Software Design Document

(SDD)

GUI for Denim Express

Date: 02/11/2018

Contributors:

|  |  |  |
| --- | --- | --- |
| Name | Enroll-Number | Section |
| Syed Tasleem | U101116FCS141 | S-6 |
| 1. Venkat | U101116FCS148 | S-6 |
| Tavva G N R S N Prudhvith | U101116FCS142 | S-6 |
| V. Jashwanth | U101116FCS151 | S-6 |

Mentors:

*Mr. Amit Kumar Sir.*

*Mr. Chaitanya Surabhi.*

**Table Of Contents**

**1. Introduction ..................................................................................** Error! Bookmark not defined.

1.1 Purpose of this document ................................................................. **Error! Bookmark not defined.**

1.2 Scope of the development project .................................................... **Error! Bookmark not defined.**

1.3 Definitions, acronyms, and abbreviations ........................................ **Error! Bookmark not defined.**

1.4 References ........................................................................................ **Error! Bookmark not defined.**

1.5 Overview of document ..................................................................... **Error! Bookmark not defined.**

**2. Conceptual Architecture/Architecture Diagram ......................** Error! Bookmark not defined.

2.2 Structure and relationships ............................................................... **Error! Bookmark not defined.**

2.3 User interface issues ......................................................................... **Error! Bookmark not defined.**

**3. Logical Architecture (Class Diagram, Sequence Diagram, State Diagram)**

**3.1 Logical Architecture Description .............................................**

**3.1.1 Class Diagram Explanation .............................................**

**3.1.2 Sequence Diagram Explanation .............................................**

3.1.2.1 Class name: About ........................................................................... **defined.**

3.1.2.2 Class Name: Login.............................................................. **not defined.**

3.1.2.3 Class Name: Registration ............................................................. **not defined.**

3.1.2.4 Class Name: FAQ.................................................................. **defined.**

3.1.2.5 Class Name: Shopping Cart .......................................................... **not defined.**

3.1.2.6 Class Name: Search....................................................................... **defined.**

3.1.2.7 Class Name: Orders......................................................... **not defined.**

3.1.2.8 Class Name: Discounts.................................................................. **defined.**

3.1.2.9 Class Name: CancelOrder......................................................... **defined**

3.1.2.10 Class Name: ReturnOrder.................................................................. **defined.**

**3.1.3 State Diagram Explanation .............................................**

*3.1.3.1 Login -----------------------------------------------------------not defined*

*3.1.3.2 Registration-----------------------------------------------------------not defined*

*3.1.3.3 View Items-----------------------------------------------------------not defined*

*3.1.3.4 Contact Us-----------------------------------------------------------not defined*

*3.1.3.5 Wishlist-----------------------------------------------------------not defined*

*3.1.3.6 Cart Manager-----------------------------------------------------------not defined*

*3.1.3.7 Make purchase-----------------------------------------------------------not defined*

*3.1.3.8 Search-----------------------------------------------------------not defined*

**4.0 Execution Architecture ..............................................................** Error! Bookmark not defined.

4.1 Reuse and relationships to other products ........................................ **Error! Bookmark not defined.**

**5.0 Design decisions and tradeoffs ..................................................** Error! Bookmark not defined.

**7.0 Appendices -------------------------------------------------------------------------------Error!Bookmark not defined.**

***1. Introduction***

*This Software Design Document (SDD) describes the detailed structure of the components of the GMU and the precise implementation details required to satisfy the requirements as specified in the Software Requirements Specification (SRS). Within the Document are narrative and graphical documentation of the software design for the project including class diagrams, sequence diagrams, state diagrams and other supporting requirement information.*

* 1. ***Purpose***

*The purpose of this project is to create a website for*

*people to be able to both buy goods. Users of this website will be able to create an account which will supply them with an easy to use interface in order to both purchase items listed by other users and sell items to other users.*

* 1. ***Project Scope***

*We will define what will be done such that the final product meets expectations.*

*• Secure registration and profile management facilities for customers*

*• Adequate searching mechanisms for easy and quick access to particular products and services*

*• We will have a shopping cart in which users can save their products and checkout later.*

*• We have a feedback mechanism for the products from the customers. Feedback can also be given on a particular individual or vendor.*

*• The hardware and software should be robust and will be tested extensively to provide users a good experience.*

* 1. ***Definitions, acronyms, and abbreviations***

*GMU – Garment Manufacturing Unit*

*SDS – Software Design Specification*

* 1. ***References***

**1.4.1** *R. S. Pressman, Software Engineering: A Practioner’s Approach, 5th Ed, McGraw-Hill, 2001.*

**1.4.2** *IEEE SDS template*

**1.5 *Overview Of Document***

*It is divided into 2 sections with various sub sections. The sections are:*

***1.Introduction****: This phase discusses about the purpose, scope and some definitions and abbreviations used in this document.*

2. **Conceptual Architecture**/**Architecture Diagram:** describes the overview of components, modules, structure and relationships and user interface issues.

