email, address, contact etc.

3. Activities

This module includes the activities that can be performed by the stakeholders in the system. There are many activities that the stakeholders can perform. Activities done by admin are verifying sellers, managing categories, subcategories etc admin can view notification.

Activities of sellers include adding products, uploading posts, managing orders and editing their profile.

Sellers can report complaints and send feedback to the admin. The customer can edit their profile, view products they want , add products to wishlist, add products to cart, make payment and buy the products the customers can view their orders. They can view feeds, view and follow sellers. Customers can review the products they bought. Customers can also sent complaints and feedbacks to the admin.

4. Reports

This module allows the owner of the site to view the details of the site through various reports. This includes a pie chart, bar chart, PDF, and some table information. The reports provide valuable information from the system. In our system, all reports are important because the system eliminates the use of manipulation of the paperwork. The system generates useful information. Admin can generate many reports such as products sold by categories, most followed sellers, most product sold customer, list of orders received. So that the owner can easily manage the website. sellers can also generate reports that include sales report on their products. This will helpful for the sellers to generate useful information. The reports show the significance of the system.

3.3. DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the users. The general theme behind a database is to integrate all information. Database design is recognized as a standard of management information system and is available virtually for every computer system. In database design several specific objectives are considered:

- Ease of learning and use
- Controlled redundancy
- Data independence
- More information at low cost
- Accuracy and integrity
- Recovery from failure

- Privacy and security
- Performance

A database is an integrated collection of data and provides centralized access to the data. Usually the centralized data managing the software is called RDBMS. The main significant difference between RDBMS and other DBMS is the separation of data as seen by the program and data has direct access to stores device. This is the difference between logical and physical data.

3.3.1. Normalization

Designing a database is complete task and the normalization theory is a useful aid in the design process. The process of normalization is concerned with transformation of conceptual schema into computer representation form. There will be need for most databases to grow by adding new attributes and new relations. The data will be used in new ways. Tuples will be added and deleted. Information stored may undergo updating also. New association may also be added. In such situations the performance of a database is entirely depend upon its design. A bad database design may lead to certain undesirable things like:

- Repetition of information
- Inability to represent certain information
- Loss of information

To minimize these anomalies, Normalization may be used. If the database is in a normalized form, the data can be growing without, in most cases, forcing the rewriting application programs. This is important because of the excessive and growing cost of maintaining an organization's application programs and its data from the disrupting effects of database growth. As the quality of application programs increases, the cost of maintaining the without normalization will rise to prohibitive levels. A normalized database can also encompass many related activities of an organization thereby minimizing the need for rewriting the applications of programs. Thus, normalization helps one attain a good database design and there by ensures continued efficiency of database.

Normalization theory is built around the concept of normal forms. A relation is said to be in normal form if it satisfies a certain specified set of constraints. For example, a relation is said to be in first normal form (1NF) if it satisfies the constraint that it contains atomic values only.

Thus every normalized relation is in 1NF. Numerous normal forms have been defined. Codd defined the first three normal forms.

All normalized relations are in 1NF, some 1NF relations are also in 2NF and some 2NF relations are also in 3NF. 2NF relations are more desirable than 1NF and 3NF are more desirable than 2NF. That is, the database designer should prefer 3NF than 1NF or 2NF.

Normalization procedure states that a relation that is in some given normal form can be converted into a set of relations in a more desirable form. We can define this procedure as the successive reduction of a given collection of relations to some more desirable form. This procedure is reversible. That is, it is always possible to take the output from the procedure and convert them back into input. In this process, no information is lost. So it is also called “no loss decomposition”.

First Normal Form

A relation is in first normal form (1NF) if and all its attributes are based on single domain. The objective of normalizing a table is to remove its repeating groups and ensure that all entries of the resulting table have at most single value.

Second Normal Form

A table is said to be second Normal Form (2NF), when it is in 1NF and every attribute in record is functionally dependent upon the whole key, and not just a part of the key.

Third Normal Form

A table is in third Normal Form (3NF), when it is in 2NF and every non-key attribute is functionally dependent on just the primary key.

3.3.2. Table Structure

Table is a collection of complete details about a particular subject. These data are saved in rows and Columns. The data of each Row are different units. Hence, rows are called RECORDS and Columns of each row are called FIELDS.

Data is stored in tables, which is available in the backend the items and data, which are entered in the input, form id directly stored in this table using linking of database. We can link more than one table to input forms. We can collect the details from the different tables to display on the output.

There are mainly 25 tables in the project. They are

1. tbl_admin
2. tbl_district
3. tbl_city
4. tbl_category
5. tbl_subcategory
6. tbl_seller
7. tbl_type
8. tbl_product
9. tbl_product_gallery
10. tbl_post
11. tbl_likes
12. tbl_comments
13. tbl_follow
14. tbl_seller_bio
15. tbl_stock
16. tbl_post_report
17. tbl_notification
18. tbl_customer
19. tbl_wishlist
20. tbl_cart
21. tbl_order
22. tbl_payment
23. tbl_complaint
24. tbl_feedback
25. tbl_rating

1. Table : tbl_admin

Description: This table is used to store details of admin.

Field name	Data Type	Constraints	Description
admin_id	int	Primary key	Unique id of admin
ad_name	varchar(40)	Not NULL	Name of the admin
ad_email	varchar(40)	Not NULL	Email of the admin
ad_password	Varchar(12)	Not NULL	Password set by the admin

Table 3.1 tbl_admin

2. Table : tbl_district

Description: This table is used to store districts.

Field name	Data Type	Constraints	Description
district_id	int	Primary key	Unique id of district
district_name	varchar(30)	Not NULL	Name of the district

Table 3.2 tbl_district

3. Table : tbl_city

Description: This table is used to store cities and corresponding pincode.

Field name	Data Type	Constraints	Description
city_id	int	Primary key	Unique id of city
city_name	varchar(40)	Not NULL	Name of the city
pincode	varchar(8)	Not NULL	pincode of the city
district_id	int	Foreign key	Unique id of district

Table 3.3 tbl_city

4. Table: tbl_category

Description : This table is used to store categories.

Field name	Data Type	Constraints	Description
category_id	int	Primary key	Unique id of category
category_name	varchar(30)	Not NULL	Name of the category

Table 3.4 tbl_category

5. Table: tbl_subcategory

Description : This table is used to store subcategories.

Field name	Data Type	Constraints	Description
subcat_id	int	Primary key	Unique id of subcategory
subcat_name	varchar(30)	Not NULL	Name of the subcategory
category_id	int	Foreign key	Unique id of category

Table 3.5 tbl_subcategory

6. Table: tbl_seller

Description : This table is used to store details of the seller.

Field name	Data Type	Constraints	Description
seller_id	int	Primary key	Unique id of seller
sell_name	varchar(30)	Not NULL	Name of the seller
sell_email	varchar(40)	Not NULL	Email of the seller
sell_address	varchar(200)	Not NULL	Seller address
sell_contact	varchar(30)	Not NULL	Contact number of the seller
sell_photo	varchar(200)	Not NULL	Id proof of seller for verification
sell_proof	varchar(300)	Not NULL	passport size photo of Seller
sell_doj	varchar(12)	Not NULL	Seller's Date of joining
sell_password	varchar(30)	Not NULL	Password set by the seller
sell_otp_status	int	Not NULL	Indicates the verification status by otp
sell_ver_status	int	Not NULL	Indicates the verification status by admin
sell_bio_status	int	Not NULL	Indicates if the seller bio is set or not
city_id	int	Foreign key	Unique id of city

Table 3.6 tbl_seller

7. Table: tbl_type

Description : This table is used to store type of the product.

Field name	Data Type	Constraints	Description
type_id	int	Primary key	Unique id of type
type_name	varchar(30)	Not NULL	Name of the type

Table 3.7 tbl_type

8. Table : tbl_product

Description : This table is used to store details of the products sold.

Field name	Data Type	Constraints	Description
product_id	int	Primary key	Unique id of product
prod_name	varchar(75)	Not NULL	Name of the product
prod_color	varchar(40)	Not NULL	Color of the product
prod_material	varchar(30)	Not NULL	Material the product made of.
type_id	varchar(20)	Not NULL	Product type
prod_img	varchar(200)	Not NULL	Preview image of product
prod_price	int	Not NULL	price of the product
prod_tag	varchar(200)	Not NULL	Search keys of the products
prod_description	varchar(200)	none	description of the product
prod_date	varchar(12)	Not NULL	Date of product added
seller_id	int	Foreign key	Unique id of seller
subcat_id	int	Foreign key	Unique id of subcategory

Table 3.8 tbl_product

9. Table : tbl_product_gallery

Description: This table is used to store product images.

Field name	Data Type	Constraints	Description
product_gallery_id	int	Primary key	Unique id of product gallery
product_img	varchar(400)	Not NULL	name of the product image
product_id	int	foreign key	Unique id of product

Table 3.9 tbl_product_gallery

10. Table: tbl_post

Description: This table is used to store the posts by the sellers.

Field name	Data Type	Constraints	Description
post_id	int	Primary key	Unique id of post
post_caption	varchar(40)	Not NULL	caption of the post
post_type	Varchar(20)	Not NULL	Whether the content is image or video
post_media	varchar(300)	Not NULL	Image/video posted
post_content	Varchar(500)	Not NULL	Detailed content post
created_on	Varchar(30)	Not NULL	Posted date
seller_id	int	Foreign Key	Unique id of seller
product_id	int	foreign key	Unique id of product

Table 3.10 tbl_post

11. Table : tbl_likes

Description: This table is used to store the count of likes of posts.

Field name	Data Type	Constraints	Description
like_id	int	Primary key	Unique id of like
customer_id	int	Foreign key	Unique id of customer
post_id	int	foreign key	Unique id of post

Table 3.11 tbl_likes

12. Table : tbl_comments

Description: This table is used to store the comments of posts.

Field name	Data Type	Constraints	Description
comment_id	int	Primary key	Unique id of comment
comment_text	varchar(40)	Not NULL	Comment by customers
customer_id	int	Foreign key	Unique id of customer
post_id	int	foreign key	Unique id of post
created_on	varchar(20)	Not NULL	Date commented on

Table 3.12 tbl_comment

13. Table : tbl_follow

Description: This table is used to store the count of followers of each seller.

Field name	Data Type	Constraints	Description
follow_id	int	Primary key	Unique id of follow
seller_id	int	Foreign key	Unique id of seller
customer_id	int	foreign key	Unique id of customer

Table 3.13 tbl_follow

14. Table : tbl_seller_bio

Description: This table is used to store the details of seller bio set by seller.

Field name	Data Type	Constraints	Description
seller_bio_id	int	Primary key	Unique id of seller bio
bioNickname	varchar(30)	Not NULL	Nickname on bio.
bio_email	varchar(30)	Not NULL	email on bio.
bio_details	varchar(200)	Not NULL	About self on bio.
sell_profilepic	varchar(200)	Not NULL	Profile pic set on bio.
seller_id	int	Foreign key	Unique id of seller

Table 3.14 tbl_seller_bio

15. Table : tbl_stock

Description: This table is used to store product stock details.

Field name	Data Type	Constraints	Description
stock_id	int	Primary key	Unique id of stock
product_id	int	foreign key	Unique id of product
stock_count	int	Not NULL	Count of the stock added
added_on	varchar(20)	Not NULL	Date the stock added.

Table 3.15 tbl_stock

16. Table : tbl_post_report

Description: This table is used to store the reported posts.

Field name	Data Type	Constraints	Description
report_id	int	Primary key	Unique id of report
report_content	varchar(100)	Not NULL	Reason for reporting.
customer_id	int	Foreign key	Unique id of customer
post_id	int	foreign key	Unique id of post
report_time	varchar(20)	Not NULL	time reported on

Table 3.16 tbl_post_report

17. Table: tbl_notification

Description : This table is used to store the details of the notifications.

Field name	Data Type	Constraints	Description
notification_id	int	Primary key	Unique id of notification
notification_type	varchar(50)	Not NULL	What the notification is about
notification_class	varchar(30)	Not NULL	Icon of each notification
message	varchar(100)	Not NULL	Content of the notification
seller_id	int	Foreign key	Unique id of seller
customer_id	int	Foreign key	Unique id of customer
admin_id	int	Foreign key	Unique id of admin
notification_status	int	Not NULL	Unique id of notification
Sent_on	varchar(20)	Not NULL	Date the notification was sent.

Table 3.17 tbl_notification

18. Table: tbl_customer

Description : This table is used to store the details of the customers.

Field name	Data Type	Constraints	Description
customer_id	int	Primary key	Unique id of customer
cus_name	varchar(50)	Not NULL	Name of the customer
cus_email	varchar(40)	Not NULL	Email of the customer
cus_dob	varchar(12)	Not NULL	date of Birth of the customer
cus_gender	varchar(6)	none	Gender of the customer
cus_address	varchar(200)	Not NULL	address of the customer
cus_contact	varchar(12)	Not NULL	Contact number of the customer
cus_password	varchar(30)	Not NULL	Password set by the customer
cus_photo	varchar(200)	none	profile photo of the customer
city_id	int	Foreign key	Unique id of city
cus_otp_status	int	Not NULL	Status of otp verification

Table 3.18 tbl_customer

19. Table : tbl_wishlist

Description: This table is used to store products that are added to wishlist by customers.

Field name	Data Type	Constraints	Description
wishlist_id	int	Primary key	Unique id of wishlist
product_id	int	foreign key	Unique id of product
customer_id	int	foreign key	Unique id of customer

Table 3.19 tbl_wishlist

20. Table : tbl_cart

Description: This table is used to store product details that are added to cart by customers.

Field name	Data Type	Constraints	Description
cart_id	int	Primary key	Unique id of cart
product_id	int	foreign key	Unique id of product
order_id	int	foreign key	Unique id of order
cart_quantity	int	Not NULL	Count of each product in cart
cart_status	int	Not NULL	Status of the product ordered

Table 3.20 tbl_cart

21. Table : tbl_order

Description : This table is used to store details of the orders received.

Field name	Data Type	Constraints	Description
order_id	int	Primary key	Unique id of order
order_date	varchar(12)	Not NULL	Date the order was made
order_status	int	Not NULL	Status of the order
order_price	int	Not NULL	Total price of the order
order_quantity	int	Not NULL	Number of the products ordered
delivery_address	varchar(50)	Not NULL	Delivery address of order
customer_id	int	foreign key	Unique id of customer

Table 3.21 tbl_order

22. Table : tbl_payment

Description: This table is used to store products that are added to wishlist by customers.

Field name	Data Type	Constraints	Description
payment_id	int	Primary key	Unique id of payment
order_id	int	foreign key	Unique id of order
pay_method	varchar(50)	Not NULL	Method of payment
pay_amount	int	Not NULL	Amount to be paid
pay_time	varchar(30)	Not NULL	Time the payment was made

Table 3.22 tbl_payment

23. Table : tbl_complaint

Description: This table is used to store the complaints of the users.

Field name	Data Type	Constraints	Description
complaint_id	int	Primary key	Unique id of complaints
complaint_title	varchar(40)	Not NULL	Title of the complaint
complaint_content	varchar(200)	Not NULL	Complaint in detail
complaint_status	int	Not NULL	Status of the complaint
seller_id	int	Foreign key	Unique id of seller
customer_id	int	foreign key	Unique id of customer
created_on	varchar(20)	Not NULL	Date of complaint

Table 3.23 tbl_complaint

24. Table : tbl_feedback

Description: This table is used to store the feedbacks by the users.

Field name	Data Type	Constraints	Description
feedback_id	int	Primary key	Unique id of feedback
feedback_content	varchar(200)	Not NULL	Feedback by the users
feedback_time	varchar(20)	Not NULL	Timestamp the feedback was received
product_id	int	foreign key	Unique id of product
customer_id	int	foreign key	Unique id of customer

Table 3.24 tbl_feedback

25. Table : tbl_rating

Description: This table is used to store the ratings by the users.

Field name	Data Type	Constraints	Description
rating_id	int	Primary key	Unique id of rating
rating_value	float	Not NULL	rating by the users
rating_comment	varchar(50)	Not NULL	Review according to rating
customer_id	int	foreign key	Unique id of customer
cart_id	int	foreign key	Unique id of cart
rating_date	varchar(20)	Not NULL	Date rating recorded on.

Table 3.25 tbl_rating

3.3.3. Data Flow Diagram

3.3.3.1. Introduction to Data Flow Diagrams

Data Flow Diagram is a network that describes the flow of data and processes that change, or transform, data throughout the system. This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

There are various symbols used in a DFD. Bubbles represent the processes. Named arrows indicate the data flow. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. Each component in a DFD is labelled with a descriptive name. Process names are further identified with a number.

The Data Flow Diagram shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the input (source), outputs (destination), database (files) and procedures (data flow), all in a format that meet the user's requirements.

The main merit of DFD is that it can provide an overview of system requirements, what data a system would process, what transformations of data are done, what files are used, and where the results flow.

This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

Rules for constructing a Data Flow Diagram

1. Arrows should not cross each other
2. Squares, circles and files must bear names.
3. Decomposed data flow squares and circles can have same time
4. Choose meaningful names for data flow
5. Draw all data flows around the outside of the diagram

Basic Data Flow Diagram Symbols

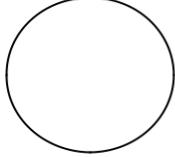
	<p>A data flow is a route, which enables packets of data to travel from one point to another. Data may flow from a source to a process and from data store or process. An arrow line depicts the flow, with arrow head pointing in the direction of the flow.</p>
	<p>Circles stands for process that converts data into information. A process represents transformation where incoming data flows are changed into outgoing data flows.</p>
	<p>A data store is a repository of data that is to be stored for use by a one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names. If a process merely uses the content of store and does not alter it, the arrowhead goes only from the store to the process. If a process alters the details in the store then a double-headed arrow is used.</p>
	<p>A source or sink is a person or part of an organization, which enters or receives information from the system, but is considered to be outside the contest of data flow model.</p>

Table 3.26 Data Flow Diagram Symbols

3.3.3.2. Data Flow Diagram

Each component in a DFD is labelled with a descriptive name. Process name are further identified with number. Context level DFD is draw first. Then the process is decomposed into several elementary levels and is represented in the order of importance. A DFD describes what data flow (logical) rather than how they are processed, so it does not depend on hardware, software, and data structure or file organization.

A DFD methodology is quite effective; especially when the required design.

