**Online Jewellery System**

**TABLE OF CONTENTS PROJECT**

SYNOPSIS.......................................................................................................................8

CHAPTER 1: INTRODUCTION.................................................................................... 11

1.1 Background ............................................................................................................... 11

1.2 Objectives................................................................................................................…11

1.3 Purpose, Scope and applicability ...........................................................…………….11

1.3.1 Purpose ............................................................................................……….11

1.3.2 Scope............................................................................................................ 12

1.3.3Aplicability.....................................................................................................13

CHAPTER 2: SURVEY OF TECHNOLOGY.................................................................. 14

CHAPTER 3: REQUIREMENTS AND ANALYSIS....................................................... 15

3.1 Problem definition: ........................................................................................………..15

3.1.1 Sub-System ...............................................................................……………15

3.2 Requirement specification: ......................................................................................... 15

3.2.1 Requirement Gathering……………............................................................. 15

3.2.2 Requirement analysis……….…….............................................................. 16

3.2.2.1 Functional Requirements ….…….............................................................24

3.2.2.2 Non-Functional Requirements ….…….....................................................24

3.2.2.3 System Requirements….……...................................................................24

3.3 Planning and Scheduling…………............................................................................ 28

CHAPTER 4: System Design........................................................................................... 30

4.1 Business rules…………............................................................................................. 30

4.2 Module diagram......................................................................................................... 30

4.3 Er diagram: ................................................................................................................ 30

4.3.1 Entity sets: ...................................................................................................33

4.3.2 Relationship sets......................................................................................... .37

4.3.3. ER Diagram.................................................................................................40

4.4 Schema diagram.......................................................................................................... 41

4.5 Data Flow Diagram (DFD)………….......................................................................... 42

4.5.1 Level 0 (Context level DFD)…….................................................................42

4.5.2 Level 1 DFD. ................................................................................................43

4.5.3 Level 2 DFD………………………………...………………………...……44

4.6 Use Case Diagram…………........................................................................................ 45

4.6.1Diagram…………………………………………………..………….………46

4.6.2 Use Case Description……………………………………………….………46

4.7 Scenario………………………………………………………………………..……..50

4.8 Sequence diagram........................................................................................................ 51

4.9 Activity Diagram-........................................................................................................ 58

4.10 State Diagram.............................................................................................................60

4.11 User interface Design................................................................................................. 61

4.12 Test cases Design........................................................................................................ 66

**List of tables**

Table 3.1 Task Table...........................................................................................................29

Table 4.1 Entity Relationship Diagram Symbols.................................................................32

Table 4.2 Schema Diagram Symbols....................................................................................41

Table 4.3 Data Flow Diagram Symbols................................................................................42

Table 4.4 Use Case Diagram Symbols.................................................................................45

Table 4.5 Sequence Diagram Symbols.................................................................................51

Table 4.6 Activity diagram Symbols....................................................................................58

Table 4.7 State Diagram Symbols........................................................................................60

Table 4.8 Test cases...............................................................................................................65

**List of figures**

Figure 0.1 Architecture................................................................................................9

Figure 3.1 Responses Through Google Forms............................................................24

Figure 3.2 Gantt Chart.................................................................................................29

Figure 4.1 module diagram..........................................................................................30

Figure 4.2 Customer Entity Set...................................................................................31

Figure 4.3 Product Entity Set......................................................................................34

Figure 4.4 Order Entity Set.........................................................................................35

Figure 4.5 Payment Entity Set....................................................................................35

Figure 4.6 Chatbot Entity Set......................................................................................36

Figure 4.7 Reviews Entity Set....................................................................................36

Figure 4.8 Reviews Entity Set....................................................................................37

Figure 4.9 Customer – customisation Relationship....................................................37

Figure 4.10 Customer – order Relationship................................................................38

Figure 4.11 Customer – Payment Relationship..........................................................38

Figure 4.12 Customer – Review Relationship............................................................38

Figure 4.13 Order - Product Relationship...................................................................39

Figure 4.14 order - payment Relationship...................................................................39

Figure 4.15 Review - Product Relationship.................................................................39

Figure 4.16 Product - Customization Relationship......................................................39

Figure 4.17 ER diagram...............................................................................................40

Figure 4.18 Schema diagram.......................................................................................41

Figure 4.19 Level 0 DFD.............................................................................................42

Figure 4.20 Level 1 DFD.............................................................................................42

Figure 4.21 Level 2 DFD.............................................................................................44

Figure 4.22 Use Case Diagram.....................................................................................46

Figure 4.23 Sequence Diagrams for customer registration..........................................52

Figure 4.24 Sequence Diagrams for customer Login...................................................52

Figure 4.25 Sequence Diagrams for admin Login........................................................53

Figure 4.26 Sequence Diagrams for search product......................................................53

Figure 4.27 Sequence Diagrams for chatbot..................................................................54

Figure 4.28 Sequence Diagrams for Cart.......................................................................54

Figure 4.29 Sequence Diagrams for Order......................................................................55

Figure 4.30 Sequence Diagrams for Payment.................................................................55

Figure 4.31 Sequence Diagrams for Manage Product....................................................56

Figure 4.32 Sequence Diagrams for Review...................................................................56

Figure 4.33 Sequence Diagrams for Customization........................................................57

Figure 4.34 Activity diagram for admin.........................................................................59

Figure 4.35 Activity diagram for customer......................................................................59

Figure 4.36 State Diagram for customer.........................................................................60

Figure 4.37 User Interface Design Registration................................................ ..........61

Figure 4.38 User Interface Design Login........................................................... ..........61

Figure 4.39 User Interface Design home........................................................... ..........62

Figure 4.40 User Interface Design Category............................................................. ......62

Figure 4.41 User Interface Design Order.................................................................. ..........63

Figure 4.42 User Interface Design cart.................................................................... ..........63

Figure 4.43 User Interface Design Payment...........................................................64

Figure 4.44 User Interface Design chatbot..............................................................64

Figure 4.45 User Interface Design Customisation...................................................65

**THE APPROVAL PROJECT PROPOSAL**

|  |  |
| --- | --- |
| 1. **Name of the Student** | |
| Vandana Ramkrishna Padhi | |
| 2. **Title of the Project** | |
| Online Jewellery System | |
| 3. **Name of the Guide** | |
| Mrs. Rakhee Rane | |
| 4. **Is this your first submission?** | |
| Yes | |
| **Student Control ID:** | 2021080316 |
| **Student Roll No.:** | 4045A028 |

**SYNOPSIS**

**TITLE**:

Ethnic jewellery system.

**STATE ABOUT THE PROBLEM:**

Customer will be able to examine jewellery and accessories of various variety for individuals of various age groups and gender as well as varied cost materials and metals while also being able to see other customers review and ratings. Additionally, the customer can place order for customised jewellery as well.

**WHY THIS TOPIC:**

The goal is to give a client a platform where they can shop online utilising the internet, browse a large section of jewellery and then purchase it while sitting at home. Additionally, people can purchase personalised and customise their desired jewellery while relaxing at home. The customer would find it more convenient to place such orders. Thus, jewellery website computerisation will improve the convenience of efficiency while also lowering the possibilities of human errors. In addition, this will help the seller to increase and expanding its market and customer.

**OBJECTIVE AND SCOPE:**

1. Online jewellery shop.
2. Variety of jewellery available also different types of jewelleries.
3. Viewing the current trending jewellery.
4. Search the availability of product according to occasion.
5. Can get the desired jewellery by customising them.
6. Can have variety of materials.
7. Can be gifted.
8. Help the seller to expand the market all over.
9. Also consist a chat boat which would help the user for querries and other information.

**SCOPE:**

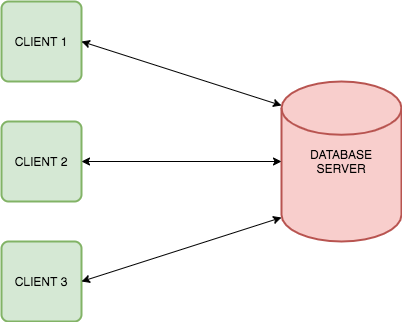
The scope of the website could be two provide a platform for buying and selling jewellery with in Maharashtra.

**METHODOLOGY:**

To develop the website, I am going to use Iterative model.

**PROPOSED ARCHITECTURE:**

To create the system, we will use centralised and Two-Tier Architecture.



**Figure 0.1 Architecture**

**REQUIREMENTS:**

**Software requirements:**

1. Front-end: HTML5, Tailwind CSS, JavaScript, PHP, JQuery.
2. Back-end: SQL.
3. OS: Windows 8.

**Hardware requirements:**

1. RAM: 1GB or more
2. Processor: Intel core 2.0 GHz Monitor: 17 CRT or LCD
3. Hard disk: 250 GB and keyboard, mouse.

**Platform Requirements:**

Visual Studio Code.

**CONTRIBUTION:**

The website will assist user in purchasing jewellery from a white range of possibilities. Through their device, the user makes quickly order any personalized jewellery in a comfortable way. As every transaction is online, the jewellery will be authentic and genuine. This will serve as a platform for both the seller and the customer. Additionally, it will save the time and the efforts required to physically visit various shops. Both the buyer and seller will find it more convenient.

**CONCLUSION:**

In conclusion, developing a website for a jewellery system might be use full for marketing as well as selling jewellery items online. Customer can easily get access to view the products at a greatest possible price. The buyer as well as the seller would receive a platform for buying and selling as well as expanding their market.

**CHAPTER 1**

**INTRODUCTION**

**1.1 BACKGROUND**

From ancient times jewels have been part of human culture and life. Most businesses have now gone online by other internet and the world wide web and as a result of this trend there are now luminous website that are selling jewellery on internet. Many well-known websites offer jewellery including the tribe Amrapali Zaveri pearls, Voula, the Luxor, Sukhu, Peoria, Mir raw, You Bella and others. The website allows buyers to purchase jewellery from the comfort of their own house 24/7 and they may also contact us for the customised jewellery. The primary purpose of the website is to offer jewellery to consumers at the leisure and convenience.

**1.2 OBJECTIVES**

The objective are as follows-

1. An online jewelleries system provides customer a simple and smooth shopping experience, allowing them to explore products and add things to the card and pay with minimum efforts and easily.
2. The website will provide a wide range of jewellery such as earrings, bracelet, anklets, earrings etc in a variety of designs and material at various prices.
3. Customer may customise their jewellery items by selecting alternative materials, colour, sizes and styles resulting in more customised stopping experience with the help of customization option.
4. The website will provide a clear and details photograph of each product as well as full description that include material information, review and pricings.
5. Customers review and ratings will be provided for each product on the website allowing consumers to see the comments from the other customers.
6. The website will offer a safe and a trustworthy payment system, protecting the customer sensitive information and offering a variety of online payment choice.
7. The website won’t be providing exchange policy allowing the customer to replace things that are broken or damage during the delivery process.
8. The website will provide a friendly customer support and service instantly providing customer request or technical difficulties and responding to complaints.
9. The website helps the seller for expanding its market and buyer receives wide range of option while buying that jewellery.

**1.3 PURPOSE, SCOPE AND APPLICABILITY**

**1.3.1 Purpose**

The website will enable the customer in exploring jewelleries of all styles occasion, brands for various people for multiple age group and gender of various prices, materials, metals etc and doing online shopping. The purpose is to provide the customer with a platform where they can shop digitally via the internet, view a large range of jewellery and then purchase while relaxing at home. Also, the seller will benefit it from expanding of its market and customers.

**1.3.2 Scope**

The website scope is to provide a platform for ordering the jewellery within Maharashtra. Combining different kinds of jewellery to one location enables the buyer to simply purchase different kinds of jewellery from their house. It has various limitation for users such as notification although these can be modified in the future.

**1.3.3 Applicability**

The main goal of the system is to give us and easy to understand interface that makes it simple to use anyone from India will be able to access the system. Increasing the e business and inspiring more individuals to adopt the e-business and the daily businesses.

**CHAPTER 2**

**SURVEY OF TECHNOLOGIES**

**Front-end technology's:**

* HTML
* CSS
* JavaScript
* Bootstrap
* jQuery
* Asp.net
* Python
* Types script
* Java
* React
* Angular
* Word press
* Tailwind CSS

**HTML, CSS, JavaScript**

HTML is a marker language. It provides the structure of a website so that the web browser knows what to show. HTML is a text-based approach to describing how content contained within an HTML. It shows how to display text images and other forms to display text images and other forms of multimedia on web page. CSS is a cascading style sheet. CSS lets the web designer change colour font, animation and transition on the web. They make the web look good. It helps to make web pages more attractive. Java script is a programming language for the web script can be calculated manipulated and validity the data. It is a dynamic computer programming language. To interact with the user and make dynamic pages.

**Bootstrap**

Bootstrap is a giant collection of handy reusable bits of code written in HTML CSS and JavaScript. It is also a front and development framework that enables the developer and designer to quickly build fully responsive website. Bootstrap includes user interface components layout and JS tools along with the framework of implementation.

**Tailwind CSS**

Tailwind CSS is a utility-first CSS framework that streamlines web development by providing a comprehensive set of pre-designed, low-level utility classes. Developers can efficiently build user interfaces by applying these classes directly in the HTML, enabling quick and flexible styling without adhering to a predefined design. Tailwind promotes readability, maintainability, and a rapid development workflow.

**jQuery**

jQuery is one of the earliest front and framework and despite its launch date it continuous to be relevant in today's tech World. This framework offers are up to use and simplicity along with minimising the used to write the extensive JavaScript codes. There are also an extensive jQuery community that developers can rely on for solution.

**ASP.NET**

It is a web development platform, which provides a programming model and a comprehensive software infrastructure and various services required to build robot web application for PC as well as mobile devices. Asp.net applications are compiled codes written using the extensible and reusable components or objects present in .net framework.

**Python**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. It is a high-level built-in data structures combined with the dynamic typing and dynamic binding makes it very attractive for rapid application development as well as for use as a script language to connect existing components together. Python is simple easy to learn syntax emphasis readability or there for reduces the cost of programming maintenance.

**Typescript**

It is a tight superset of JavaScript that adds optional static typing to the language. It makes it easier to develop large scale JavaScript application by catching errors at compile time rather than run time. Type scripts allow specifying the type of data being passed around within the code and lies the ability to report errors where the types don't match.

**Java**

Java is both a programming language and a platform. Java is a high-level object oriented and secure programming language. Application programming interface API and Java runtime environment JRE hence it is a most suitable and fully functional platform to develop an application. Java is used to create stand-alone application, web application, enterprise application and mobile application.

**React**

React is a UI development library written in JavaScript. Facebook and open source developer communities run it. Although react is a library and not a language it uses in web development is rampant. React goes beyond simply and has many extensive life flux and react native for complete application architecture support. It is used to build interactive user interfaces web applications quickly and efficiently with significantly less code than you would write with vanilla JavaScript.

**Angular**

Angular is an open-source JavaScript framework based on type script. Google run set and one of its main goals is to build single page apps. As a framework angular offers apparent advantages and it also provides a standard structure for developers in a team to work with. It allows users to create use applications that are easy to manage.

**WordPress**

WordPress uses four languages: HTML CSS JavaScript and PHP. The first three are executed in the browser while PHP works on the web server to generate and serve the HTML CSS and JavaScript the browser uses. Word press is a content management system (CMS) is that allows you to host and build websites.

**Back-end languages:**

* PHP
* My SQL
* Ruby
* C++
* Rust
* C#
* Go

**PHP**

PHP recruited acronym for PHP is hypertext preprocessor is widely used open-source general purpose scripting language that is especially suited for the web development and can be embedded into HTML. PHP is in server-side scripting language that is used to develop static websites or dynamic websites or web applications. PHP script can only be interpreted on a server that has PHP installed.

**My SQL**

My SQL is a relational database management system RDBMS developed by Oracle that is based on structural query language SQL. My SQL is a fast, easy to use, RDBMS being used for many small and big businesses. My SQL provides an implementation of SQL database very well suited for small to medium web pages. Database is just a structure collection of data that is organised for easy use and retrieval.

**Ruby**

Ruby is a dynamic object-oriented programming language known for its simplicity and development features. Ruby offers a standard library that provides a wide range of built in functionalities such as string manipulation i/o and network programming. Ruby is a cross platform compatible running on major operating systems such as windows, Mac OS and Linux.

**C++**

While primarily used for system level programming, C++ can also be used for web development. It offers high performance and low-level context but requires more memory management compared to higher level languages.

**Rust**

Rust is a system programming language that offers high performance memory safe and concurrency. While primarily used for system level programming rest is increasingly being adapted for web development due to its security features and performance advantages.

**C#**

C# is a language by Microsoft and is often used with the .net frameworks for web development. Asp.net is a popular framework for building web application using C# .

**Go**

Go is also known as Golang, is a language by Google it is known for its simplicity concurrency features and efficiency. Go is often used for building high performance web applications.

**WHY THESE LANGUAGES?**

To provide a user-friendly front end I would use HTML, CSS, jQuery, Tailwind CSS. Tailwind CSS because it rapidly build user interfaces with a utility-first approach. Simple, flexible, and highly customizable, Tailwind encourages efficient development through intuitive styling directly in the HTML. My SQL and PHP are the bacon languages used for developing the system as they are easy and simple to learn. Adding and deleting and manipulation of data is simple.

**CHAPTER 3**

**REQUIREMENT AND ANALYSIS**

**3.1 PROBLEM DEFINITION**

The system is built to make online shopping of jewelleries easy for the customer so that they can buy jewellery online just like how they do other shopping online. The user will be able to browse different types of jewelleries and request for customised items. The website will provide a range of jewelleries such as earrings, bracelet, anklets, rings and necklaces. The system would also provide a chat bought for enquiry or request for customising items else they can also contact us through phone number or email id.

**3.1.1 Sub-System**

* **Login and register**
  + Users will be able to create a new account that is registered themselves.
* **Chatbot**
  + Would answer any query related to customisation or any product to the user.
* **Shopping Cart**
  + It allows you to add products to their card and review the cart content.
* **Customer support**
  + It would provide help regarding any issue. It includes features such as a chat, email support and contact number.
* **Payment**
  + It handles the online payment.