***3.Logical Architecture****: This phase discusses about the Logical Architecture Description and Components.*

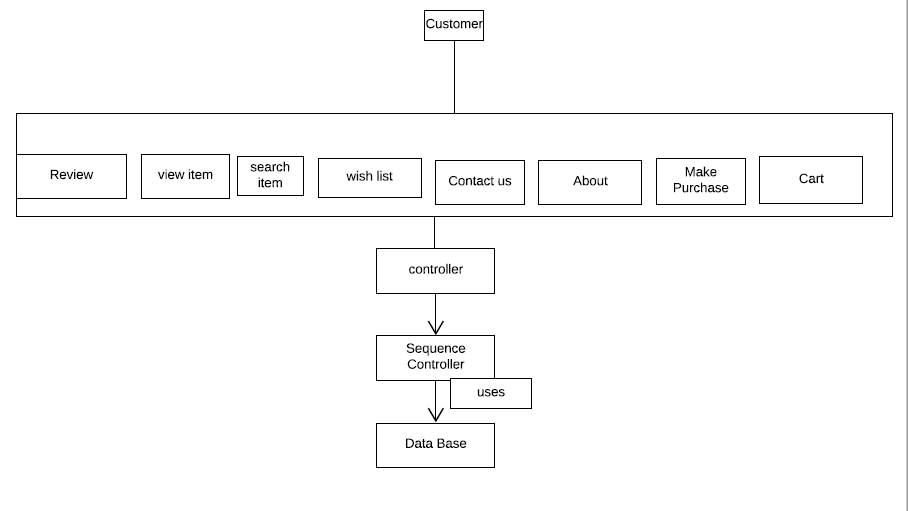
*4.* ***Execution Architecture:*** *defines the runtime environment, processes, deployment view.*

*5.* ***Design Decisions and Trade-offs:*** *describes the decisions taken along with the reason as to why they were chosen over other alternatives.*

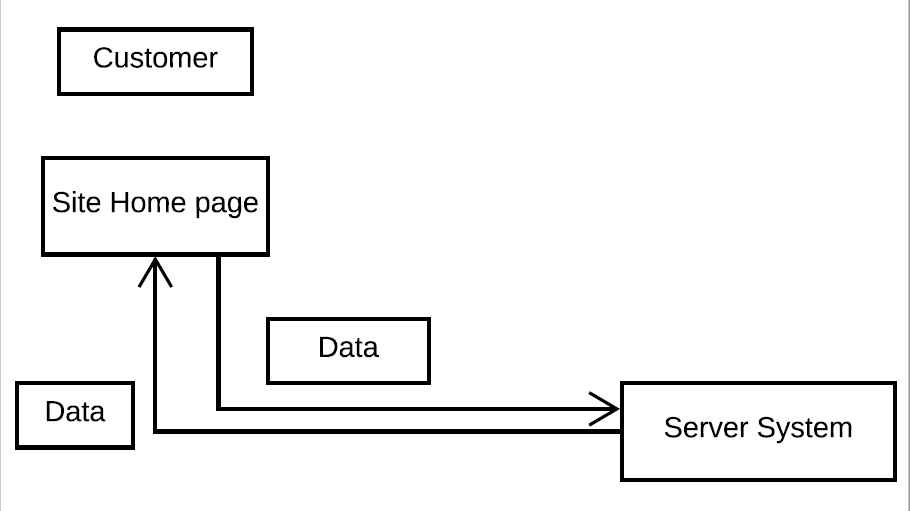
***7. Appendices:*** *describes subsidiary matter if any.*