3.3.3.2.1. Zeroth level DFD for Artisanal

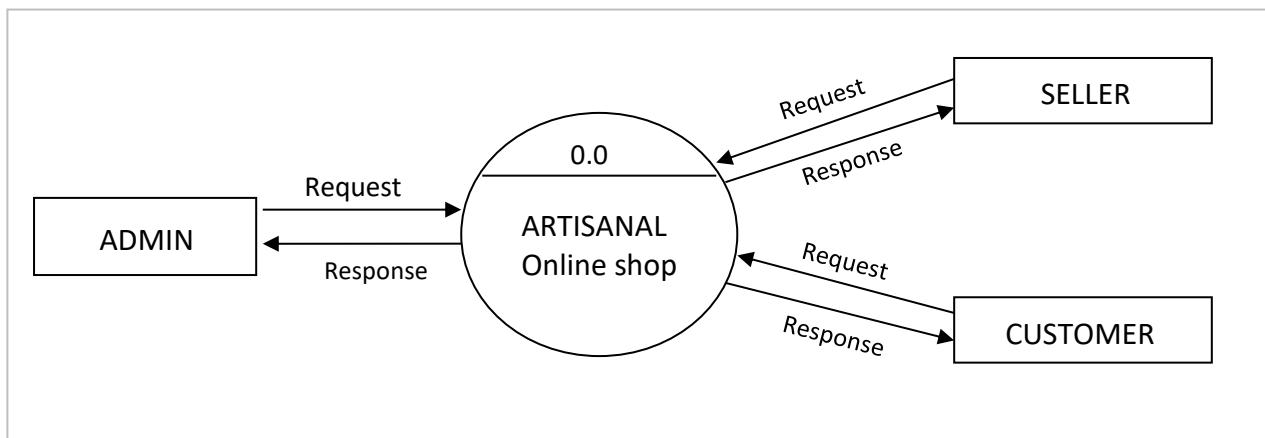


Fig 3.1 Zeroth level DFD for Artisanal

3.3.3.2.2. First level DFD for Artisanal

1. Admin side

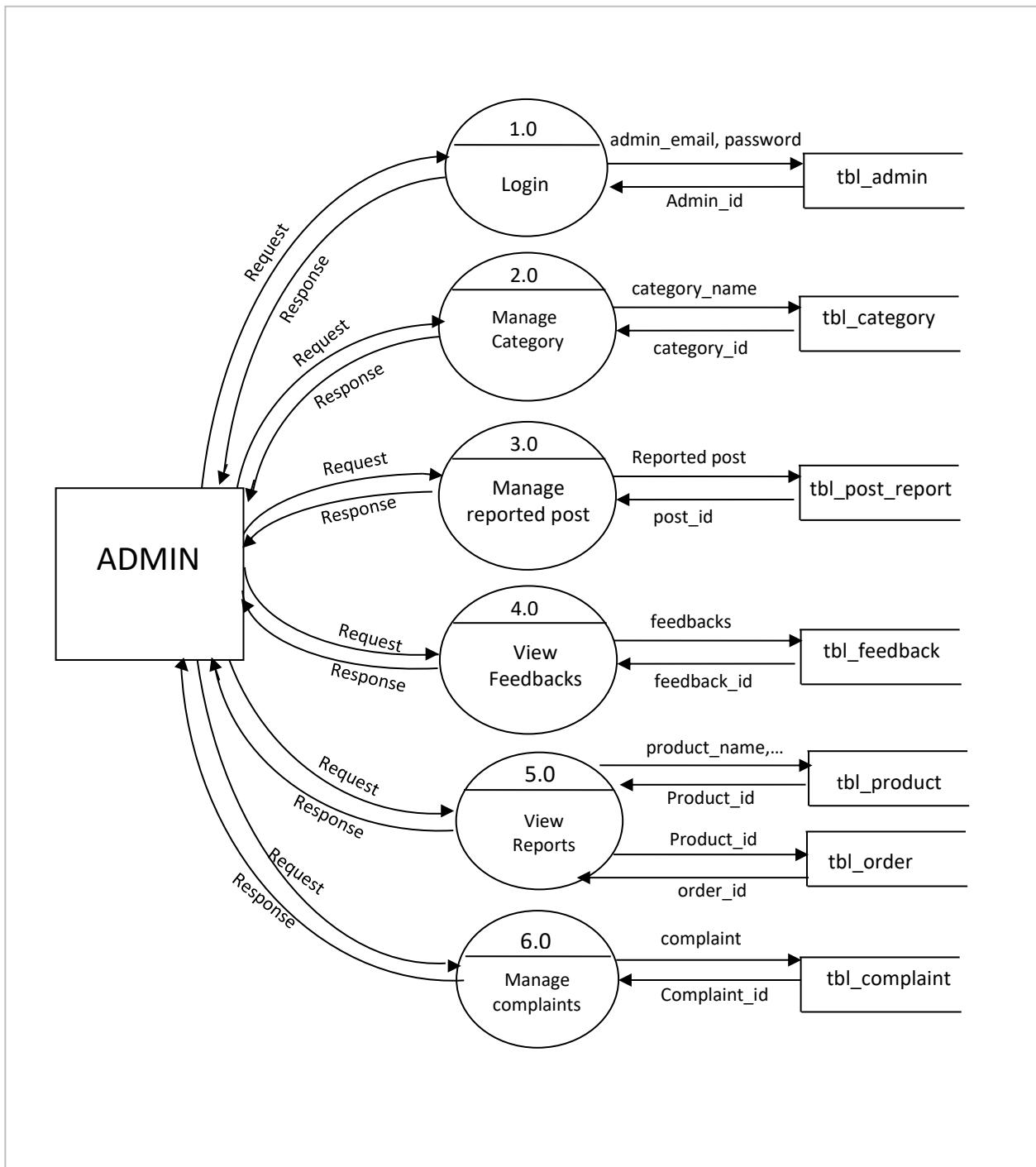


Fig 3.2 1st level DFD for Artisanal admin side

2. Seller side

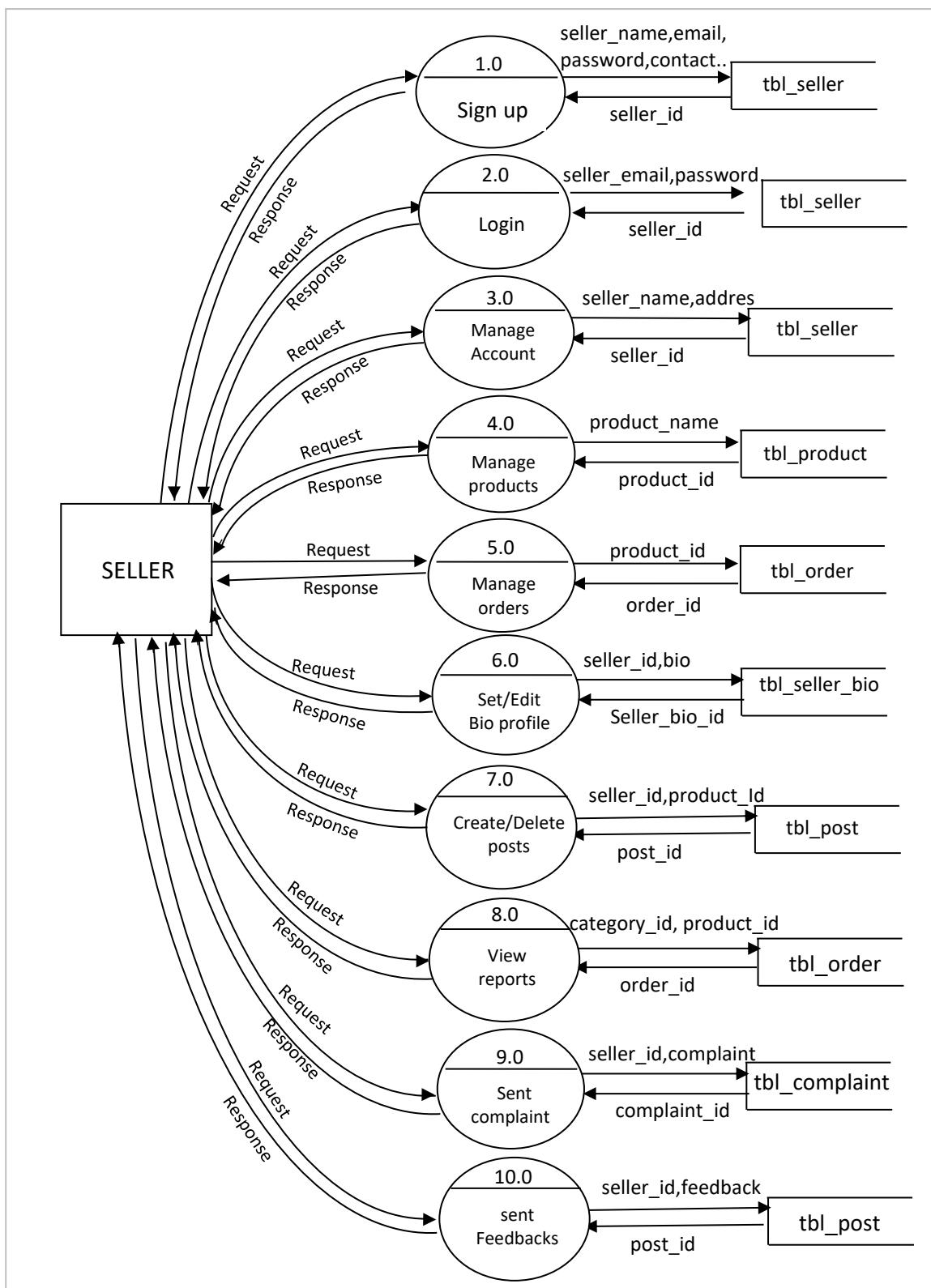


Fig 3.3 1st level DFD for Artisanal seller side

3. Customer Side

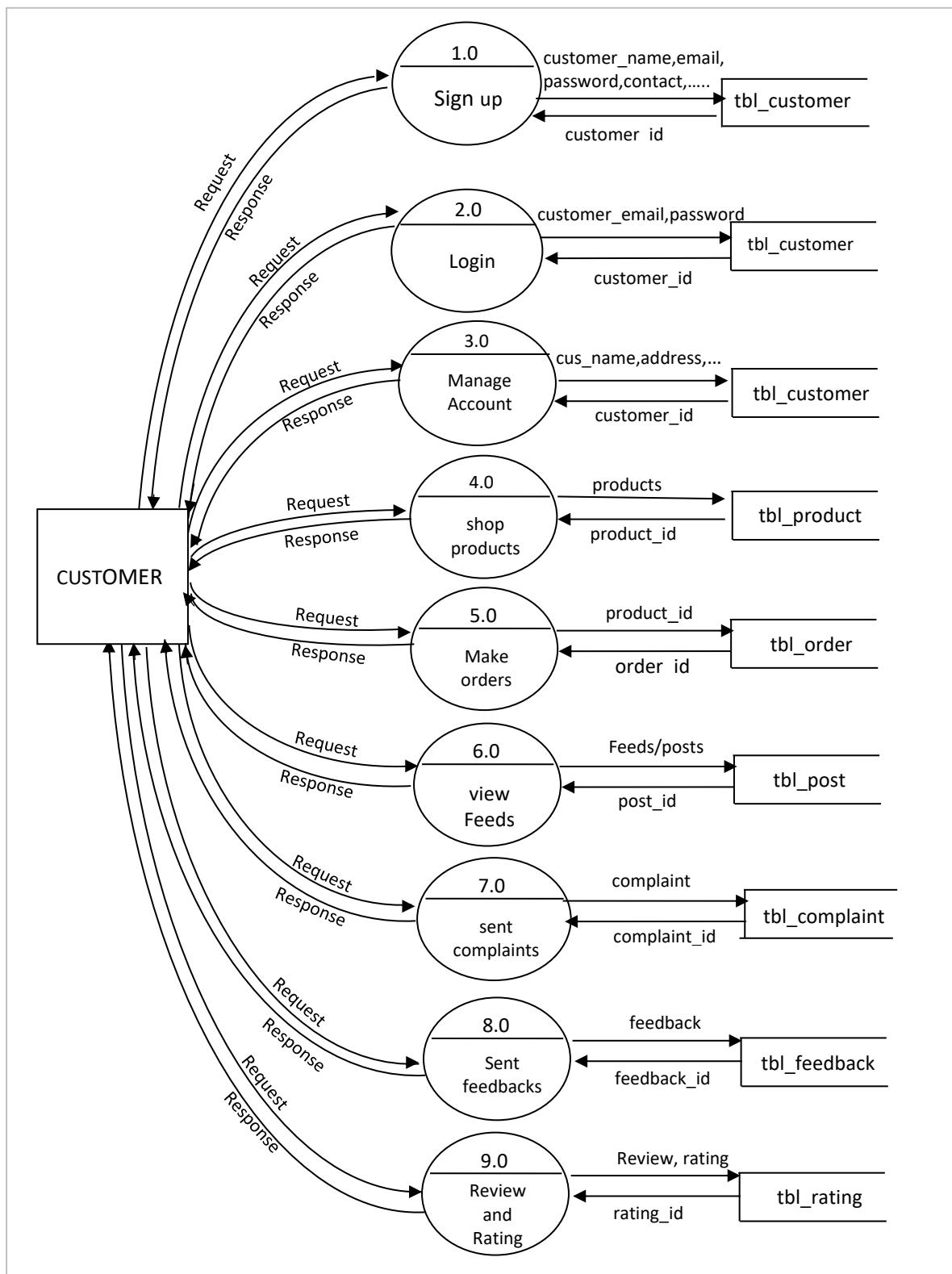


Fig 3.4 1st level DFD for Artisanal customer side

3.3.3.2.3. Second level DFD for Artisanal

1. Admin Side

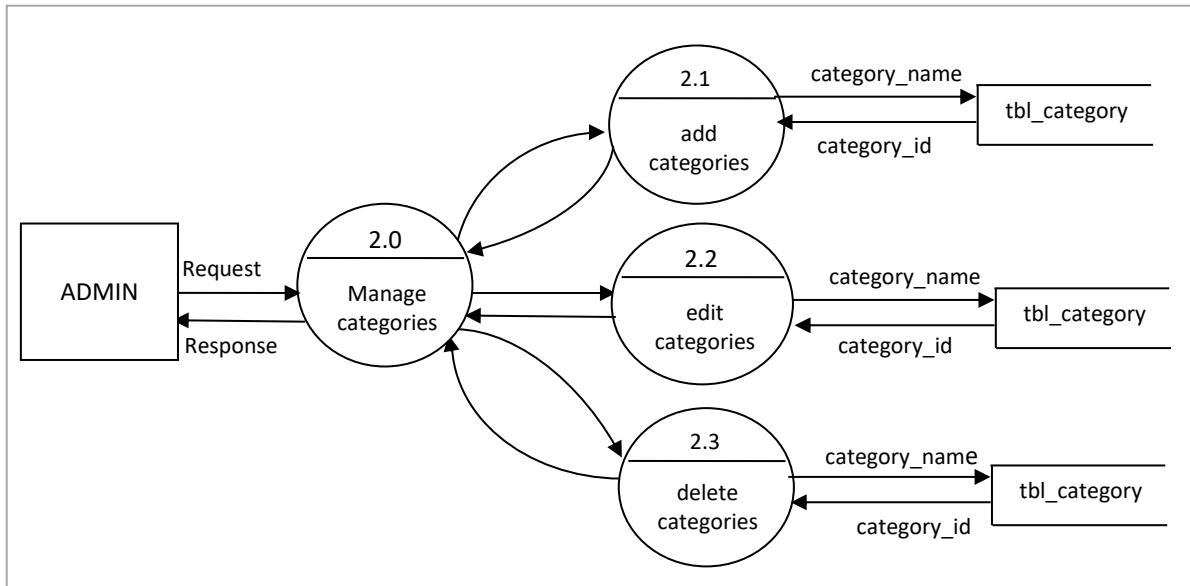


Fig 3.5 2nd level DFD for Artisanal admin side Manage Categories

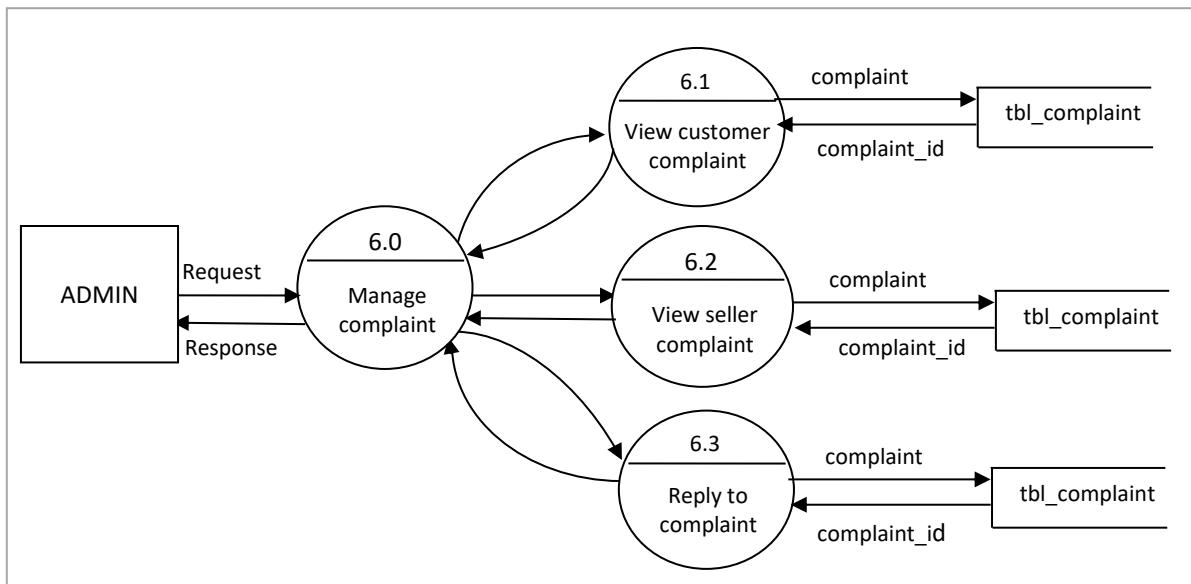


Fig 3.6 2nd level DFD for Artisanal admin side Manage Complaint

2. Seller side

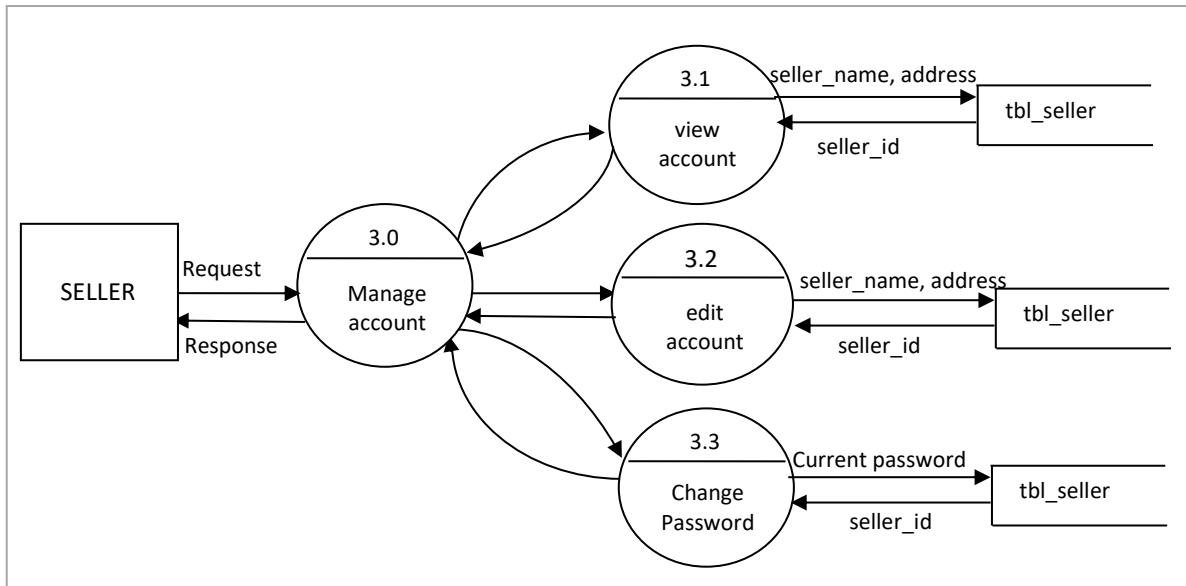


Fig 3.7 2nd level DFD for Artisanal seller side Manage Account

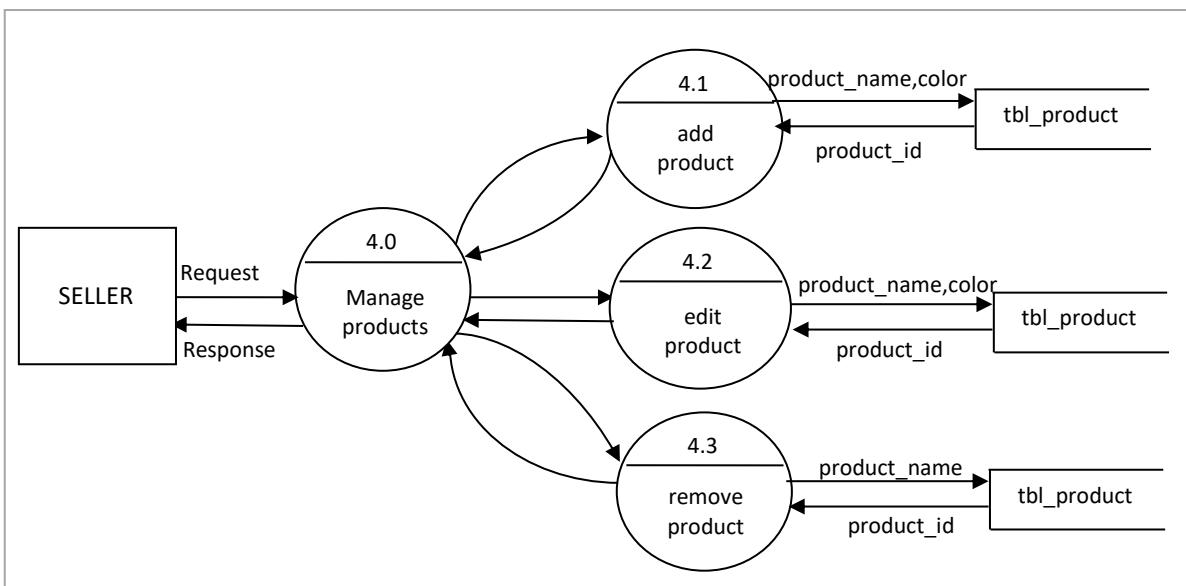


Fig 3.8 2nd level DFD for Artisanal seller side Manage Products

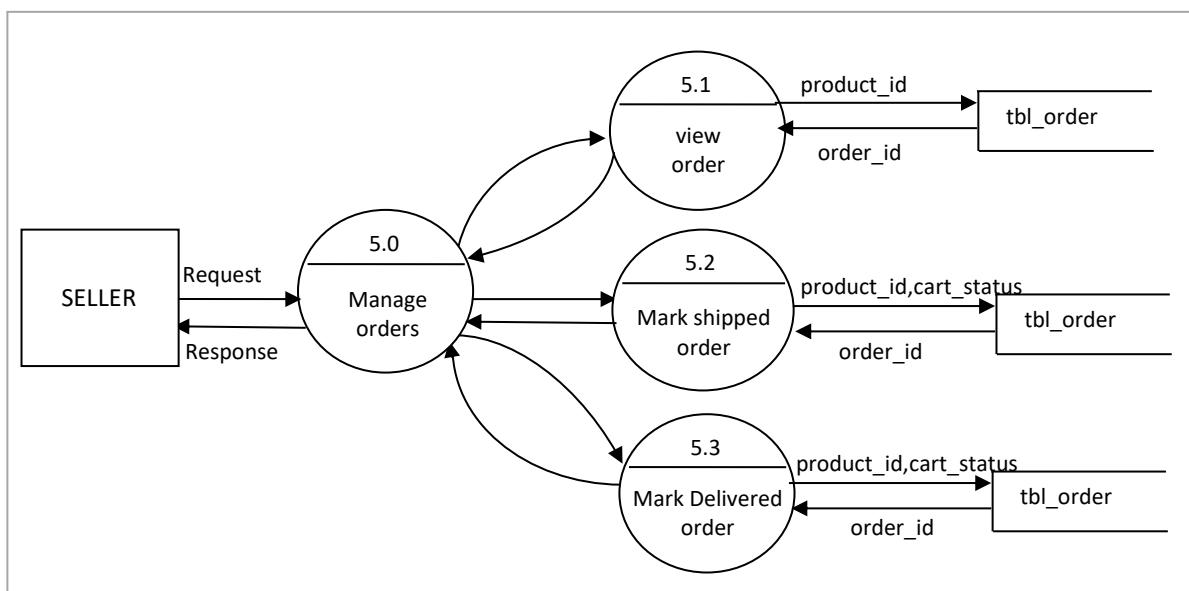


Fig 3.9 2nd level DFD for Artisanal seller side Manage Orders

3. Customer side

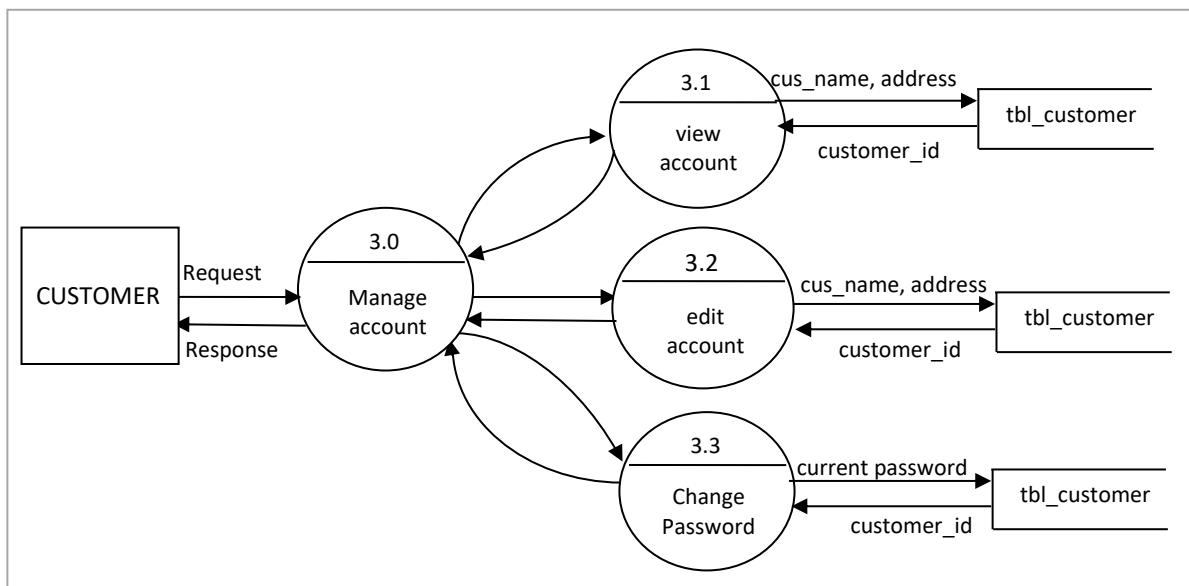


Fig 3.10 2nd level DFD for Artisanal customer side

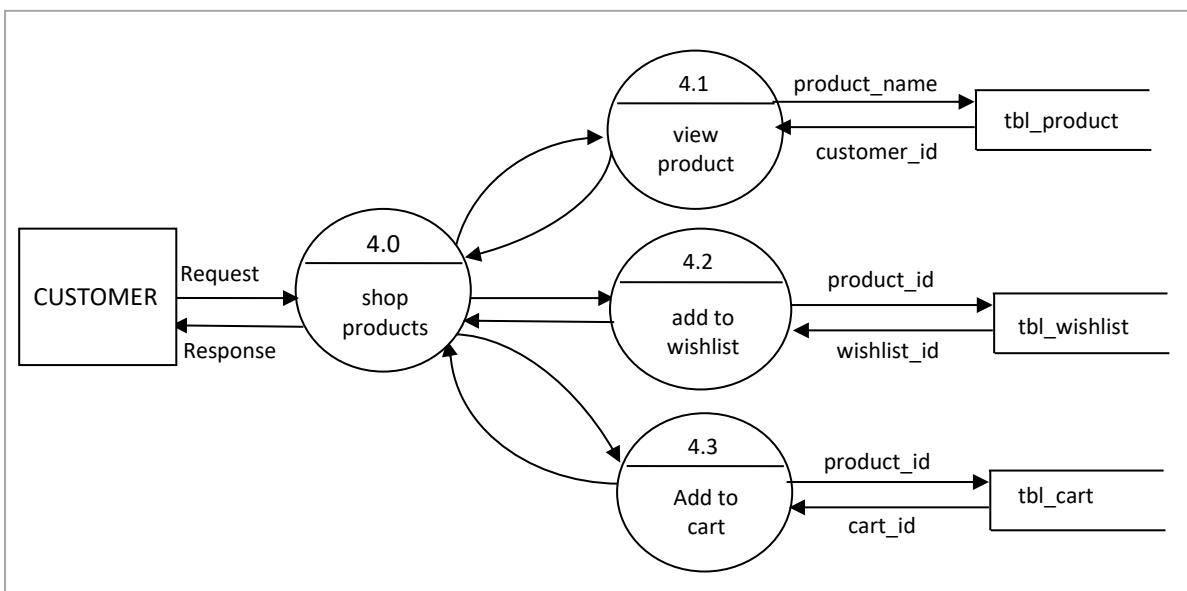


Fig 3.11 2nd level DFD for Artisanal customer side Shop Products

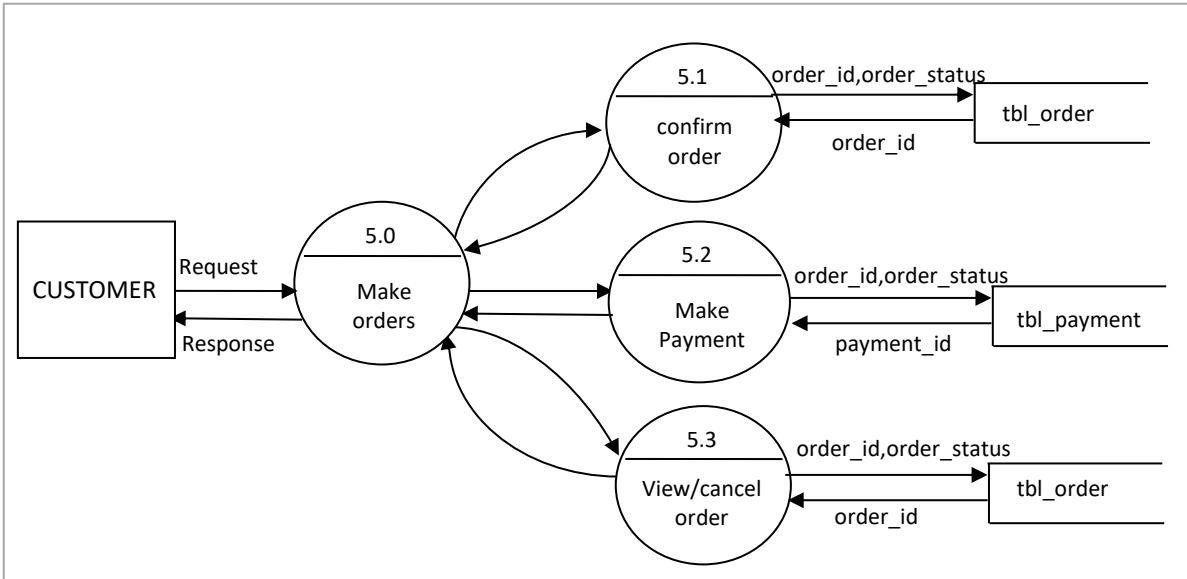


Fig 3.12 2nd level DFD for Artisanal customer side Make Orders

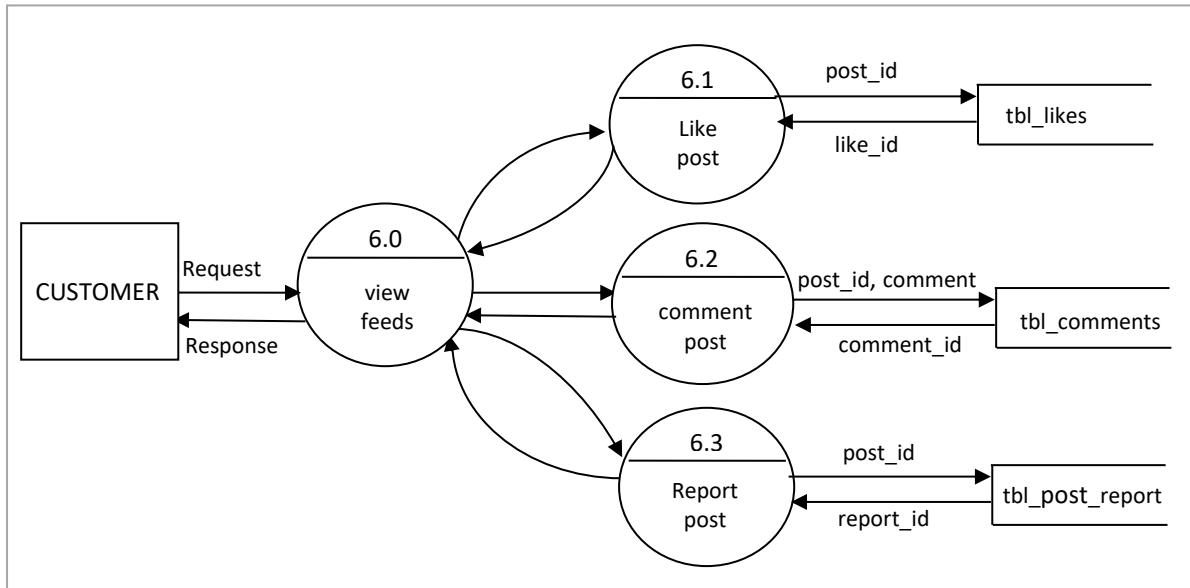


Fig 3.13 2nd level DFD for Artisanal customer side View Feeds

3.4. INTERFACE DESIGN

These modules can apply to hardware, software or the interface between a user and a machine. An example of a user interface could include a GUI, a control panel for a nuclear power plant, or even the cockpit of an aircraft. In systems engineering, all the inputs and outputs of a system, subsystem, and its components are listed in an interface control document often as part of the requirements of the engineering project. The development of a user interface is a unique field.

3.4.1. User Interface Screen Design

The user interface design is very important for any application. The interface design describes how the software communicates within itself, to system that interpreted with it and with humans who use it. The input design is the process of converting the user-oriented inputs into the computer-based format. The data is fed into the system using simple inactive forms. The forms have been supplied with messages so that the user can enter data without facing any difficulty. They data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into system. The goal of designing input data is to make the automation as easy and free from errors as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right messages and help for the user at right are also considered for development for this project.