**3.2 REQUIREMENTS SPECIFICATIONS**

**3.2.1 Requirements gathering**

**3.2.1.1 Ways of requirement gathering**

Various requirement gathering technology include

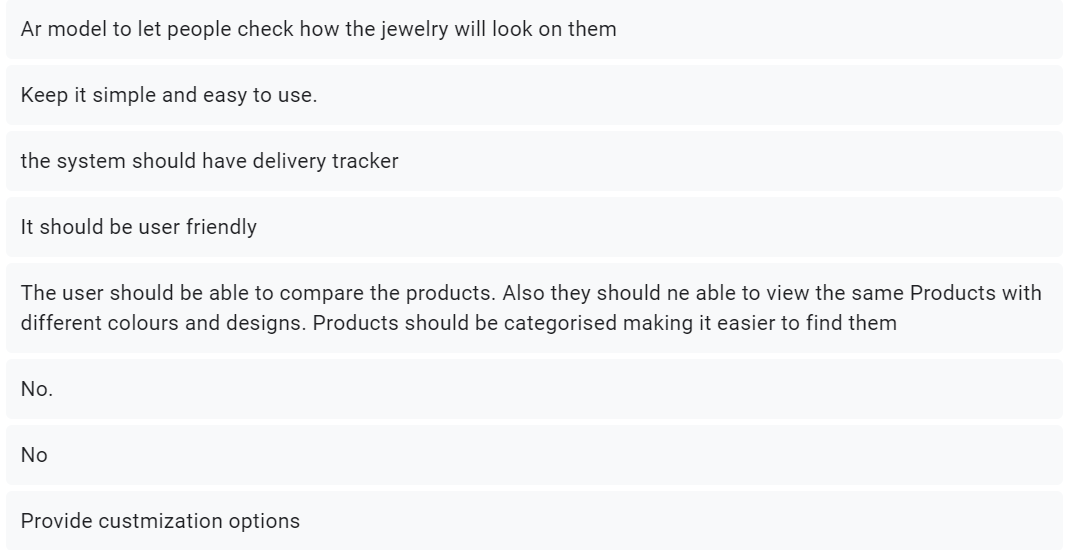
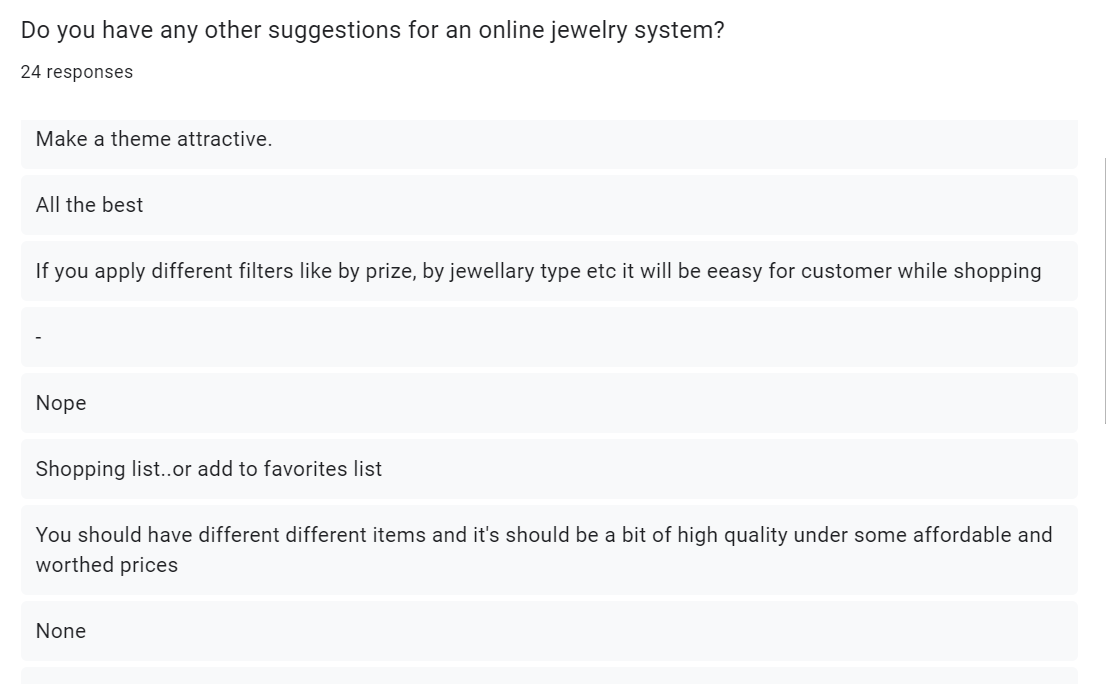
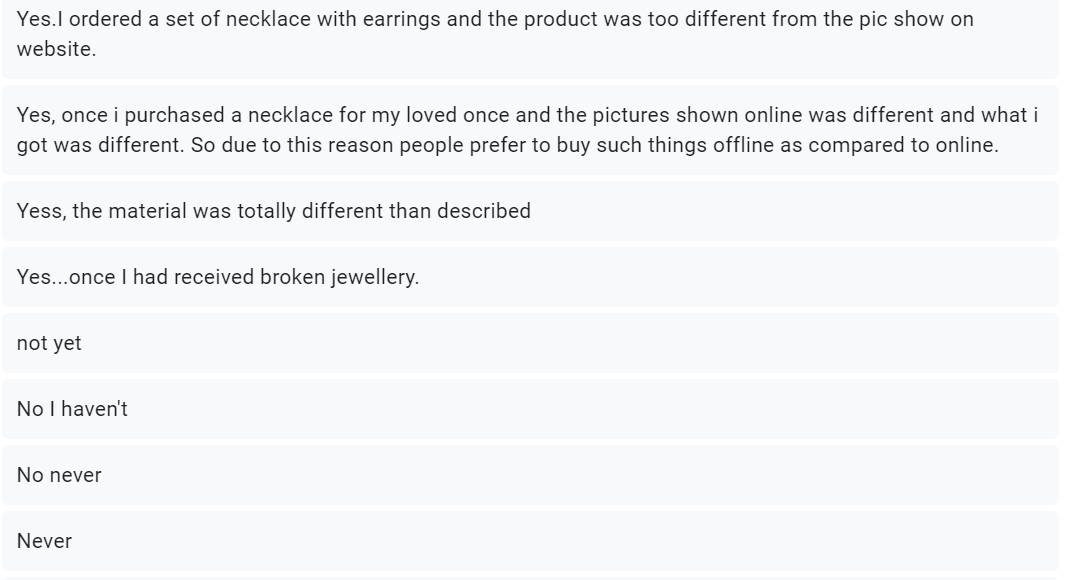
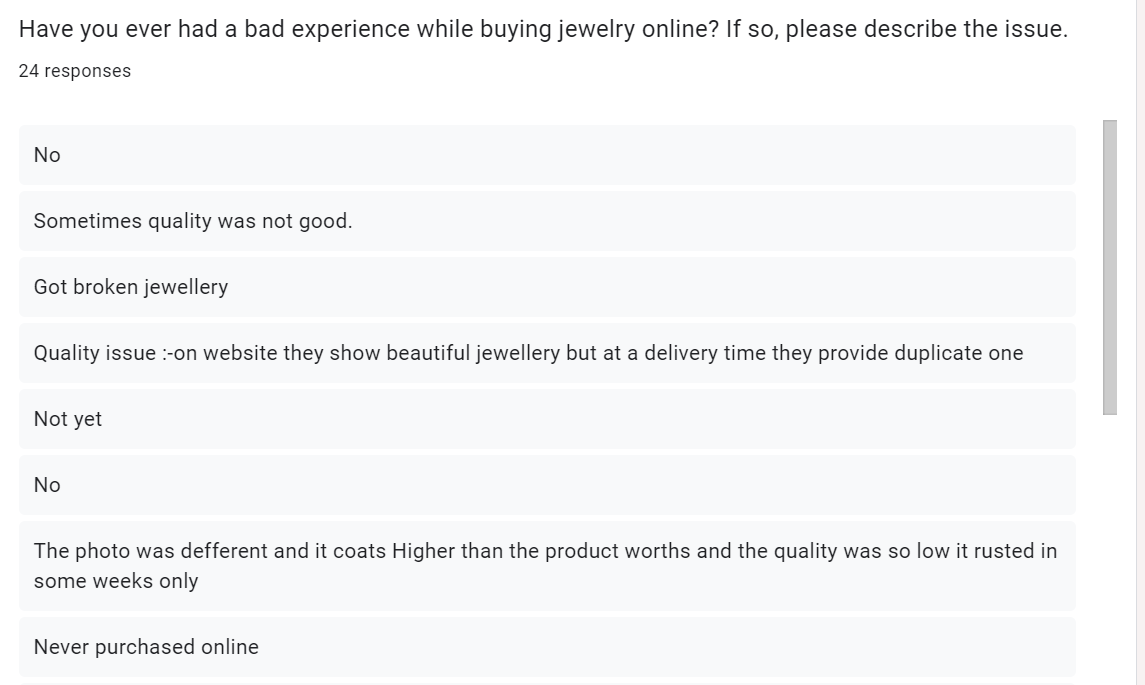
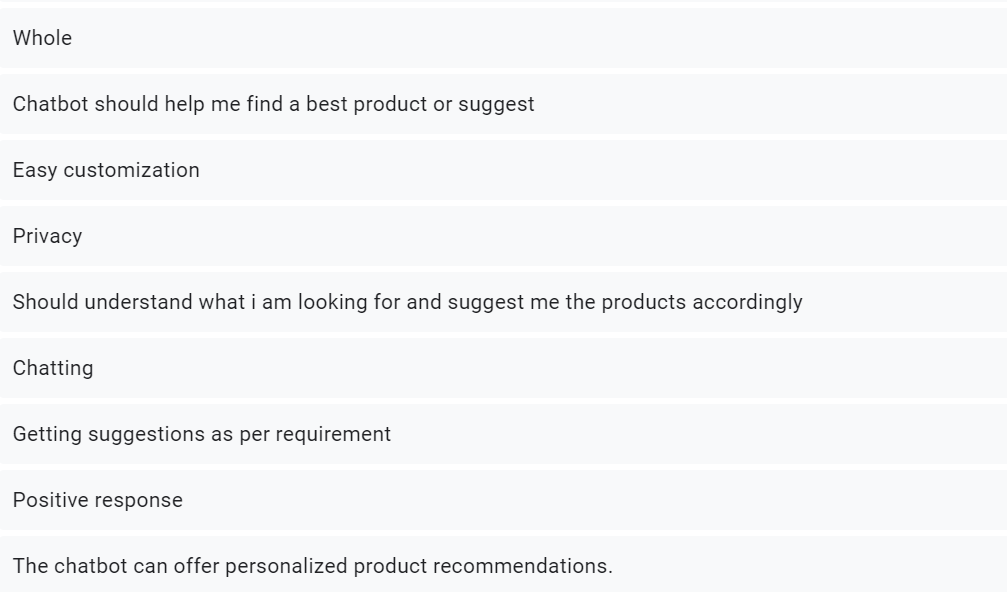
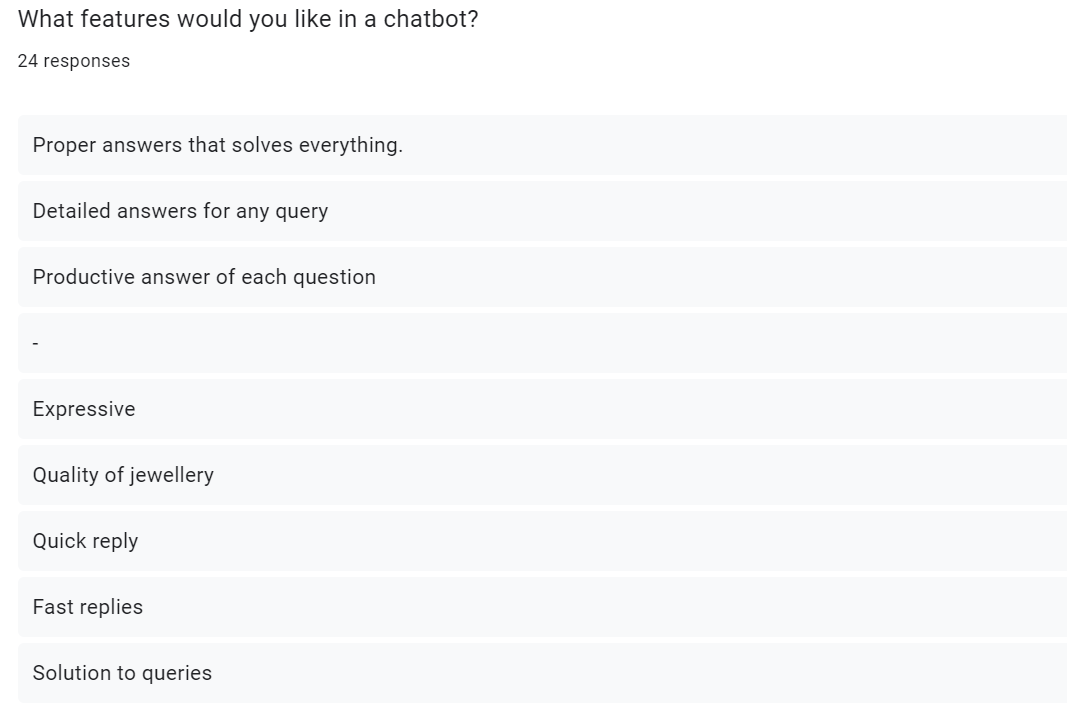
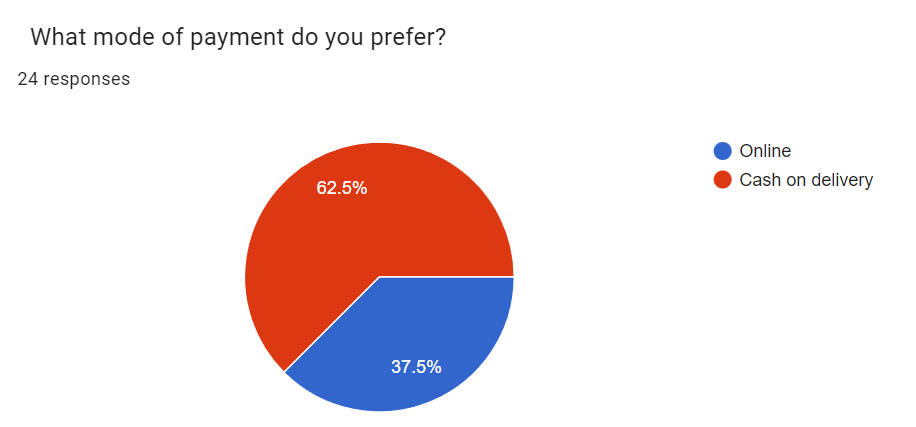
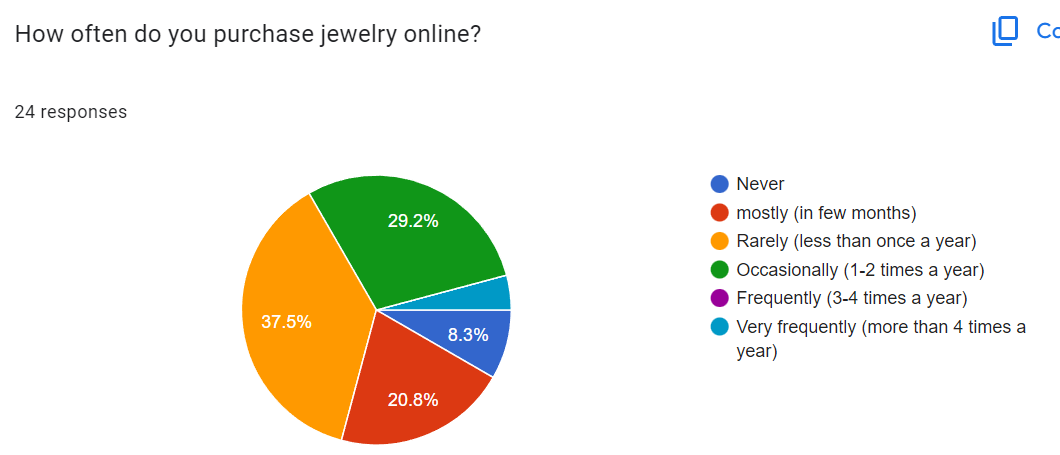
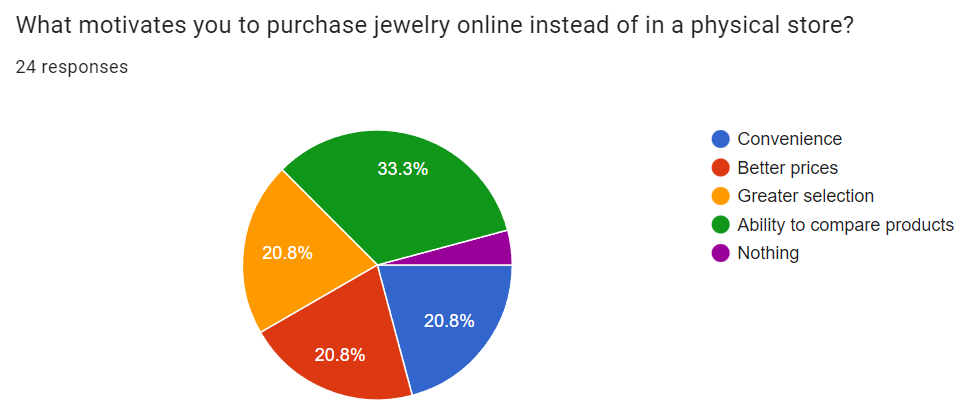
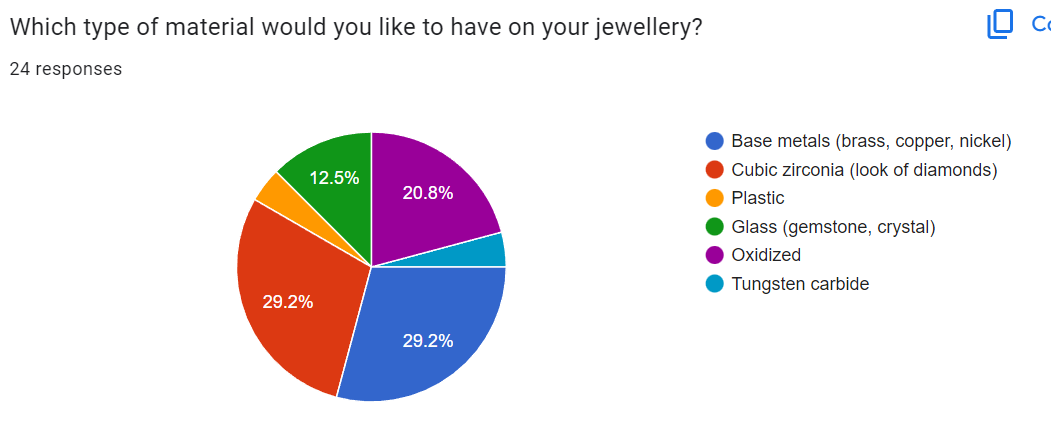
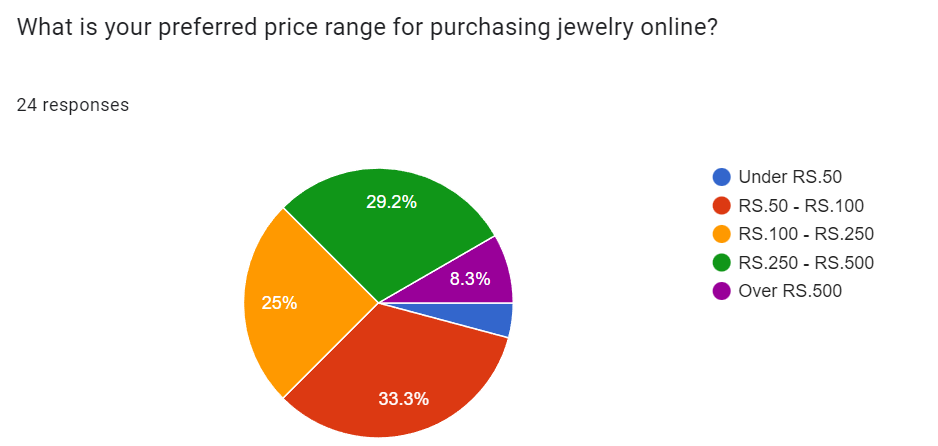
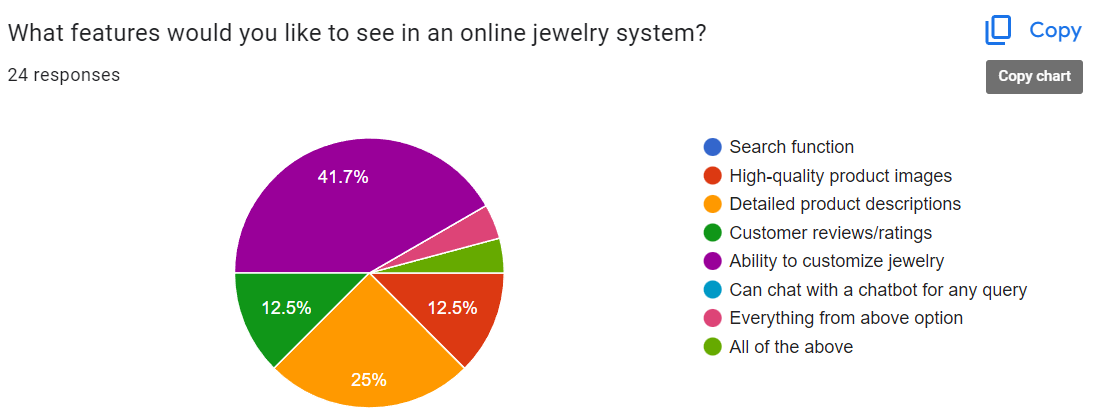
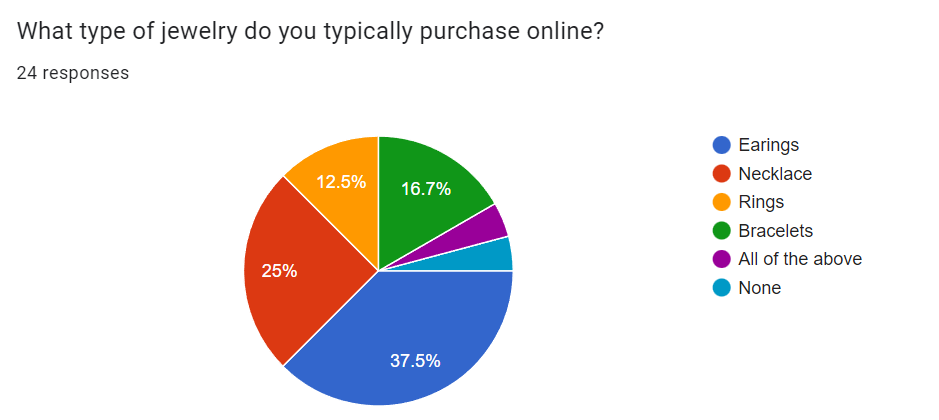
* Interview- Conducting interviews with stakeholders such as clients, users and subject matter experts to gather their input and understand the requirements. These interviews can be conducted one to one in a group setting.
* Surveys and questions- Distributing surveys and questionnaires to the street holders together their feedback preference is and requirements. This can be done through online forms or printed questionnaires.
* Workshop and Focus group- Organising workshop or Focus group with stay holders to facilitate collaborative discussion and brains streaming session. This session can help in identifying and private rising requirements while promoting active participation and concerns building. Observation-Actively observing users and stakeholders in their environment to understand their work flows points and requirement. This can provide valuable insight into the current processes and potential improvements.
* Document analysis-reviewing existing document such as business documents user manuals or technical specification to extract relevant requirements and understand the context of the project.
* Prototyping and mock-ups-creating proto types or mock-ups of the system other interface and gathering feedback from stakeholders. This visual representation can help electric requirements and validate designed decisions.
* Us Case Analysis- Developing detailed use case or user’s stories that captures specific interaction and requirement of the system. This technique helps in defining the system functionality from the uses perspective.
* Benchmarking- Researching and analysing similar systems or competitors offering to identify potential requirements and best practice that can be incorporated into the project.
* Requirement workshops- Organising dedicated workshops focus only on requirement gathering. This involves bringing together stay holder and requirement analyst to collaboratively define and priority is requirement.
* Prototype and feedback- Creating early sessions and versions of the system or the component and gathering feedback from stakeholders through typing. This allows for early validation and refinement of requirements based on user’s feedback.