**2.0 Conceptual Architecture/Architecture Diagram**

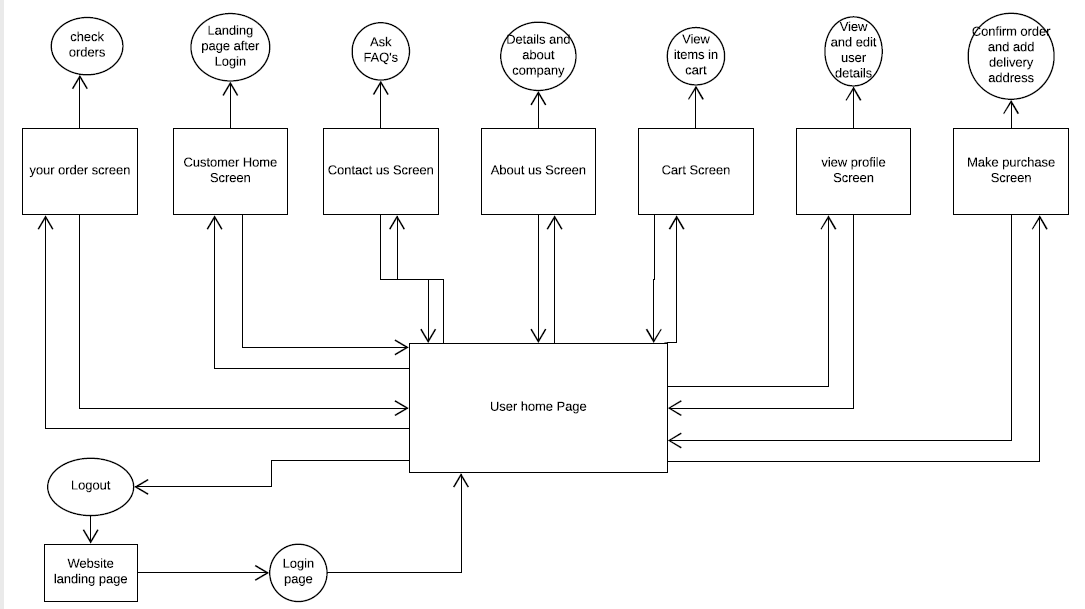
*Architecture Diagram 1*

**

*Architecture Diagram 2*

**

*2.2 Structure and Relationships*

**

*2.3 User Interface issues*

\* **Quality issues**

*The biggest problem while buying things online is that you have no guarantee of a product's quality. Reviews are not always true and all the research can't guarantee you the product's quality; fraud sellers who mislead customers to increase sales are the prime reason for faulty products being sold online. It can be quite difficult for them to conduct quality checks on each and every one of the products they're selling. Additionally, the issue of getting the correct size remains a serious drawback for buying clothing online. Sizes vary from brand to brand, and since you can't try out the products before buying them, selecting the size is always a gamble.*

*\** ***Delivery and logistics***

*One issue that constantly turns up while shopping online is when the order will be delivered. Delivery personnel often turn up at our homes when we're at work or out somewhere as there's no way to fix a particular time slot for the delivery to take place. This same issue exists while returning products. Another problem is that the vast majority of the Indian population which lives in rural areas and Tier-III cities is unable to shop online because not all they provide delivery services to their locations.*

*\** ***Additional charges***

*How many times has it happened that you've spotted a great deal on a product and when you're one click away from purchasing it you noticed an additional shipping charge. This is common place on all e-commerce sites when your order amount isn't high enough to qualify for free shipping. And even when it is, sometimes these shipping charges are added on each individual product and not the collective order.*

*\** ***Unclear return and guarantee policies***

*Since you have no idea of product's quality until you hold it in your hands, returning things bought online is quite common. Unless you're buying from one of the established e-commerce companies, it's important to go through the return policy while making a purchase. But most sites have vague return policies that can leave you with a low-quality product and no way to return it. The same applies for guarantees, as most sites don't clearly mention what the policy is for a product and then refuse to carry out replacements if you receive a damaged product.*