Input Design is a part of the overall design. The input methods can be broadly classified into batch and online. Internal controls must be established for monitoring the number of inputs and for ensuring that the data are valid. The basic steps involved in input design are:

- Review input requirements.
- Decide how the input data flow will be implemented.
- Decide the source document.
- Prototype on line input screens.
- Design the input screens.

The quality of the system input determines the quality of the system output. Input specifications describe the manner in which data enter the system for processing. Input design features can ensure the reliability of the system and produce results from accurate data. The input design also determines whether the user can interact efficiently with the system.

These are the two sample input forms

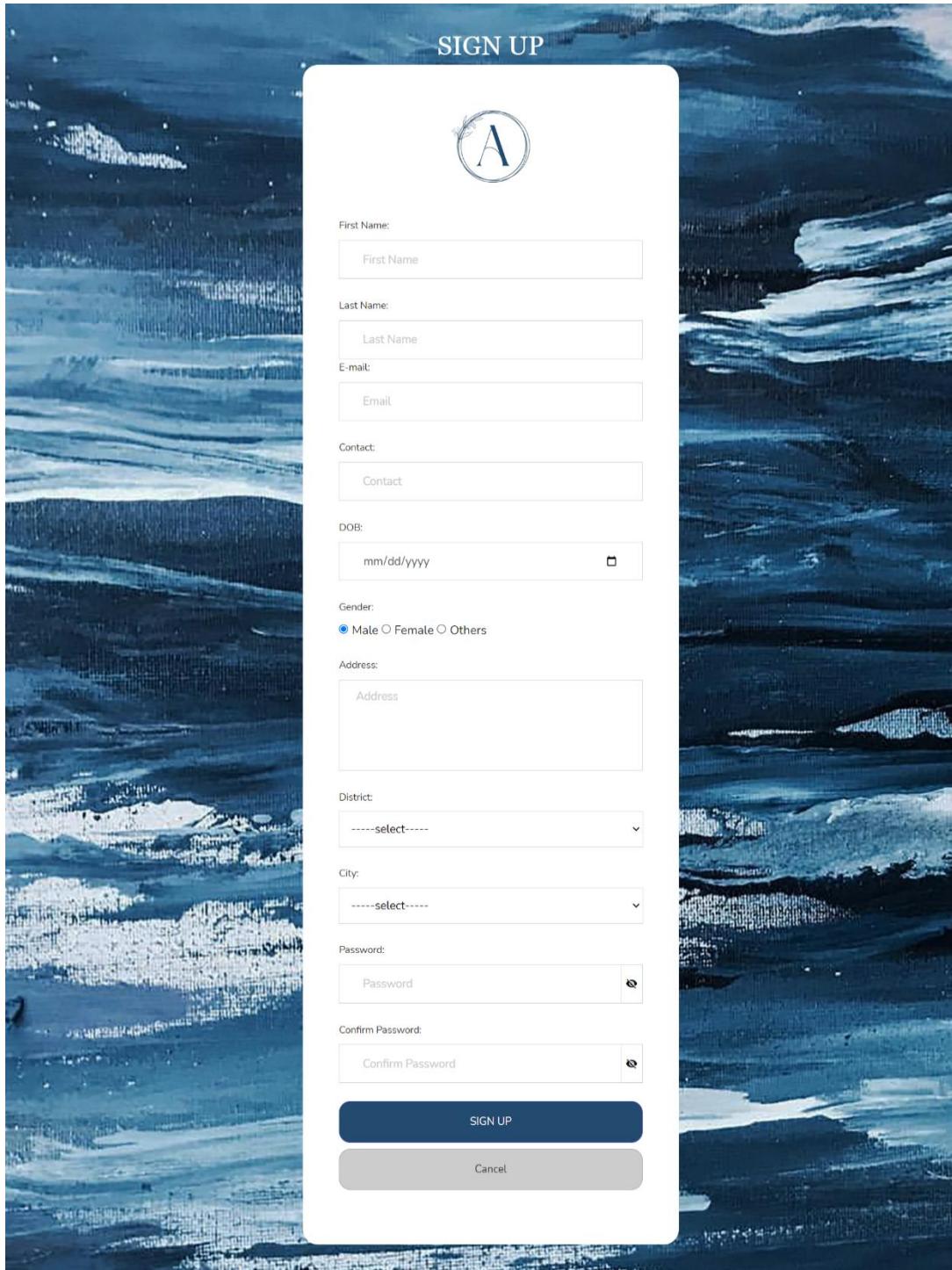


Fig 3.14 Customer registration form

This input form is for the registration of a new customer. It contains textboxes for inputting First Name, Last Name, Email, contact, Date of Birth, Gender, Address , Password and Confirm

Password. This is to keep their information. The form also provide a provision select City, District from drop down.

After selecting the district automatically return the cities in that district and selecting the city will automatically fill the pincode of that city. The customer registration form is very important in the project. This allows the user to enter their details and register in the system before login in to the system. This is helpful for the customers to prove their credential. Each customer must have to fill the full details that are given in the form to register into the system and log in to it. Each field have its own label that denotes the value need to enter in that box and type of value which need to enter in the box. The form also has a button that allows the user to pass the contents entered in the form to the database table. The data entered in the form should be correct according to the type of that field. All labels are arranged in the same alignment line and all boxes to enter values are also in the same line. After clicking the Register button the customer will get a registration successful with OTP and after the enter the OTP successfully they can log. The customer will redirected to the home page of the customer after the log in in first time.

ARTISANAL

Add New Product

product Name:	<input type="text"/>
Color	<input type="text"/>
Material	<input type="text"/>
Type	<input type="checkbox"/> MEN <input type="checkbox"/> WOMEN <input type="checkbox"/> UNISEX <input type="checkbox"/> KIDS <input type="checkbox"/> BOYS <input type="checkbox"/> GIRLS
Price	Indian Rupees
image	<input type="button" value="Choose file"/> No file chosen
Search Keys/ tags	eg. #blue #crochet amigurumi #knitting #lion plushie #blue lion doll #cute stuffed toy
description	<input type="text"/>
category	-----select-----
Sub Category	-----select-----
<input type="button" value="Cancel"/> <input type="button" value="ADD"/>	

Copyright ©2023 All rights reserved | Made with ❤ by Sinju

Hand-crafted & made with ❤

Fig 3.15 Product Adding form

This form is used to add new products to the inventory by the seller. The seller can include new products they want to sell through this form. The form has fields to add the Product Name, product color, material, price, description of the product and search keys for the product, the customer can select the gender type of the product from the checkbox and can select the category and subcategory from the dropdown box. When the category is selected the corresponding subcategories would be available to select. The customer should upload the image of the product when the product is added and can add more images later. There are two buttons in this form one is “ADD” which on click will add the product to the database. And other button is “cancel” which on click will clear the form to fill again.

3.4.2. Output Design

A quality output is one, which meets the requirements of end user and presents the information clearly. In any system result of processing are communicated to the user and to the other system through outputs. In the output design it is determined how the information is to be displayed for immediate need.

It is the most important and direct source information is to the user. Efficient and intelligent output design improves the system’s relationships with the user and helps in decision -making. The objective of the output design is to convey the information of all the past activities, current status and to emphasize important events. The output generally refers to the results and information that is generated from the system. Outputs from computers are required primarily to communicate the results of processing to the users.

Output also provides a means of storage by copying the results for later reference in consultation. There is a chance that some of the end users will not actually operate the input data or information through workstations, but will see the output from the system.

Two phases of the output design are:

1. Output Definition
2. Output Specification

Output Definition takes into account the type of output contents, its frequency and its volume, the appropriate output media is determined for output. Once the media is chosen, the detail specification of output documents are carried out. The nature of output required from the proposed system is determined during logical design stage. It takes the outline of the output from the logical design and produces output as specified during the logical design phase.

In a project, when designing the output, the system analyst must accomplish the following:

- Determine the information to present.

- Decide whether to display, print, speak the information and select the output medium.
- Arrange the information in acceptable format.
- Decide how to distribute the output to the intended receipt.
- Thus by following the above specifications, a high quality output can be generated.

This is an example for outputs generated in this project

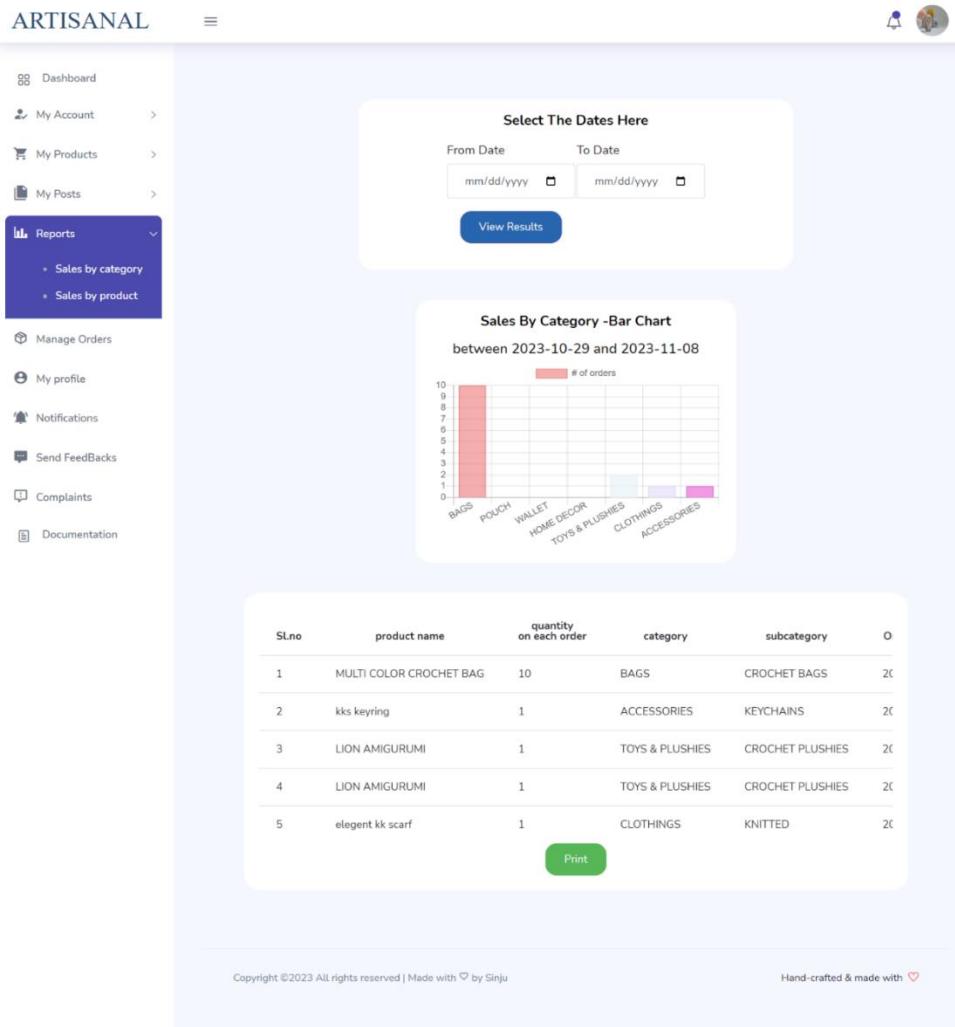


Fig 3.16 Report sales By category -Bar Chart

This bar chart is generated for both admin and seller. This bar chart gives in which categories products are sold the most. In the given example bags are sold the most with 10 sales. This report is of between 2023-10-29 and 2023-11-08.

4. IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned into a working system. The implementation stage is a systems project in its own right. It includes careful planning, investigation of current system and its constraints on implementation, design of methods to achieve the changeover, training of the staff in the changeover procedure and evaluation of changeover method.

4.1. CODING STANDARDS

Writing an efficient software code requires a thorough knowledge of programming. This knowledge can be implemented by following a coding style which comprises several guidelines that help in writing the software code efficiently and with minimum errors. These guidelines, known as coding guidelines, are used to implement individual programming language constructs, comments, formatting, and so on. These guidelines, if followed, help in preventing errors, controlling the complexity of the program, and increasing the readability and understandability of the program.

A set of comprehensive coding guidelines encompasses all aspects of code development. To ensure that all developers work in a harmonized manner (the source code should reflect a harmonized style as a single developer had written the entire code in one session), the developers should be aware of the coding guidelines before starting a software project. Moreover, coding guidelines should state how to deal with the existing code when the software incorporates it or when maintenance is performed.

Since there are numerous programming languages for writing software codes, each having different features and capabilities, coding style guidelines differ from one language to another. However, there are some basic guidelines which are followed in all programming languages. These include naming conventions, commenting conventions, and formatting conventions.

1. **File header comments** are useful in providing information related to a file as a whole and comprise identification information such as date of creation, name of the creator, and a brief description of the software code.
2. **Trailing comments** are used to provide explanation of a single line of code. These comments are used to clarify the complex code. These also specify the function of the abbreviated variable names that are not clear. In some languages, trailing comments are used with the help of a double slash (//).

3. **Indentation:** This refers to one or more spaces left at the beginning of statements in the program. Indentation is useful in making the code easily readable. However, the spaces used for indentation should be followed in the entire program.
4. **Implementing coding guidelines:** If coding guidelines are used in a proper manner, errors can be detected at the time of writing the software code. Such detection in early stages helps in increasing the performance of the software as well as reducing the additional and unplanned costs of correcting and removing errors. Moreover, if a well-defined coding guideline is applied, the program yields a software system that is easy to comprehend and maintain.

4.2. SAMPLE CODE

```
<?php
ob_start();
include('Head.php');
include("../Assets/Connection/Connection.php");
$cusid = $_SESSION['uid'];

if (isset($_POST['btnupdate'])) {
    $cusname = strtoupper($_POST["txtname"]);
    $cuscontact = $_POST["txtcontact"];
    $cusemail = strtolower($_POST["txtemail"]);
    $cusdob = $_POST["txtdob"];
    $cusadd = $_POST["txtaddress"];
    $cityid = $_POST["ddlcity"];

    if (file_exists($_FILES['filephoto'][ "tmp_name"])) {
        $cusphoto = $_FILES["filephoto"] [ "name" ];
        $tempphoto = $_FILES["filephoto"] [ "tmp_name" ];
        move_uploaded_file($tempphoto, '../Assets/File/Customer/' . $cusphoto);

        $photoUpQry = "update tbl_customer set cus_photo=''' . $cusphoto . ''' where
customer_id=''' . $cusid . '''";
        $conn->query($photoUpQry);
    }
}
```

```
$upQry = "update tbl_customer set cus_name='".$cusname."',cus_email='".$cusemail
. "',cus_dob='".$cusdob."',cus_contact='".$cuscontact."',cus_address='".$cusadd."',",
city_id='".$cityid."' where customer_id='".$cusid."'";
if ($conn->query($upQry)) {
    ?>
    <script>
        alert("updated");
        window.location = "MyAccount.php"
    </script>
    <?php
} else {
    ?>
    <script>
        alert("failed");
    </script>
    <?php
}
?>
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Edit Account</title>
<link rel="icon" type="image/x-icon" href="../Assets/File/Admin/Artisanal_icon.png" />
<link rel="stylesheet" type="text/css" href="../Assets/CssPages/headerNavigation.css">
<style>
.container-account {
    width: 470px;
    background-color: #ffffff;
    padding: 20px;
```

```
border-radius: 20px;  
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
display: flex;  
flex-direction: column;  
margin-top: 10px;  
}  
  
.image-container {  
width: 150px;  
height: 150px;  
border-radius: 50%;  
overflow: hidden;  
margin: 0 auto 20px;  
}  
  
.image-container img {  
width: 100%;  
height: 100%;  
object-fit: cover;  
}  
  
</style>  
</head>  
<body>  
<center>  
<form id="form1" name="form1" method="post" action="" enctype="multipart/form-data" onsubmit="return validateTextarea();">  
    <?php  
        $selQry = "select * from tbl_customer u inner join tbl_city c on u.city_id=c.city_id  
        inner join tbl_district d on c.district_id=d.district_id where customer_id='$_GET[cusid]';  
        $res = $conn->query($selQry);  
        while ($row = $res->fetch_assoc()) {  
            ?>  
            <div class="container-account">  
                <center>
```

```
<table width="324">
<tr>
<td colspan="2" align="center">
<div class="image-container">

</div>
<input type="file" name="filephoto" id="filephoto"/>
</td>
</tr>
<tr>
<td width="123"> Name :</td>
<td><label for="txtname"></label>
<input type="text" class="form-control" name="txtname" id="txtname" value="<?php echo $row['cus_name']; ?>" required pattern="[A-Za-z]{2,60}" title="Must contain only alphabets and length should be between 2 - 60 characters.">
</td>
</tr>

<tr>
<td>E-mail :</td>
<td><label for="txtemail"></label>
<input type="email" class="form-control" name="txtemail" id="txtemail" value="<?php echo $row['cus_email']; ?>" required />
</td>
</tr>

<tr>
<td>Contact :</td>
<td><label for="txtcontact"></label>
<input type="text" class="form-control" name="txtcontact" id="txtcontact" value="<?php echo $row['cus_contact']; ?>" required pattern="\d{10}" title="Must contain exactly 10 digits.">
```

```
</td>
</tr>
<tr>
<td>Date Of Birth:</td>
<td><label for="txtdob"></label>
<input type="date" class="form-control" name="txtdob" id="txtdob"
value=<?php echo $row['cus_dob']; ?>
required />
</td>
</tr>

<tr>
<td>Address :</td>
<td>
<textarea name="txtaddress" class="form-control" rows="6" cols="20"
required><?php echo $row['cus_address']; ?></textarea>
<span id="error" style="color:#BD0101; font-size: 15px;"></span>
</td>
</tr>
<tr>
<td>District :</td>
<td>
<select name="ddldistrict" id="ddldistrict" onChange="getCity(this.value)">
<?php
$dselqry = "select * from tbl_district";
$dreslt = $conn->query($dselqry);
while ($drow = $dreslt->fetch_assoc()) { ?>
<option <?php if($row['district_id']==$drow['district_id']) ?>
{?>
selected
<?php } ?>
value=<?php echo $drow['district_id'] ?>">
<?php echo $drow['district_name'] ?>
</option>
<?php } ?>
```

```
</select>
</td>
</tr>

<tr>
<td>City :</td>
<td>
<select name="ddlcity" id="ddlcity" onChange="getPin(this.value)">
<option value=<?php echo $row['city_id']; ?>>
<?php echo $row['city_name']; ?>
</option>
<?php
$cselqry = "SELECT *FROM tbl_city where district_id=" . $row['district_id']
. "";
$creslt = $conn->query($cselqry);
while ($crow = $creslt->fetch_assoc()) { ?>
<option value=<?php echo $crow['city_id'] ?>>
<?php echo $crow['city_name'] ?>
</option>
<?php } ?>
</select>
</td>
</tr>
<tr>
<td>Pincode :</td>
<td id="pincode">
<span class="form-control">
<?php echo $row['pincode']; ?>
</span>
</td>
</tr>
<tr>
<td colspan="2" align="center">
<div style="margin-top:25px;text-align:center;"><input class="btn btn-primary" />
```

```

name="btnupdate" id="btnupdate" value="Update" />
</div>
</td>
</tr>
</table>
</center>
</div>
<?php
} ?>
</form>
</center>
</body>
<script src=..../Assets/JQ/jQuery.js "></script>
<script>
function getCity(ddid) {
$.ajax({
url: "..../Assets/AjaxPages/AjaxCity.php?pid=" + ddid,
success: function (html) {
$("#ddlcity").html(html);
}
});
}
function getPin(ctid) {
$.ajax({
url: "..../Assets/AjaxPages/AjaxPincode.php?cid=" + ctid,
success: function (html) {
$("#pincode").html(html);
}
});
}

const dobInput = document.getElementById("txtdob");
dobInput.addEventListener("change", function () {
const selectedDate = new Date(dobInput.value);
```

```
const today = new Date();
const age = today.getFullYear() - selectedDate.getFullYear();
// Check if the user is at least 18 years old
if (selectedDate > today || age < 15) {

    alert("You must be at least 15 years old.");
    dobInput.value = ""; // Clear the input field
} else {
    document.getElementById("dobError").textContent = "";
}
});

// Function to disable past dates
function disablePastDates() {
    const today = new Date().toISOString().split('T')[0];
    dobInput.setAttribute('max', today);
}

disablePastDates();
document.getElementById("filephoto").addEventListener("change", function () {
    var fileInput = this;

    if (fileInput.files.length > 0) {
        var file = fileInput.files[0];
        var allowedTypes = ["image/jpeg", "image/jpg", "image/png", "image/gif"];
        var maxSize = 5 * 1024 * 1024; // 5 MB in bytes

        if (allowedTypes.indexOf(file.type) === -1) {
            alert("Invalid file type. Please select a jpg, jpeg, png, or gif file.");
            fileInput.value = ""; // Clear the input
        } else if (file.size > maxSize) {

            alert("File size exceeds 5 MB. Please select a smaller file.");
            fileInput.value = ""; // Clear the input
        }
    }
})
```

```
        }
    }
});

function validateTextarea() {
    const textarea = document.getElementById('txtaddress');
    const error = document.getElementById('error');

    // Define a regular expression pattern to allow only alphabets and spaces
    const pattern = /^[A-Za-z0-9,.\\n ]+$/;

    // Check if the input matches the pattern
    if (pattern.test(textarea.value)) {
        error.textContent = ""; // Clear any previous error message
        return true; // Allow form submission
    } else {
        error.textContent = 'Only alphabets, numbers, white space, comma, and dot are
allowed. Line breaks are also accepted.';
        return false; // Prevent form submission
    }
}

</script>
<?php
ob_end_flush();
include('Foot.php');
?>
</body>
</html>
```

5. TESTING

Coding conventions are a set of guidelines for a specific programming language that recommend programming style, practices and methods for each aspect of a piece program written in this language. These conventions usually cover file organization, indentation, comments, declarations, statements, white space, naming conventions, programming practices, programming principles, programming rules of thumb, architectural best practices, etc. These are guidelines for software structural quality. Software programmers are highly recommended to follow these guidelines to help improve the readability of their source code and make software maintenance easier.

5.1. TEST CASES

The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together. The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results. The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results. A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by person other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

Parallel running is often regarded as the final phase of system testing. Since the parallel operation of two systems is very demanding in terms of user resources it should be embarked

on only if the user is satisfied with the results of testing -- it should not be started if problems are known to exist. Testing is the major quality control measure during software development. Its basic function is to detect errors in the software.

Thus the goal of testing is to uncover requirement design and coding errors in the program. Testing is the process of correcting a program with intends of finding an error. Different types of testing are,

1. Unit Testing
2. Integrated Testing
3. Black Box Testing
4. White Box Testing
5. Validation Testing
6. User Acceptance Testing

5.1.1. Unit Testing

In computer programming, unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit for use In this testing we test each module individual and integrated the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as module testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to working satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user which both the formal and validity of the entered. It is very easy to find error debug the system.

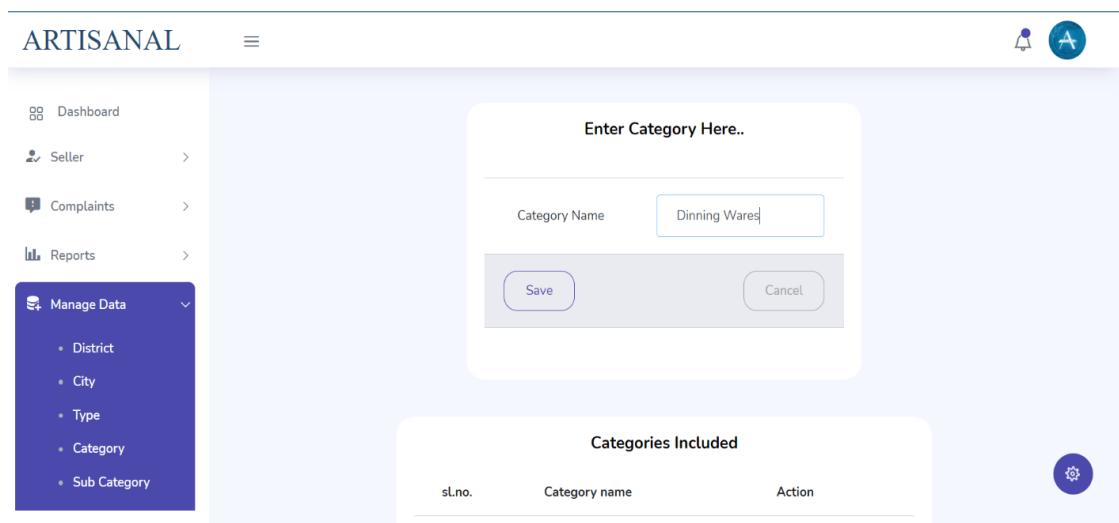


Fig 5.1 Unit testing

The screenshot shows the 'ARTISANAL' application interface. On the left, a sidebar menu includes 'Dashboard', 'Seller', 'Complaints', 'Reports', 'Manage Data' (with sub-options: District, City, Type, Category, Sub Category), 'Reported Posts', 'Notifications', 'FeedBack', and 'Documentation'. The main content area has two sections. The top section is a modal titled 'Enter Category Here..'. It contains a 'Category Name' input field with 'DINNING WARES' typed in, a 'Save' button, a 'Cancel' button, and a message 'record inserted..' at the bottom. The bottom section is titled 'Categories Included' and displays a table with columns: sl.no., Category name, and Action. The table rows are:

sl.no.	Category name	Action
1	BAGS	<button>Delete</button> <button>Edit</button>
2	POUCH	<button>Delete</button> <button>Edit</button>
3	WALLET	<button>Delete</button> <button>Edit</button>
4	HOME DECOR	<button>Delete</button> <button>Edit</button>
5	TOYS & PLUSHIES	<button>Delete</button> <button>Edit</button>
6	CLOTHINGS	<button>Delete</button> <button>Edit</button>
7	ACCESSORIES	<button>Delete</button> <button>Edit</button>
8	DINNING WARES	<button>Delete</button> <button>Edit</button>

At the bottom of the page, there is a copyright notice 'Copyright ©2023 All rights reserved | Made with ❤ by Sinju' and a footer note 'Hand-crafted & made with ❤'.

Fig 5.2 Unit testing

We have continued Unit Testing from the starting of the coding phase itself. Whenever we completed one small sub module, some amount of testing was done based on the requirements to see if the functionality is aligned to the gathered requirements.

5.1.2. Integration Testing

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Data can be lost across an interface; one module can have an adverse effect on the other sub functions when combined by, may not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The developed system has run success full for this sample data. The need for integrated test is to find the overall system performance.

Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit. Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis. Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight." Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.