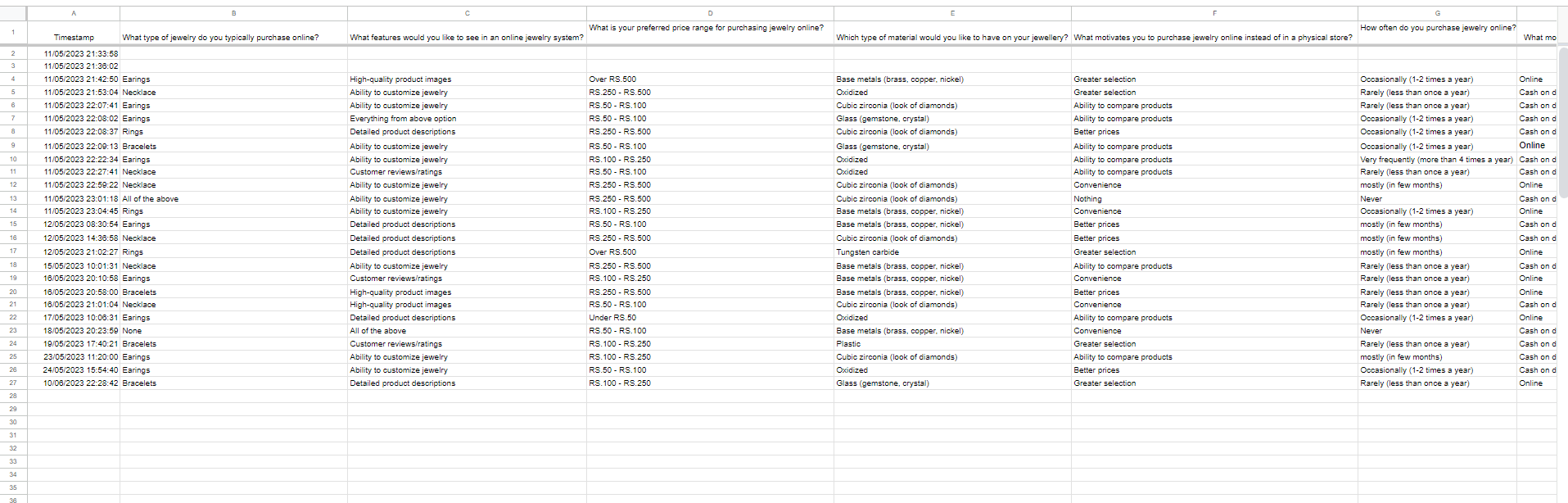
For the current situation I used survey and questionnaires methods to identify the requirements for the project using Google forms as the mains to collect the data. The link for the Google Form is given below – <https://forms.gle/rnhzUadPcFDq3PL97>

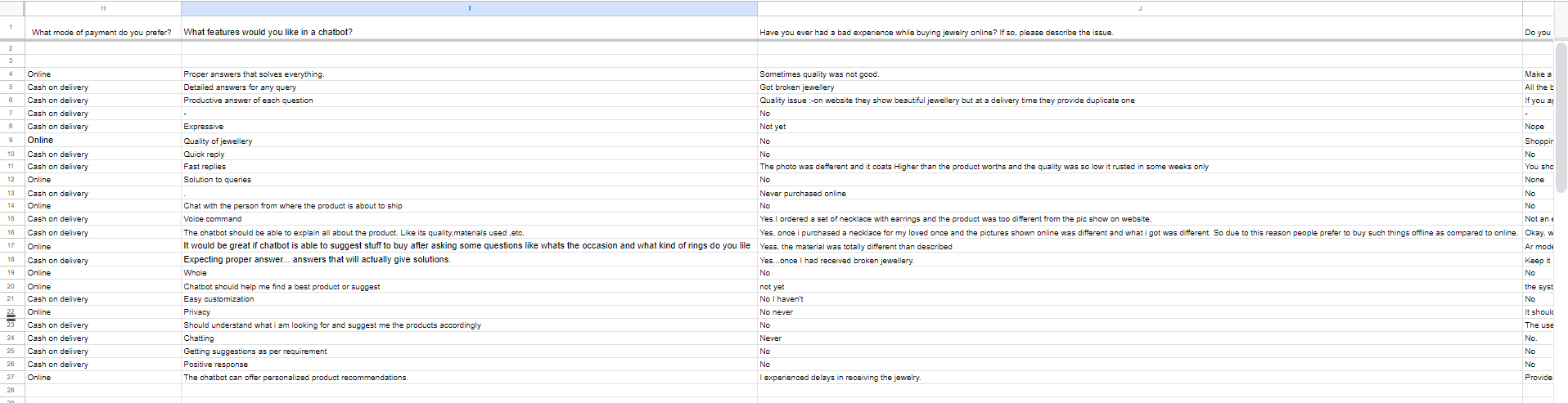
The quotients asked in this survey are as follows: -

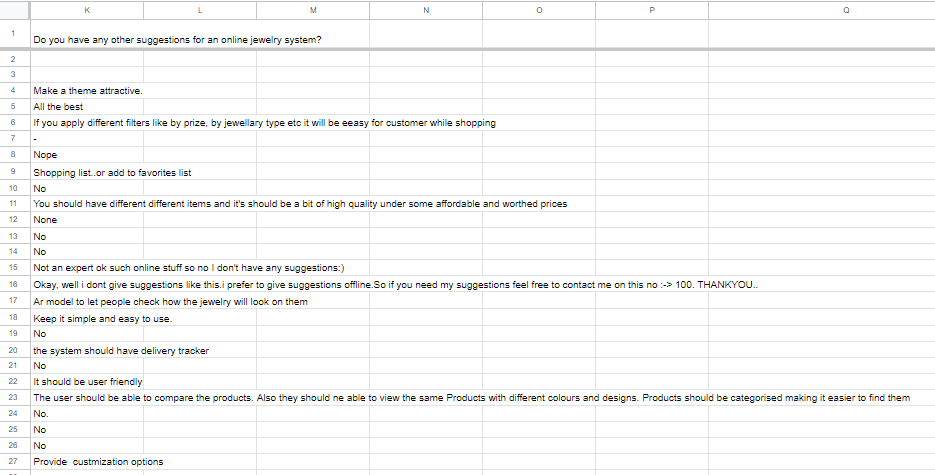
1. What type of jewellery do you typically purchase online?
2. What features would you like to see in an online jewellery system?
3. What is your preferred price range for purchasing jewellery online?
4. Which type of material would you like to have on your jewellery?
5. What motivates you to purchase jewellery online insure of physical stores?
6. How often do you purchase jewellery online?
7. What mode of payment do you prefer ?
8. What features would you like in a chatbot?
9. How have you been ever having a bad experience while buying the jewellery online? If so, please describe the issue.
10. Do you have any other suggestion for the online system?

The Responses are as follows: -









**Figure 3.1 Responses Through Google Forms**

**3.2.2 Requirements Analysis**

**3.2.2.1 Functional Requirements**

* Enables the user to register to the system by filling the details and creating a password.
* Authenticate and allows the user to login.
* Users can login into the accounts by using the user id and password.
* Users can search the desired product if available.
* Users can go through the product and add his desired product to the cart.
* They can edit those choices later if they want.
* User and products according to price range.
* Users would be able to chat with the chat boat to solve any queries.
* The system would provide detailed information for each type of jewellery.
* The system would have separate sections for rings earrings necklace anklets etc will provide reviews for each product.
* The selected product would be added to the card and the total price will be calculated.
* The system will enable the user to pay for the order through online mode.
* The user can also submit feedback which contains the feedback from the user.
* The system will provide a help customer service to the user to convey any problem if they have.

**3.2.2.2 Non-functional requirements**

* Security

The system should be able to verify the user based on the username and password. If the login credentials match with the database the user will be logged into the system. Thus, only the registered users can browse the website to place an order.

* Usability

Usability means how can the user easily use the website. It will be providing is in navigation bar and also a chart bought feature to solve any query.

**3.2.2.3 System requirements**

**1. Registration**

The user must first register its account in order to use the site without registering the user will be not able to use the website but they can't order any product

* Input: name contact, password
* Output: registered successfully
* Source: user
* Destination: enter data is stored in database
* Action: after registration and account is created for user
* Pre-condition: User must provide details such as name and contact
* Post condition: users will be able to login with the appropriate credentials

**2. Login**

After registering the user will be able to login to the system by the username and password and after verification the user will be logged in to the system.

* Input: username and password.
* Output: successfully login.
* Source: user
* Destination: enter data is stored in database
* Action: After entering the login ID and password system will check whether it matches or not. If not, the customer has to login again and if it matches the customer will be directed to the home page.
* Pre-condition: user be registered.
* Post condition: user can browse the website and order the product.

**3. Search**

The user can simply type the name of the product inside the search bar and search the product.

* Input: name of the product.
* Output: If the product is available the result is shown.
* Source: user
* Destination: system will fetch the details from the database.
* Action: System will fetch the product details from the database for the search product entered by the user and display on the result in the web page.
* Pre-condition: product name must be entered in the search bar.
* Post condition: customer will get the desired product if available.

**4. Category**

Different products will be placed in different categories to which it belongs so it will be easy to find a certain type of product.

* Input: search category option
* Output: show the category items
* Source: user
* Destination: None
* Action: clicking the category options different types of product category will get open and the user can select the type of product they want
* Pre-condition: Access valid category
* Post condition: All items in the category are shown.

**5. Filter**

While browsing and searching the product the user can applied different filters they want like price, range, brand, metal, gems etc which would be helpful to the user to select the product as per the choice

* Input: apply filter
* Output: product will be displayed according to the filter
* Source: user
* Destination: none
* Action: buy applying the filter the product will be sort according to the applied filter by the user
* Pre-condition: user must apply the valid filter
* Post condition: filters will get applied and the product would be displayed

**6. Add to cart**

The customer will add the product which they want to buy an art to cart

* Input: select the product they want
* Output: product will be added to cart
* Source: user
* Destination: None
* Action: whatever the product the customer desires to buy can add the product to add to cart
* Pre-condition: the product must be available
* Post condition: they can proceed for confirming the order and proceeding it to payment

**7. Remove from cart**

User can select the product and remove from cart if they don't want it to buy

* Input: select the product that should be removed from the cart
* Output: product selected will be removed from the cart
* Source: user
* Destination: None
* Action: any product user wants to remove they can remove it from the cart
* Pre-condition: product must be present in the cart
* Post condition: product will be removed from the cart

**8. Transaction**

The payment would be done by the user for the order

* Input: Amount
* Output: product purchased
* Source: user
* Destination: enter data is stored in database
* Action: after proceeding to payment the user will pay the amount displayed through online mode
* Pre-condition: user must proceed to the payment option after selecting the product in cart
* Post condition: if payment field the order is cancelled

**9. Order**

After adding the desired product to the cart, the user can proceed for order

* Input: add products to cart to order it
* Output: the order is placed
* Source: user
* Destination: order request will get store in the database.
* Action: when you proceed for order it will ask for an address and contact with a product should be delivered
* Pre-condition: The product must be in cart
* Post condition: order will be placed

**10. Delivery**

After the order and payment is done the product will be procedure for delivery

* Input: place an order
* Output: the product gets delivered
* Source:
* Destination: delivery record has been stored in the database
* Action: once the order is confirmed and the payment is done the product goes for delivery.
* Pre-condition: payment should be done
* Post condition: product is delivered

**11. Review and Ratings**

The user who has already purchase the product can give the review and rating for the products

* Input: reviews and rating for reviews they can write a short comment and for ratings we can give stars out of five
* Output: reviews and rating uploaded
* Source: user
* Destination: it will store in the database
* Action: after the product is purchased, they can comment down the feedback of the product in the reviews section and rate the product
* Pre-condition: product must be purchased by the user
* Post condition: review and ratings will be uploaded

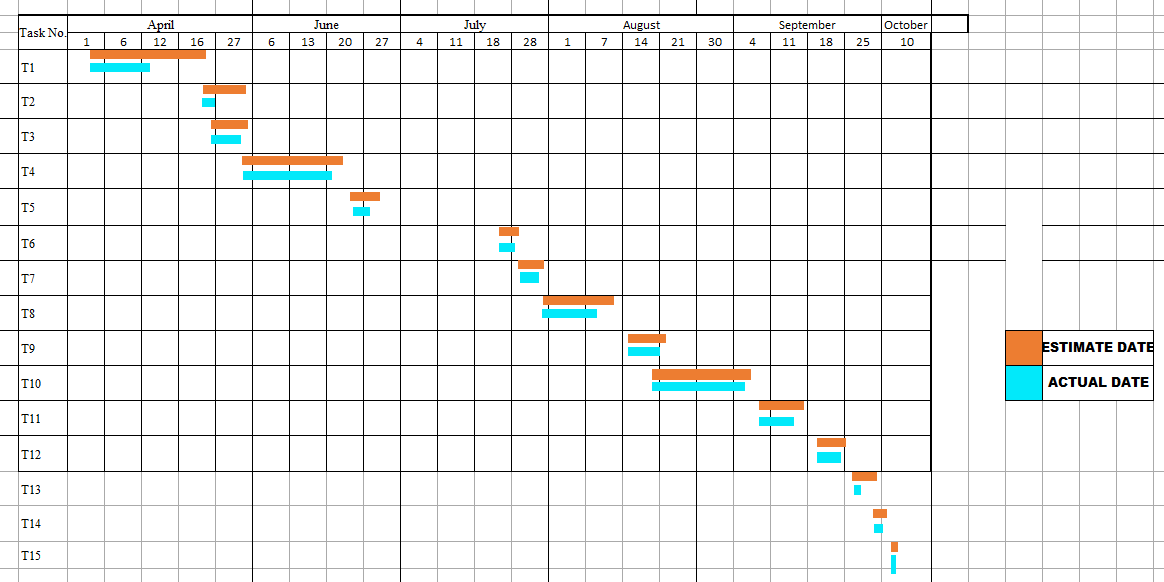
**12. Contact us / help services**

If the user has any problem regarding to the product, they can contact with the help service provided.

* Input: comment down the problem in the help section
* Output: the problem will be uploaded successfully
* Source: user
* Destination: system will fetch the details from the database.
* Action: after problem is uploaded in the help service then the solution will be provided or inform to the user
* Pre-condition: user must have any query or problem related to the product
* Post condition: solution for the problem will be provided
  1. **Planning and Scheduling**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task No.** | **Task Name** | **Actual Start Date** | **Actual End Date** | **Start Date** | **End Date** |
| T1 | Project Synopsis | 02/04/23 | 11/04/23 | 02/04/23 | 17/04/23 |
| T2 | Chapter-1: Introduction | 17/04/23 | 19/04/23 | 17/04/23 | 20/04/23 |
| T3 | Chapter-2: Survey of Technology | 20/04/23 | 26/04/23 | 20/04/23 | 30/04/23 |
| T4 | Chapter-3: Requirement Analysis  3.1 Problem Definition  3.2 Requirement Specification | 30/04/23 | 19/06/23 | 30/04/23 | 20/06/23 |
| T5 | Chapter-4: System Design  4.1 Business Rules  4.2 Module Diagram | 12/08/23 | 14/08/23 | 12/08/23 | 18/08/23 |
| T6 | Chapter-4: System Design  4.3.1 Entity Sets | 19/07/23 | 21/07/2023 | 19/07/23 | 22/07/23 |
| T7 | 4.3 ER Diagram  4.4 Schema Diagram | 27/07/23 | 1/08/23 | 27/07/23 | 08/08/23 |
| T8 | 4.5 Data Flow Diagram | 18/08/23 | 29/08/23 | 18/08/23 | 01/09/23 |
| T9 | 4.6 Use Case Diagram | 04/09/23 | 06/09/23 | 04/08/23 | 08/09/23 |
| T10 | 4.7 Scenarios | 09/09/23 | 11/09/23 | 09/09/23 | 14/09/23 |
| T11 | 4.8 Sequence Diagram | 15/09/23 | 16/09/23 | 15/09/23 | 18/09/23 |
| T12 | 4.9 Activity Diagram  4.11 User Interface Diagram | 18/09/23 | 03/10/23 | 03/10/23 | 03/10/23 |
| T14 | 4.10 State Diagram  4.12 Test Cases | 03/10/23 | 04/10/23 | 03/10/23 | 06/10/23 |
| T15 | **Re-engineering**  Chapter 1 | 02/11/23 | 03/11/23 | 01/11/23 | 10/11/23 |
| T16 | Chapter 2 | 04/11/23 | 05/11/23 | 01/11/23 | 10/11/23 |
| T17 | Chapter 3 | 06/11/23 | 07/11/23 | 01/11/23 | 10/11/23 |
| T18 | Chapter 4 | 08/11/23 | 09/11/23 | 01/11/23 | 10/11/23 |
| Iteration 1 | | | | | |
| Module 1: Landing Page | | | | | |
| T19 | Coding and Testing | 17/11/23 | 20/11/23 | 16/11/23 | 19/11/23 |
| Module 2: Registration Page | | | | | |
| T20 | Coding and Testing& Debugging |  |  | 20/11/23 | 22/11/23 |
| T21 | Integrating Module 1 and testing |  |  | 23/11/23 | 24/11/23 |
| Module 3: Login Page | | | | | |
| T22 | Coding and Testing& Debugging |  |  | 25/11/23 | 27/11/23 |
| T23 | Integrating Module 2 and testing |  |  | 28/11/23 | 29/11/23 |
| Iteration 2 | | | | | |
| Module 4: Customer Home Page | | | | | |
| T24 | Coding and Testing& Debugging |  |  | 01/12/23 | 03/12/23 |
| T25 | Integrating Module 3 and testing |  |  | 03/12/23 | 04/12/23 |
| Module 5: Admin Home Page | | | | | |
| T26 | Coding and Testing& Debugging |  |  | 05/12/23 | 07/12/23 |
| T27 | Integrating Module 4 and testing |  |  | 07/12/23 | 08/12/23 |
| Iteration 3 | | | | | |
| Module 6: Browse Product | | | | | |
| T29 | Coding and Testing& Debugging |  |  | 08/12/23 | 10/12/23 |
| T30 | Integrating Module 5 and testing |  |  | 10/12/23 | 11/12/23 |
| Module 7: Browse Categories | | | | | |
| T31 | Coding and Testing& Debugging |  |  | 12/12/23 | 14/12/23 |
| T32 | Integrating Module 6 and testing |  |  | 14/12/23 | 15/12/23 |
| Module 8: View Product Details | | | | | |
| T33 | Coding and Testing& Debugging |  |  | 16/12/23 | 18/12/23 |
| T34 | Integrating Module 7 and testing |  |  | 19/12/23 | 20/12/23 |
| Module 9: Add Product to Cart | | | | | |
| T35 | Coding and Testing& Debugging |  |  | 21/12/23 | 23/12/23 |
| T36 | Integrating Module 8 and testing |  |  | 23/12/23 | 24/12/23 |
| Module 10: Remove Product from Cart | | | | | |
| T37 | Coding and Testing& Debugging |  |  | 25/12/23 | 27/12/23 |
| T38 | Integrating Module 9 and testing |  |  | 27/12/23 | 28/12/23 |
| Module 11: Update Product From Cart | | | | | |
| T39 | Coding and Testing& Debugging |  |  | 29/12/23 | 31/12/23 |
| T40 | Integrating Module 10 and testing |  |  | 31/12/23 | 01/01/24 |
| Iteration 4 | | | | | |
| Module 12: Product Customization | | | | | |
| T41 | Coding and Testing& Debugging |  |  | 02/01/24 | 04/01/24 |
| T42 | Integrating Module 11 and testing |  |  | 04/01/24 | 05/01/24 |
| Iteration 5 | | | | | |
| Module 13: Order Product | | | | | |
| T43 | Coding and Testing& Debugging |  |  | 06/01/24 | 08/01/24 |
| T44 | Integrating Module 12 and testing |  |  | 09/01/24 | 10/01/24 |
| Module 14: Product Payment | | | | | |
| T45 | Coding and Testing& Debugging |  |  | 11/01/24 | 13/01/24 |
| T46 | Integrating Module 13 and testing |  |  | 13/01/24 | 14/01/24 |
| Iteration 6 | | | | | |
| Module 15: Give Review and Rating | | | | | |
| T47 | Coding and Testing& Debugging |  |  | 14/01/24 | 16/01/24 |
| T48 | Integrating Module 14 and testing |  |  | 17/01/24 | 18/01/24 |
| Module 16: Contact Us | | | | | |
| T49 | Coding and Testing& Debugging |  |  | 19/01/24 | 21/01/24 |
| T50 | Integrating Module 15 and testing |  |  | 21/01/24 | 22/01/24 |
| Iteration 7 | | | | | |
| Module 17: Chatbot | | | | | |
| T51 | Coding and Testing& Debugging |  |  | 23/01/24 | 25/01/24 |
| T52 | Integrating Module 16 and testing |  |  | 25/01/24 | 30/01/24 |
| Iteration 8 | | | | | |
| T53 | Final Testing |  |  | 31/01/24 | 02/02/24 |

**Table 3.1 Task table**



**Figure 3.2 Gantt Chart**

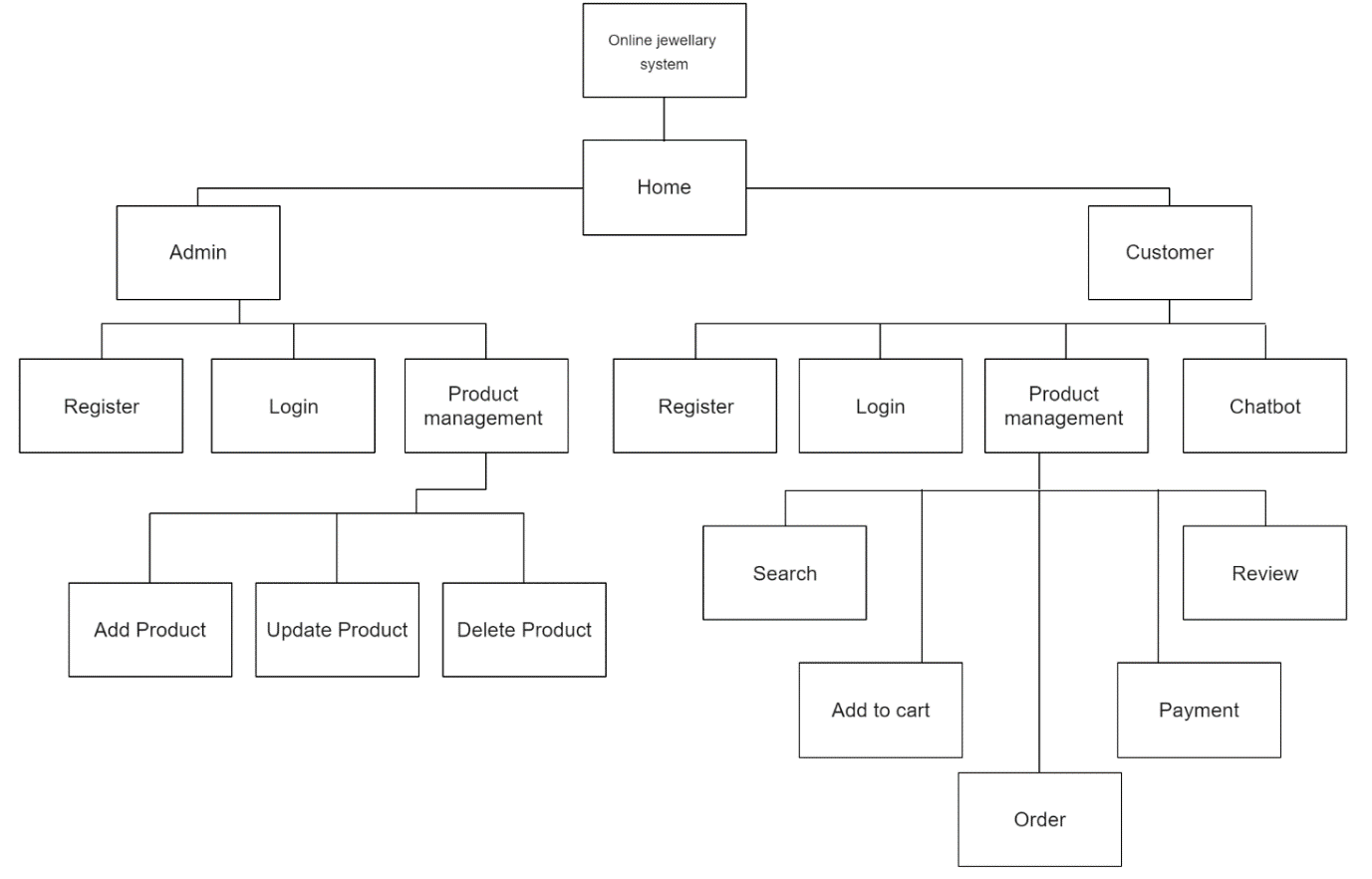
**Chapter 4**

**System Design**

**4.1 Business rules**

* User must provide a valid email address and the mobile number while registration.
* While login in the user must provide the same password as the password given during the registration.
* Only the user who have registered would be able to place the order.
* The website won’t be providing exchange policy allowing the customer to replace things that are broken or damage during the delivery process.