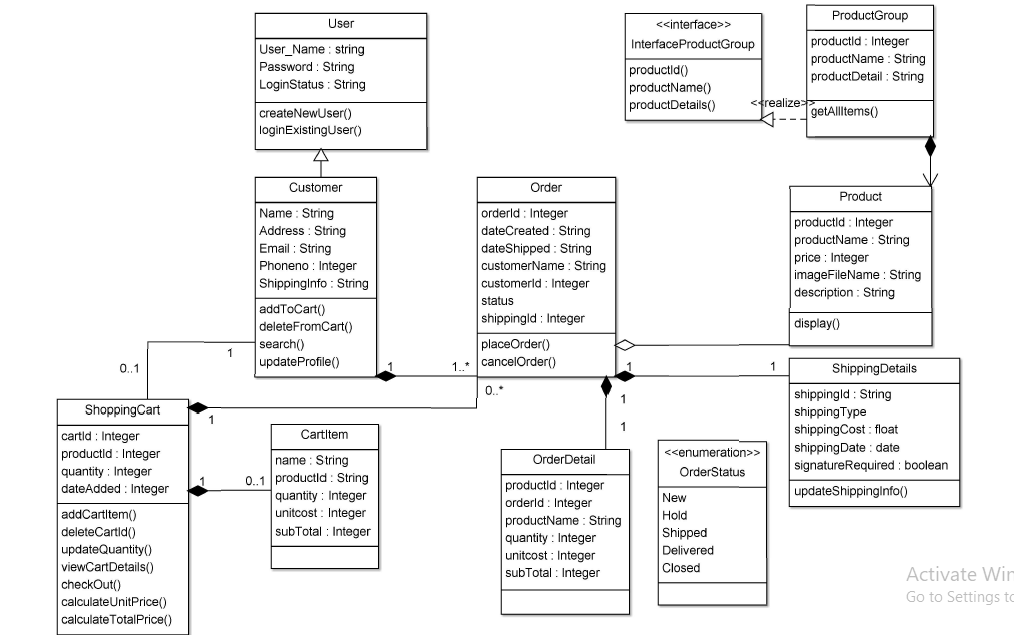
*\** ***Lack of security***

*The lack of security, is a major problem on the internet today. E-commerce sites record important customer data like name, phone number, address, and bank details. If these sites don't implement stringent cyber security measures, your data is at risk of falling into the wrong hands who can then wreak havoc on your bank account. Most of the big players in online shopping certainly have the best-in-class security measures to protect their customers' details, but the same can't be said about the countless smaller sites who may not have the expertise to do so.*

*Online shopping is far too convenient to get hindered by these problems. But if e-commerce sites can fix these issues, they will certainly improve customer experience and hence generate more sales.*

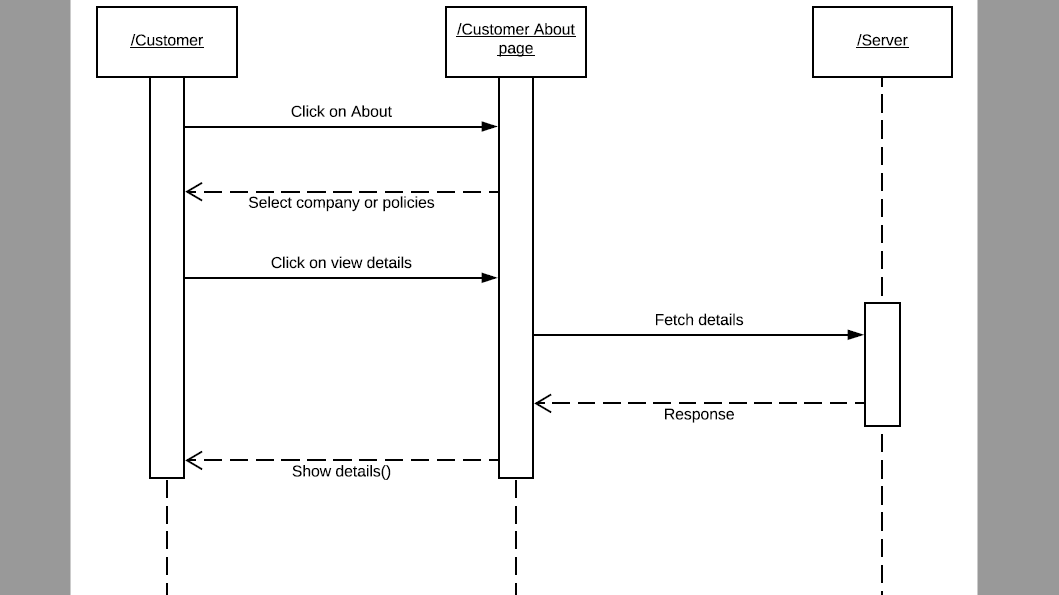
***3.0 Logical Architecture (Class Diagram, Sequence Diagram, State Diagram)***