The screenshot shows the 'Add New Product' form within the Artisanal application. The left sidebar includes links for Dashboard, My Account, My Products (with Add Products and View Products options), My Posts, Reports, Manage Orders, My profile, Notifications, Send FeedBacks, Complaints, and Documentation. The main form has fields for product Name (AJ wallet), Color (brown), Material (leather), Type (checkboxes for MEN, WOMEN, UNISEX, KIDS, BOYS, GIRLS, where MEN is checked), Price (300), image (Choose file button), Search Keys/tags (#brown wallet #men wallet), description (quality leather wallet for men. 100% hand made), category (WALLET dropdown), and Sub Category (LEATHER WALLET dropdown). Buttons for Cancel and ADD are at the bottom.

Fig 5.3 Integration testing

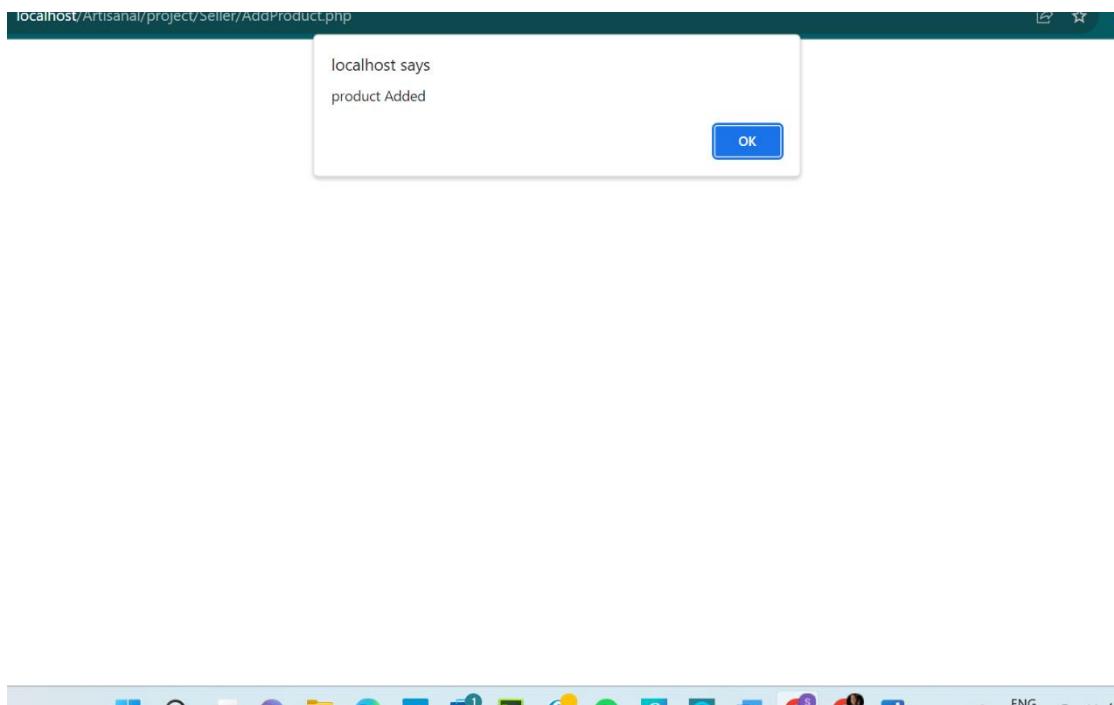


Fig 5.4 Integration testing result

The screenshot shows the ARTISANAL application's user interface. On the left is a sidebar with navigation links: Dashboard, My Account, My Products (selected), My Posts, Reports, Manage Orders, My profile, Notifications, Send FeedBacks, Complaints, and Documentation. The main area is titled 'My Products' and displays a table with two rows of product information. The columns are Product Name, manage Images, color, material, type, price, and tags.

Product Name	manage Images	color	material	type	price	tags
RABBIT CROCHET	 Click here	blue	wool	UNISEX	300	#cutecrochets #couplecrochets
AJ WALLET	 Click here	brown	leather	MEN	300	#brown wallet #men wallet

Fig 5.5 Integration testing result

We have performed integration testing whenever we have combined two modules together. When two modules are combined we have checked whether the functionality works correctly or not through integration testing.

5.1.3. Black Box Testing

Black-box testing is a method of software testing that examines the functionality of an application (e.g. what the software does) without peering into its internal structures or workings. This method of test can be applied to virtually every level of software testing: unit, integration, system and acceptance. It typically comprises most if not all higher level testing, but can also dominate unit testing as well. In black box testing the structure of the program is not considered. Test cases are decided solely on the basis of the requirements or the specification of the program or module, and the internals of the module or program are not considered for selection of the test cases.

In the Black Box testing tester only knows the input that can be given to the system and what output the system should give. In other words, the basis of deciding test cases in functional testing is requirements or specifications of the system or module. This form of testing is also called functional or behavioural testing. One advantage of the black box technique is that no

programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the

5.1.4. White Box Testing

White-box testing (also known as clear box testing, glass box testing, and transparent box testing and structural testing) is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g. in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system-level test. Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements. White Box testing is concerned with testing the implementation of the program. The intent of this testing is not to exercise all the different input or output conditions but to exercise the different programming structures and data structures used in the program.

White-box test design techniques include:

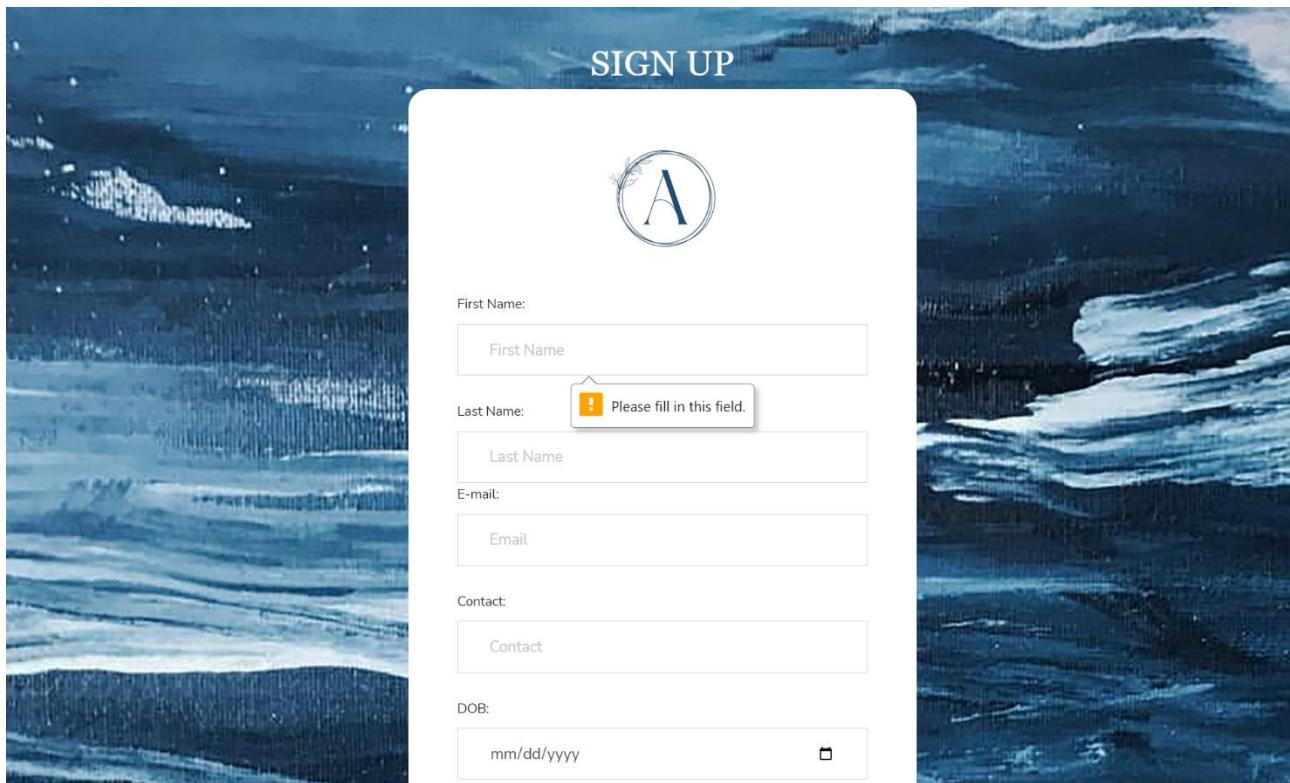
- Control flow testing
- Data flow testing
- Branch testing
- Path testing
- Statement coverage
- Decision coverage

5.1.5. Validation Testing

At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, Validation tests begins. Validation testing can be defined many was but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer. After validation test has been conducted one of the two possible conditions exists.

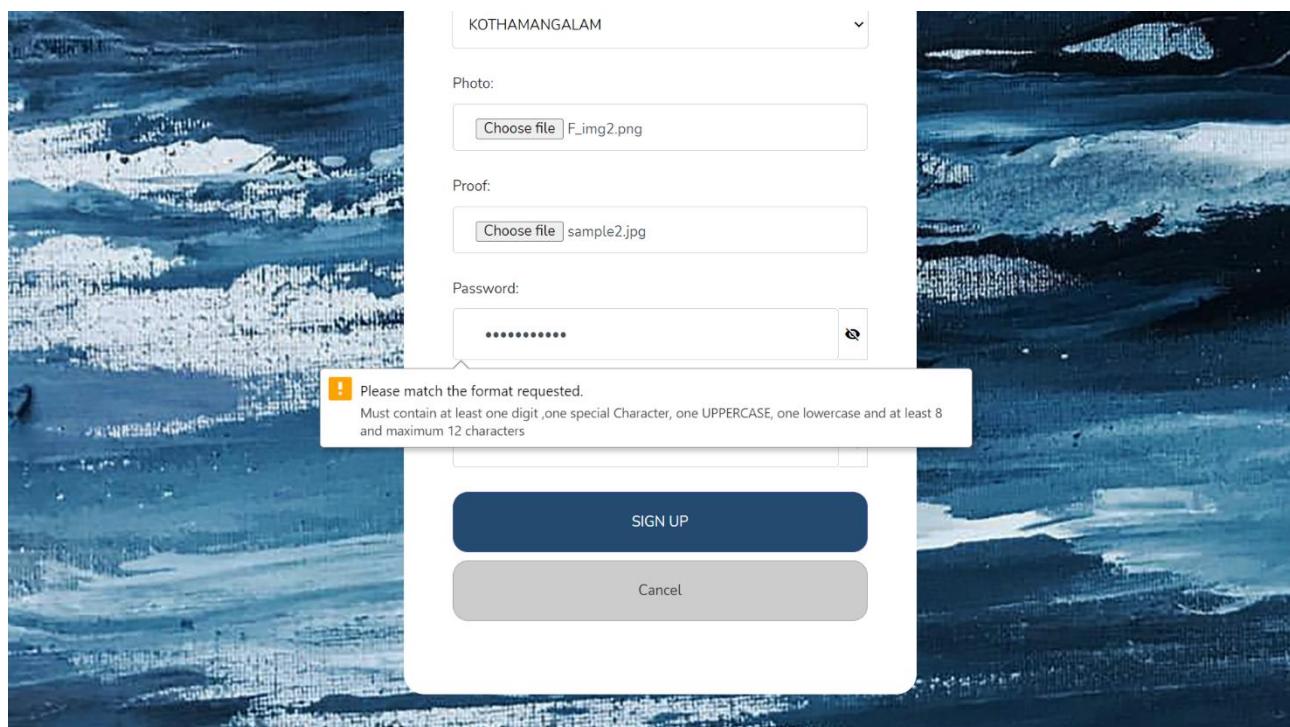
1. The function or performance characteristics confirm to specification and are accepted.

2. A derivation from specification uncovered and a deficiency list is created.



The image shows a sign-up form titled "SIGN UP" set against a background of blue and white abstract waves. The form includes fields for First Name, Last Name, E-mail, Contact, and DOB. The "Last Name" field has an error message: "Please fill in this field." The "First Name" field contains "First Name". The "E-mail" field contains "Email". The "Contact" field contains "Contact". The "DOB" field contains "mm/dd/yyyy".

Fig 5.6 Username validation



The image shows a sign-up form titled "SIGN UP" set against a background of blue and white abstract waves. The form includes fields for Name, Photo, Proof, and Password. The "Name" field contains "KOTHAMANGALAM". The "Photo" field has a file chosen: "F_img2.png". The "Proof" field has a file chosen: "sample2.jpg". The "Password" field contains "*****". A validation message at the bottom left states: "Please match the format requested. Must contain at least one digit, one special Character, one UPPERCASE, one lowercase and at least 8 and maximum 12 characters".

Fig 5.7 Password validation

The screenshot shows the 'ARTISANAL' application interface. On the left, a sidebar menu includes 'Dashboard', 'Seller', 'Complaints', 'Reports', 'Manage Data' (with sub-options: District, City, Type, Category, Sub Category), 'Reported Posts', 'Notifications', 'FeedBack', and 'Documentation'. The main content area has two sections: 'Enter District Here..' (with a 'District Name' input field containing 'Ernakulam' and 'Save'/'Cancel' buttons) and 'Districts Included' (a table listing districts with columns: sLno., District name, and Action). The table data is as follows:

sLno.	District name	Action
1	ALAPPUZHA	<button>Delete</button> <button>Edit</button>
2	ERNAKULAM	<button>Delete</button> <button>Edit</button>
3	IDUKKI	<button>Delete</button> <button>Edit</button>
4	KANNUR	<button>Delete</button> <button>Edit</button>
5	KASARAGOD	<button>Delete</button> <button>Edit</button>
6	KOLLAM	<button>Delete</button> <button>Edit</button>
7	KOTTAYAM	<button>Delete</button> <button>Edit</button> <button>Print</button>

In the browser's address bar, the URL is 'localhost/Admin/District.php'. A modal dialog box displays the message 'localhost says already exist' with an 'OK' button.

Fig 5.8 Already Existing Validation testing.

We have given various validations in our forms so that there will be a neat format for the data's that are entered on to the website. We have also given an already existing validation so that the data redundancy is reduced; same data is not entered twice.

5.1.6. User Acceptance Testing

Acceptance Testing is a level of the software testing process where a system is tested for acceptability. User Acceptance testing is the software testing process where system tested for acceptability & validates the end to end business flow. Such type of testing executed by client in separate environment & confirms whether system meets the requirements as per requirement specification or not.

UAT is performed after System Testing is done and all or most of the major defects have been fixed. This testing is to be conducted in the final stage of Software Development Life Cycle (SDLC) prior to system being delivered to a live environment. UAT users or end users are concentrating on end to end scenarios & typically involves running a suite of tests on the completed system.

User Acceptance testing also known as Customer Acceptance testing (CAT), if the system is being built or developed by an external supplier. The CAT or UAT are the final confirmation from the client before the system is ready for production. The business customers are the primary owners of these UAT tests. These tests are created by business customers and articulated in business domain languages. So ideally it is collaboration between business customers, business analysts, testers and developers. It consists of test suites which involve multiple test cases & each test case contains input data (if required) as well as the expected output. The result of test case is either a pass or fail.

5.2. TEST CASE DOCUMENTS

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application. A sample of test case document format is given below.

TC No.	Test Steps	Expected Result	Actual Result	Status	Comment
1.	Run application and navigate to login screen	Login screen is displayed. A field for entering email, a field for entering password and a button to submit should be present	Login screen has been displayed, fields for entering email and password together with a log in button is available.	Pass	
2.	Enter an invalid email and invalid password and press the button	A message should be displayed stating that email and password are invalid	A message has been displayed stating that email and password are invalid	Pass	
3.	Enter a valid email and password and press the button	User must successfully login to the WebPages.	User successfully log in and navigate into home page	Pass	
4.	Enter a valid email and leave password and press the button	A message should be displayed stating that ‘Password Required’ in beside of Password textbox.	A message has been displayed stating that please ‘Password Required’ in beside of Password textbox	Pass	
5.	Enter a valid email and invalid password and press the button	A message should be displayed stating that email and password are invalid	A message has been displayed stating that email and password are invalid	Pass	
6.	Leave email and password and press the button	A message should be displayed stating that ‘email required’ in beside of email textbox and ‘Password Required’ in beside of Password textbox	A message has been displayed stating that ‘email required’ in beside of email textbox and ‘Password Required’ in beside of Password textbox	Pass	

7.	Leave email and enter a valid password and press the button	A message should be displayed stating that 'email required' in beside of Username textbox	A message should be displayed stating that 'email required' in beside of Username textbox	Pass	
8.	Enter Invalid email and enter a valid password and press the button	A message should be displayed stating that email and password are invalid	A message has been displayed stating that email and password are invalid	Pass	

Table 5.1 Test Case

6. CONCLUSION

The project entitled “**ARTISANAL**” was completed on time. This project provided maximum interaction and flexibility. The system was tested and the performance of the system was provided to be much efficient and data maintenance is achieved partially. The system has been developed in attractive fashion. The modules in the system help in faster development, implementation and maintenance of the software. This system has been developed as versatile and user friendly as possible keeping in mind the advanced features. Using HTML, CSS, Bootstrap, Ajax, jQuery, JavaScript, PHP, MySQL and XAMPP Server, the system was developed and tested with all possible samples of data. As a whole, the system was well planned and designed. The performance of the system is proved to be efficient. And it already provide all the objectives we have identified before. All modules are tested separately and put together to form the main system. Finally the system is tested with the real data and everything worked successfully. Thus the system has fulfilled the entire objective identified. The system required least hardware requirement to work on. So I can state, I have developed such a good environment for communication, to connect with more people. And it provides a number of advantages too as I described in the previous sections. To conclude this, I thank all people who help me to complete this project work successfully.

6.1. FUTURE ENHANCEMENTS

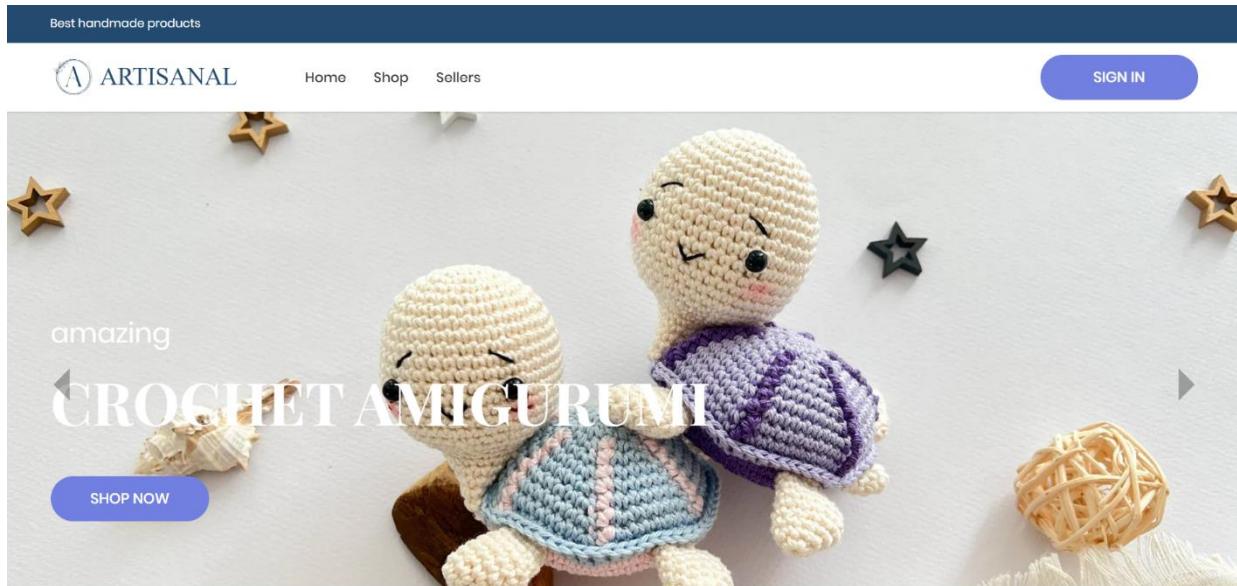
The system has been designed in such a way that it can be modified with very little effort when such needs arise in the future. New features can be added with slight modifications of software which make it easy to expand the scope of this project. Though the system is working on various assumptions, it can be modified easily to any kind of requirements. The system is also expected to be improvised by adding various features. Now the system is designed in English. In future this will converted into the mother tongue language. The system will be modified to have a automatic language converting that helpful for the peoples. In future the sellers and customers can have more interactions by adding direct message feature. The system will also add features that help the sellers to support and interact with each other. Introduction of a mobile application can also be done so that this system becomes more users friendly.

7. REFERENCES

1. SYSTEM ANALYSIS AND DESIGN
Elias M Award,
Second Edition, 1999,
Galgotica Publications, Delhi
2. DATABASE SYSTEM CONCEPT
Silberschatz, Korth,
Sudarshan, Fourth
Edition.
3. WEB PROGRAMMING USING PHP
Dr. Jeeva Jose,
Kalyani Publication
4. https://en.wikipedia.org/wiki/Operating_system
5. <http://ecomputernotes.com/software-engineering/feasibilitystudy>
6. https://en.wikipedia.org/wiki/Third_normal_form
7. <https://www.guru99.com/user-acceptance-testing.html>
8. https://www.tutorialspoint.com/system_analysis_and_design.html
9. <http://softwaretestingfundamentals.com/test-case/>
10. <http://softwaretestingfundamentals.com/black-box-testing/>
11. <https://www.geeksforgeeks.org/asp.net-coding-standards/>
12. <https://www.geeksforgeeks.org/software-engineering-sdlc-v-model/>

8. APPENDIX

8.1. SCREENSHOTS



OUR SELLERS



ABOUT US

The Objective of Artisanal is to support small handicraft business and make a website that is fully dedicated to handmade products which attracts customer who like or prefer unique handmade products over machine made mass produced products. Artisanal sells a wide range of handmade products like paintings, bags, wallets, hats, key chains, cups, home decors etc. And simple cloths like scarves, mufflers, sweaters, Beanies etc. Artisanal wants to give an amazing online shopping experience to everyone who visits the site.

GET IN TOUCH

Any questions? Let us know in artisanalhelp@gmail.com or call us on (+91) 6235009367

CONNECT WITH US

[LinkedIn](#)

[FEEDBACK](#)

PayPal VISA MasterCard American Express Discover

Copyright © 2023 All rights reserved | Made with ❤ by Sinju

Fig 8.1 Guest page

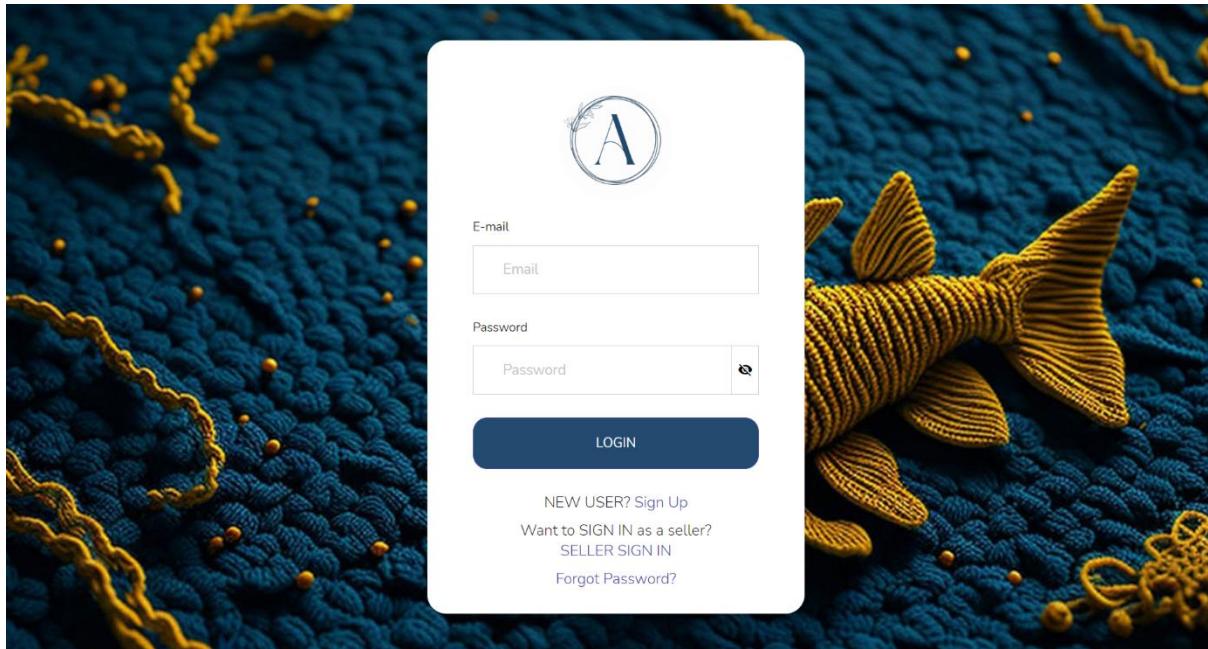


Fig 8.2 Login Page

Fig 8.3 Admin Dashboard

Sellers Not Verified Yet

sL.no.	Name	photo	contact	E-mail	address
1	ANDREWS STEPHEN		5687453487	sinjumathews53@gmail.com	uppkandam

List Of Verified

sL.no.	Name	photo	contact	E-mail	
1	KRISHNA PRIYA		87657657599	krishna123@gmail.com	kr
2	GANGA GOPI		7867567687	Ganga123@gmail.com	

Fig 8.4 Seller Verification

Enter District Here..