**4.2 Module Diagram**



**Figure 4.1 module diagram**

**4.3 Entity Relationship Diagram**

An Entity Relationship Diagram (ER Diagram) pictorially explains the relationship between entities to be stored in a database. Fundamentally, the ER Diagram is a structural design of the database. It acts as a framework created with specialized symbols for the purpose of defining the relationship between the database entities. ER diagram is created based on three principal components: entities, attributes, and relationships. ER diagrams are essential tools for database design, helping developers and stakeholders understand the logical organization of data, identifying key relationships, and ensuring the proper implementation of a database schema. By providing a clear and intuitive overview, ER diagrams play a crucial role in improving communication and ensuring the accuracy of database systems in various domains, such as business, software development, and data management. ER model becomes an abstract data model that defines a data or information structure which can be implemented typically in a relational database.

**Symbol references:** Silberschatz−Korth−Sudarshan

**Diagram Notations:**

|  |  |  |
| --- | --- | --- |
| Name | Symbol | Description |
| Rectangle |  | Entity set |
| Double Rectangle |  | Weak Entity set |
| Diamond |  | Relationship set |
| Double Diamond |  | Identifying relationship set for weak entity set |
| Eclipse |  | Attribute |
| Double Eclipse |  | Multivalued attribute |
| Dotted Eclipse |  | Derived attribute |
| Primary key | PK | Primary key |
| Discriminator | PK | Discriminating attribute of weak entity set |
| Many-to-many |  | Many-to-many relationship |
| One-to-one |  | One-to-one relationship |
| Many-to-one |  | Many-to-one relationship |
| Double line |  | Total participation of entity set in a relationship |
| Line |  | Links attribute to entity set or represents Partial participation of entity set in a relationship. |

**Table 4.1 Entity Relationship Diagram Symbols**

**4.3.1 Entity Sets:**

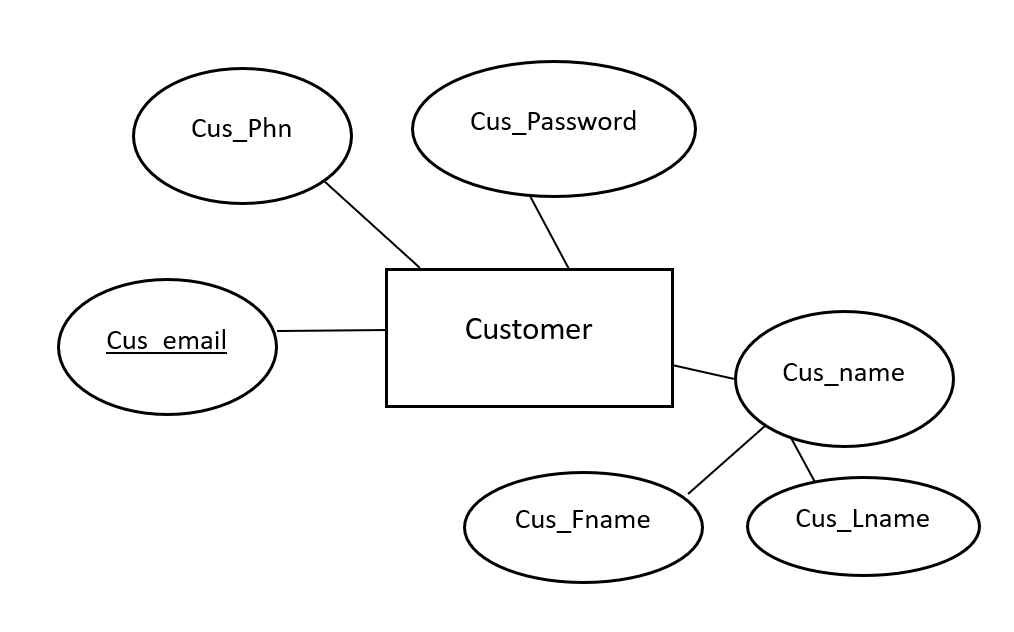
* Customer.
* Products.
* Order.
* Payment.
* Chatbot.
* Reviews.
* Customisation.

1. Customer:

The entity set customer contains the detail of every customer who purchase any product.

*Attributes:-*

* Cus\_Name
  + Composite attribute
  + It contains the name that is the first and last name of customer.
  + Component attributes-
    - Cus\_Fname
    - Cus\_Lname
* Cus\_Password.
  + Single valued and simple attribute.
  + It contains the password of customers.
* Cus\_Phn.
  + Single valued and simple attribute.
  + It contains the mobile number of customers.
* Cus\_Email
  + Single valued attribute and simple attribute
  + Simple attribute, primary key.
  + It contains the unique attribute for each customer.
  + It contains email of customer.



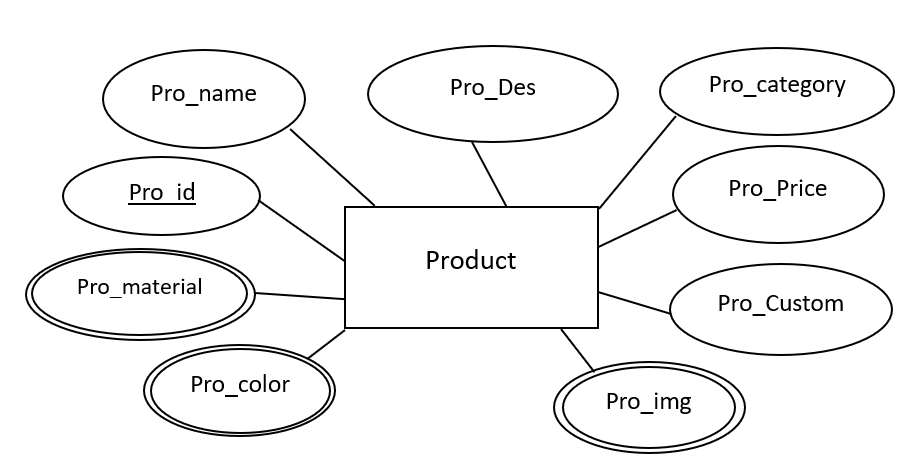
**Figure 4.2 Customer Entity Set**

1. Products.

Product entity set will have all the information related to products.

*Attributes: -*

* Pro\_ID
  + simple attribute and primary key.
  + A unique ID for each product.
* Pro\_Name.
  + Simple attribute.
  + The name of the product.
* Pro\_Des.
  + Simple attribute.
  + A brief description or details about the product.
* Pro\_Category
  + Simple attribute.
  + It is the type of product the item belongs to such as ring, hearing, necklace.
* Pro\_Price
  + Simple attribute and single value attribute.
  + The cost of the product.
* Pro\_Material
  + Simple attribute and multivalued attribute.
  + The materials that are used to make such as pearl diamond, silver etc.
* Pro\_Image.
  + Simple attribute and multivalued attribute.
  + The image of the product.
* Pro\_Colour
  + Simple attribute and multivalued attribute.
  + The colour of the product.
* Pro\_Custom
  + simple attribute.
  + Would have the option of yes or no.
  + if yes customisation is available and if no customisation is not available.



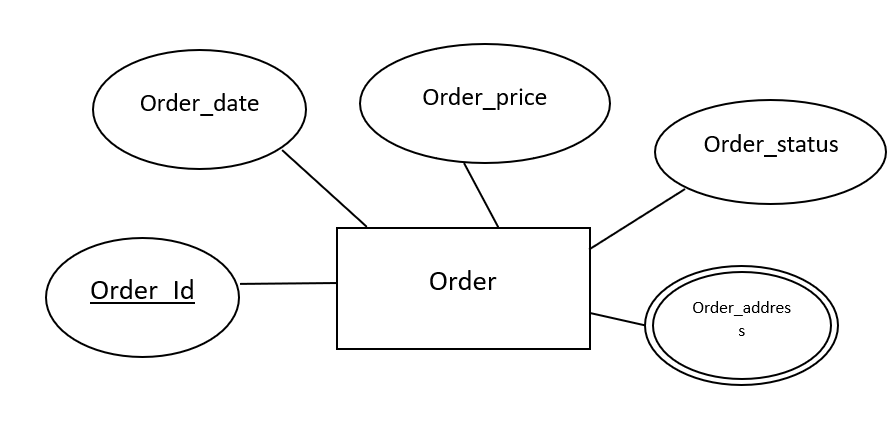
**Figure 4.3 Product Entity Set**

1. Order

It represents individual order of customer who have procedure to order the product.

*Attributes: -*

* Order\_ID.
  + Simple attribute and primary key.
  + It is the unique ID for each order placed.
* Order\_date.
  + Simple attribute.
  + The date when the order was placed.
* Order\_price.
  + Simple attribute, single valued attribute.
  + It is the total amount of the product.
* Order\_address.
  + It is simple attribute and multi-valued attribute.
  + The delivery address where it must be delivered.
* Order\_status
  + Simple attribute.
  + It would show that it is shipped, processing, delivered.

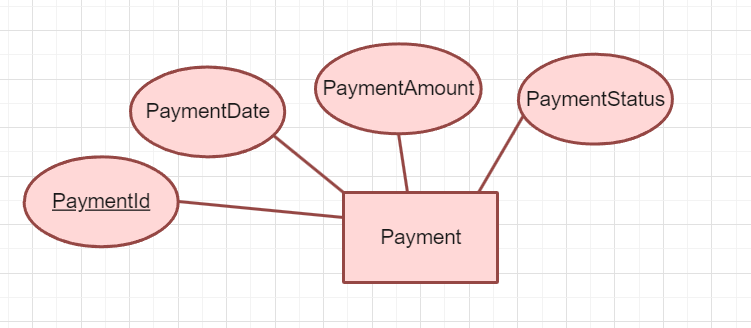


**Figure 4.4 Order Entity Set**

1. Payment.

It represents the payment done for an order by the customer.

*Attributes: -*

* PaymentID.
  + Simple attribute primary key.
  + The unique key with represents the payment ID.
* Payment date
  + Simple attribute
  + It shows the date on which the payment is done.
* Payment amount.
  + Single valued and simple attribute.
  + The amount that is the total amount to be paid.
* Payment status.
  + Simple attribute.
  + It indicates whether it is successful pending failed etc.

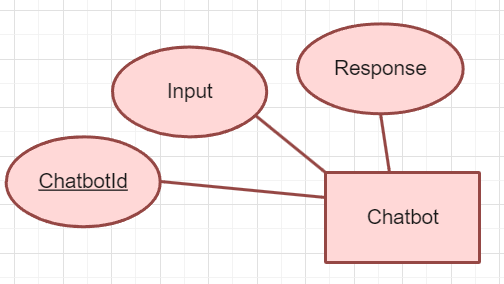
**Figure 4.5 Payment Entity Set**

1. Chatbot.

It answers the queries of the user.

*Attributes: -*

* ChatbotID.
  + Simple attribute and primary key.
  + The unique ID for chatbot.
* Input.
  + Simple attribute.
  + The query that is ask by the user.
* Response.
  + Simple attribute.
  + The response which is given by the chatbot.



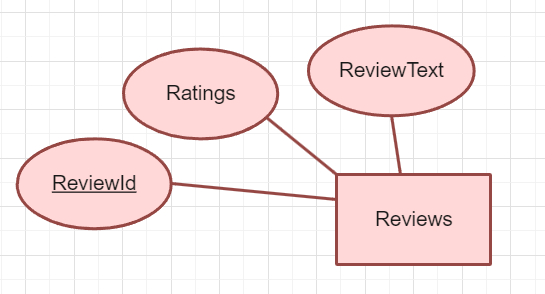
**Figure 4.6 Chatbot Entity Set**

1. Reviews

They are the feedback provided by the user to share their opinion and experience with the product and service.

*Attributes*

* ReviewID.
  + Simple attribute and primary key.
  + The unique key to identify the review.
* Ratings.
  + Simple attribute.
  + The user can rate the product or service on the scale of 1to5.
* Reviewtext.
  + Simple attribute.
  + A brief description that is the feedback about the product.



**Figure 4.7 Reviews Entity Set**

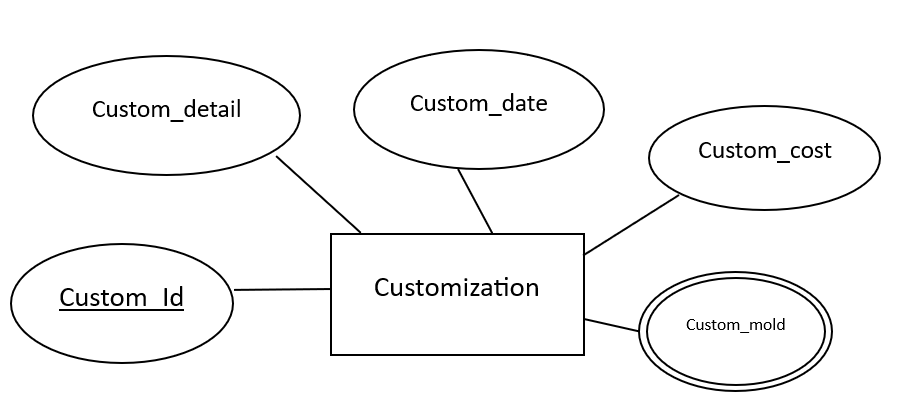
1. Customization.

The user wants to customise jewellery of the choice then they can go for customisation option.

If the customisation is available then they can proceed for customisation or else they can't.

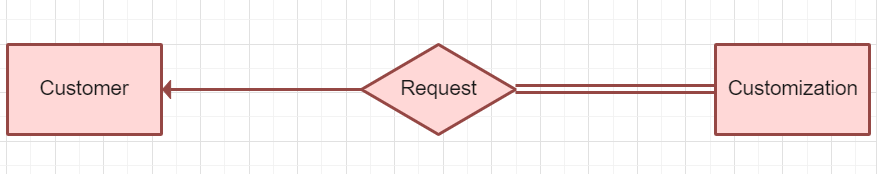
*Attributes: -*

* Custom\_ID.
  + Simple attribute and primary key.
  + The unique key for customisation.
* Custom\_Detail.
  + Simple attribute.
  + They can provide a brief detail about how they want it to be.
* Custom\_Date.
  + Simple attribute.
  + The date on which customisation is done.
* Custom\_Cost.
  + Simple attribute.
  + Additional cost if required is the additional cost.
* Custom\_mold
* Simple attribute and multivalued attribute.
* It shows images of the mold.

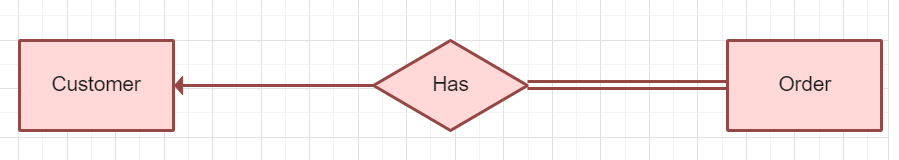


**Figure 4.8 Reviews Entity Set**

**4.3.2. Relationship Sets:**

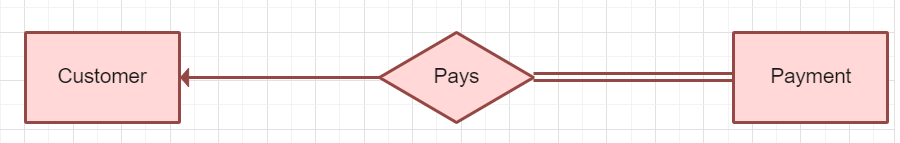
1. Customer – customisation.
   1. Mapping Cardinality:1 to many relationships.
   2. Customer requests for customisation.
   3. Participation: Partial participation.
   4. One customer can request multiple customizations.

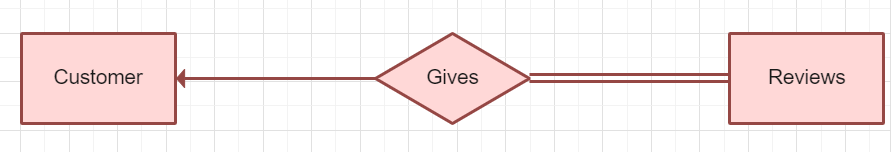
**Figure 4.9 Customer – customisation Relationship**

1. Customer – order.
   1. Mapping Cardinality:1 to many relationships.
   2. Customer has order.
   3. Participation: Partial participation.
   4. One customer can place many orders, but each order associates one customer.

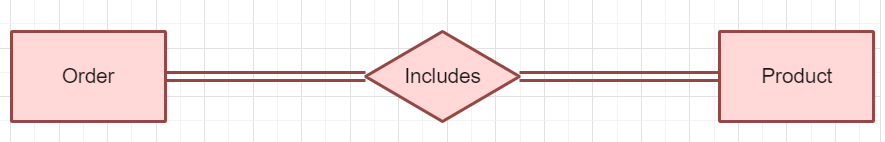
**Figure 4.10 Customer – order Relationship**

1. Customer – payment.
   1. Mapping Cardinality:1 to many relationships.
   2. Customer pays payment.
   3. Participation: Partial participation.
   4. A customer can make multiple payments, but each payment is of one customer.

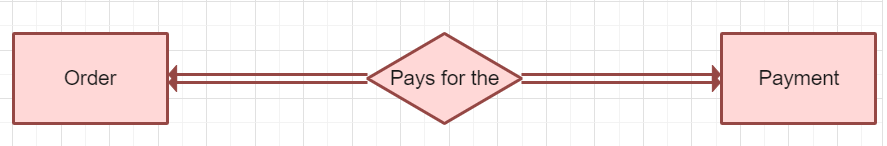
**Figure 4.11 Customer – Payment Relationship**

1. Customer – review.
   1. Mapping Cardinality:1 to many relationships.
   2. Customer gives review.
   3. Participation: Partial participation.
   4. One customer can write multiple reviews.

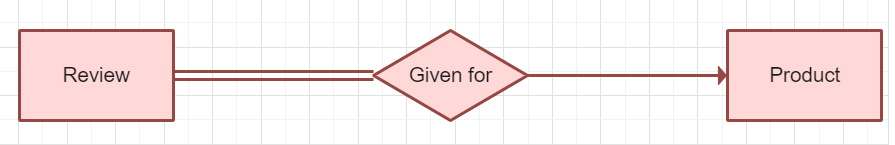
**Figure 4.12 Customer – Review Relationship**

1. Order - product.
   1. Mapping Cardinality: Many to many relationships.
   2. Product included in order.
   3. Participation: Total participation.
   4. Many products can be there in a single order and A single product can be there in multiple orders.

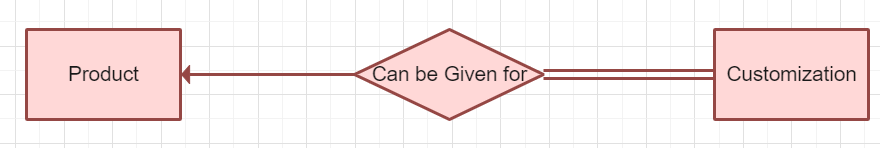
**Figure 4.13 Order - Product Relationship**

1. Order – payment.
   1. Mapping Cardinality:1 to 1 relationship.
   2. Order pays for the payment.
   3. Participation: Total participation.
   4. Each order is associated to a payment and each payment is associated with a respective order

**Figure 4.14 order - payment Relationship**

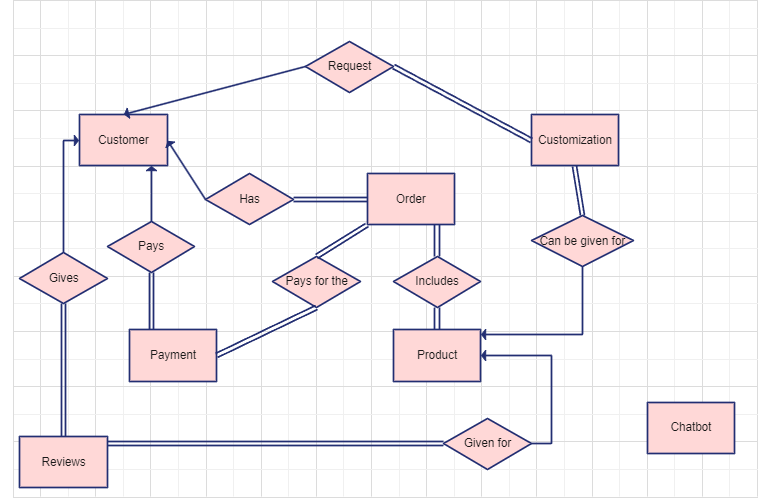
1. Product - review.
   1. Mapping Cardinality:1 to many relationships.
   2. Review given for product.
   3. Participation: Participation: Partial participation.
   4. A single product can have multiple reviews.