**Class Diagram:**

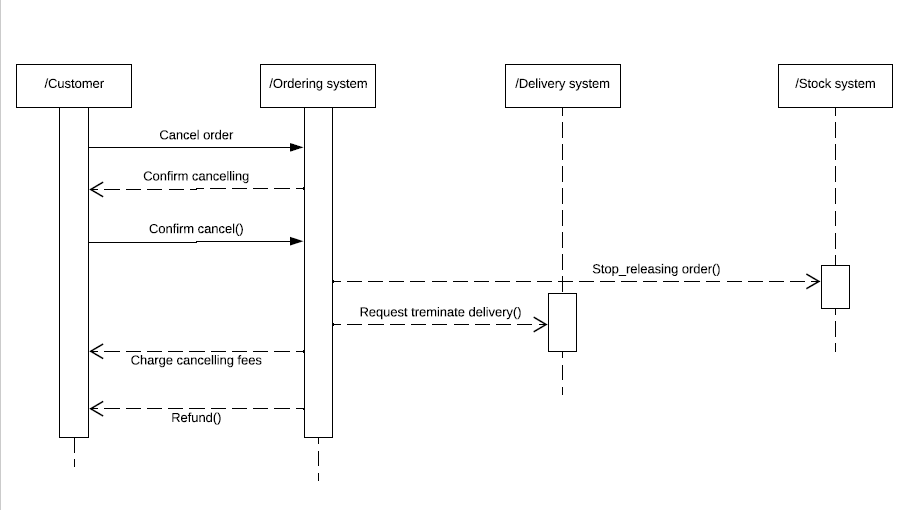
******

***Sequence Diagrams:***

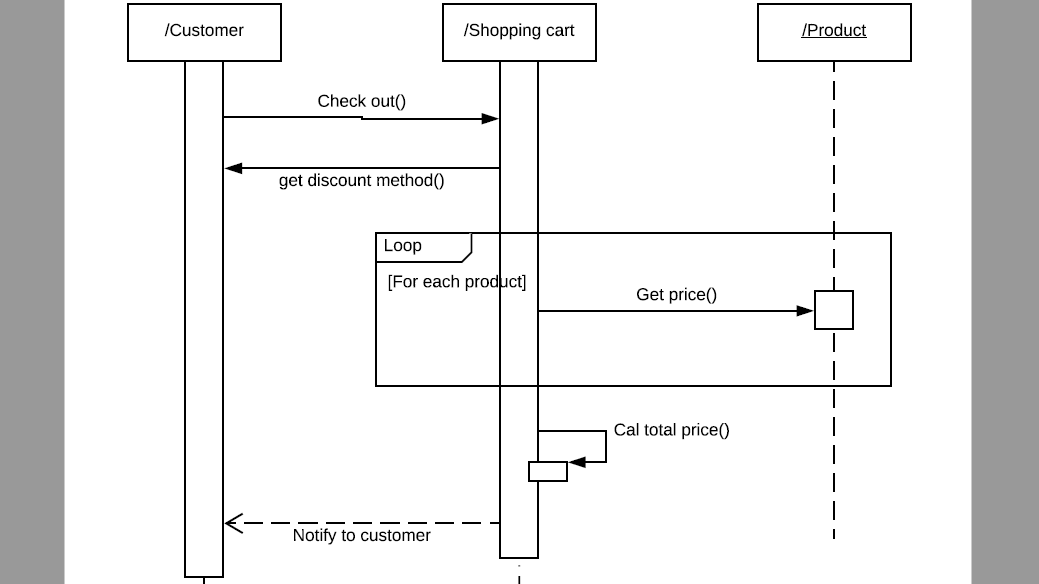
**Sequence Diagram: About Page**

******

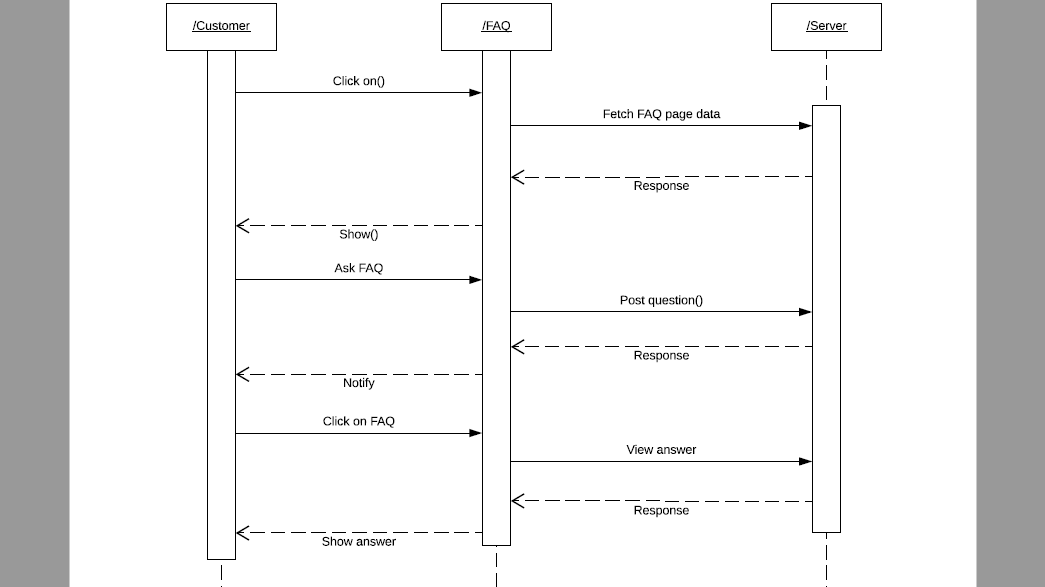
**Sequence Diagram: Cancelling Order**

******

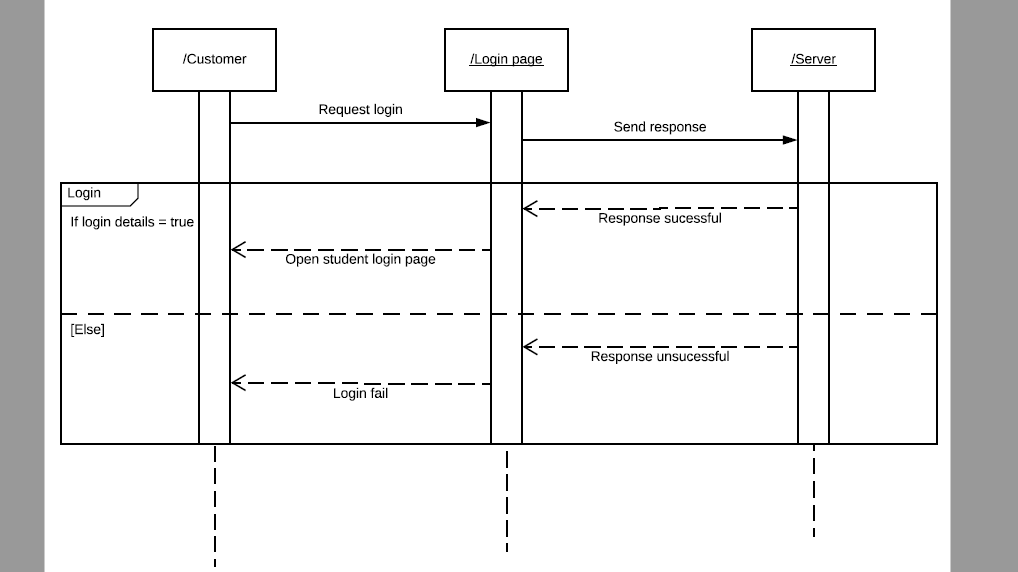
**Sequence Diagram: Discounts**

******

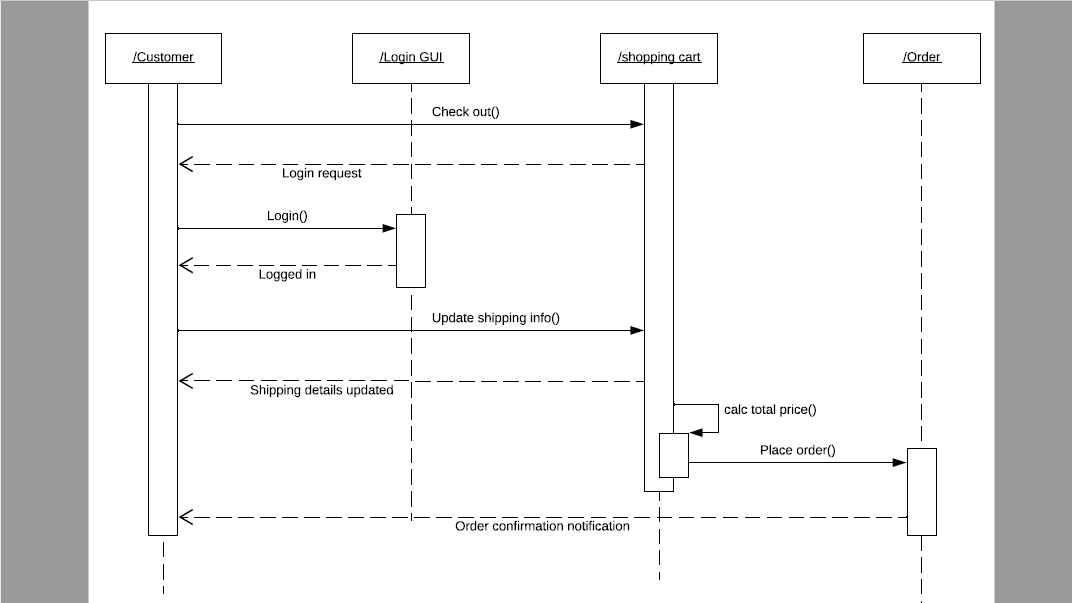