District Name
<input type="text"/>

Districts Included

sL.no.	District name	Action
1	ALAPPUZHA	Delete Edit

Fig 8.5 Adding Districts

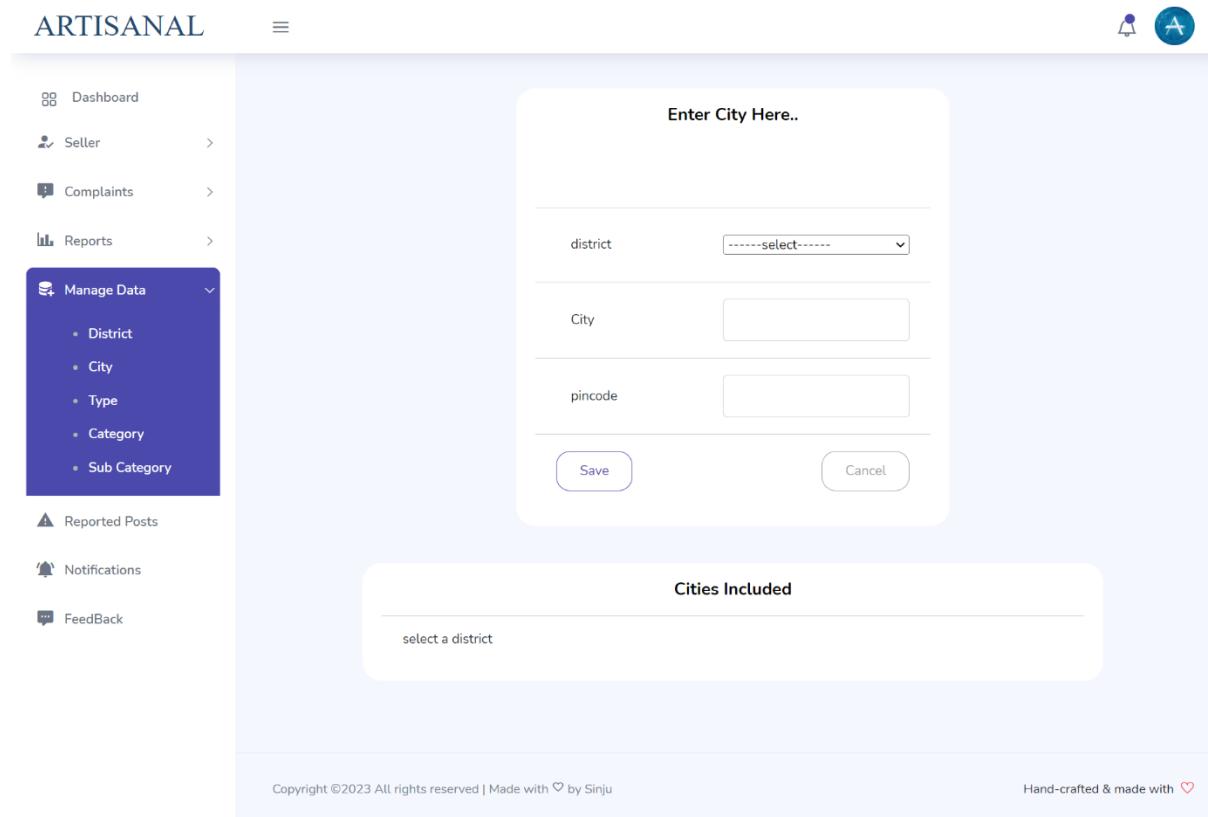


Fig 8.6 Adding Cities

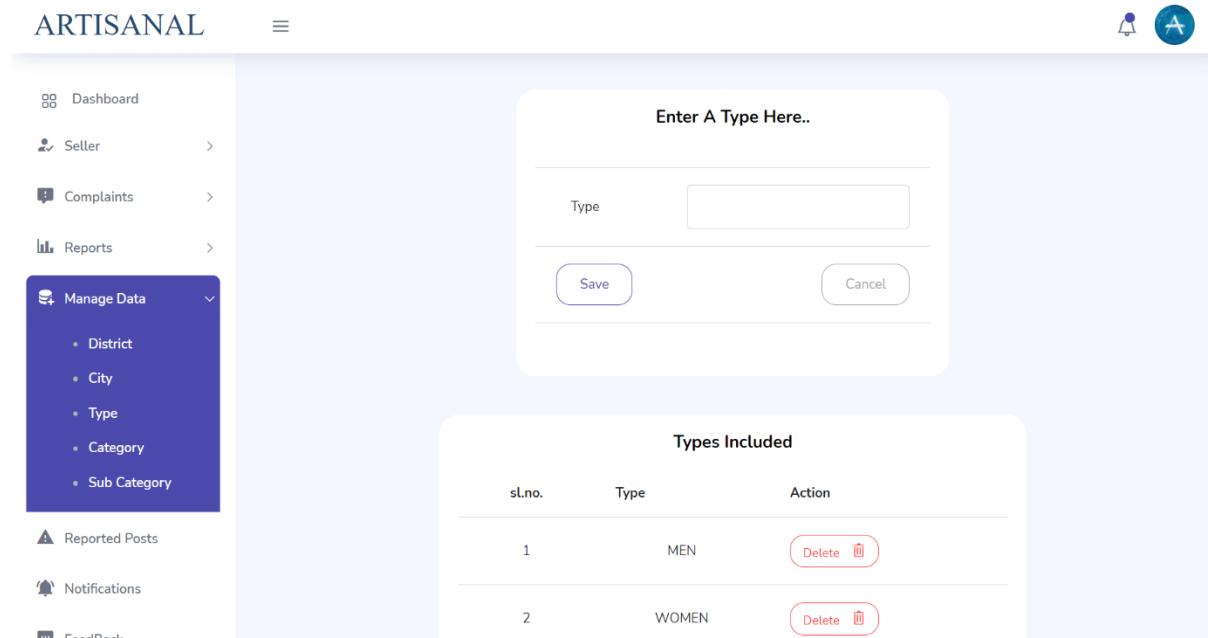


Fig 8.7 Adding Types

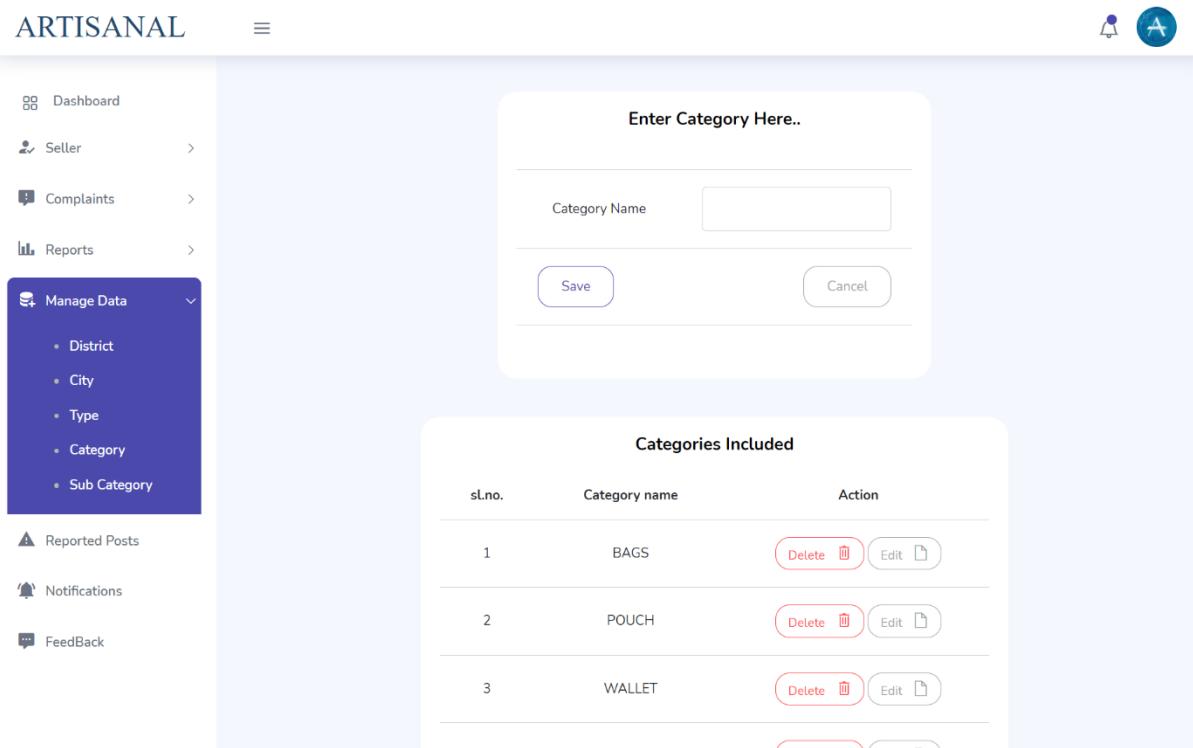


Fig 8.8 Adding Categories

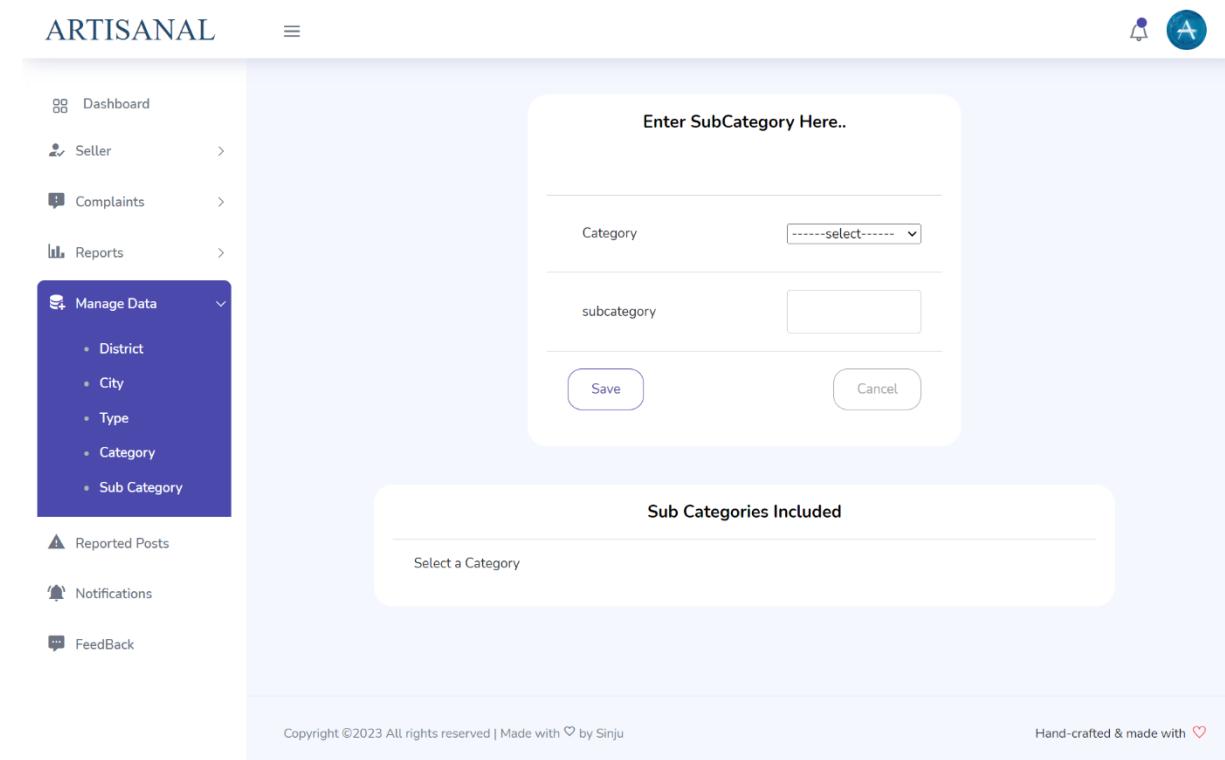


Fig 8.9 Adding Subcategories

Posts Being Reported

sl.no.	post	reason to report	posted by	reported by	reported on
1	working on my New product	report testing	FELIX LEE felix123@gmail.com	PAILY SAJI pailysaji08@gmail.com	2023-11-06 23:59:59

Posts Removed

sl.no.	post	reason to report	posted by	reported by	reported on	Action

Posts Kept

sl.no.	post	reason to report	posted by	reported by	reported on	Action

Copyright ©2023 All rights reserved | Made with ❤ by Sinju

Hand-crafted & made with ❤

Fig 8.10 Reported Posts

Feedbacks From Sellers

SL No:	Seller Name:	Seller Contact:	Seller Email:	Feedback Content	Feedback Time:
1	GANGA GOPI	7867567687	Ganga123@gmail.com	not bad	2023-11-06 23:29:01

Feedbacks From Customers

SL No:	Customer Name:	Customer Contact:	Customer Email:	Feedback Content	Feedback Time:
1	PAULSON ELDHO	7994681529	paulsoneldho877@gmail.com	nice website	2023-11-06 20:59:59
2	PAILY SAJI	8281860108	pailysaji08@gmail.com	great website	2023-11-06 23:59:59

Copyright ©2023 All rights reserved | Made with ❤ by Sinju

Hand-crafted & made with ❤

Fig 8.11 View Feedbacks

The screenshot shows the 'ARTISANAL' application interface. On the left, a sidebar menu includes 'Dashboard', 'Seller', 'Complaints' (selected), 'Reports', 'Manage Data', 'Reported Posts', 'Notifications', and 'FeedBack'. The main content area is titled 'Complaints From Customers' and displays a table with three rows of customer data:

Customer Contact	Customer Email	Complaint title	Complaint Description	Action
7994681529	paulsonelh0877@gmail.com	this is complaint demo	here complaint details	reply for complaint
8281860108	pailysaji08@gmail.com	complaint example	this is a complaint	Reply
7867567889	lashi123@gmail.com	complaint example2	complaint example2 content	Reply

At the bottom of the main content area, there are copyright and attribution notices: 'Copyright ©2023 All rights reserved | Made with ❤ by Sinju' and 'Hand-crafted & made with ❤'.

Fig 8.12 View Complaints

The screenshot shows the 'ARTISANAL' application interface. The sidebar menu is identical to Fig 8.12. The main content area is titled 'Reply To Complaints' and contains a form with the following fields:

- A text input field labeled 'Enter your reply:' containing the placeholder text 'reply For complaint'.
- A blue 'SEND REPLY' button at the bottom of the form.

At the bottom of the main content area, there are copyright and attribution notices: 'Copyright ©2023 All rights reserved | Made with ❤ by Sinju' and 'Hand-crafted & made with ❤'.

Fig 8.13 Reply to complaints

The screenshot shows the Artisanal application's interface. On the left is a sidebar with navigation links: Dashboard, Seller, Complaints, Reports (with sub-options: Sales by category, Sales by Sellers, Seller Followers, General Report), Manage Data, Reported Posts, Notifications, and FeedBack. The main content area has a header "Select The Dates Here" with "From Date" and "To Date" fields. Below this is a bar chart titled "Sales By Category -Bar Chart" showing sales between 2023-10-29 and 2023-11-05. The categories on the x-axis are BAGS, POUCH, WALLET, HOME DECOR, TOYS & PLUSHIES, CLOTHINGS, ACCESSORIES, and DINNING WARES. The y-axis represents the number of orders, ranging from 0.0 to 4.0. The chart shows that ACCESSORIES has the highest sales at approximately 4.0 orders. Below the chart is a table of sales data:

SL.no	product name	quantity on each order	category	subcategory
1	KK MULTI COLOR SCARF	1	CLOTHINGS	KNITTED
2	LEATHER KEY CHAIN	2	ACCESSORIES	KEYCHAINS
3	LINO THE BUNNY	1	TOYS & PLUSHIES	CROCHET PLUSHIES
4	AJ ASSORTED GREEN BRACELET	1	ACCESSORIES	BRACELETS
5	AJ BLUE BRACELETS	1	ACCESSORIES	BRACELETS

A green "Print" button is located below the table. At the bottom of the page, there are copyright and hand-crafted notices.