**Figure 4.15 Review - Product Relationship**

1. Product – customisation.
   1. Mapping Cardinality:1 to many relationships.
   2. Product can be given for customisation.
   3. Participation: Partial participation.
   4. A single product can be customised multiple times.

**Figure 4.16 Product - Customization Relationship**

**4.3.3. ER Diagram:**



**Figure 4.17 ER diagram**

**4.4 Schema:**

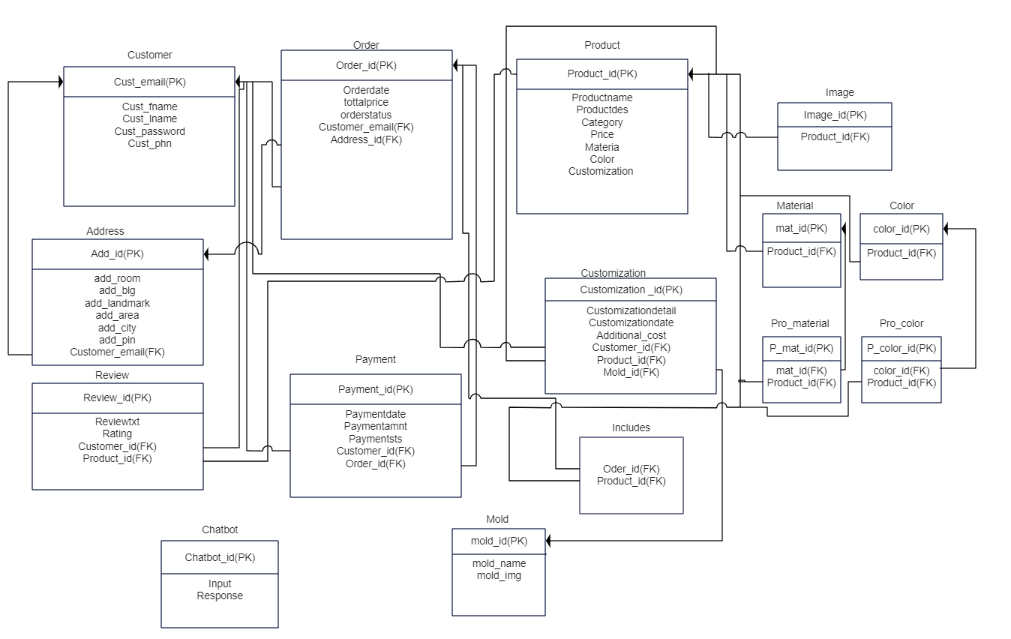
A database schema is the skeleton structure that represents the logical view of the entire database. A database schema, along with primary key and foreign key dependencies, can be depicted pictorially by schema diagrams. It formulates all the constraints that are to be applied on the data.

**Symbol Reference:** https://www.lucidchart.com/

|  |  |  |
| --- | --- | --- |
| Name | Symbol | Description |
| Table |  | A table is a collection of related data held in table format within a database. |
| Relation |  | In a relational database system, a one-to-one table relationship links two tables based on a Primary Key column in the child which is also a Foreign Key referencing the Primary Key of the parent table row. Therefore, we can say that the child table share the Primary Key with the parent table. |

**Table 4.2 Schema Diagram Symbols**

**Diagram:**



**Figure 4.18 Schema diagram**

**4.5 Data Flow Diagram**

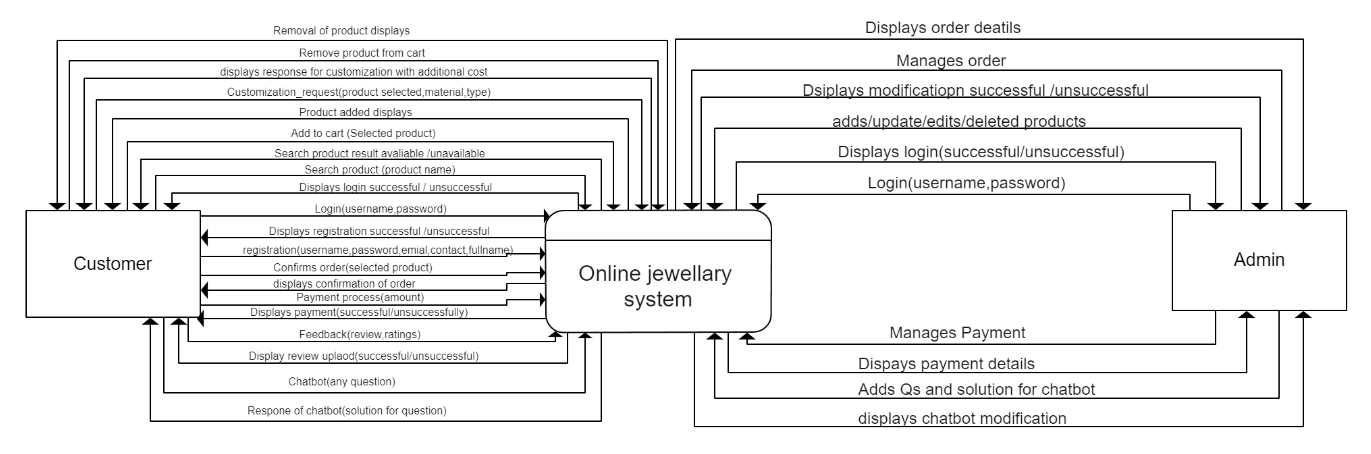
Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation. Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

**Symbol Reference**: <https://www.lucidchart.com/>

|  |  |  |
| --- | --- | --- |
| Name | Symbol | Description |
| Process |  | A process transforms incoming data flow into outgoing data flow. |
| Database |  | Data stores are repositories of data in the system. |
| Data Flow |  | Data flows are pipelines through which packets of information flow. Label the arrows with the name of the data that moves through it. |
| External Entity |  | External entities are objects outside the system, with which the system communicates. |

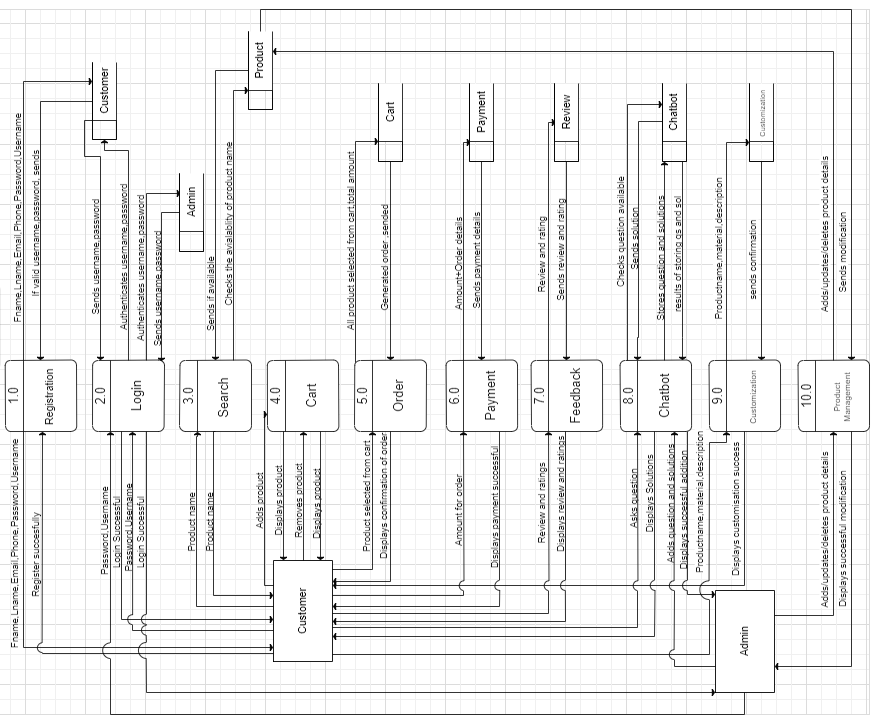
**Table 4.3 Data Flow Diagram Symbols**

**Level 0 (Context Level DFD):**



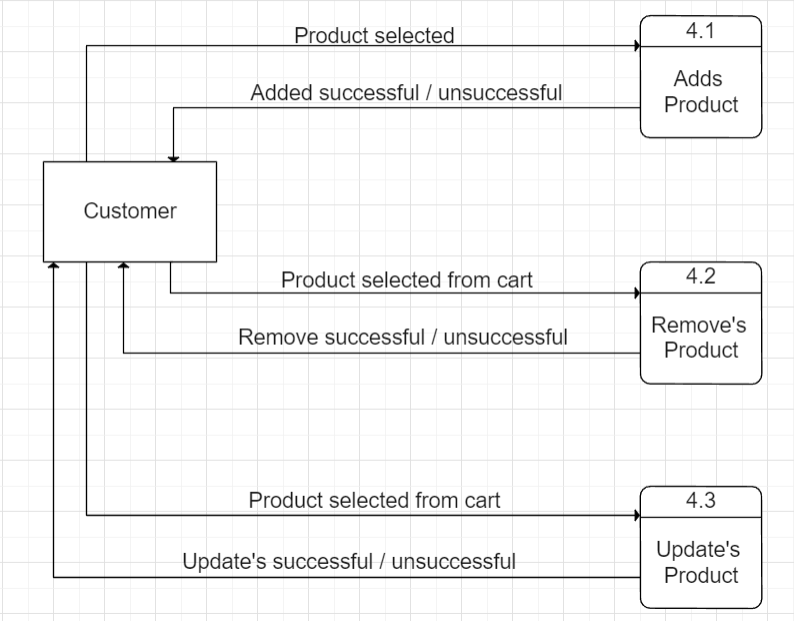
**Figure 4.19 Level 0 DFD**

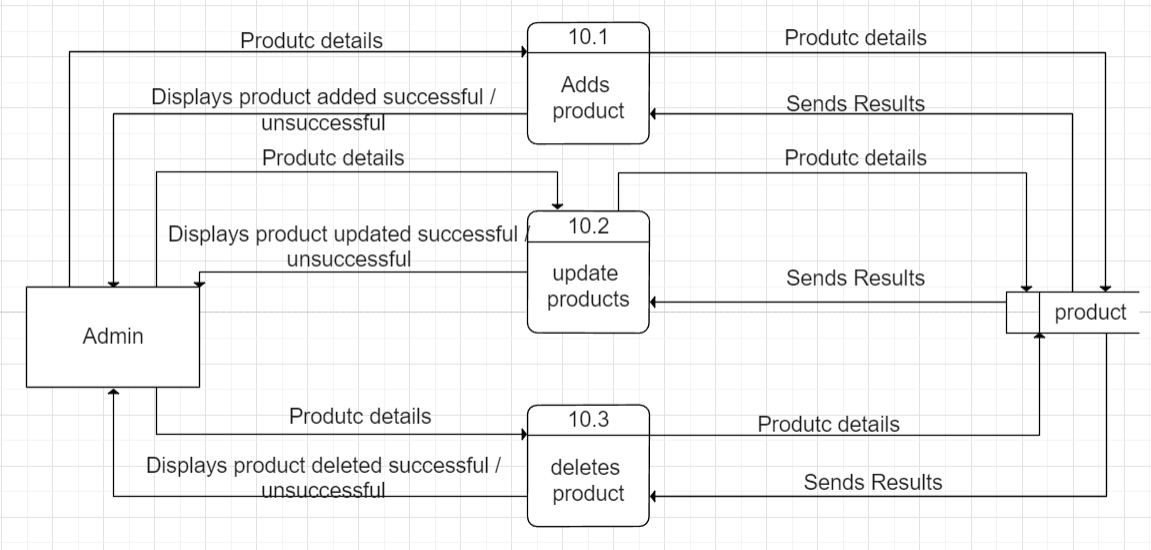
**Level 1 DFD:**

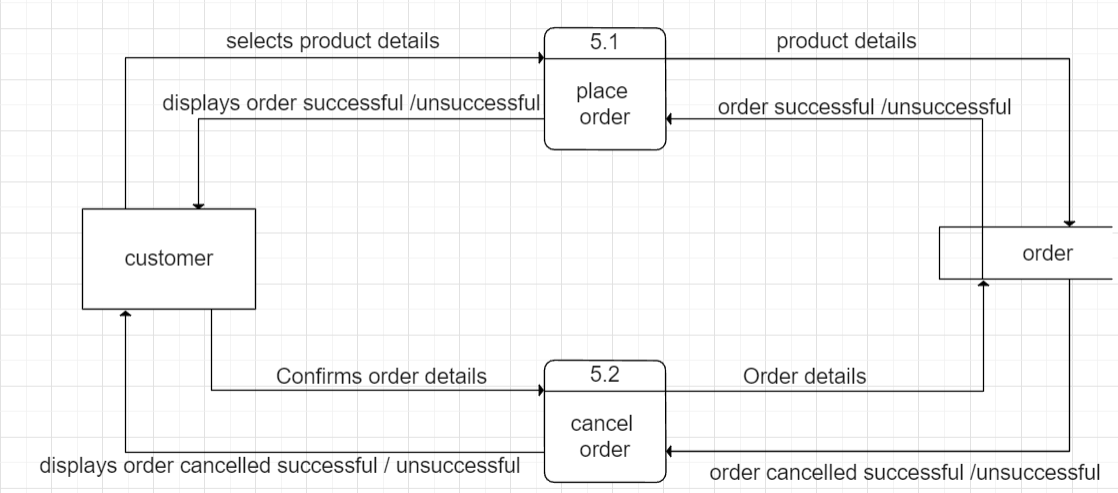


**Figure 4.20 Level 1 DFD**

**Level 2 DFD:**







**Figure 4.21 Level 2 DFD**

**4.7 Use Case Diagram**

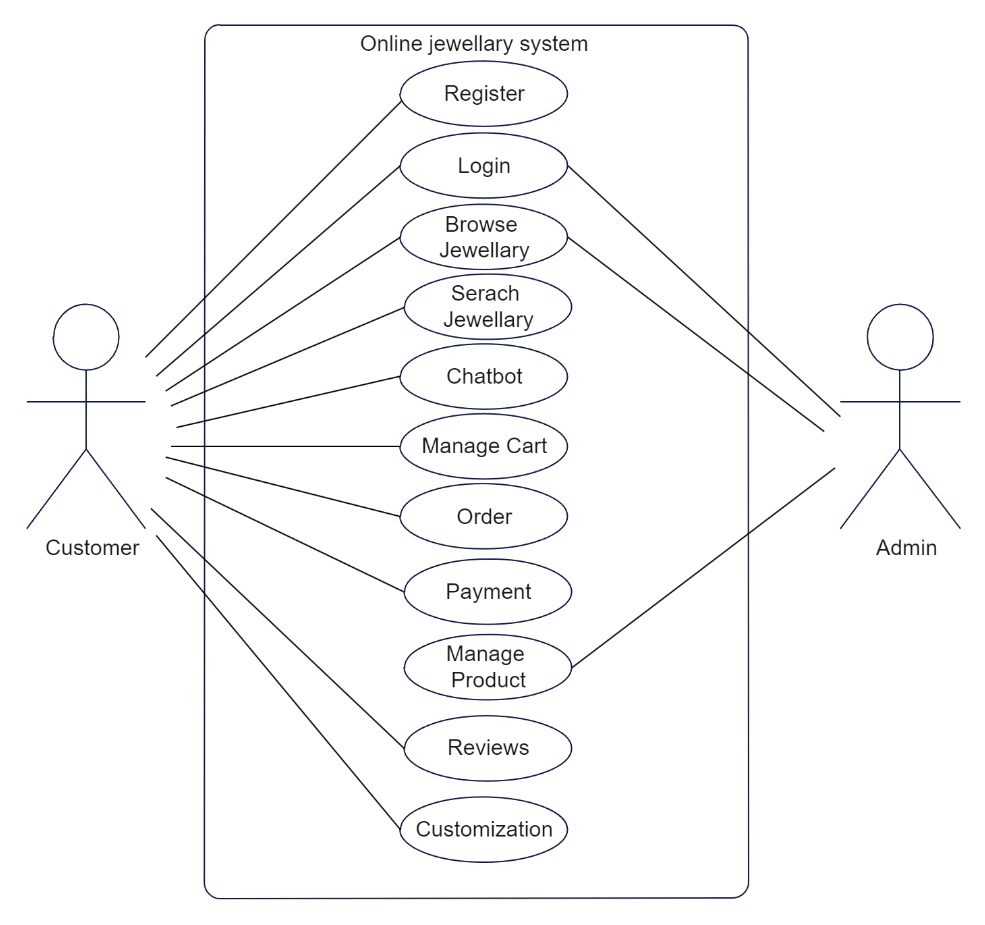
A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

Symbols Reference: <https://www.lucidchart.com/>

|  |  |  |
| --- | --- | --- |
| Name | Symbol | Description |
| Actor |  | Actor represents a user or another system that will interact with the system you are modelling |
| Use case |  | A use case is an external view of the system that represents some action the user might perform in order to complete a task. |
| Associations |  | Association between use cases. |
| Include Relationship |  | Include relationship between the use cases |

**Table 4.4 Use Case Diagram Symbols**

**4.7.1 Diagram**



**Figure 4.22 Use Case Diagram**

**4.7.2 Description**

1.Use case -Register

* 1. Summary- A user registers if the user is a new user and can access to the system for jewellery related services.
  2. Actors-Customer
  3. Preconditions- The user is not registered in the system
  4. Description-The user accesses the system. The system presents a registration form with some required fields i.e full name, email, contact, username, password. The username fills the required information. The system validates the input. If the user’s validation is successful then system creates a new account and display a confirmation message to the user indicating successful registration
  5. Exception- Invalid input: If user provides incomplete information the system displays error and says to fill again
  6. Existing User Account: If the user has already created an account, then the system would send message account already exists
  7. Post condition-The user can login to access the system services and other features

2.Use Case- login

1. Summary-Any user who has registered and created an account can login into the system
2. Actors-Customer/Admin
3. Pre-condition-The actor has already registered an account
4. Description-The actor has visited the login page and fill the necessary credentials i.e username and password. The system validates the inputs and verifying the password matches with the stored password if the user verification is successful then the system operates access to the user also say to login again with the error message invalid credentials
5. Exceptions- Invalid credentials: if the user enters incorrect login credentials the system will display error messages
6. Unverified account-the user won’t be having a registered account, the system would display error messages saying to create a account first
7. Post Condition-The user is successful logged in and can browse product and other features.

3.Use Case-Browse Products

1. Summary-The user explores and views the available products
2. Actors-Customers/Admin
3. Pre-condition-The user should logged into their account to access the system
4. Description-The user after logging into the du could able to view and access the available products in the system. The system would display various categories, search option, filter as well as the product with their name, price, material, description etc The user would able to add products to the cart
5. Exception- In case the user is not logged into the system then the message would display to login first
6. Post Condition- The user can now make orders, add, and remove product from cart.

4. Use case- Search products

1. Summary: A user can search products which the desire to view
2. Actors: Customer
3. Precondition: The user should be logged in to the system to access the system.
4. Description: The user selects the search product to find the specific product which the desire to view. This system would provide the user a search bar to input the appropriate reward which can be product name category then the system would display the products matching with the criteria. When the products are display it would show the name, price, material, image description of the product. Exception: Search result invalid: If the user inputs incomplete query or if the product is not available then the system would display not available.
5. Post condition: The user of can view all the information regarding to the product and can purchase it.

5. Use case-chatbot.

1. Summary- The user can message a chatbot for any query related to the product.
2. Actor-customer.
3. Pre-condition-the user of must logged and the system to access
4. Description- When the user select the chatbot feature the system provides a input field to type the query of question they want to ask. The chatbot process and give the valid and appropriate solution for the respective question.
5. Exception- Invalid query: if the chat what doesn't understand the users query then the airport would not able to give a valid response.
6. Post condition-If yet the query is not resolved the system redirects the user to the customer support that is the contact us page.

6. Use case-manage cart.

1. Summary- The user can view the product and add to the desired product to the card and remove from the cart when required. Actor-customer.
2. Precondition- User must be logged into the system and choose a desired product to add to the cart and while removing product from cart there should be at least one product in the cart.
3. Description- The user of select the product the wish and add to the cart then the system displays the list of product and the cart. The product in the cart would display all the information regarding to the product. The user can also add multiple products for each products that is update the quantity of the product and when the user doesn't want the product then the user can remove the selected product from the cart. After confirming the products in the cart the user can proceed for the order.
4. Exception- If the cart is empty then the user cannot proceed to order.
5. Precondition- The system would re direct the order page after confirming the products in the cart.

7. Use case-order.

1. Summary-User can confirm their order after confirming the desired product they wish to purchase.
2. Actor- Customer.
3. Pre condition-The user must be logged into the system. The user must have at least one product added to the card.
4. Description-The user conforms the product and proceeds to order. The system would display the order process. The use of must give the valid and appropriate address where the product must deliver. The system then would display all the products that are confirmed for order with the total amount price of all product including the list of product names and user conforms the order then the system displace the payment procedure. The use can view there order history to track status of order.
5. Exception- If the address field input and not field then order would not confirm.
6. Post condition- System generates order confirmation and redirects to payment procedure and user can later track the status of their order.