**Sequence Diagram: FAQ**

******

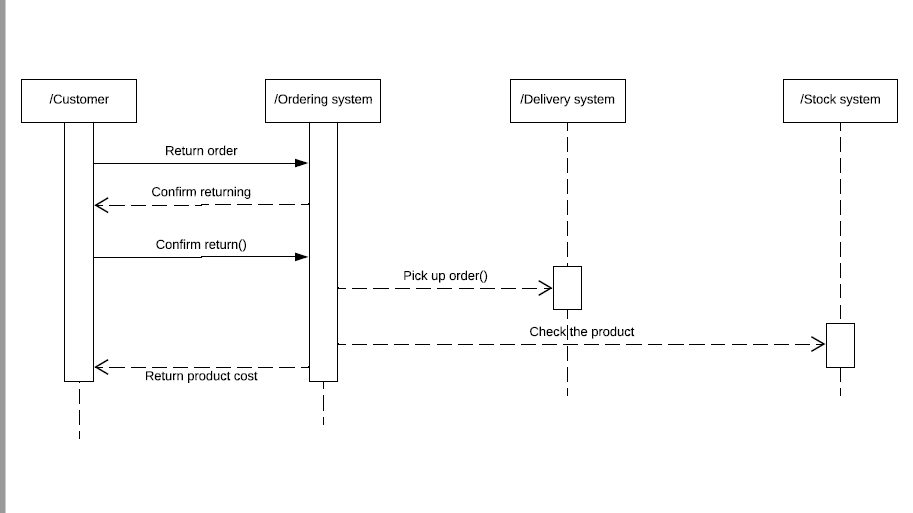
**Sequence Diagram: Login**

******

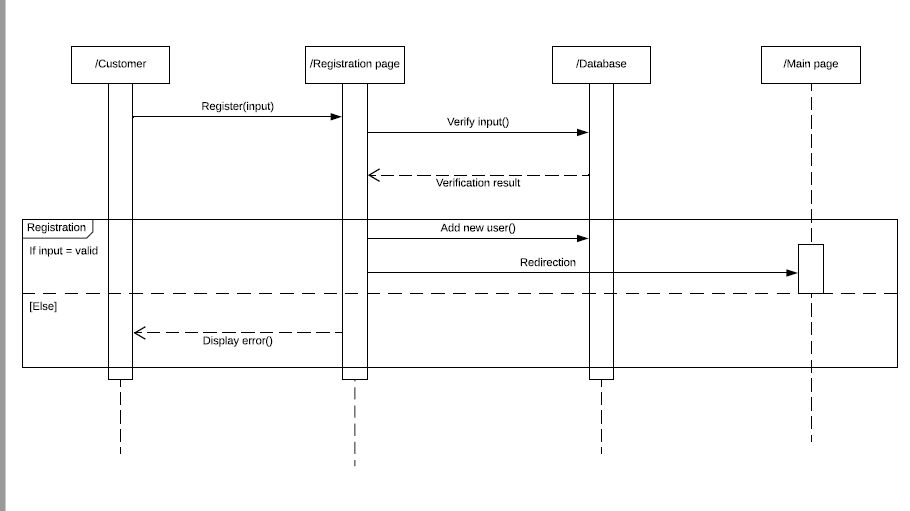
**Sequence Diagram: Order**

******

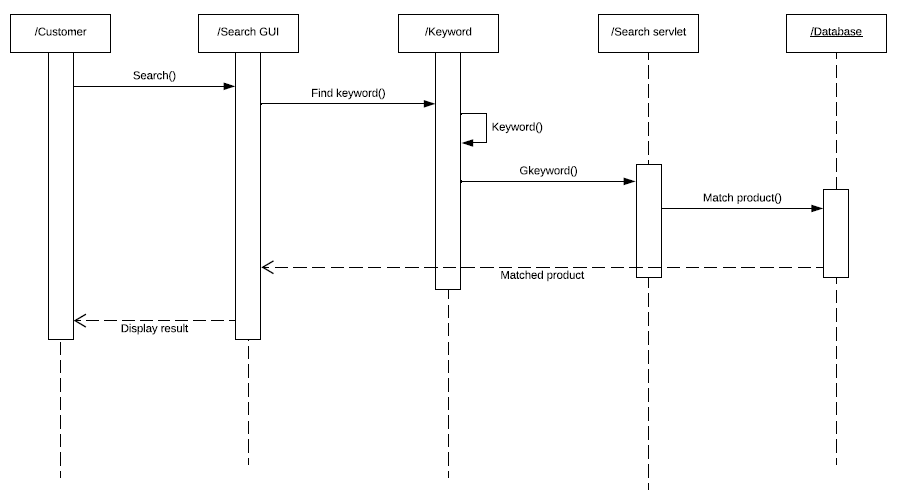
**Sequence Diagram: Return Order**

******

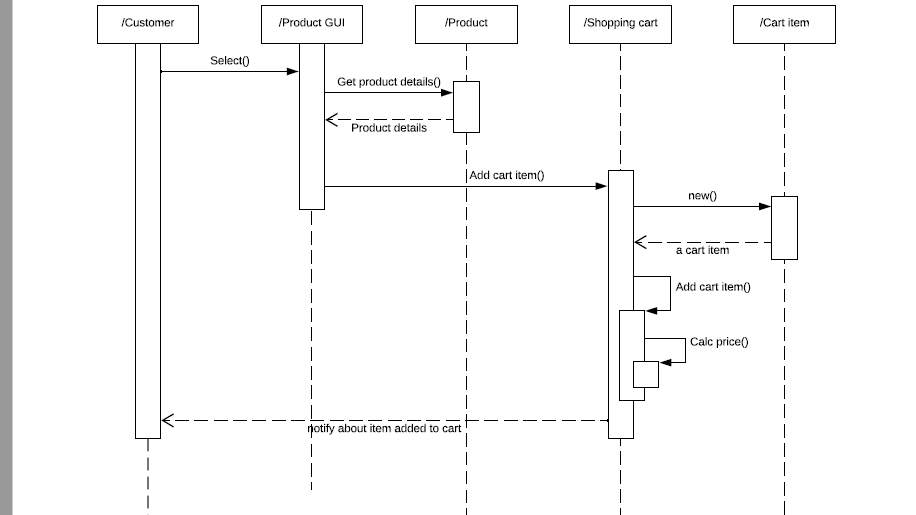