Fig 8.14 Sales report by category

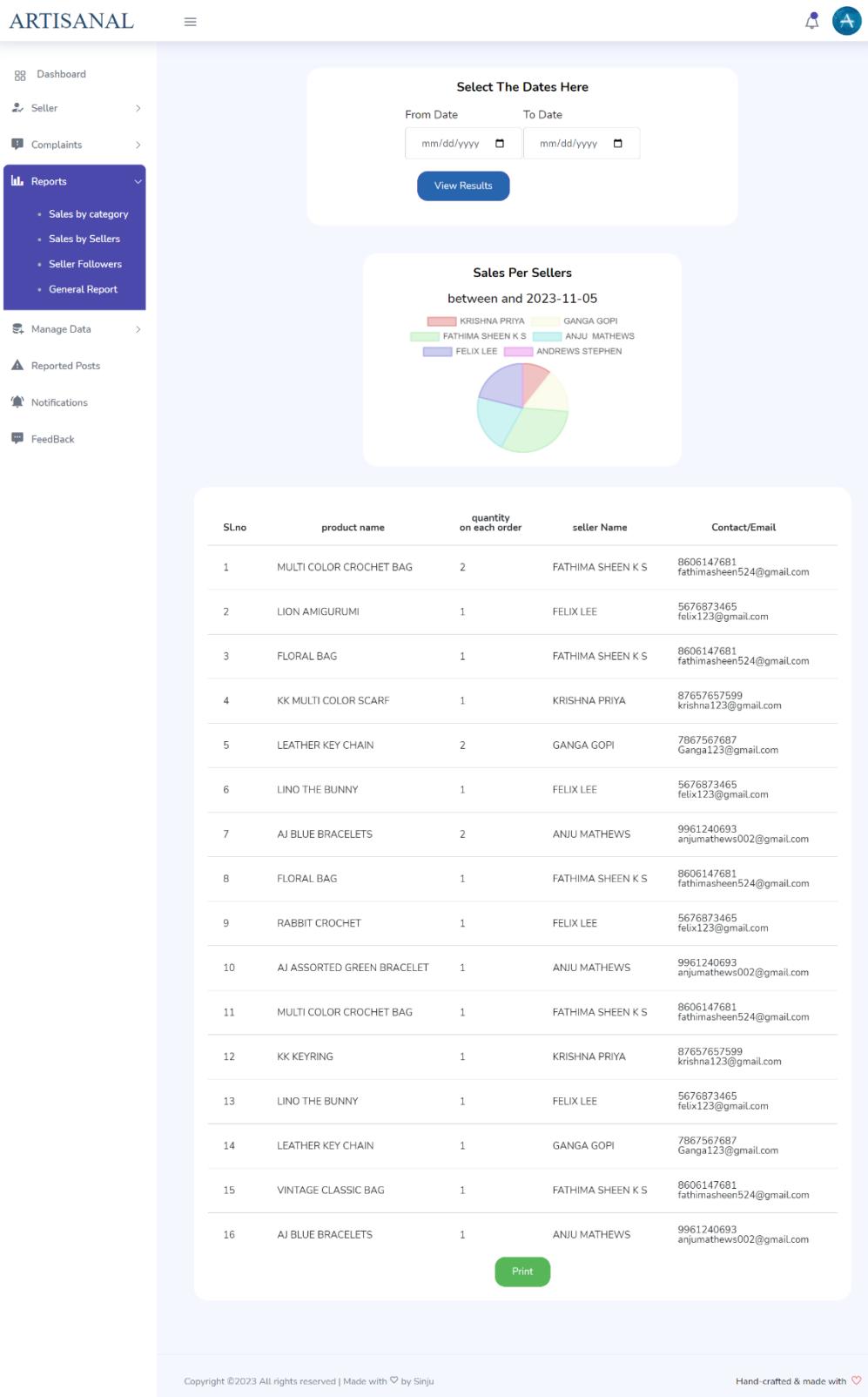


Fig 8.15 Sales Report by sellers

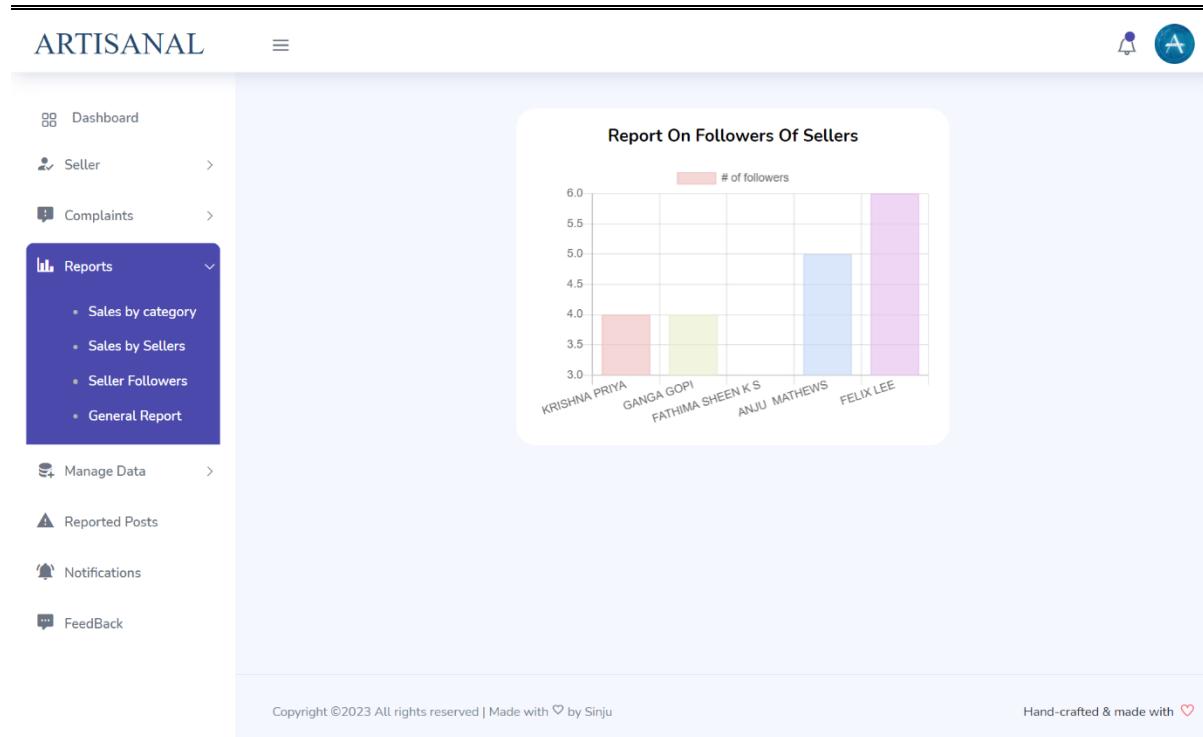


Fig 8.16 Bar chart on Followers of sellers

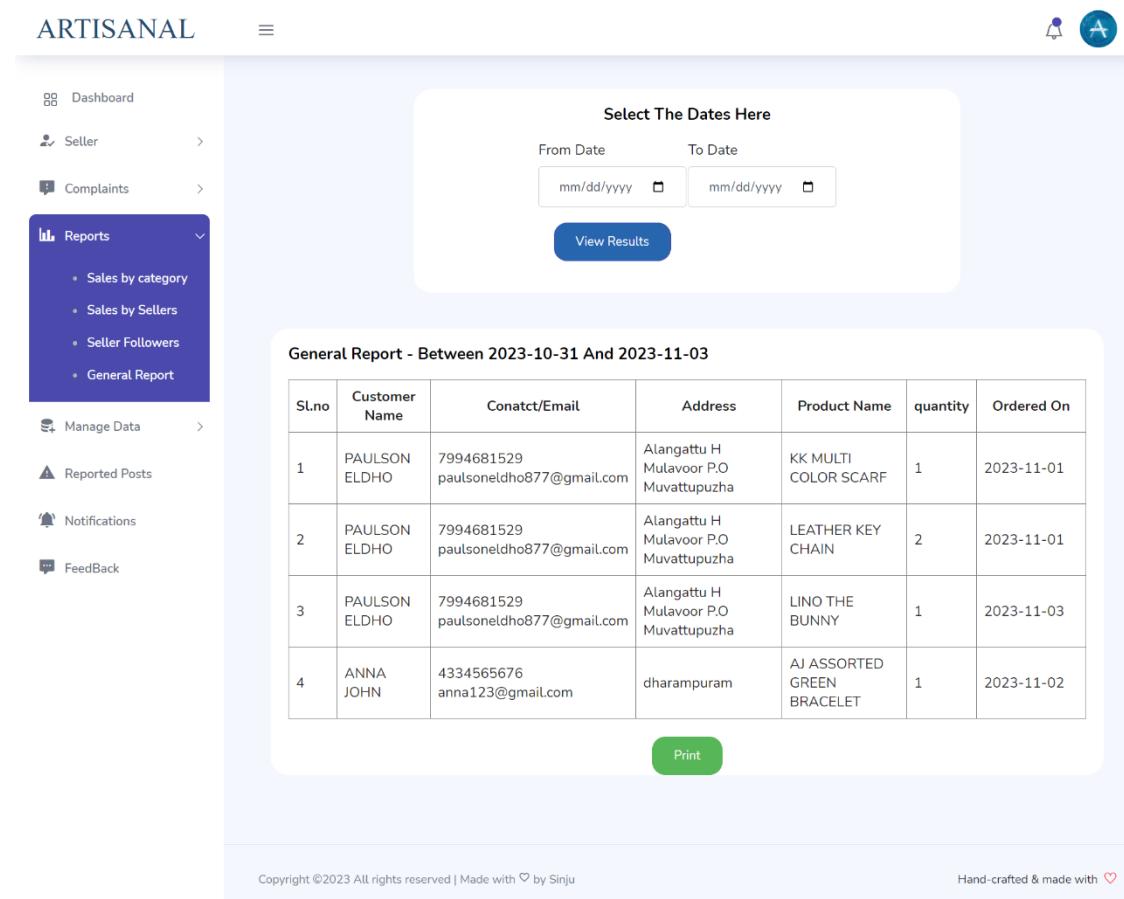


Fig 8.17 General Report on Orders received

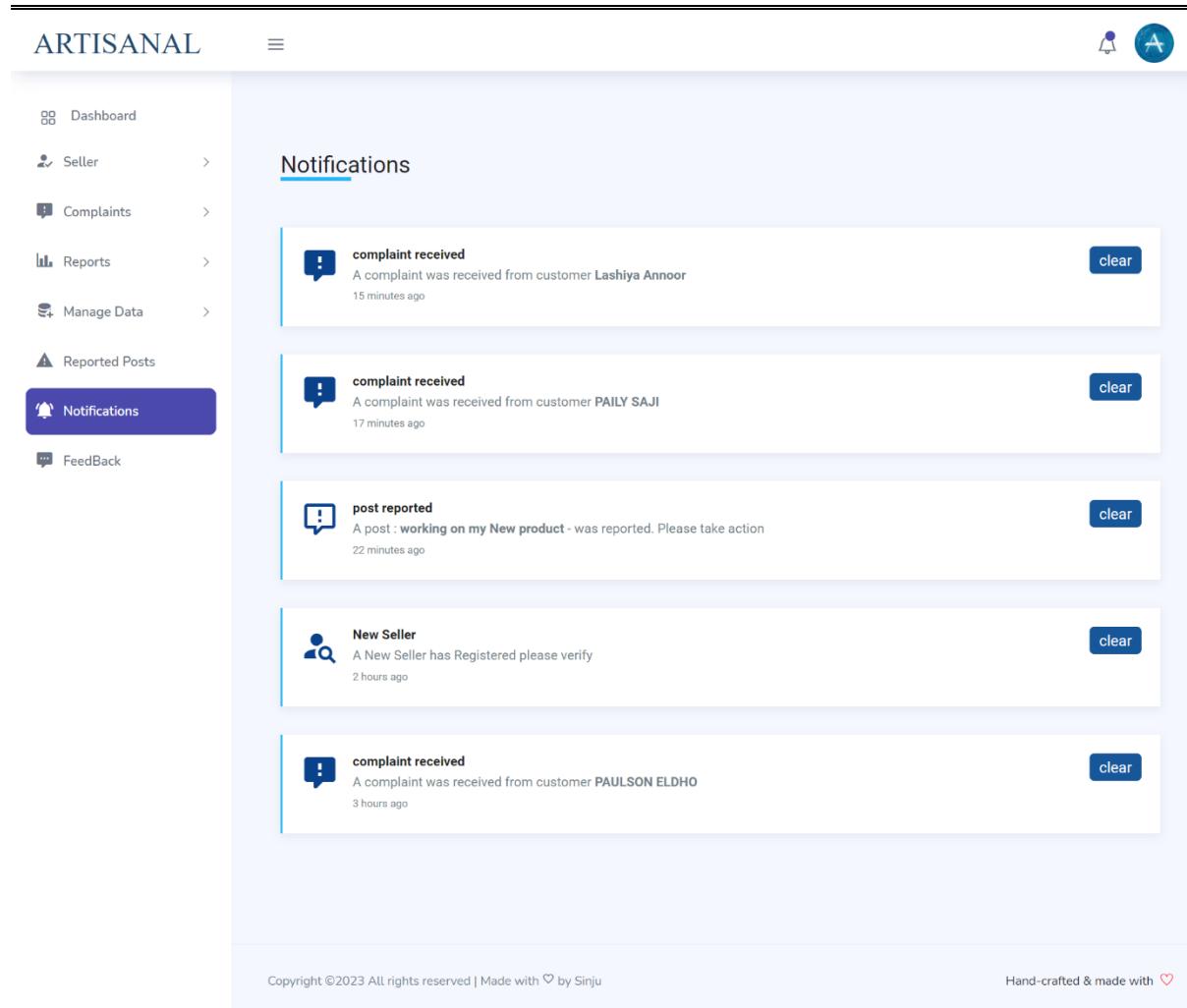


Fig 8.18 Admin Notifications

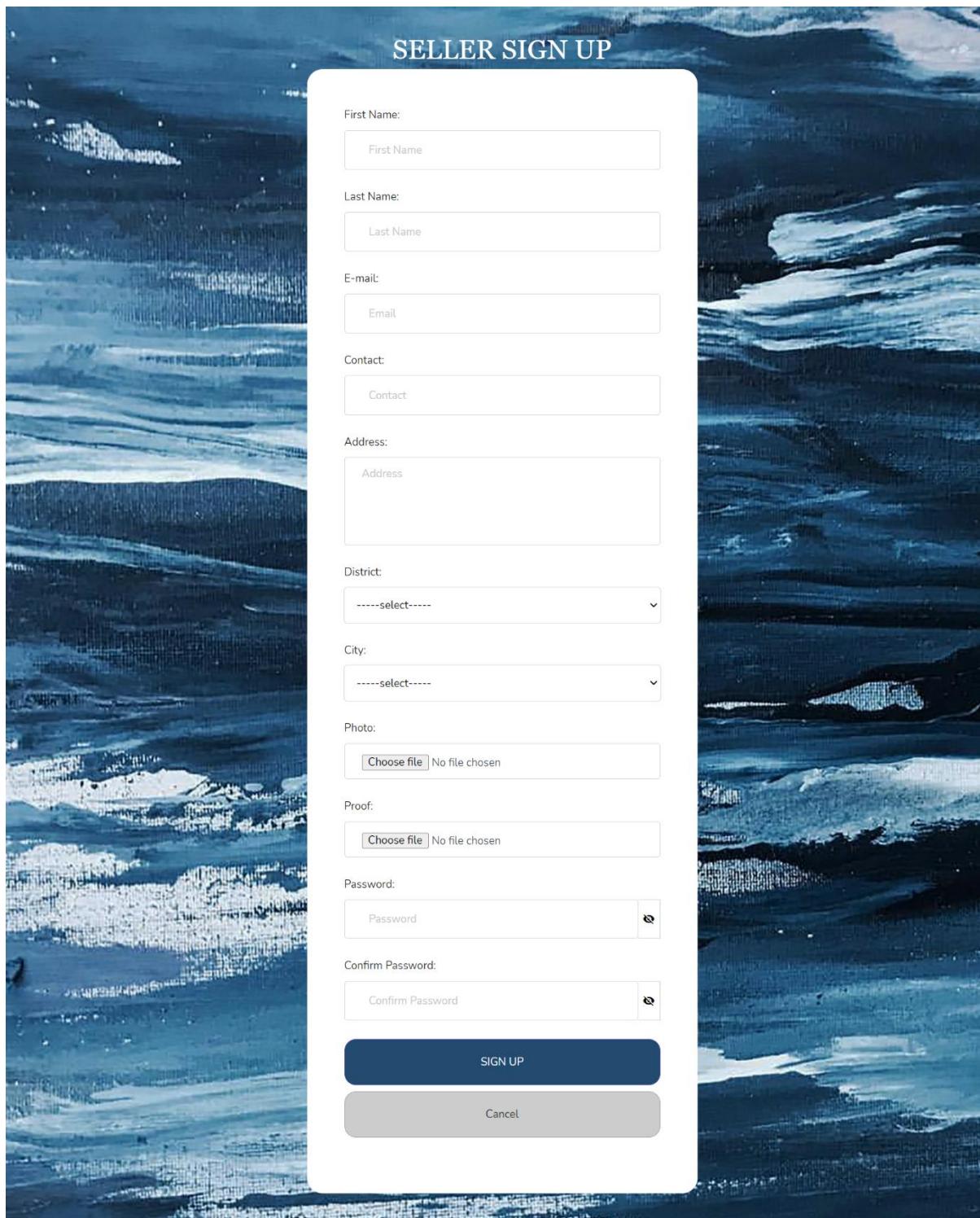
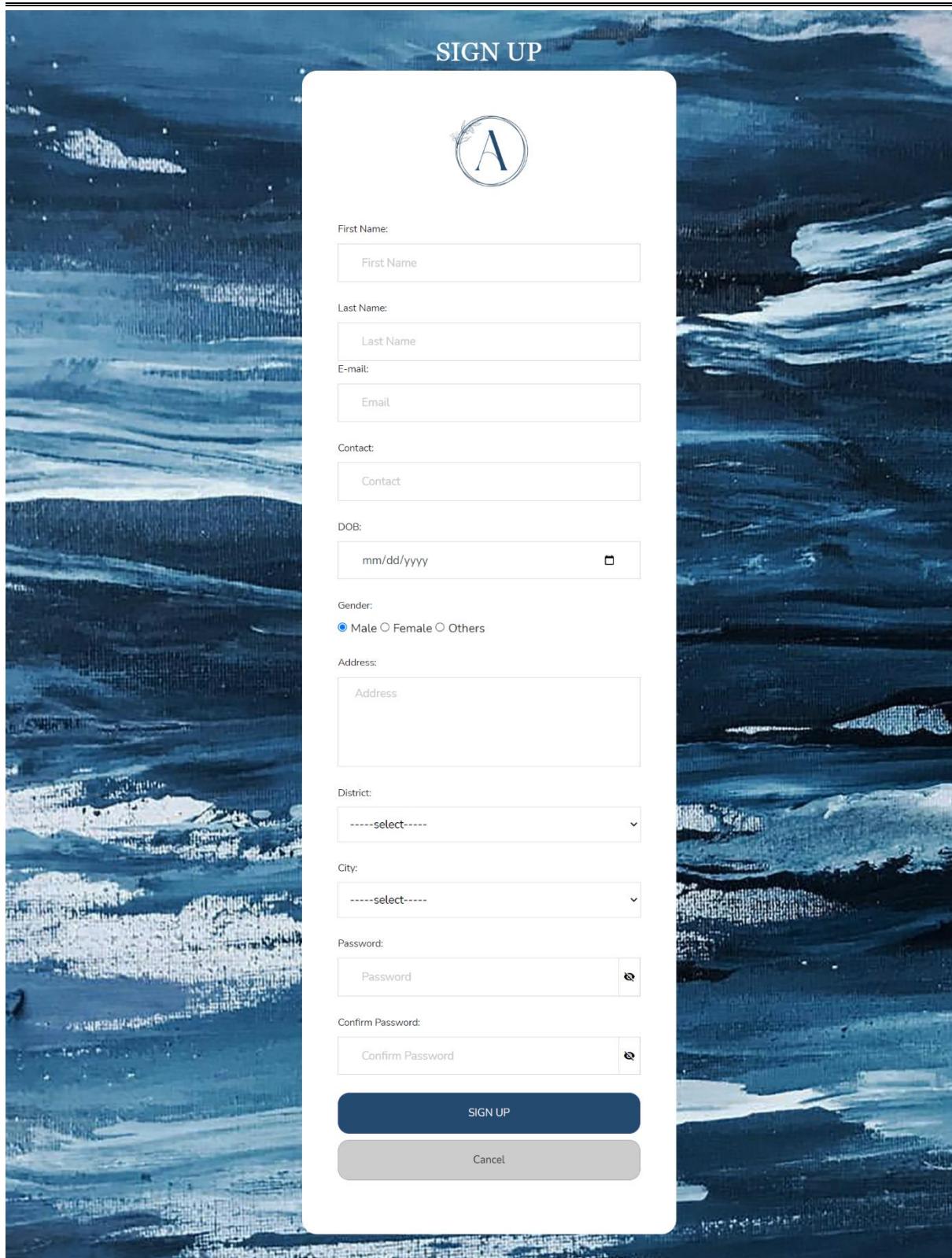


Fig 8.19 Seller Registration



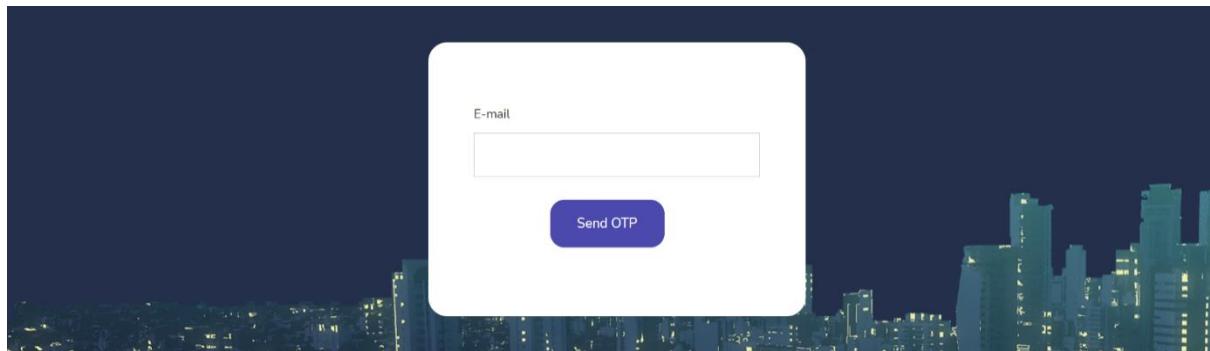


Fig 8.21 forgot Password

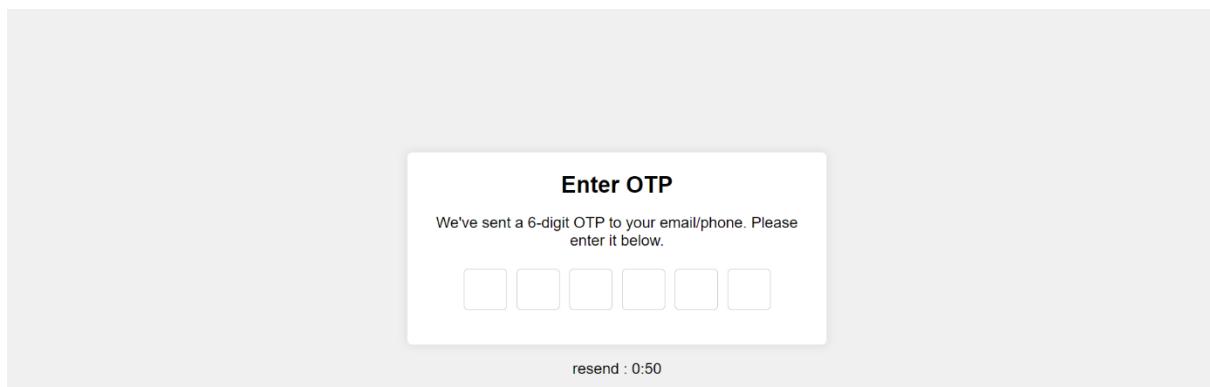


Fig 8.22 password recovery OTP

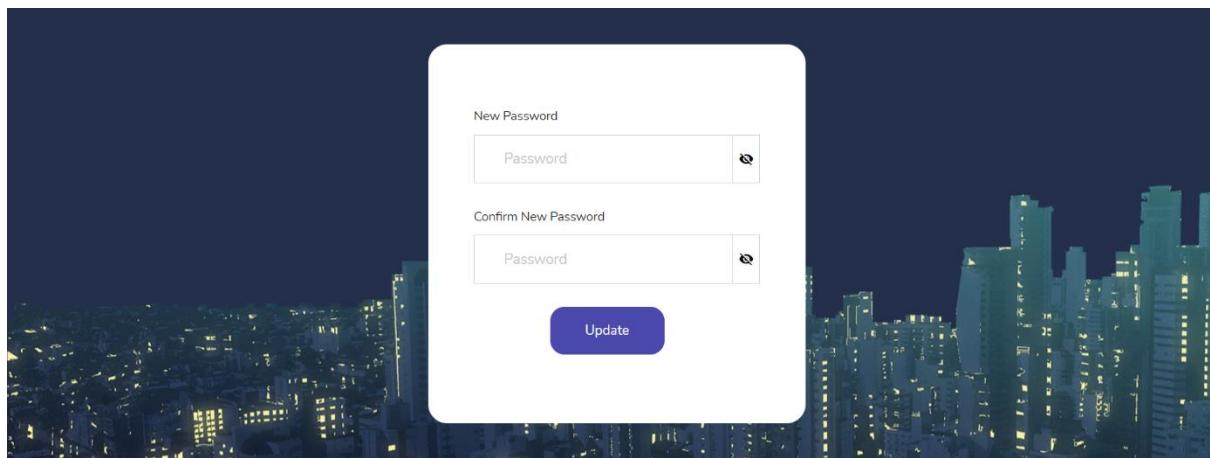


Fig 8.23 Change Password

The screenshot shows the Seller Dashboard for 'ARTISANAL'. The top navigation bar includes a profile icon and a notification bell. The dashboard features a sidebar with links like 'Dashboard', 'My Account', 'My Products', etc. The main area displays a welcome message 'Hello, ANU MATHEWS' and a promotional image of crocheted narwhals. It also shows performance metrics: 'Today's orders: 0', 'Total number of products ordered: 8', 'Number of followers: 5', and 'Number of products sold: 3'. Below these are two charts: 'Sales Per Product - Pie Chart' and 'Sales By Category - Bar Chart'. The bottom of the page includes copyright and hand-crafted notices.

Fig 8.24 Seller Dashboard

The screenshot shows the 'My Account' section of the Seller Dashboard. It features a large profile picture placeholder. Below it, account details are listed: Name: ANU MATHEWS, Email: anumathews002@gmail.com, Contact: 9961240693, Date of Join: 2023-09-05, Address: Koikakudy House Nellimattom P O Nellimattom, Pincode: 686691, City: KOTHAMANGALAM, and District: ERNAKULAM. There are two buttons at the bottom: 'Edit Account' and 'Change Password'. The sidebar on the left is identical to Fig 8.24.

Fig 8.25 Seller My Account

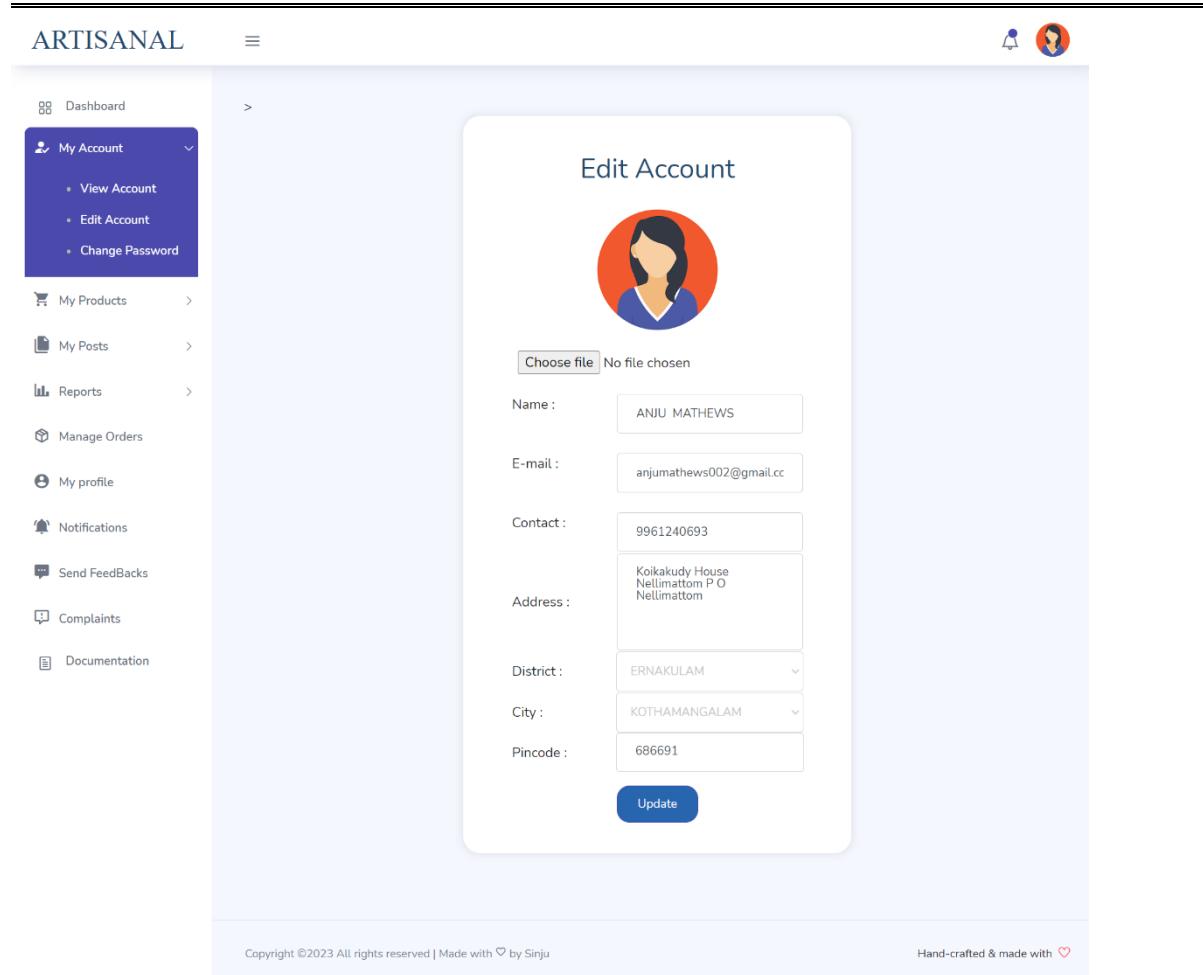


Fig 8.26 Seller Edit Account

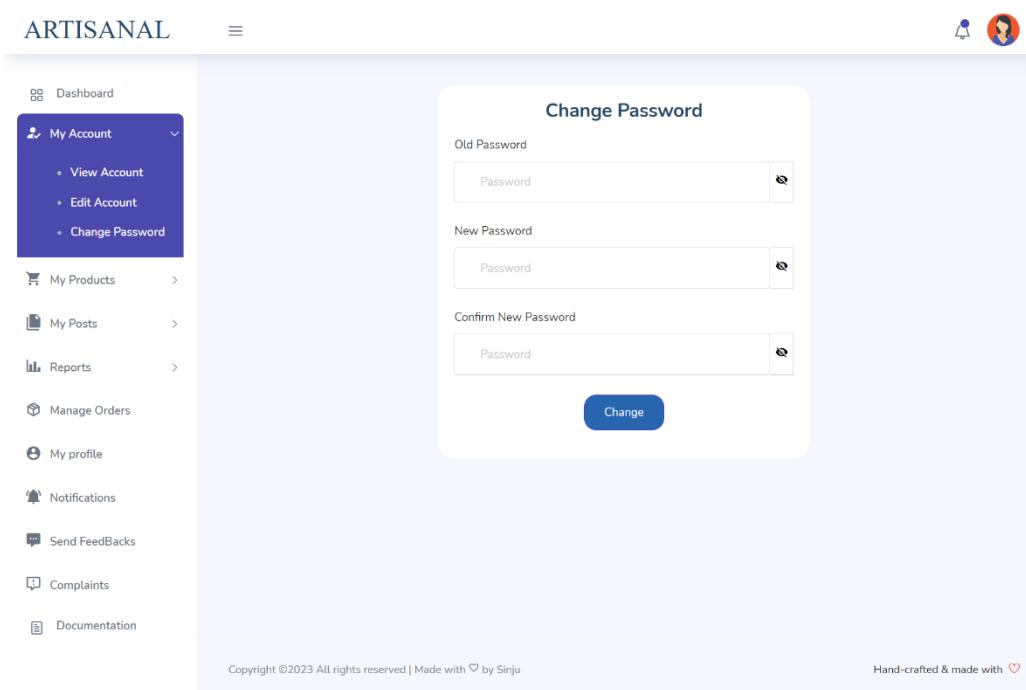


Fig 8.27 Seller Change Password

Fig 8.28 Seller Add product

Product Name	manage Images	color	material	type	price
AJ ASSORTED GREEN BRACELET	 Click here	green	beads	WOMEN GIRLS	200 #e
AJ BLUE BRACELETS	 Click here	blue	beads	WOMEN GIRLS	300 #b
AJ TWISTED BRACELET	 Click here	apricot	14k gold filled, pearl	WOMEN GIRLS	500 #p

Fig 8.29 Seller My Products

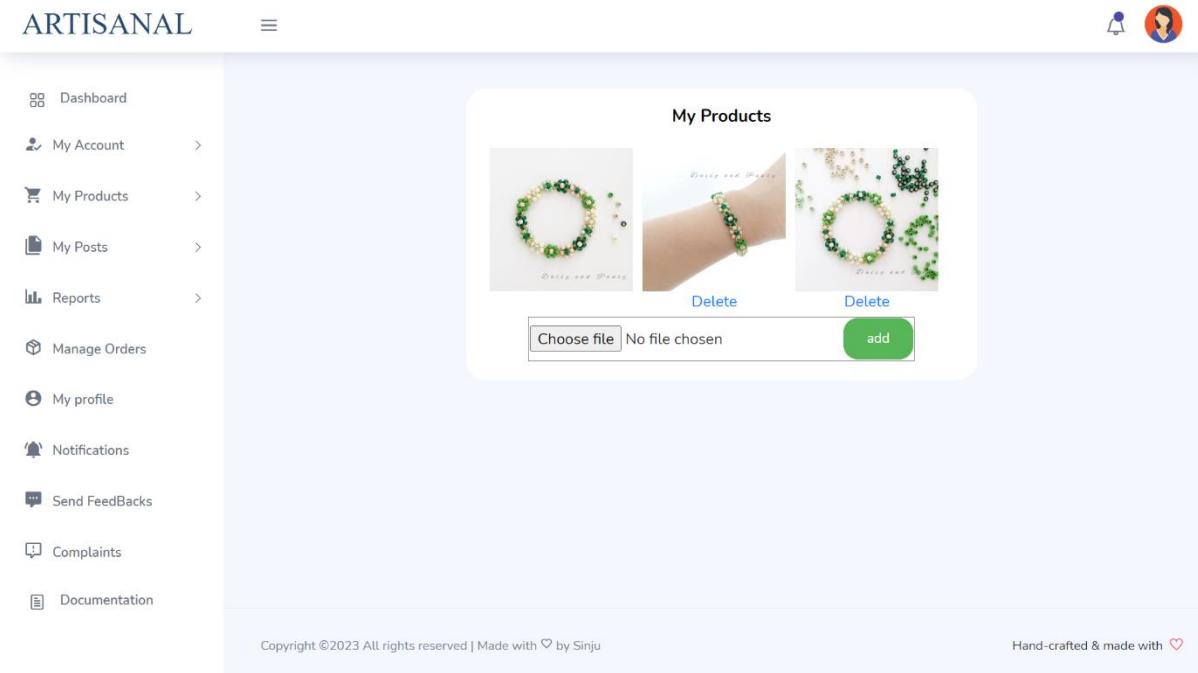


Fig 8.30 Add Product images

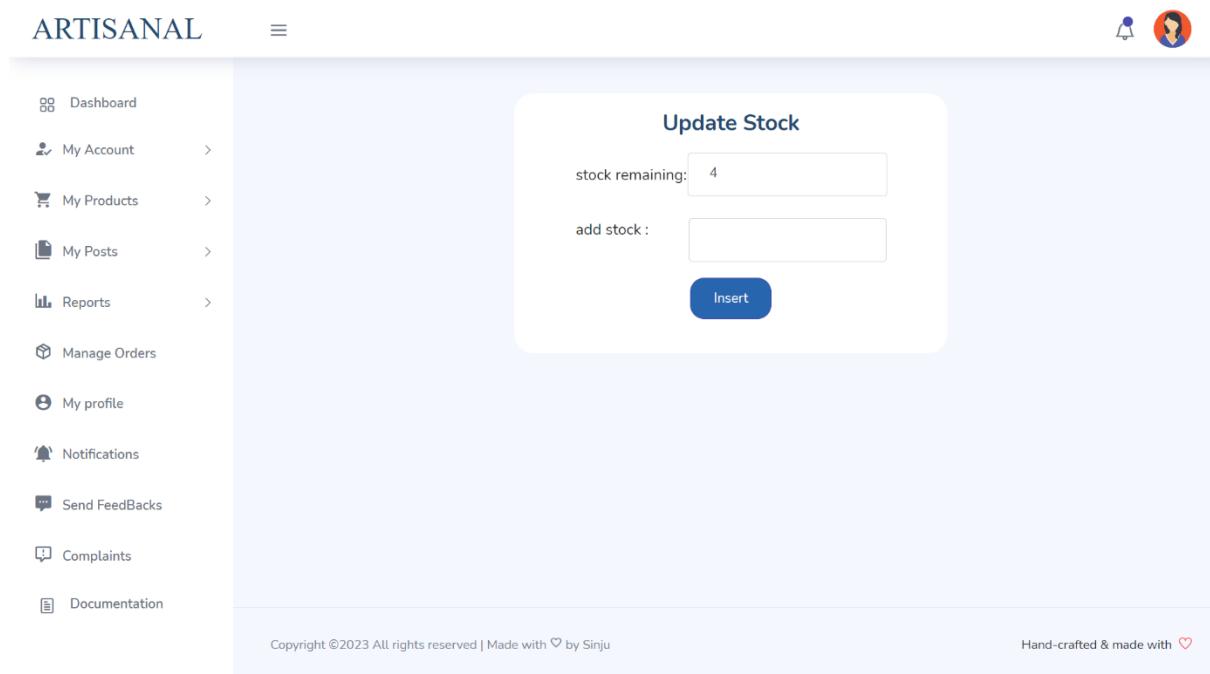


Fig 8.31 Add product Stock

The screenshot shows the 'Edit Product' interface within the Artisanal application. On the left, there is a sidebar with various navigation options: Dashboard, My Account, My Products, My Posts, Reports, Manage Orders, My profile, Notifications, Send FeedBacks, Complaints, and Documentation. The main area is titled 'Edit Product' and contains the following fields:

- product Name: AJ ASSORTED GREEN BR
- Color: green
- Material : beads
- Type:
 - MEN
 - WOMEN
 - UNISEX
 - KIDS
 - BOYS
 - GIRLS
- Price : 200
- Search Keys/ tags: #elegant bracelet #hand chain #green chain
- description: 18cm -long high quality beads
- category: ACCESSORIES
- Sub Category: BRACELETS

A blue 'update' button is located at the bottom right of the form. At the very bottom of the page, there is a copyright notice: 'Copyright ©2023 All rights reserved | Made with ❤ by Sinju' and a footer message: 'Hand-crafted & made with ❤'.