8. Use case-payment.

1. Summary-After completion of order the system displays procedure of payment for the respective products.
2. Actor-Customer.
3. Pre-conditions-The user can proceed for confirming order and proceed for payment procedure.
4. Description-The user reaches to the payment page of the system. The system displays list of product name along with number of products and total amount and says to do the payment that is the amount. The user enter the amount which is required then the system verify the payment if the payment verification is successful each and rates a message of payment successful and order confirm else displace message payment and successful and says to pay again.
5. Exception-Payment fail:If the amount is enter wrong other less amount in their account it is place error message saying to unsuccessful payment and to pay again.
6. Post condition-The user after successful payment can track their order and the status of order whether it's pending, failed or success

9. Use case-Manage product

1. Summary- An admin peforms some actions such adding, updating, deleting products in the system.
2. Actor- Admin
3. Pre-condition- The admin should be logged into their system
4. Description- Admins adds the product such As name, description, price, category, Material. Admin also update to deletes this information for the system in a correct format than these data get stored
5. Exception- If the confirmation of these upadation is denied then server The condition pops up
6. Post-condition- The admin successful adds, updates, delete the products and its visible in the datbase and to the customers

10. Use case-Review

1. Summary-The User provides feedback and sends review for a product which is the overall experience of the product
2. Actor-Customer
3. Pre Conditions- User must be logged into the system. User must have purchased the product in order to review.
4. Descriptions-After receiving the product, the user may go to the Review section for the respective product. The Review field usually contains the Ser cutting (1 to 5 stars), the written review of 3-4 lines more. User can choose any method and give reviews. These revives get stored and displays to the customer
5. Exception: (1)Invalid: If the user inputs invalid review the system should display door message.
6. Post Condition: Successful reviews get uploaded and stored in the database and other user may view whild buying the jewellery product

11. Use case - Customization.

1. Summary- The user requests for Customization of products of their choice.
2. Actor- Customer
3. Pre-condition- The user must be logged into the system and user must proceed for customisation field
4. Description- The user selects the customisation offer. They can choose the choice of jewellery, choice of material, selection of material. After selection of user choice the additional Cost can get added up then the user may proceed to the order section which shows the total price.
5. Exception- Unavailable: User selects the product which is not available for customizations then a error message pops up.
6. Post condition- The the user confirms a customization of product then the details get stored in the database and the additional cost gets add to the order.

**4.8 Scenario**

* Registration:
  + User will fill registration form. The system would check if the username already exists or not. If the user is already existing then the message pops saying username already in use else the username is unique than the system successfully generates account. User has to fill some required fields such as name, contact, email password, username these data. These data will get stored in the database and registration successful message will show to the user.
* Login:
  + User would enter its username and password. System would match the credentials with the data store in the database. If any existing data found then the user would be logged into the system and home page would get displayed. If there is no match then the message pops as user must register first in order to login or else will use your credentials invalid message pops saying invalid credentials, please login again with correct username and password. the user is able to see the home page. It places various products such as rings, earrings, bracelets. The user select any product then the system this place to purchase the product then it may can be added to the cart.
* Search product:
  + After login into the website, the system displays the homepage with various products. If the user is interested in finding a specific piece of jewellery so they can type the product name into the search bar. As the user enters the appropriate keyword and presses and the system displays the similar product which matches the keyword and if the keyword is invalid it displace product not available.
* Chatbot:
  + After login into the system the user of would be able to access the system. If the user would have any query related to the order of product or anything else then the user can select the chatbot option. The user selects the chatbot icon on the screen and type their question in the input. If the user asks appropriate question, then chat what displace the solution for the respective query else for an invalid question it displays and says invalid question.
* Manage cart:
  + The user browser the website and comes across various product. If the user is interested in buying a particular product, then user might add the product to the cart. Uset a can add multiple products in the cart. The system displays the cart showing a list of items they have added. Each item in the cart includes all the product details. The user of has option to update the card by changing the quantity of items removing the items of cart.
* Order:
  + After user has added and updated the cart according to the choices they can proceed to order after clicking proceed to order the system displays the list of products, they wish to purchase with each unit price as well as the sub total of all the product after confirming the details user may proceed to payment procedure.
* Payment:
  + After proceeding to order the user can choose and go for payment. The user of enters the specified amount for the payment. After successful payment the order gets confirmed and the user may track their order status that weather is pending, successful or failed.
* Manage product:
  + The admin logs into the system with their given credentials. Admin can manage the product of the system that is add, delete or update any product as per the requirement. For adding product, the admin has to add all the product details such as image, description, cost, material, extra. Then the admin saves the details and it gets stores in the database and get this place to the customer. Admin can update and existing product by editing the details of the product and saving the details by which it gets stores in the database and displays to the customer. Similarly, the admin can delete any product from the system if the product is in no use and saving the details.
* Review:
  + If the customer has purchased a product and their wish to share the experience with the product, we can select the review option for each product. Users select a review, and input would get display in which they can give the feedback for the product which get stores and displays to the other customer as well. The user can also give rating through the product on the scale of 1 to 5.
* Customisation:
  + If the user wishes to design their own product, they can go for customisation option if it is available for the respective product for some products the customisation is available and for some it's not. User can select the material they want for a specific product the desire to customise.

**4.9 Sequence Diagram**

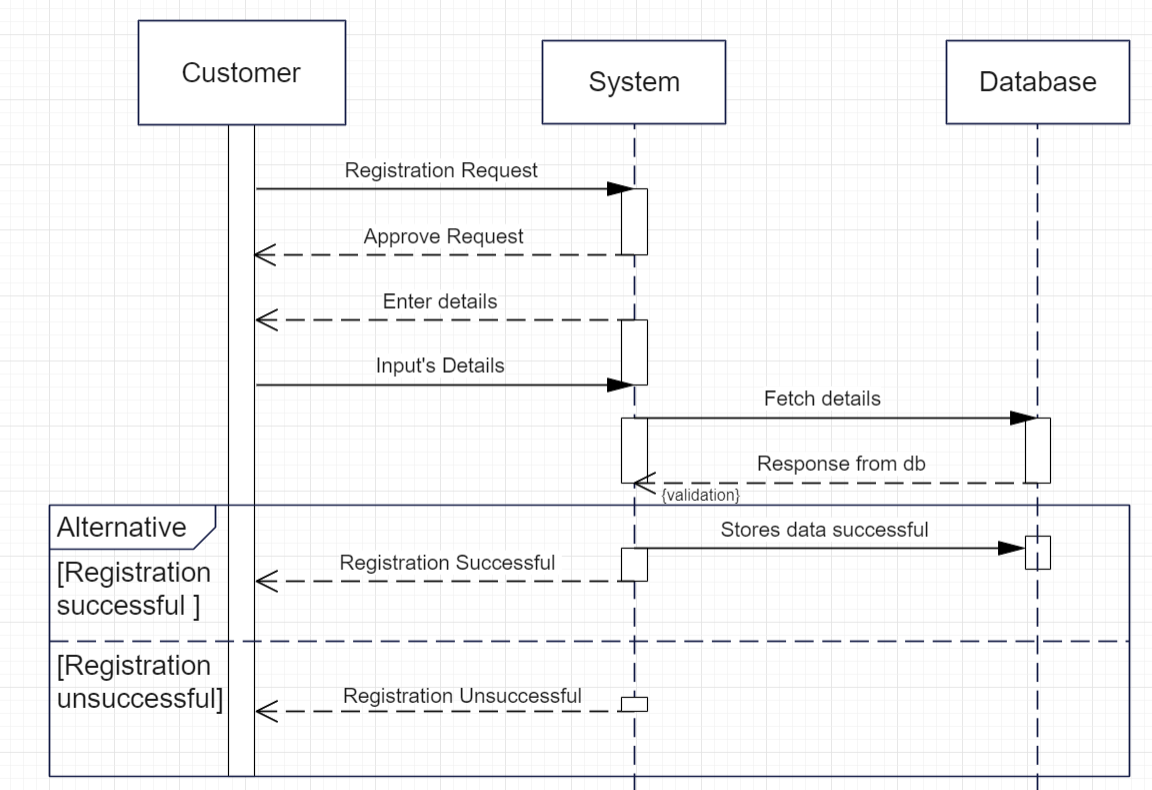
A sequence diagram in a Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams typically are associated with use case realizations in the Logical View of the system under development.

Symbol reference: <https://www.lucidchart.com/>

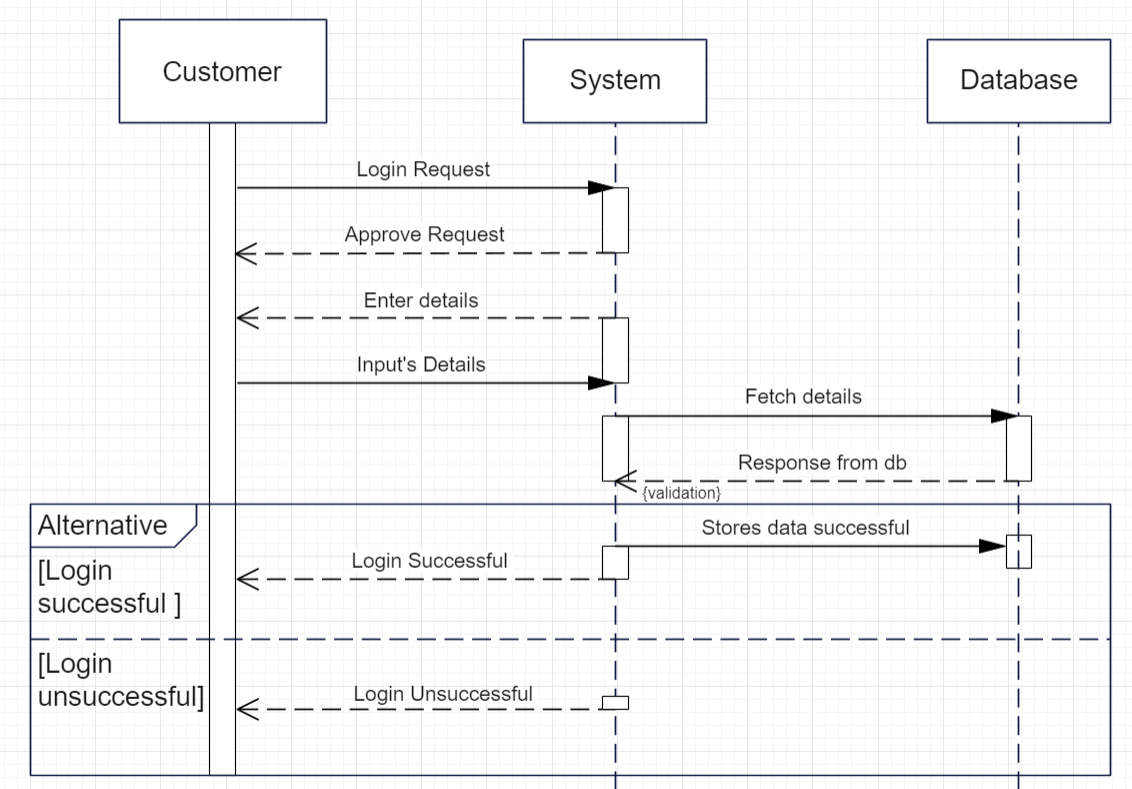
|  |  |  |
| --- | --- | --- |
| Name | Symbol | Description |
| Synchronous Message |  | Aninstantaneous communication between objects that conveys information, with the expectation that an action will be initiated as a result. |
| Activation Box |  | The period during which an object is performing an action. |
| Object |  | An object that is created, performs actions, and/or is destroyed during the lifeline |

**Table 4.5 Sequence Diagram Notation**

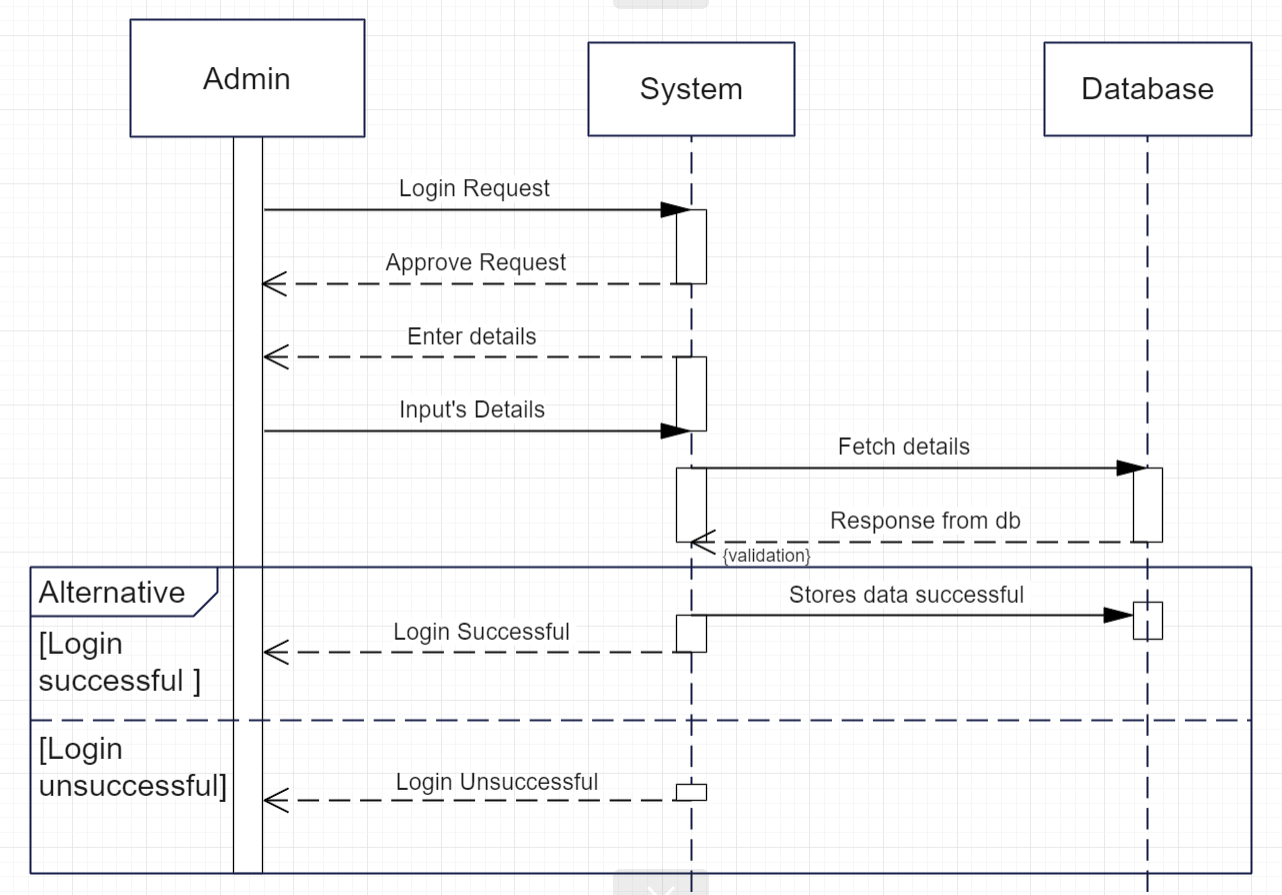
**Sequence Diagrams:**



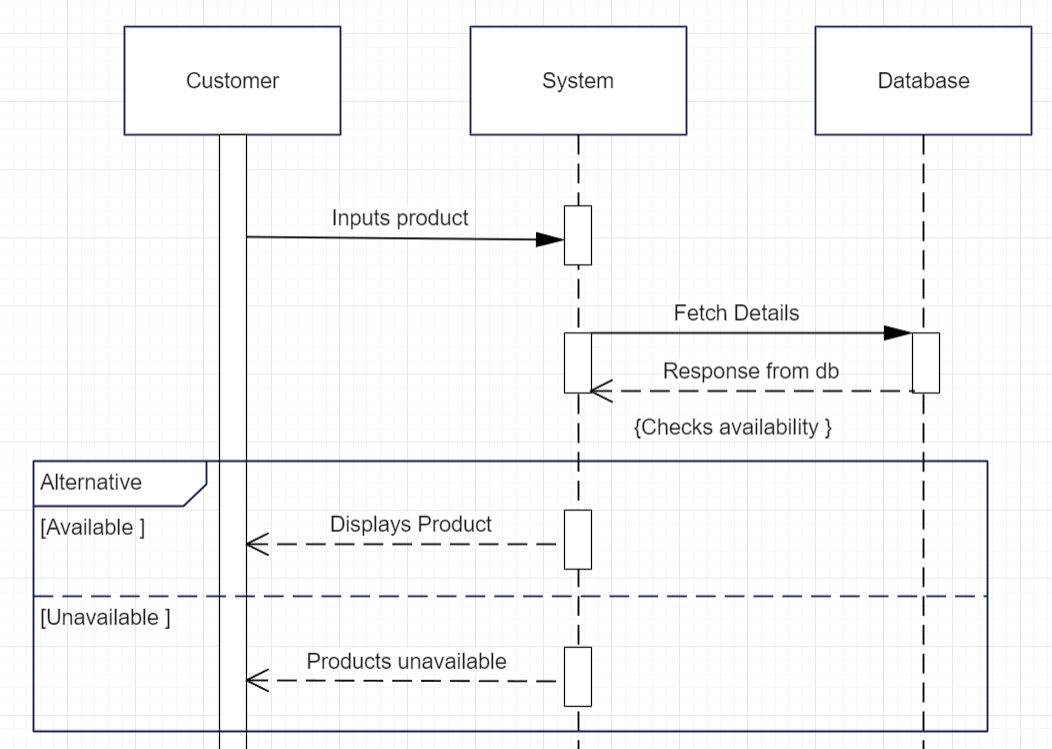
**Figure 4.23 Sequence Diagrams for customer registration**



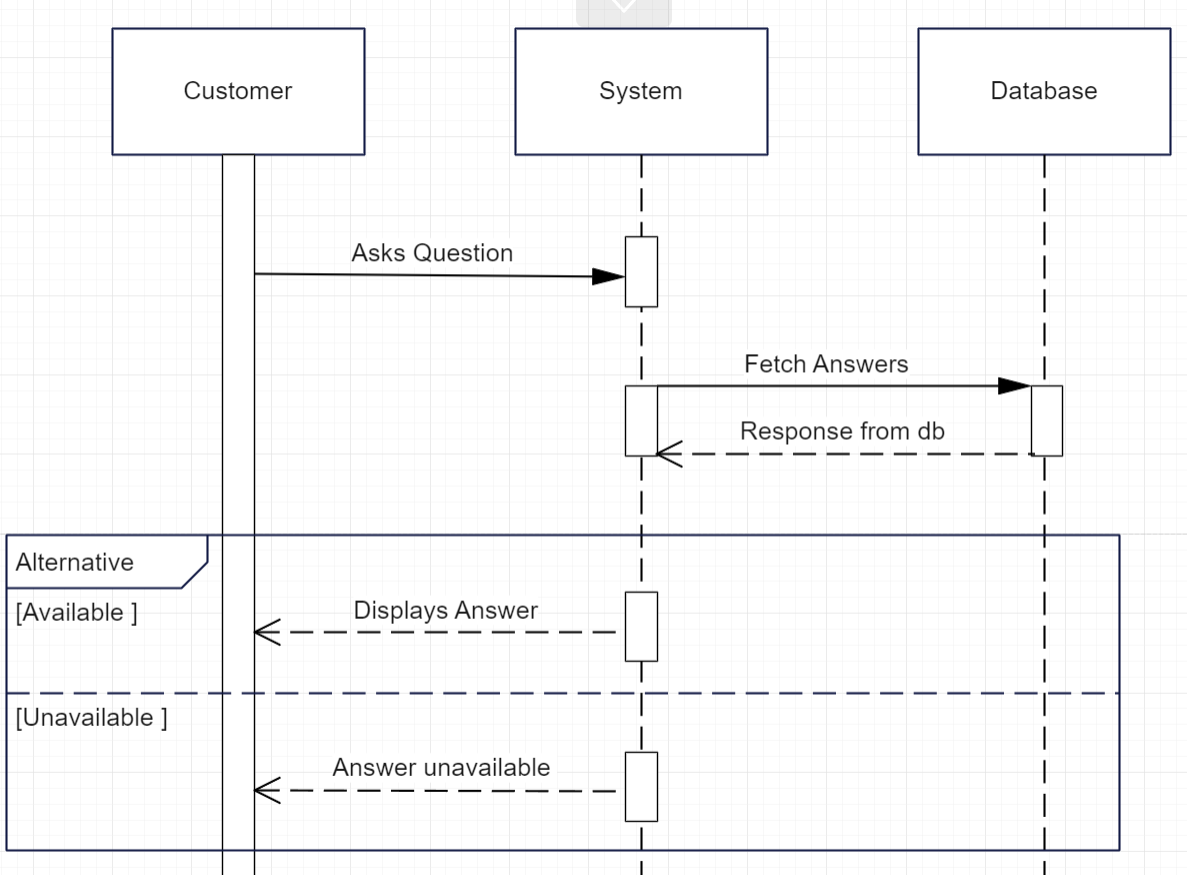
**Figure 4.24 Sequence Diagrams for customer Login**