**Sequence Diagram: Registration**

******

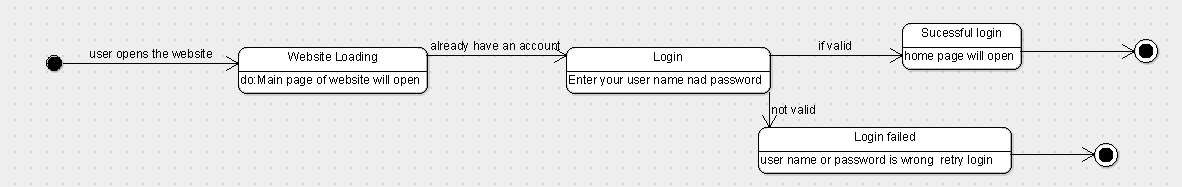
**Sequence Diagram: Search Engine**

******

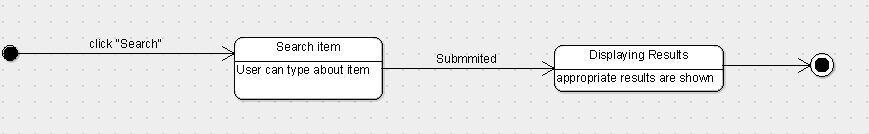
***Sequence Diagram: Shopping Cart***

******

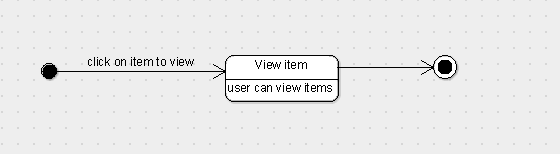
***State Diagrams:***

**State Diagram: Login****

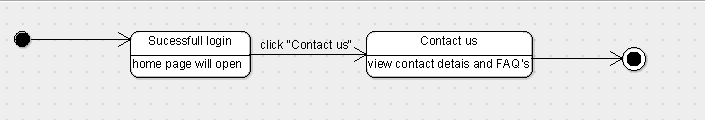