Fig 8.32 Edit product

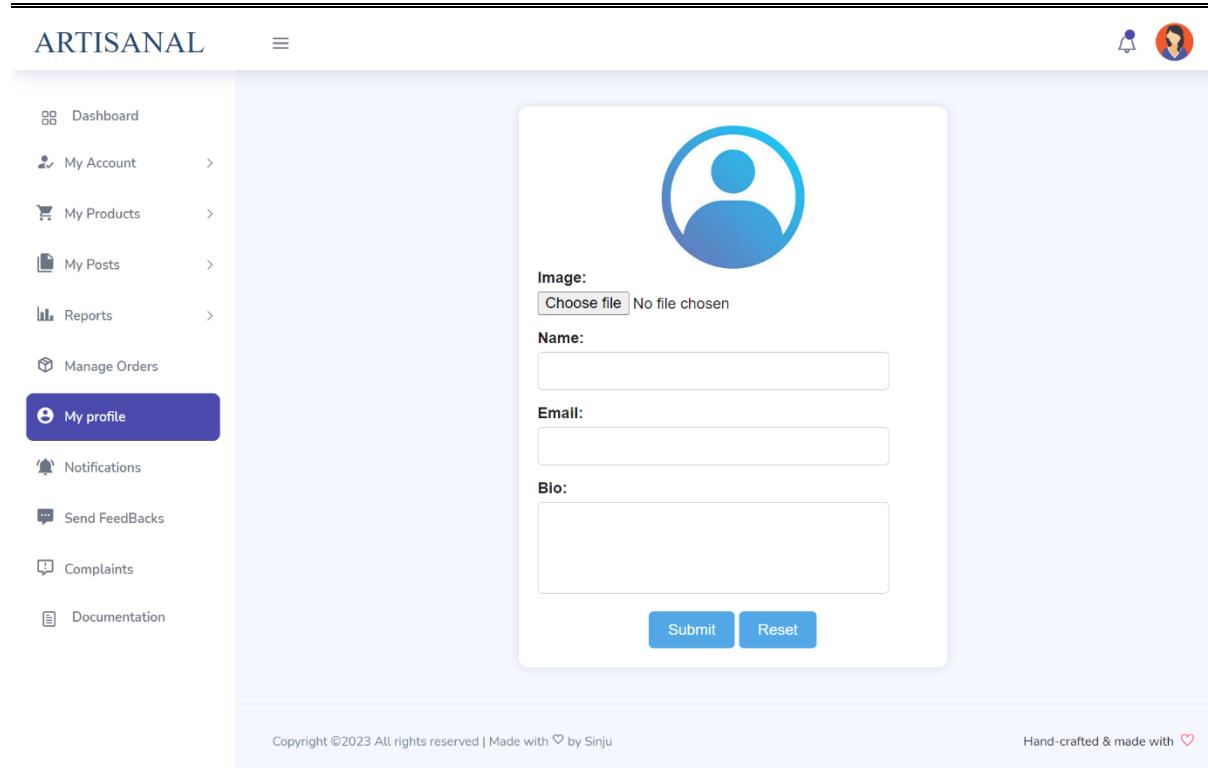


Fig 8.33 Seller Set profile

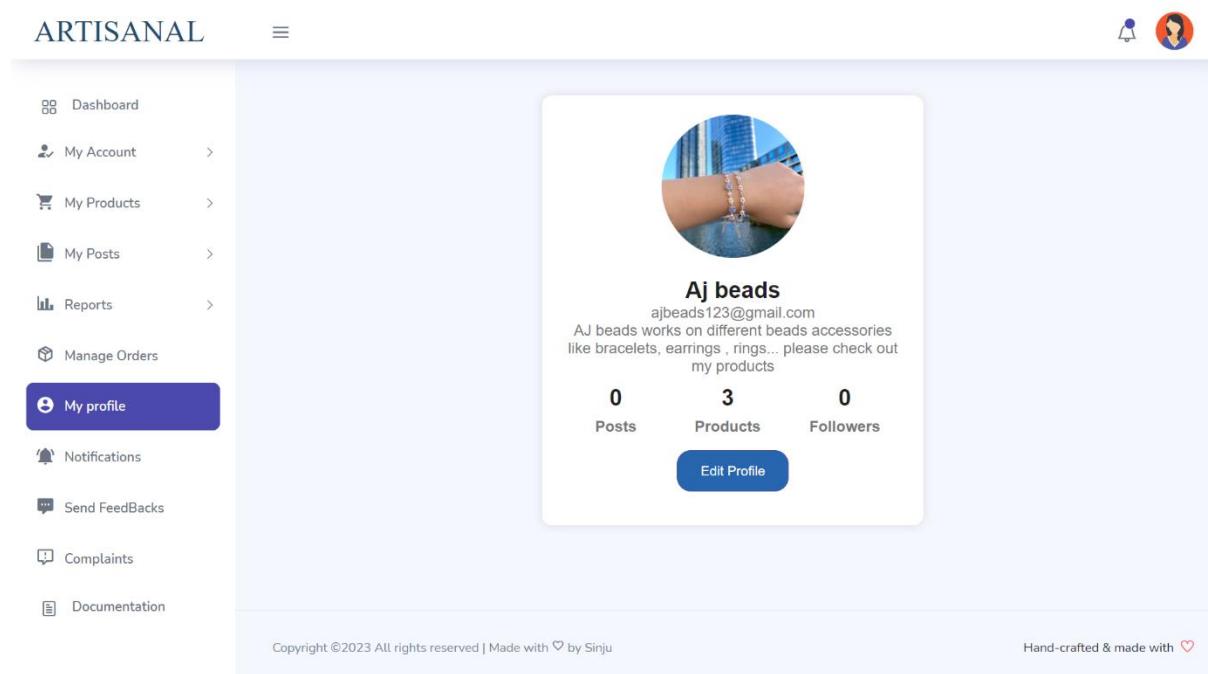


Fig 8.34 Seller My Profile

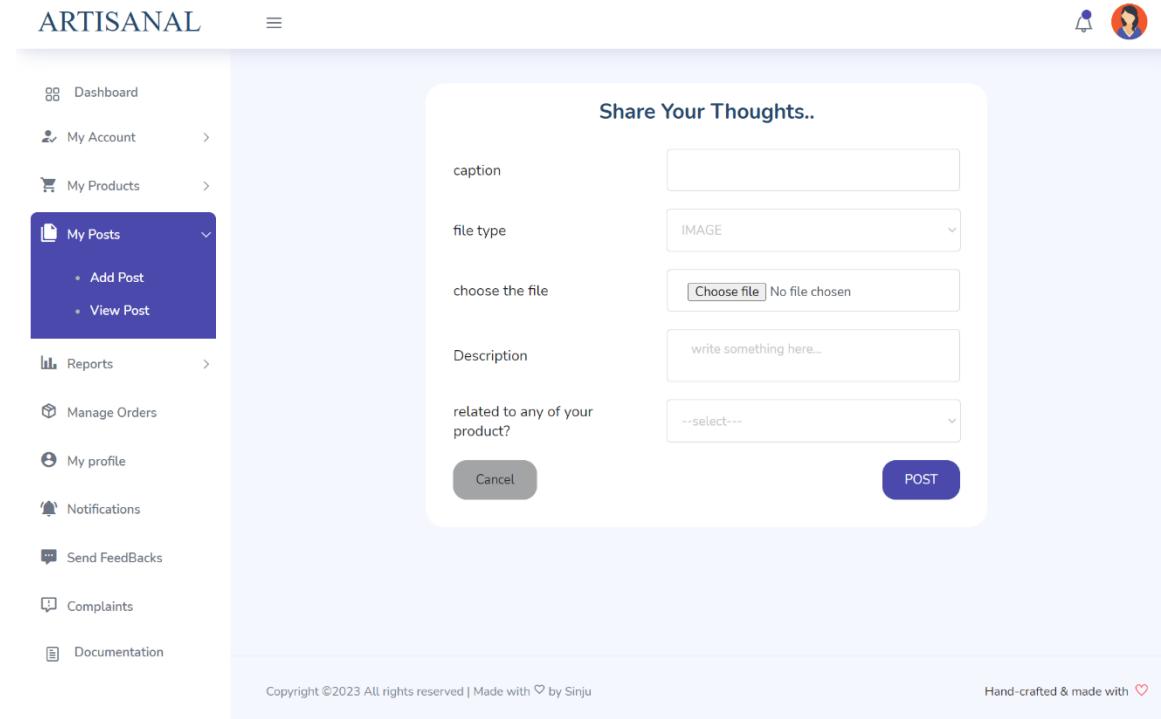


Fig 8.35 Seller Add Posts

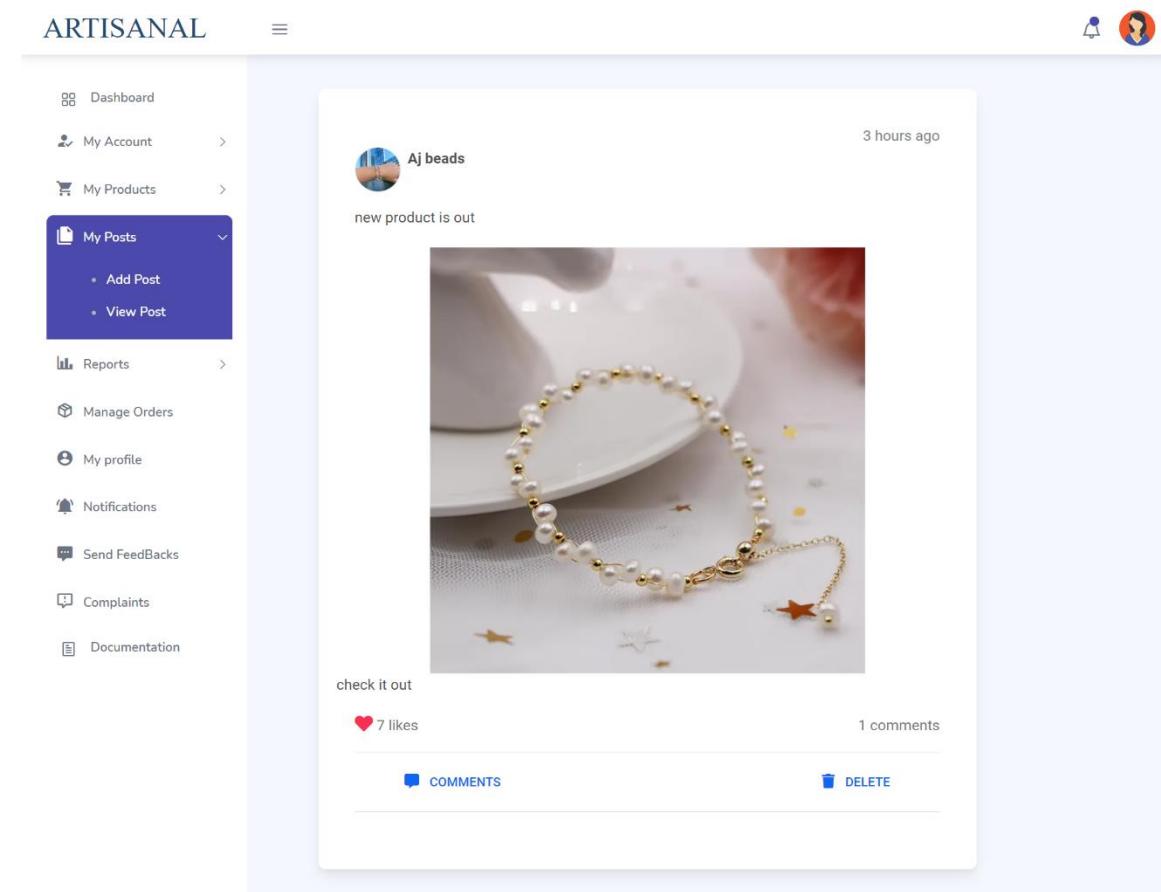


Fig 8.36 Seller My Posts

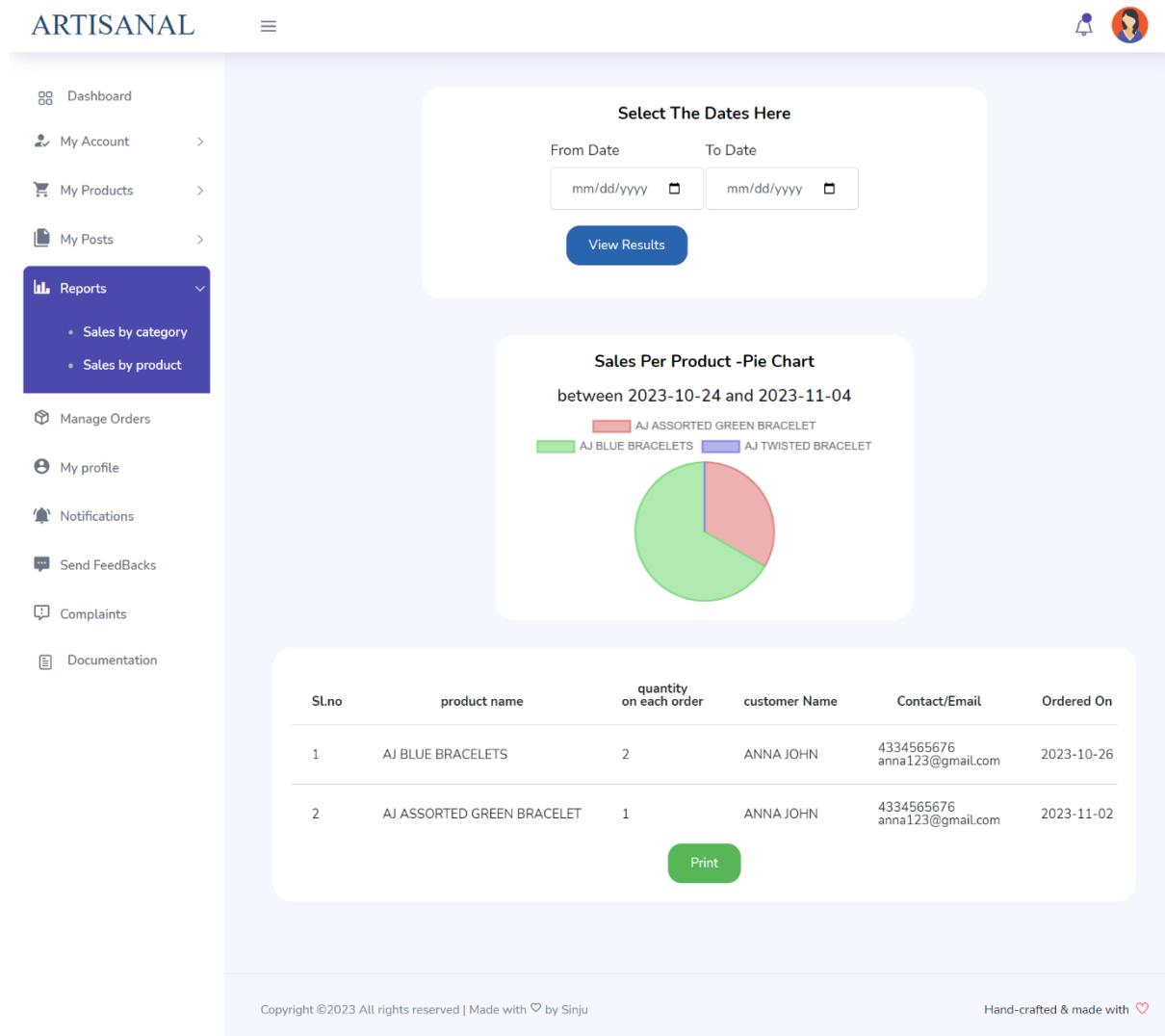


Fig 8.37 Seller sales per product

The screenshot shows the 'Orders Received' section of the Artisanal application. On the left sidebar, under the 'Manage Orders' category, there is a purple button labeled 'Mark Delivered'. In the main content area, there is a table with two rows of order data:

sL.no.	product name	quantity	customer Name	contact	E-mail
1	AJ BLUE BRACELETS	1	PAILY SAJI	8281860108	pailsaji08@gmail.com
2	AJ ASSORTED GREEN BRACELET	1	ANNA JOHN	4334565676	anna123@gmail.com

Below this, there is a 'List Of Shipped' section with one row of data:

contact	E-mail	delivery address	Ordered On	order status
254017	arsharajan02@gmail.com	ARSHA RAJAN 9497254017 Mettakottil H N.mazhuvannoor P.O Valamboor KOLENCHERY 682311 ERNAKULAM	2023-11-06	<button>Mark Delivered</button>

Then there is a 'List Of Delivered' section with one row of data:

customer Name	contact	E-mail	delivery address	Ordered On	order status
ANNA JOHN	4334565676	anna123@gmail.com	ANNA JOHN 4334565676 dharampuram ELAMKUNNAPUZHA 682503 ERNAKULAM	2023-10-26	<button>completed ✓</button>

Finally, there is a 'List Of Cancelled' section which is currently empty.

At the bottom of the page, there is a footer with the text 'Copyright ©2023 All rights reserved | Made with ❤ by Sinju' and 'Hand-crafted & made with ❤'.

Fig 8.38 View Orders

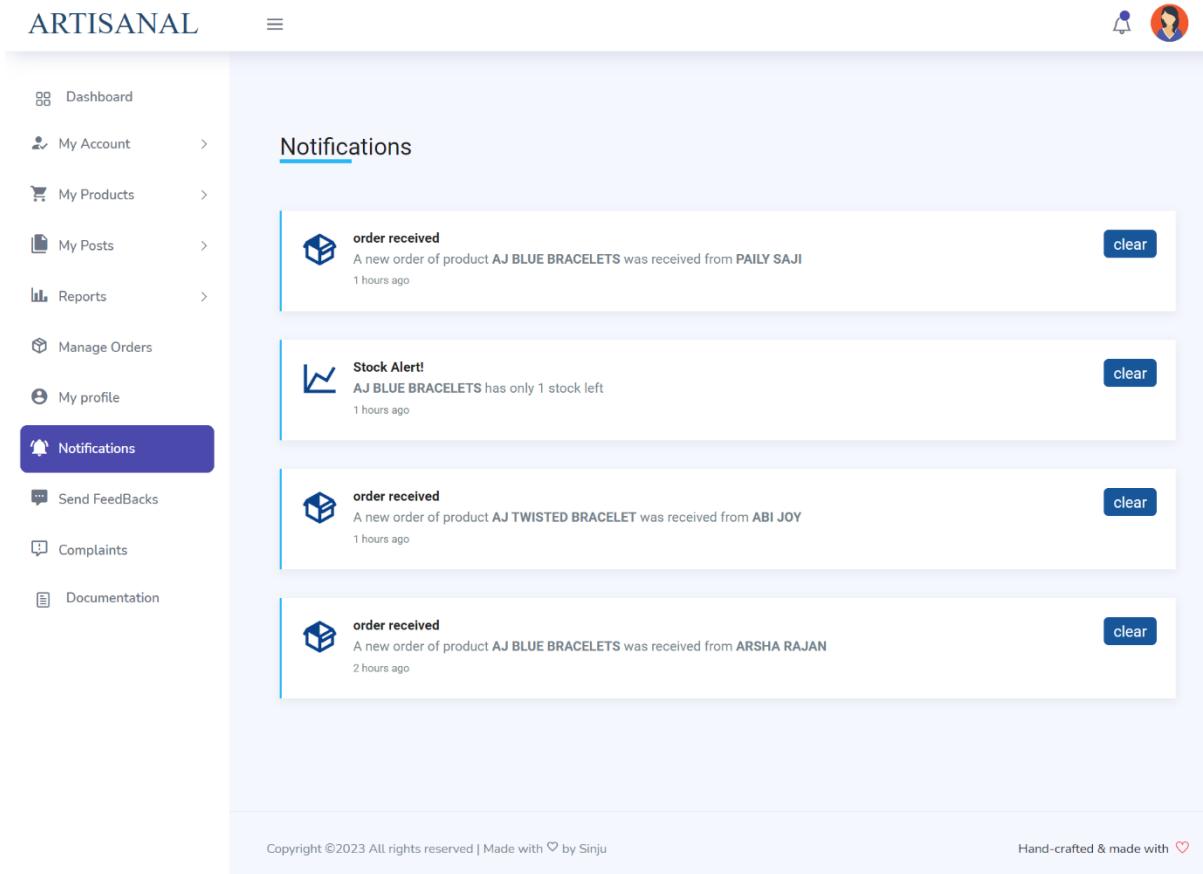


Fig 8.39 Seller notification

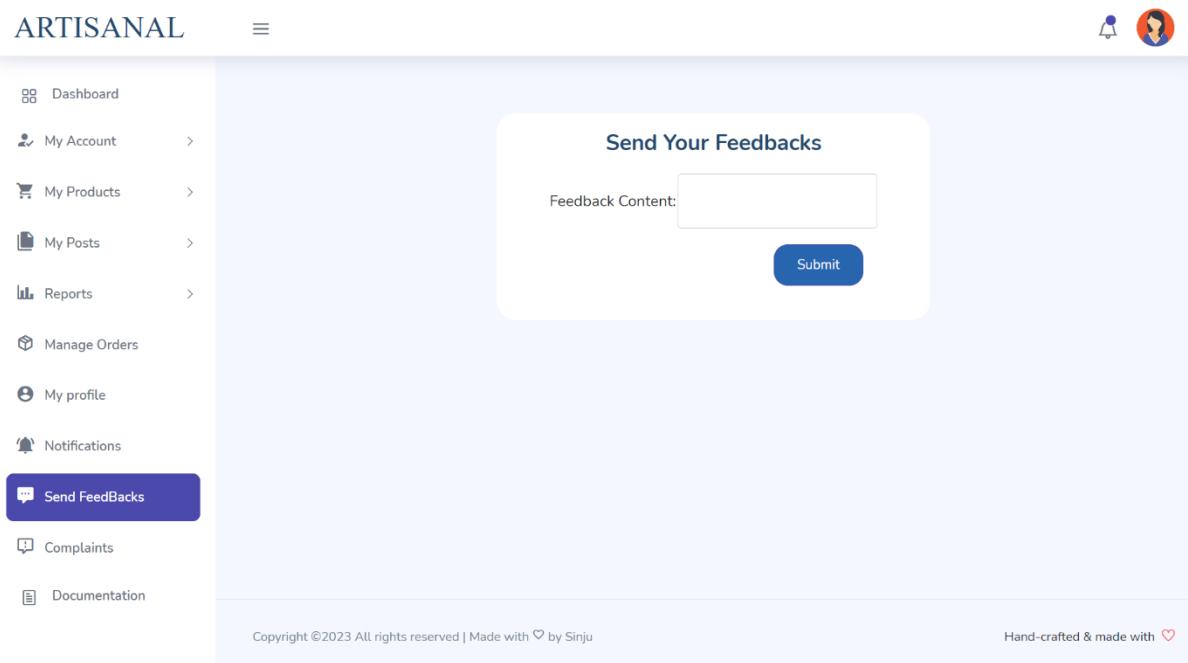


Fig 8.40 Seller send Feedback

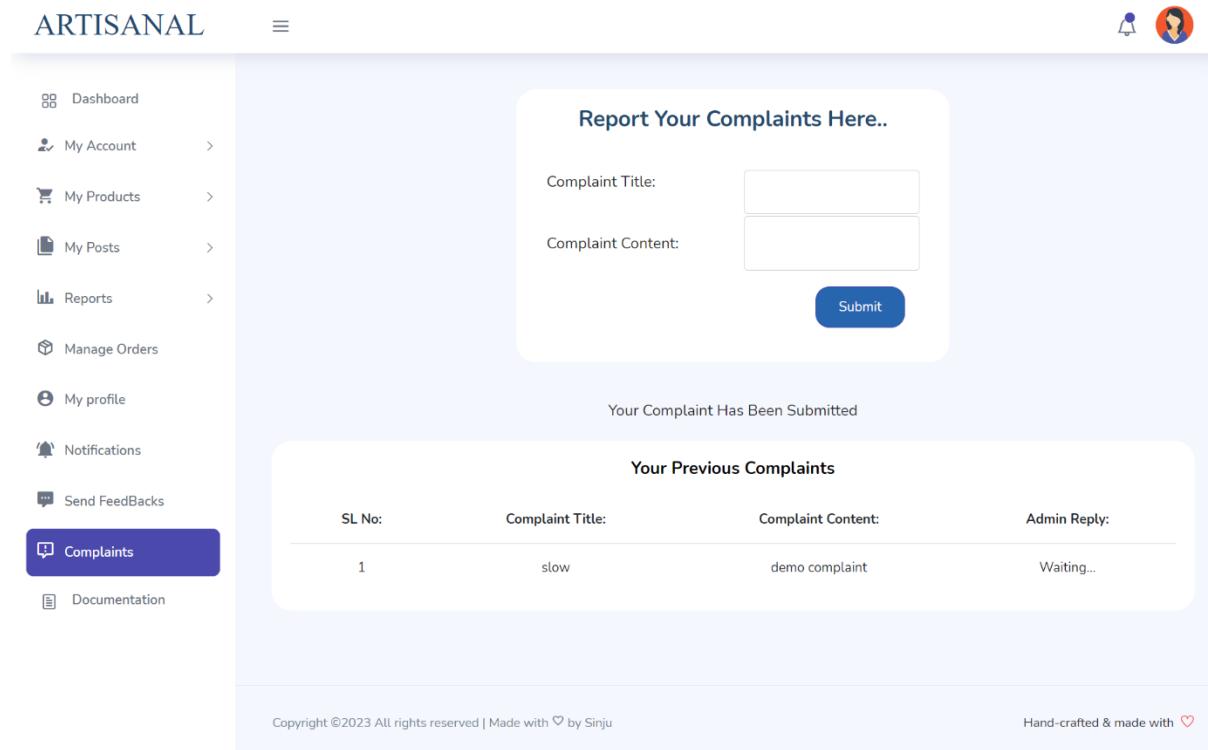


Fig 8.41 Seller send Complaint

Best handmade products

My Account | **logout**

ARTISANAL

Home Account Shop Feeds Sellers My Orders Contact us

2 0 1

Handmade JEWELLERY

SHOP NOW

OUR SELLERS

"There is something deeply satisfying in shaping something with your hands. Proper artificing is like a song made solid. It is an act of creation."