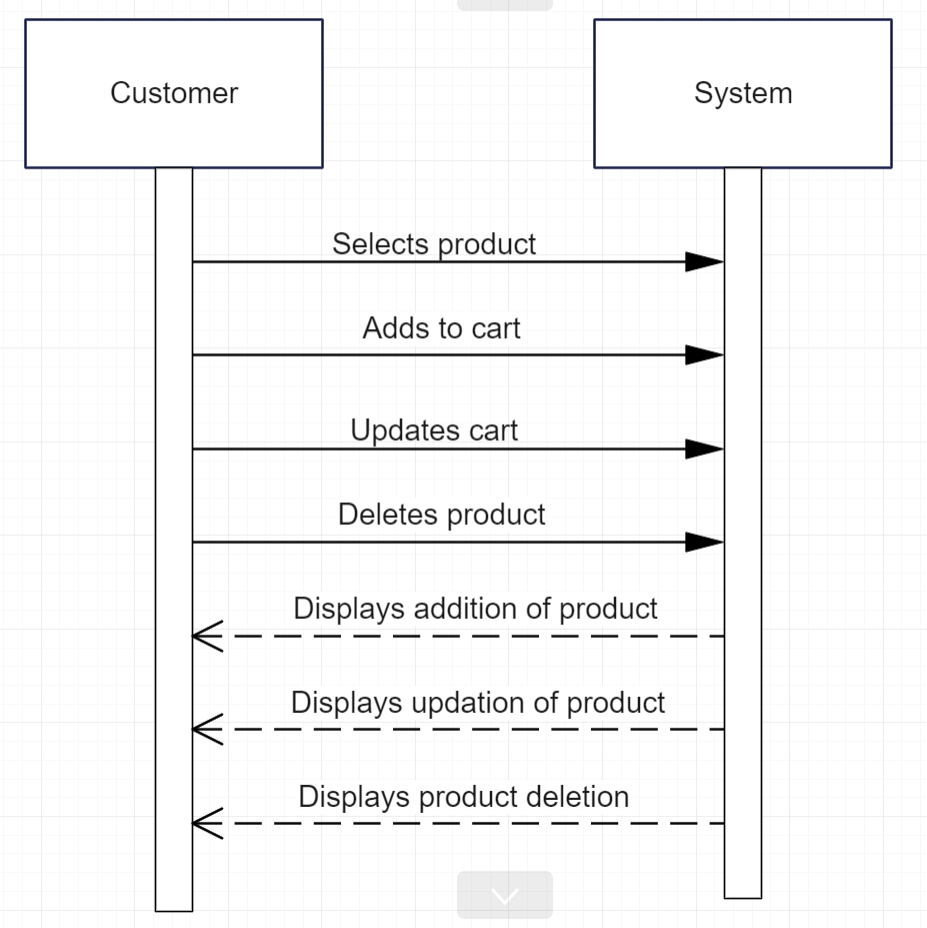
**Figure 4.25 Sequence Diagrams for admin Login**



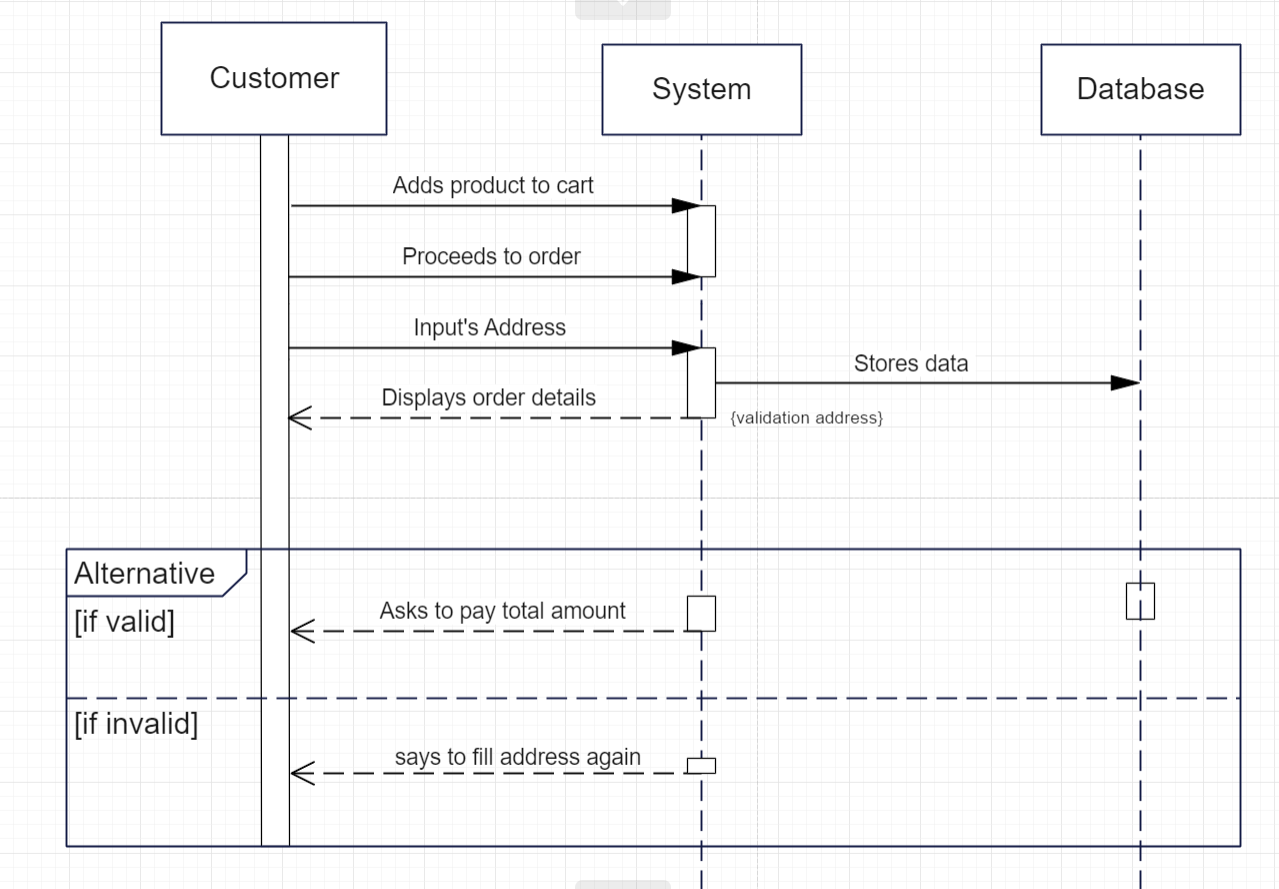
**Figure 4.26 Sequence Diagrams for search product**



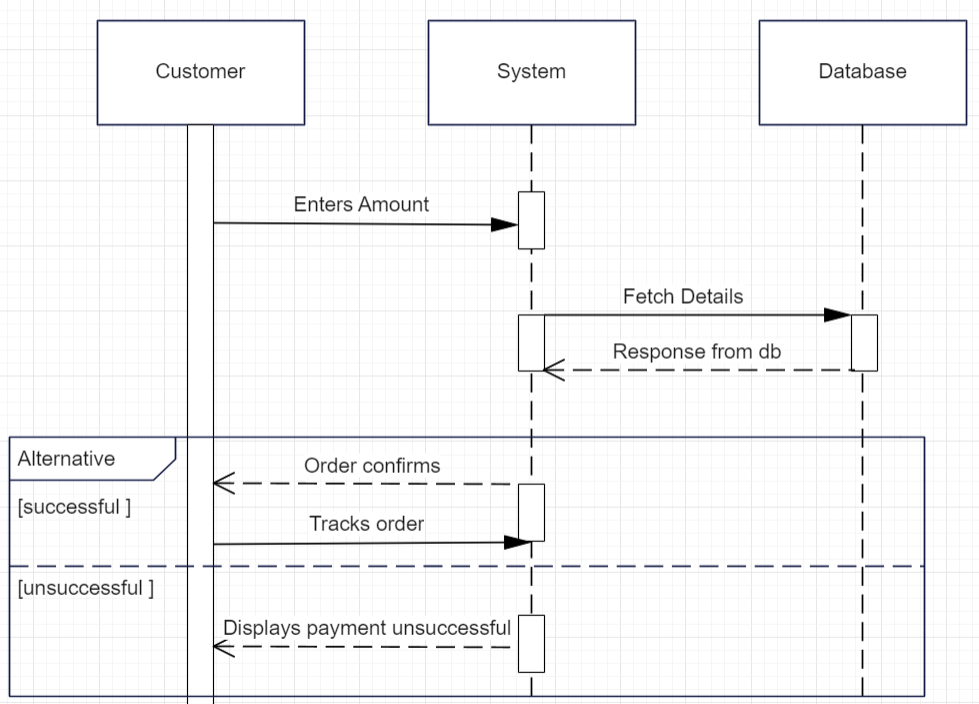
**Figure 4.27 Sequence Diagrams for chatbot**



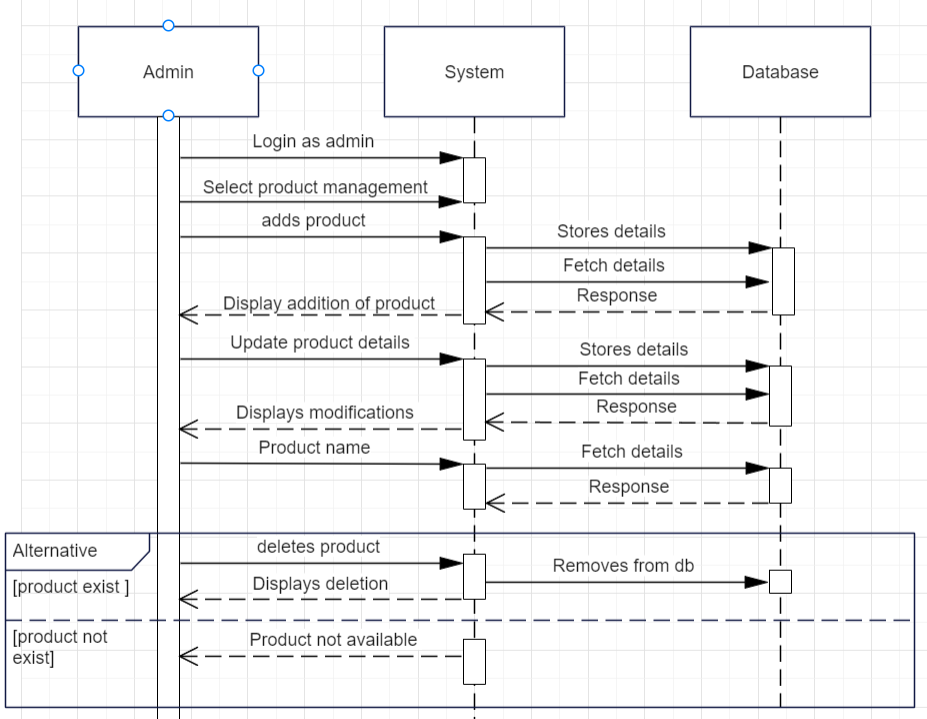
**Figure 4.28 Sequence Diagrams for Cart**



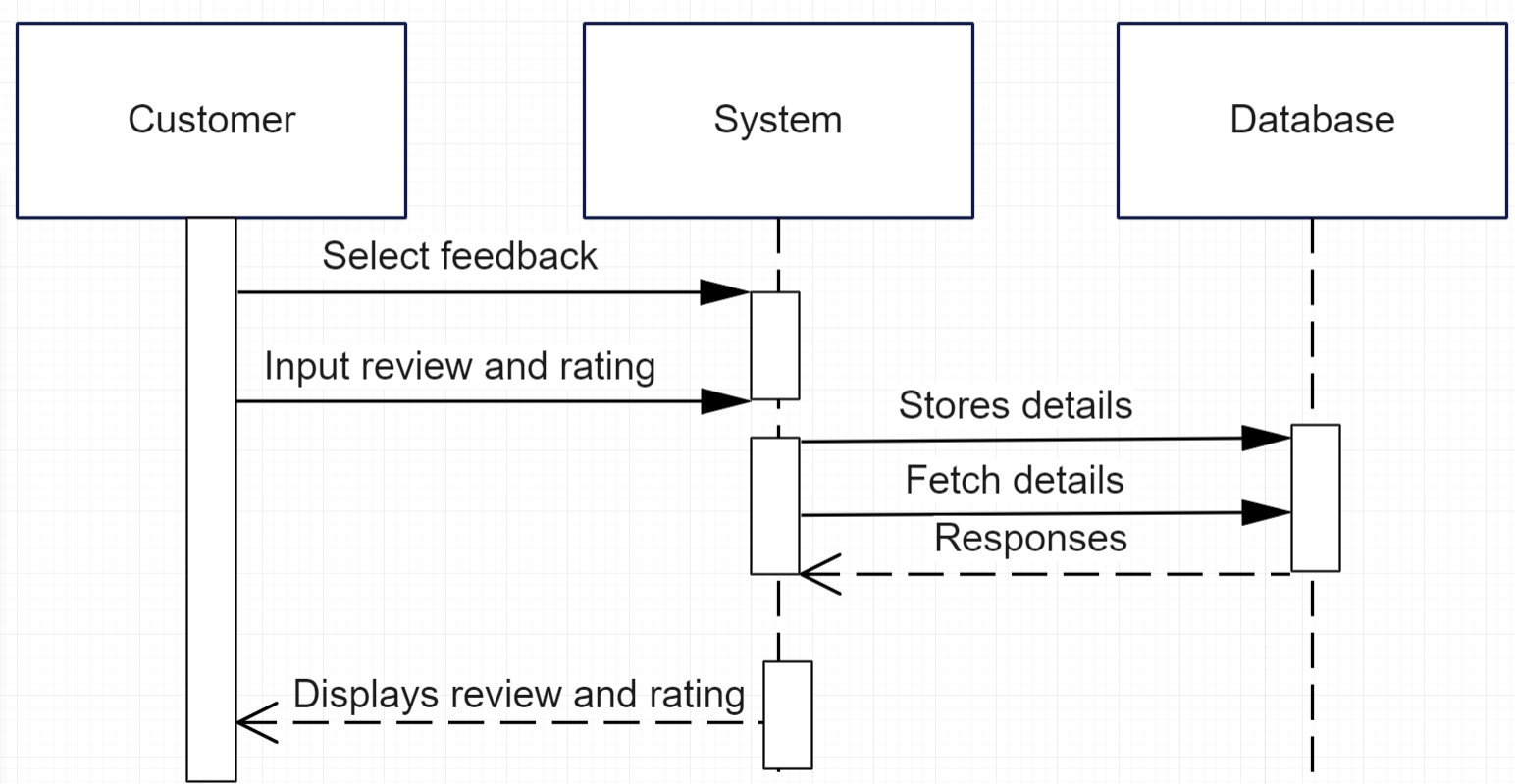
**Figure 4.29 Sequence Diagrams for Order**



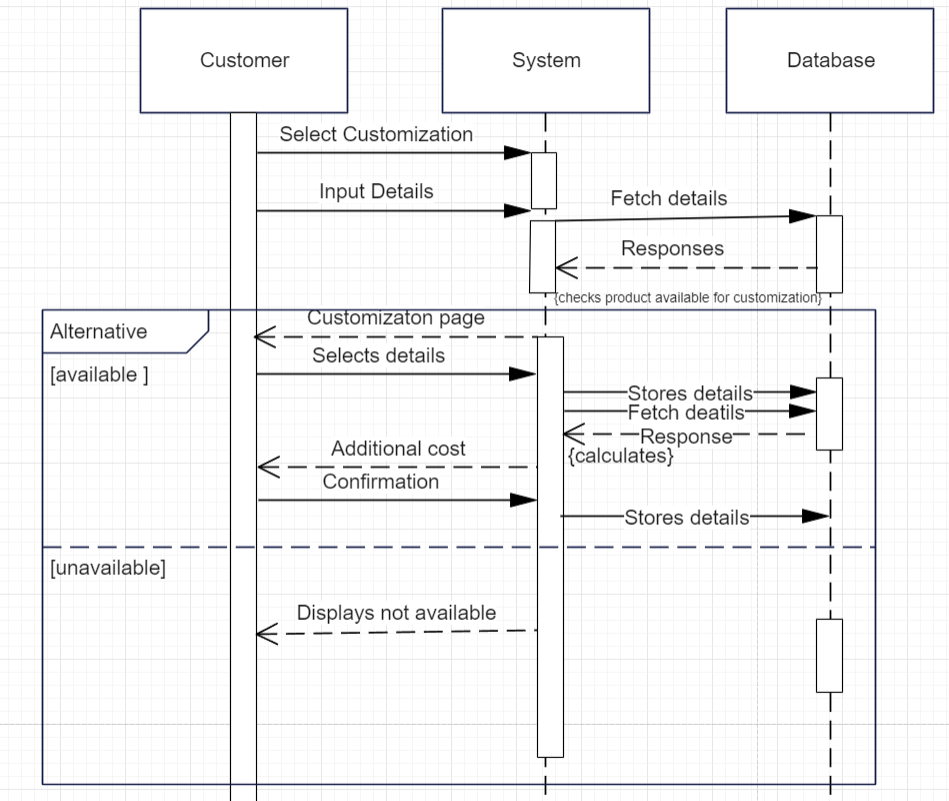
**Figure 4.30 Sequence Diagrams for Payment**



**Figure 4.31 Sequence Diagrams for Manage Product**



**Figure 4.32 Sequence Diagrams for Review**



**Figure 4.33 Sequence Diagrams for Customization**

**4.10 Activity diagram**

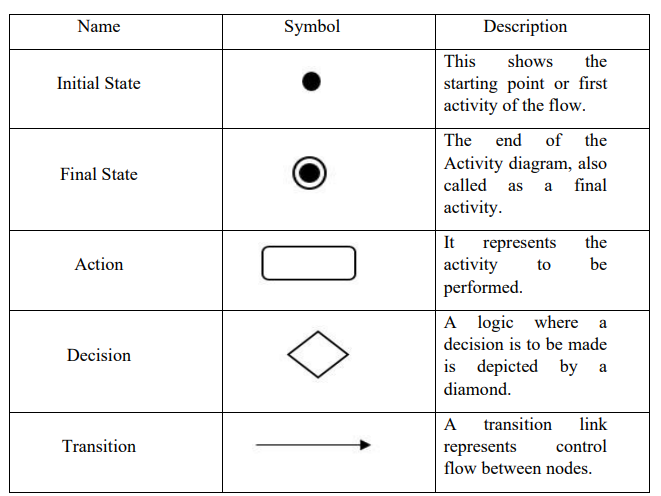
• Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.

• Activity diagram is basically a flowchart to represent the flow from one activity to another activity.

• The activity can be described as an operation of the system. The control flow is drawn from one operation to another.

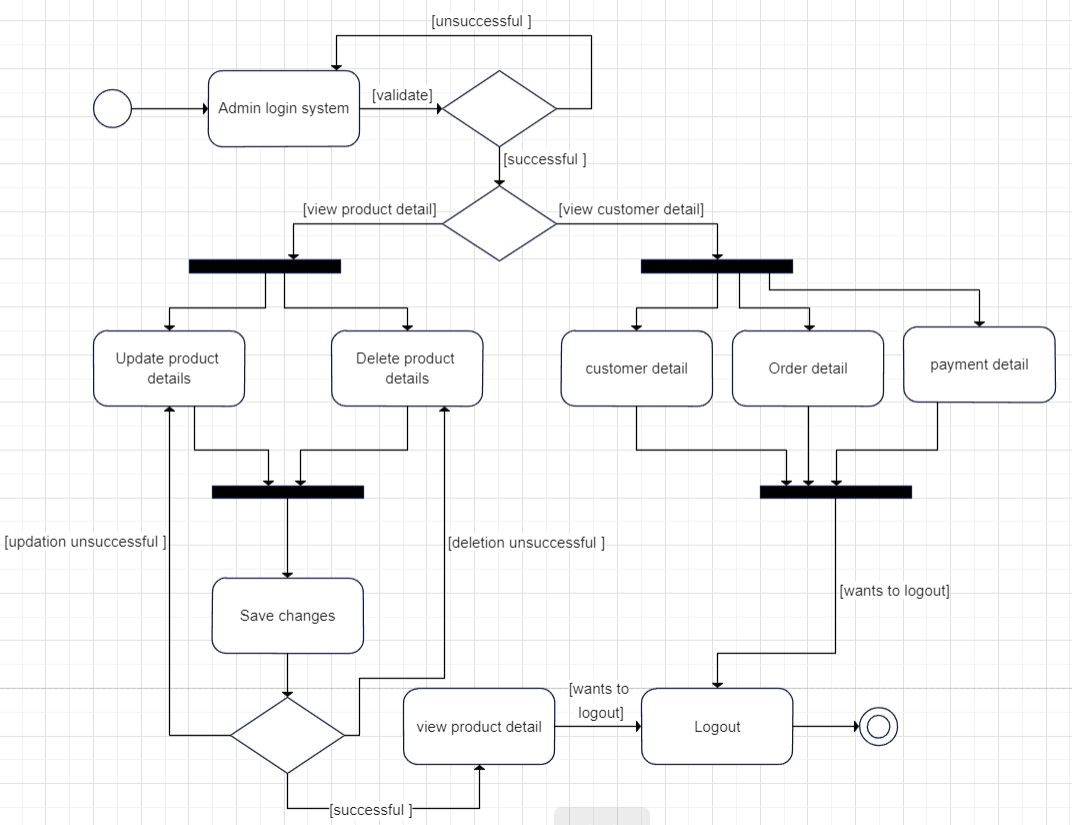
• This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