- Patrick Rothfuss

ABOUT US

The Objective of Artisanal is to support small handicraft business and make a website that is fully dedicated to handmade products which attracts customer who like or prefer unique handmade products over machine made mass produced products. Artisanal sells a wide range of handmade products like paintings, bags, wallets, hats, key chains, cups, home decors etc. And simple cloths like scarves, mufflers, sweaters, Beanies etc. Artisanal wants to give an amazing online shopping experience to everyone who visits the site.

GET IN TOUCH

Any questions? Let us know in artisanalhelp@gmail.com or call us on (+91) 6235009367

CONNECT WITH US

in LinkedIn

FEEDBACK

PayPal VISA MasterCard American Express

Copyright © 2023 All rights reserved | Made with ❤ by sinju

Fig 8.42 Customer Home Page

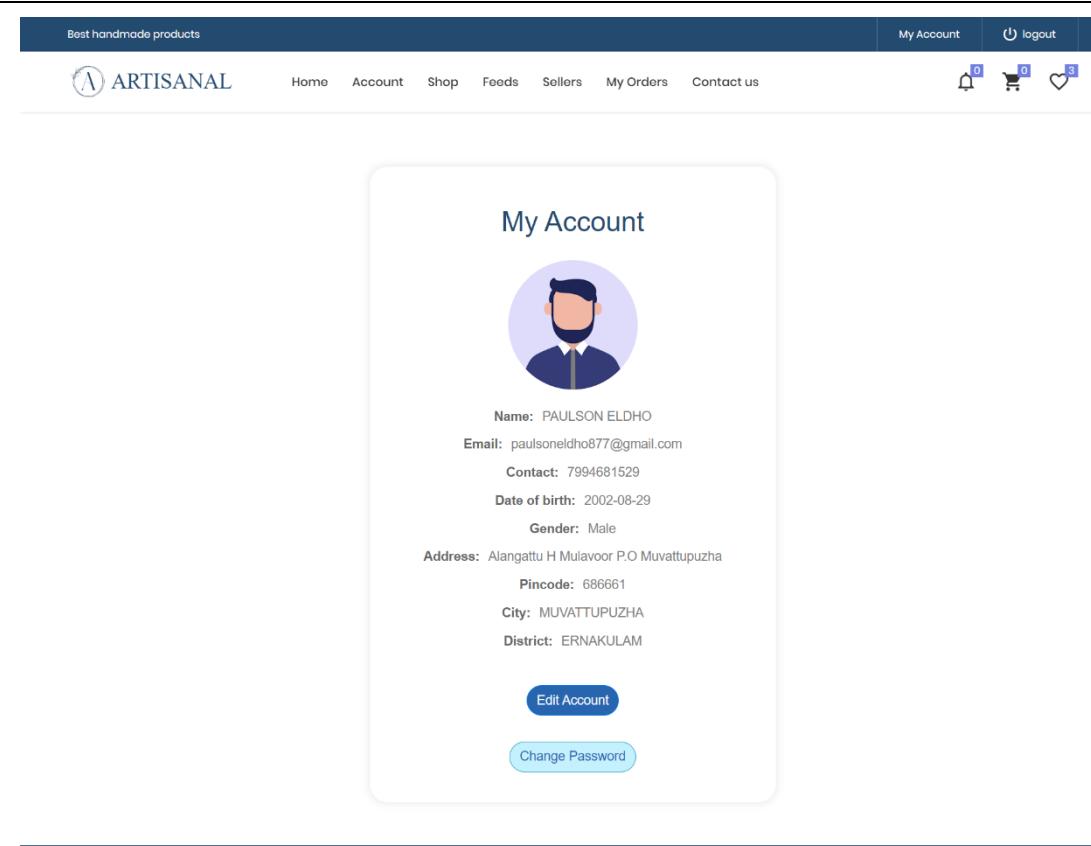


Fig 8.43 Customer My Account

This screenshot shows the 'Edit Account' page for the same customer. The top navigation bar and header icons are identical to Fig 8.43. The main form area contains the user's profile picture and several input fields for updating personal information:

- Name:** PAULSON ELDHO
- E-mail:** paulsoneldho877@gmail.com
- Contact:** 7994681529
- Date Of Birth:** 08/29/2002
- Address:** Alangattu H Mulavoor P.O Muvattupuzha
- District:** ERNAKULAM
- City:** MUVATTUPUZHA
- Pincode:** 686661

There is also a file upload field labeled 'Choose file' with the placeholder 'No file chosen'. At the bottom of the form is a blue 'Update' button.

Fig 8.44 Customer Edit Account

Best handmade products

My Account | logout

ARTISANAL

Home Account Shop Feeds Sellers My Orders Contact us

0 2 0

ALL MEN WOMEN UNISEX KIDS BOYS GIRLS

Filter Search

AJ ASSORTED GREEN BRACELET

★★★★★ 5/5 ★★★★★ 200/-

LINO THE BUNNY

★★★★★ 5/5 ★★★★★ 650/-

AJ BLUE BRACELETS

★★★★★ 5/5 ★★★★★ 300/-

AJ TWISTED BRACELET

★★★★★ 5/5 ★★★★★ 500/-

FLORAL BAG

★★★★★ 5/5 ★★★★★ 700/-

SUNFLOWER FLORAL BAG

★★★★★ 5/5 ★★★★★ 700/-

LEATHER KEY CHAIN

★★★★★ 5/5 ★★★★★ 90/-

LEATHER BAG

★★★★★ 5/5 ★★★★★ 1300/-

Fig 8.45 Search Product

Best handmade products

My Account | logout

ARTISANAL Home Account Shop Feeds Sellers My Orders Contact us 2 0 1



LINO THE BUNNY

5 stars

₹650 /-

-> Length-30cm -> Width-18cm -> Stuffed with fiberfill -> Hand wash only

Type	KIDS,
Color	pink
Material	wool yarn
category	TOYS & PLUSHIES

Buy now add to cart

Ratings

0.0 / 5

5 star (0) 4 star (0) 3 star (0) 2 star (0) 1 star (0)

★ ★ ★ ★ ★

0 Review

Product Seller

support your fav sellers

[see more](#)



Felix

felix123@gmail.com
I am crocheter. And i have been working on cute amigurumi

Fig 8.46 View Product

Best handmade products

My Account | logout

ARTISANAL Home Account Shop Feeds Sellers My Orders Contact us 2 0 3

WISHLIST



AJ BLUE BRACELETS

★★★★★ ★★★★★

300/-

add to cart

remove from wishlist



LINO THE BUNNY

★★★★★ ★★★★★

650/-

add to cart

remove from wishlist



LION AMIGURUMI

★★★★★ ★★★★★

500/-

add to cart

remove from wishlist

Fig 8.47 Wishlist

Best handmade products

My Account | logout

ARTISANAL Home Account Shop Feeds Sellers My Orders Contact us 0 3 3

Cart

Image	Product	Price	Quantity	Remove	Total
	MULTI COLOR CROCHET BAG	₹ 250	2	Remove	₹ 500
	LION AMIGURUMI	₹ 500	1	Remove	₹ 500
	FLORAL BAG	₹ 700	1	Remove	₹ 700

Grand Total
₹1700.00

Checkout

Fig 8.48 My Cart

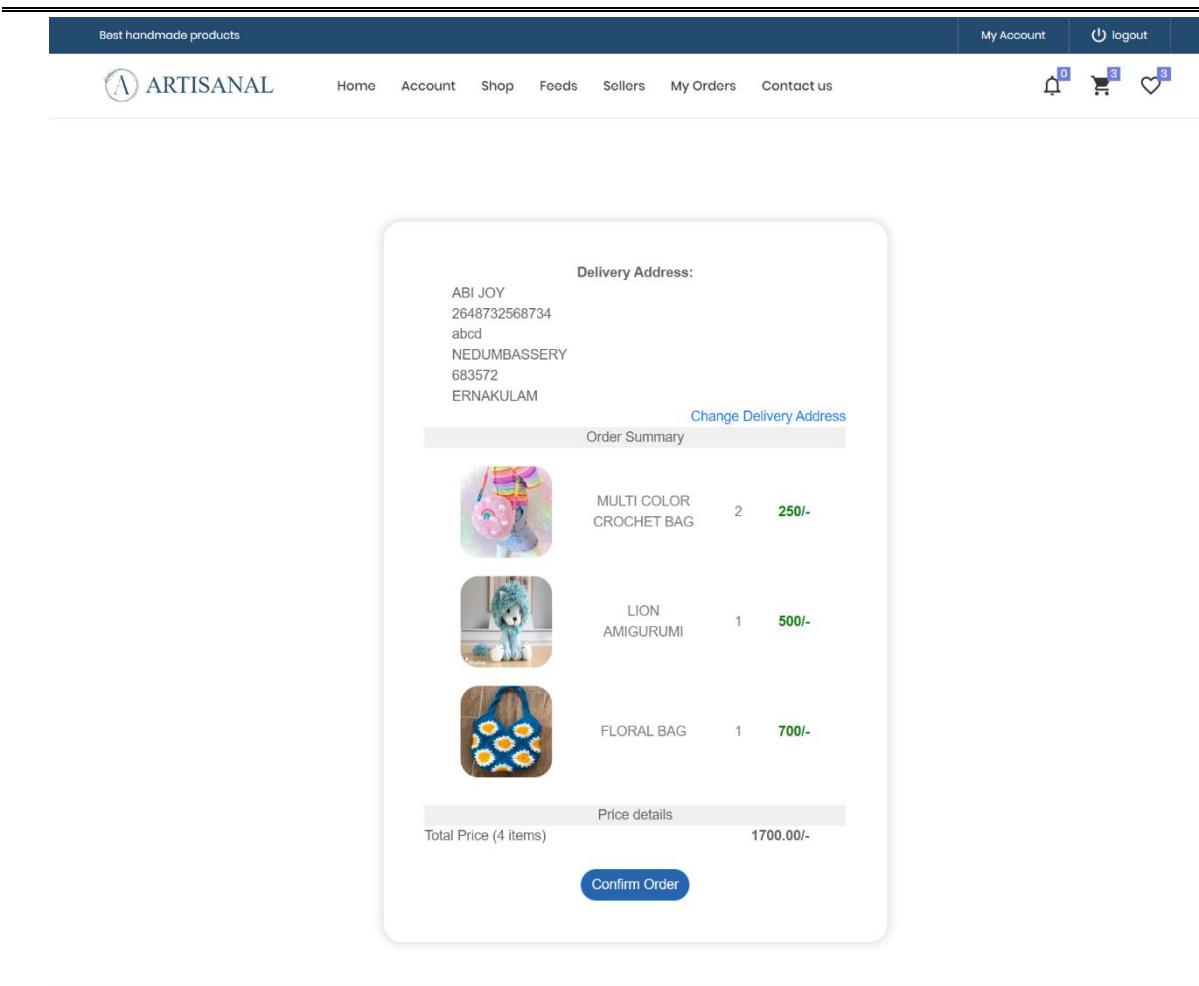


Fig 8.49 Order Summary

The screenshot shows the 'choose Payment method' page of the Artisanal website. It features four payment options: Credit Card (selected), Paypal, Cash on Delivery, and Net Banking. Below these are fields for Card Owner Name, Card number (with a placeholder 'Valid card number' and icons for Visa, Mastercard, and American Express), Expiration Date (MM and YY fields), and CVV (with a placeholder 'CVV'). A large blue 'Confirm Payment' button is at the bottom.

Fig 8.50 Payment Gateway

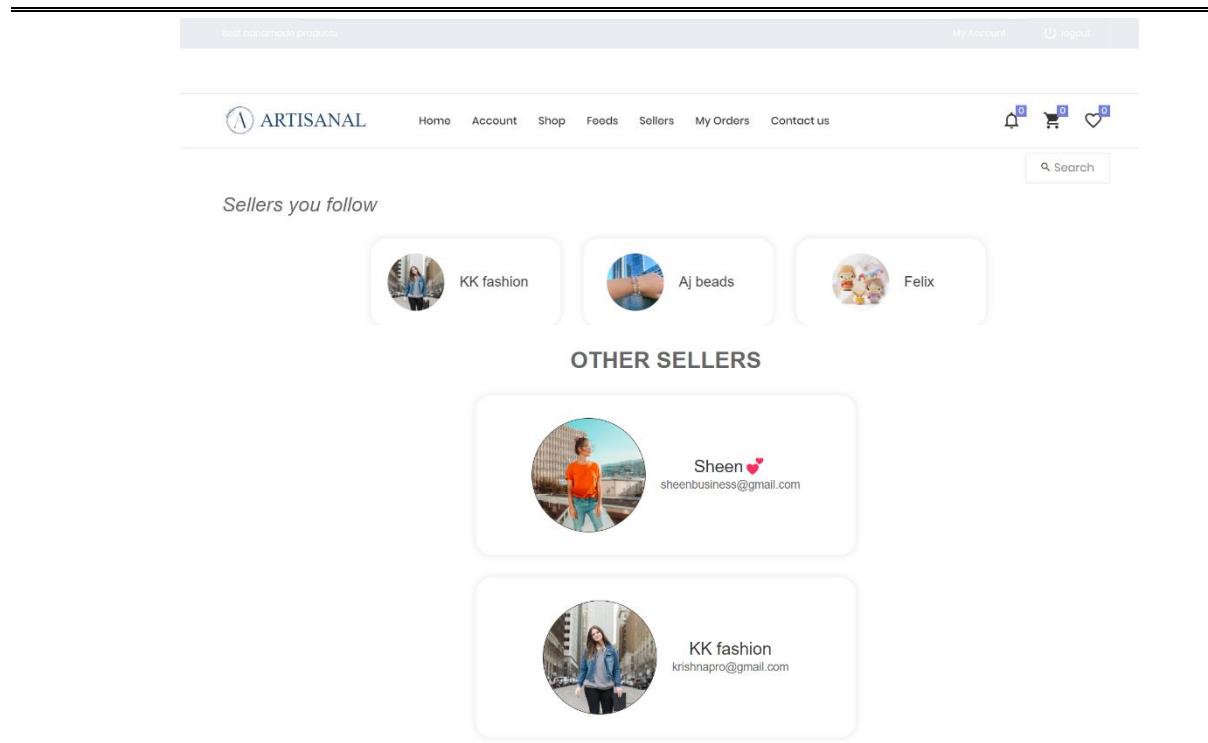


Fig 8.51 View Sellers

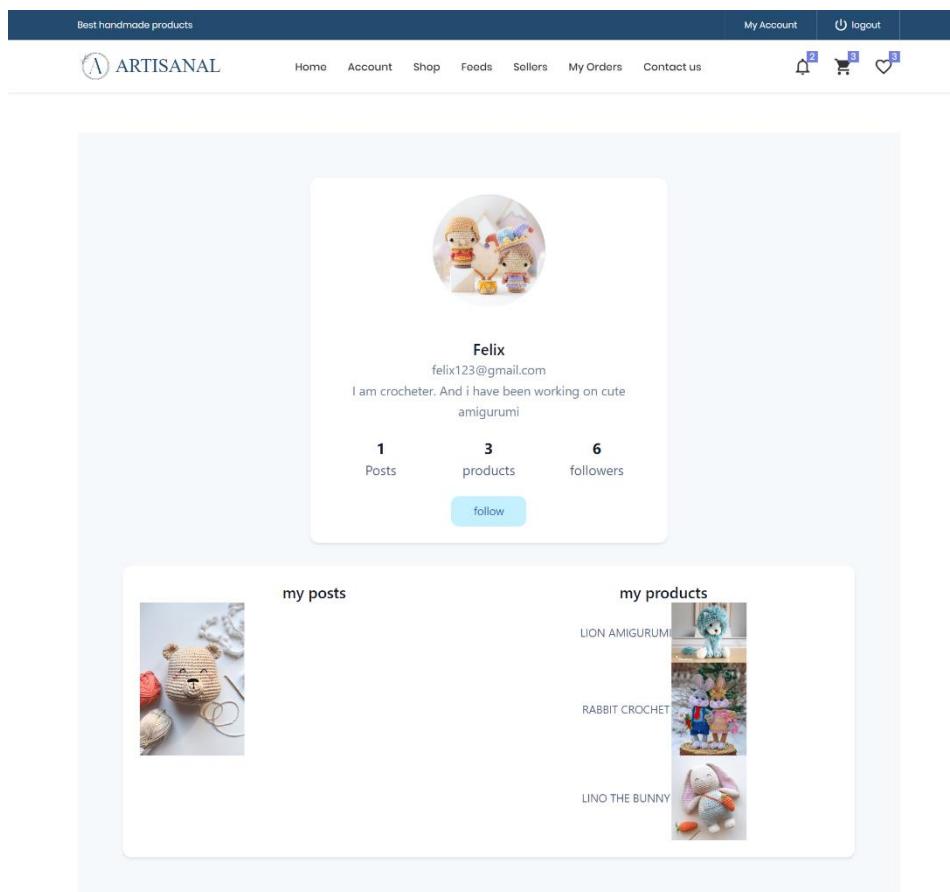


Fig 8.52 Each Seller

Best handmade products

My Account | logout

ARTISANAL Home Account Shop Feeds Sellers My Orders Contact us

2 3 3

Aj beads 4 hours ago

new product is out

check it out
See this product..

7 likes 1 comments

LIKE COMMENT REPORT

ganga 2 days ago

my Favourite keychain I ever made.

first time posting something..

8 likes 2 comments

LIKE COMMENT REPORT

Fig 8.53 Feeds

The screenshot shows the 'My Orders' section of the Artisanal app. It lists three delivered orders:

- Order 1:** Order Status: Delivered, ₹ 500/-, x 2 items at 250/- each. Seller: FATHIMA SHEEN K S, Punnamattom west Muvattupuzha, fathimasheen524@gmail.com.
- Order 2:** Order Status: Delivered, ₹ 500/-, x 1 item at 500/- each. Seller: FELIX LEE, aramana p.o, felix123@gmail.com.
- Order 3:** Order Status: Delivered, ₹ 700/-, x 1 item at 700/- each. Seller: FATHIMA SHEEN K S, Punnamattom west Muvattupuzha, fathimasheen524@gmail.com.

Total Price: ₹ 1700/-

[View More](#)

Delivery Address:
ABI JOY
2648732568734
abcd
NEDUMBASSERY
683572
ERNAKULAM

Order ID: 48
Ordered On: 2023-10-06
Payment ID: 32
Payment Method: PAY PAL
[Get Invoice](#)

Fig 8.54 My Orders

Billing Address
ABI JOY
abcd
NEDUMBASSERY
683572
ERNAKULAM
contact: 2648732568734

Shipping Address
ABI JOY
2648732568734
abcd
NEDUMBASSERY
683572
ERNAKULAM

order Date: 2023-10-06
ordered from: Artisanal

Invoice Generated on : 2023-11-06
Bill number: 202310060048

#	Product	description	Sold by	Price	Qty	total price
1	MULTI COLOR CROC HET BAG	NALLA BAAG.... ENT E BAAG... CUTE BAA G... VANG BAAG...	FATHIMA SHEEN K S Punnamattom west Muvattupuzha fathimasheen524@gmail.com	250	2	500
2	LION AMIGURUMI	nice Lion Amigurumi height: 30cm width: 20cm	FELIX LEE aramana p.o felix123@gmail.com	500	1	500
3	FLORAL BAG	35cm-length 30cm -width 20cm dth hand wash 100% cotton	FATHIMA SHEEN K S Punnamattom west Muvattupuzha fathimasheen524@gmail.com	700	1	700

Grand Total 1700 Rs

Fig 8.55 Invoice Bill

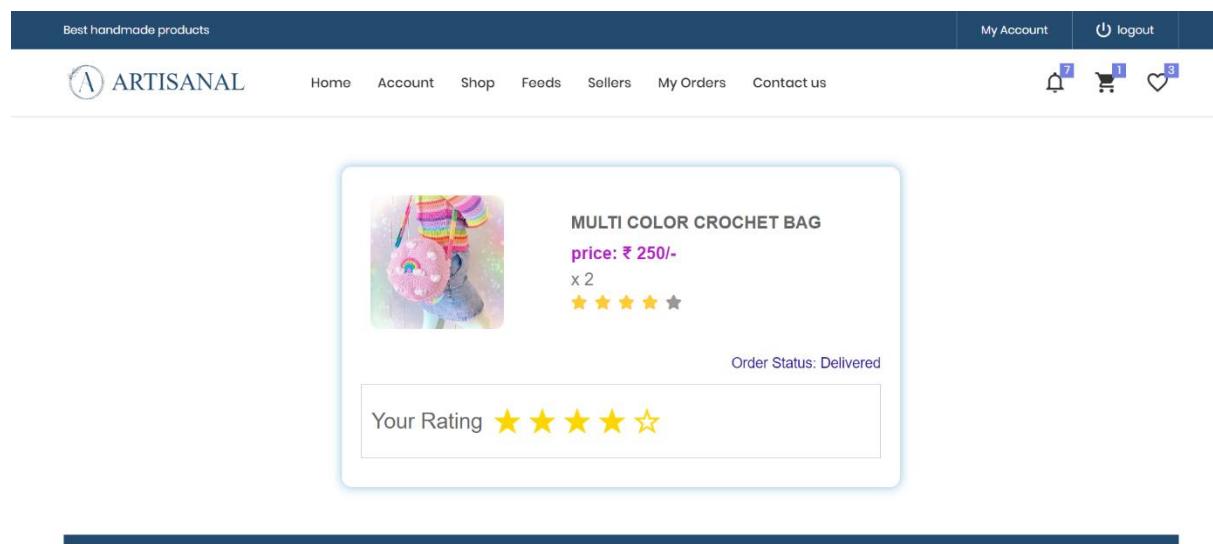


Fig 8.56 Each Order

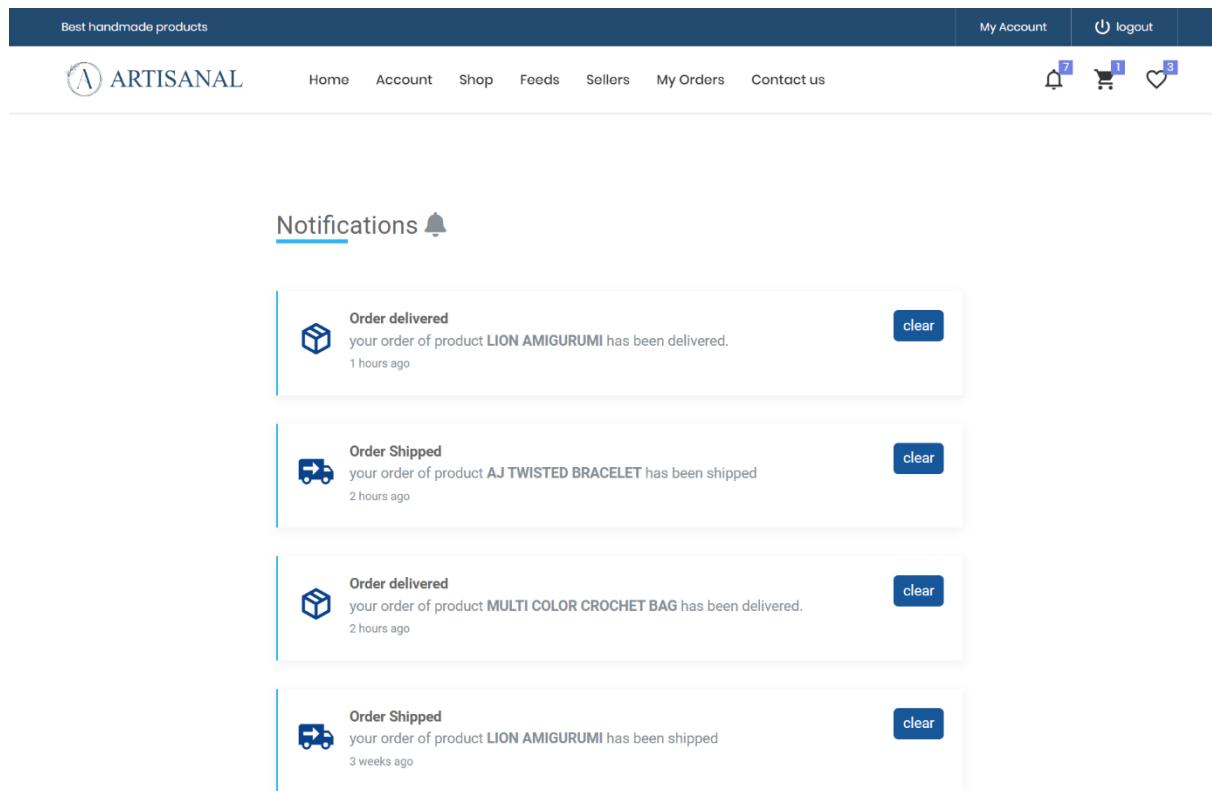


Fig 8.57 Customer Notification

The screenshot shows the Artisanal website's header with a dark blue bar containing "Best handmade products", "My Account", and a "logout" button. Below the header is a navigation bar with links: Home, Account, Shop, Feeds, Sellers, My Orders, and Contact us. To the right of the navigation are icons for a bell (0 notifications), a shopping cart (0 items), and a heart (3 likes). The main content area has a light gray background and features a centered form for sending feedback. The form includes a label "Feedback Content" above a large text input field, a "Submit" button at the bottom, and a small "x" icon in the top right corner of the input field.

Fig 8.58 Customer Feedback

The screenshot shows the Artisanal website's header and navigation bar, identical to Fig 8.58. The main content area features a form for reporting complaints. It includes labels "Complaint Title" and "Complaint Content" with their respective input fields, a "Submit" button, and a small "x" icon in the top right corner of the content input field. Below this form is a section titled "Your previous complaints" containing a table with one row of data.

SL No:	Complaint Title:	Complaint Content:	Admin Reply:
1	complaint example2	complaint example2 content	Waiting...

Fig 8.59 Customer Complaint