Symbol reference: https://www.lucidchart.com/

****

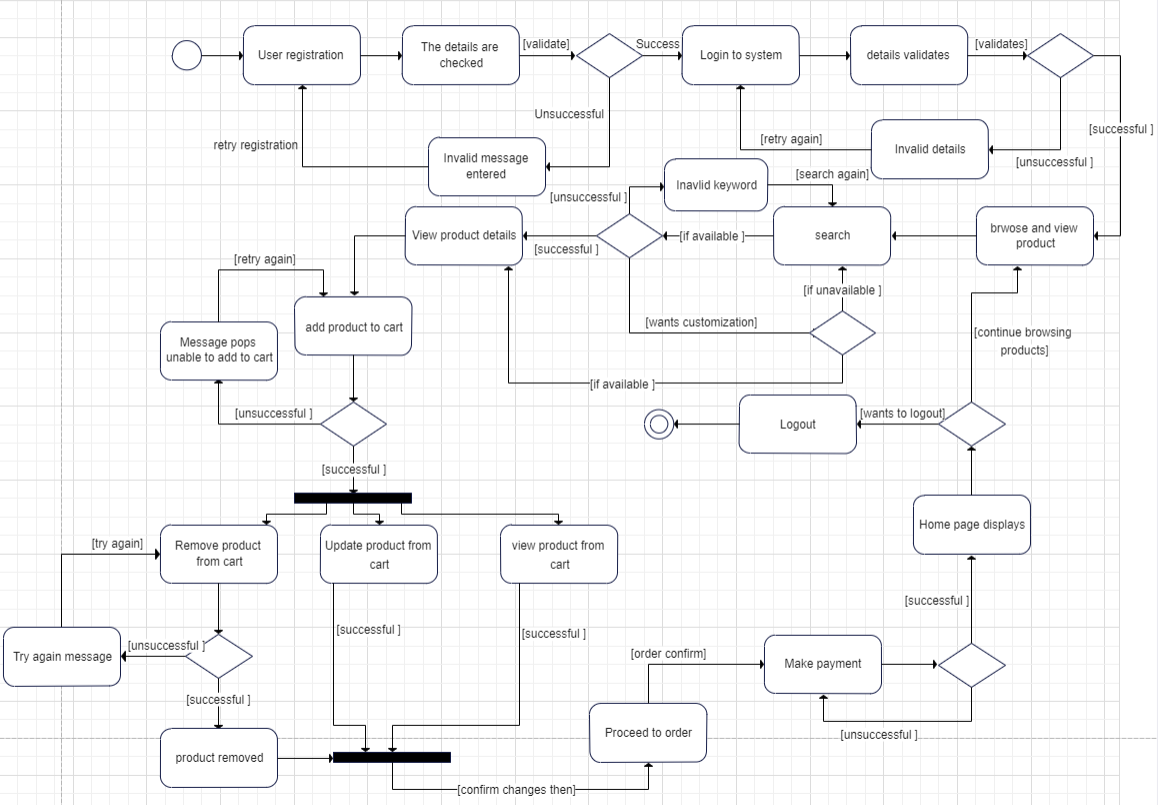
**Table 4.6 Activity diagram Symbols**

**Admin-**



**Figure 4.34 Activity diagram for admin**

**User-**

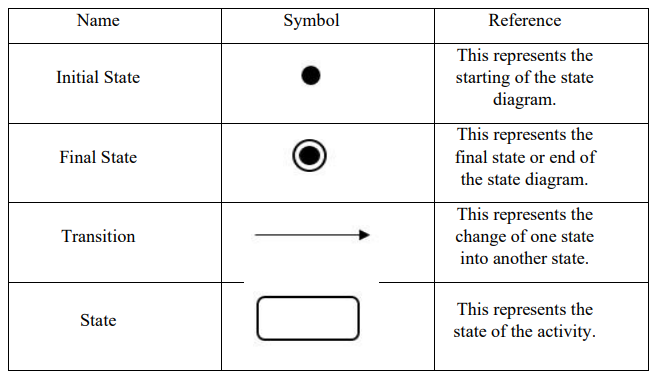


**Figure 4.35 Activity diagram for customer**

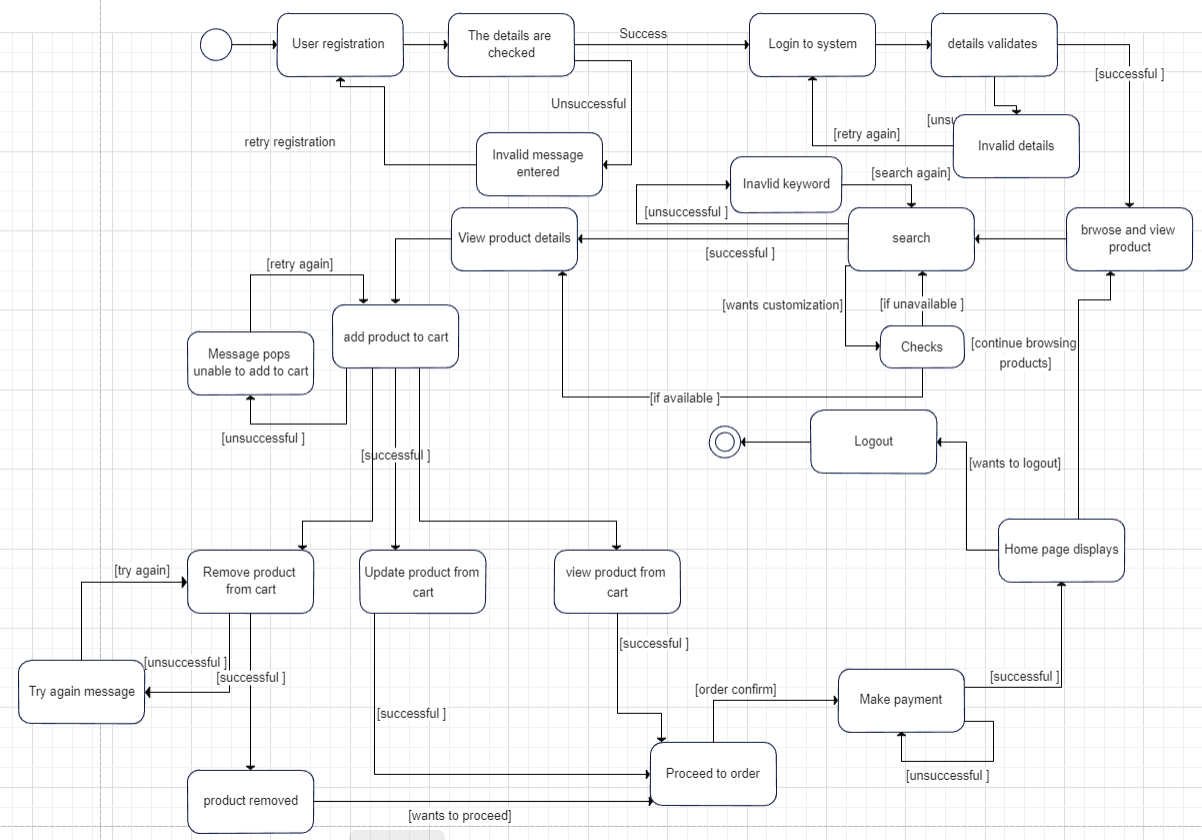
**4.11 State Diagram**

A state diagram is used to represent the condition of the system or part of the system at finite instances of time. It’s a behavioural diagram and it represents the behaviour using finite state transitions. State diagrams are also referred to as State machines and Statechart Diagrams. These terms are often used interchangeably. So simply, a state diagram is used to model the dynamic behaviour of a class in response to time and changing external stimuli.

Symbol reference: <https://www.lucidchart.com/>

****

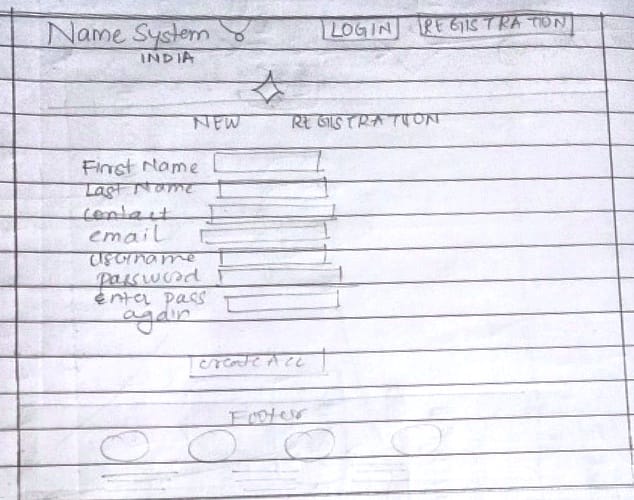
**Table 4.7 State Diagram Symbols**



**Figure 4.36 State Diagram for customer**

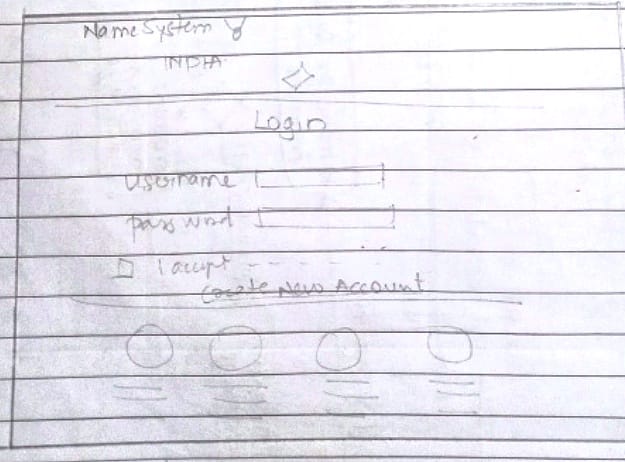
**4.12 User Interface Design**

**Registration**



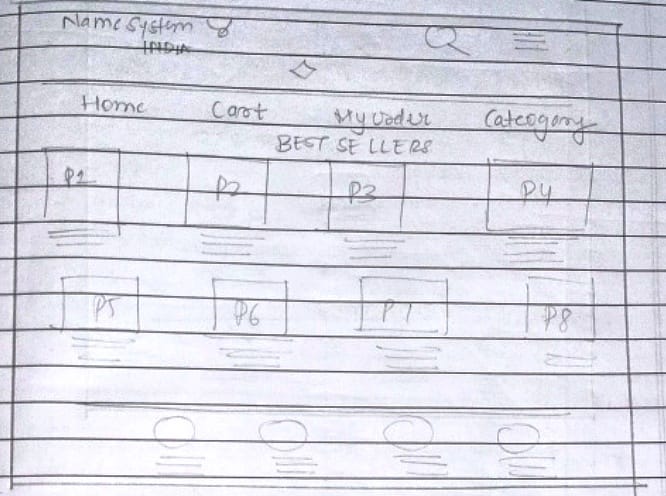
**Figure 4.37 User Interface Design Registration**

**Login**



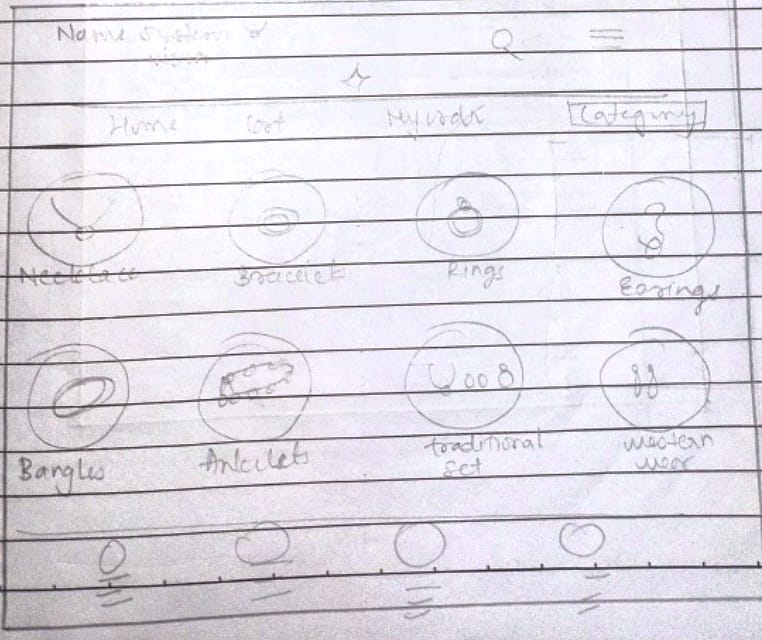
**Figure 4.38 User Interface Design Login**

**Home**



**Figure 4.39 User Interface Design home**

**Category**



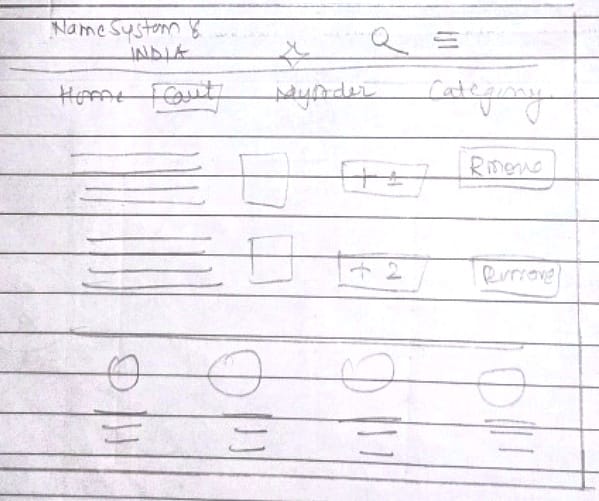
**Figure 4.40 User Interface Design Category**

**My order**



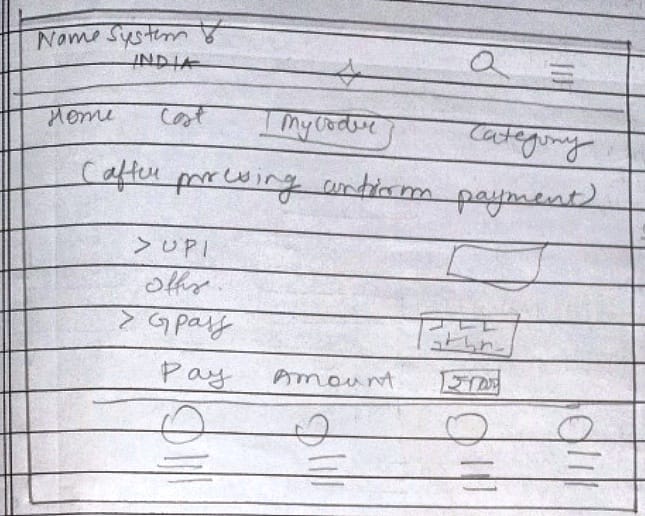
**Figure 4.41 User Interface Design Order**

**My cart**



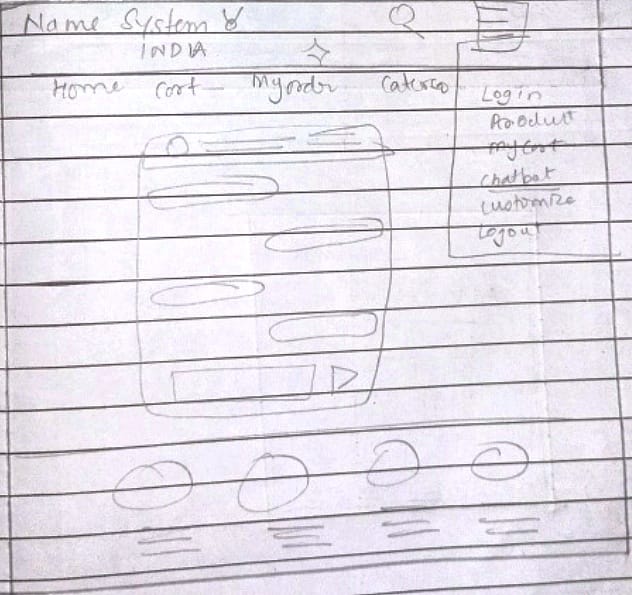
**Figure 4.42 User Interface Design cart**

**Payment**



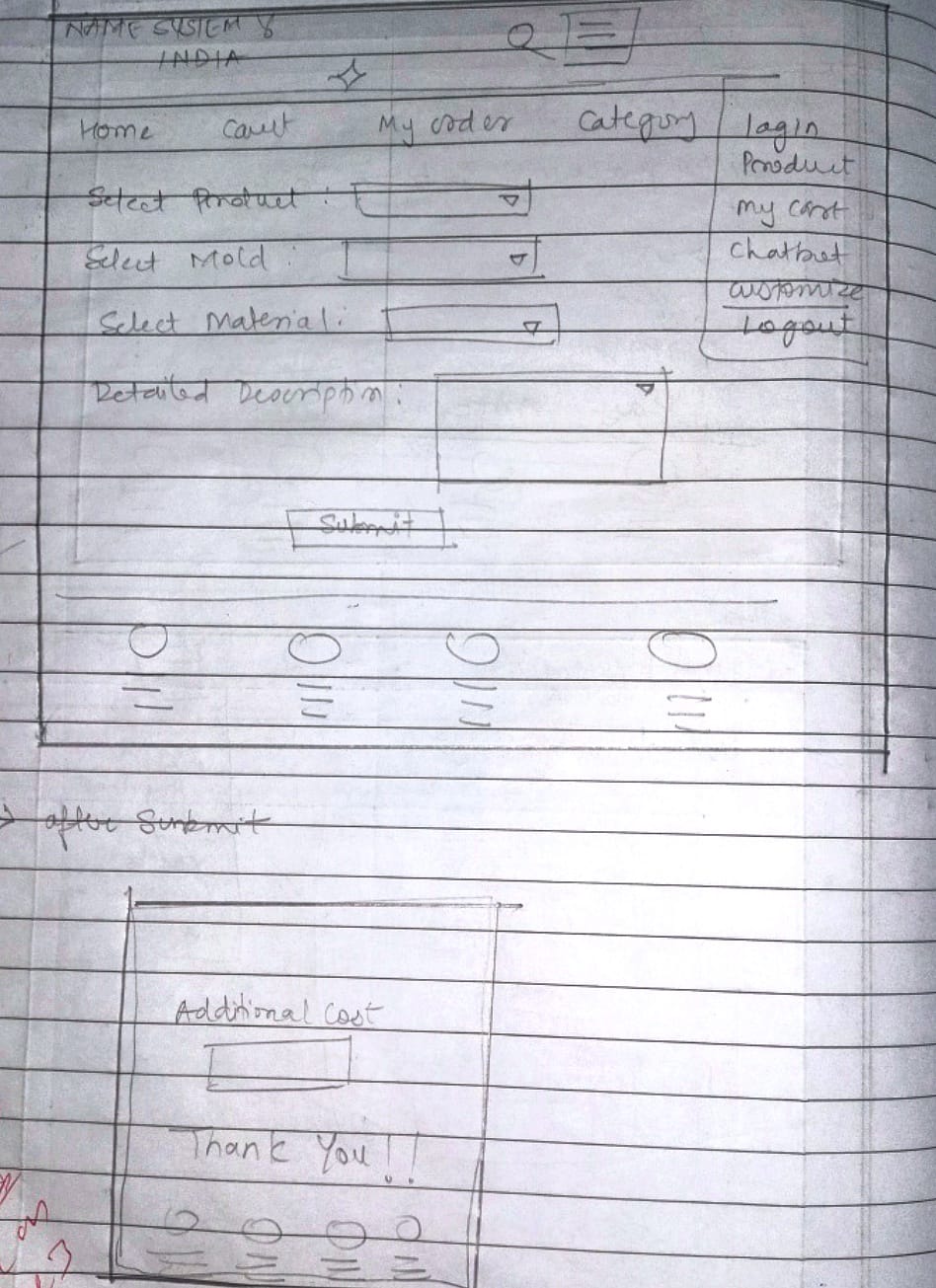
**Figure 4.43 User Interface Design Payment**

**Chatbot**



**Figure 4.44 User Interface Design chatbot**

**Customization**



**Figure 4.45 User Interface Design Customisation**

**4.13 Test Case Design**

**(1) Registration**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remark |
| 1 | Email | Abcgmail.com | Email doesn’t exist. |  |  |
| 2 | Email | Abc@gmail.com | Registers successfully. |  |  |
| 3 | Contact no | Contact: 970298042 | There must be at least 10 integers. |  |  |
| 4 | Contact no | Contact:9702980423 | Registers successfully. |  |  |
| 5 | Re-Enter Password | Password: hello02  Re-Enter Password: hello03 | Password doesn’t match. |  |  |
| 6 | Re-Enter Password | Password: hello02  Re-Enter Password: hello02 | Password matched. |  |  |
| 7 | Null value |  | Enter details for mandatory fields |  |  |

**(2) Login**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remark |
| 1 | Email | Abcgmail.com | Email doesn’t exist. |  |  |
| 2 | Login | Email: Abc@gmail.com  Password: hello02 | Login successfully. |  |  |
| 3 | Login | Email: Abc@gmail.com  Password: hello2 | Wrong email or password. |  |  |
| 4 | Login | Email: Abcgmail.com  Password: hello02 | Wrong email or password. |  |  |
| 5 | Null value |  | Enter details for mandatory fields. |  |  |

**(3) Search**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remark |
| 1 | Product name | Name: ring | Search successful. |  |  |
| 2 | Invalid Product name | Name: 888 | Enter the valid name. |  |  |
| 3 | Null value |  | Enter details for mandatory fields. |  |  |

**(4) Payment**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remark |
| 1 | Amount | Amount:500 | Payment successful. |  |  |
| 2 | Invalid Amount | Amount:0bn9 | Enter the valid amount |  |  |
| 3 | Null value |  | Enter details for mandatory fields. |  |  |

**(5) Customization**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remark |
| 1 | Customization Detail | Instead of diamonds replace it with pearls. | User can proceed with the other details of customization. |  |  |
| 2 | Null value |  | Enter details for mandatory fields. |  |  |

**(6) Address**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remark |
| 1 | Address | Room:109  Building: Virat  Landmark: Sai Nagar Kanjurmarg E  City: Mumbai  Pin code: 400078 | Address confirms |  |  |
| 3 | Null value |  | Enter details for mandatory fields |  |  |

**(7) Chatbot**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No | Test Case | Input | Expected Output | Actual Output | Remarks |
| 1 | Querry | Qs: good morningg | Enter the valid Question |  |  |
| 2 | Null value |  | Enter details for mandatory fields |  |  |

**Table 4.8 Test cases**

**References**

[1] <https://www.lucidchart.com/> 19/07/23 – 18/09/23

[2] Database System Concepts, “Henry F. Korth, Abraham Silberschatz. S.Sudarshan” McGrawHill 4th Edition

[3] The Unified Modeling Language Reference Manual, 2nd Edition, “James Rumbaugh, Ivar Jacobson, Grady Booch”