**State Diagram: Search**

******

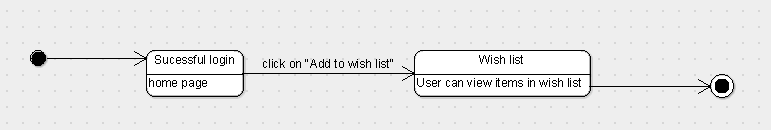
**State Diagram: View Items**

******

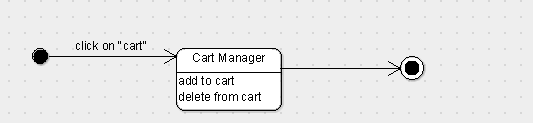
**State Diagram: Contact Us**

******

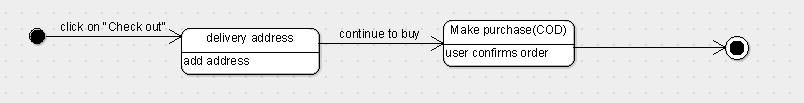
**State Diagram: Wishlist**

******

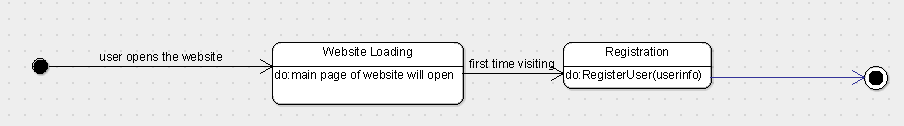
**State Diagram: Cart Manager**

******

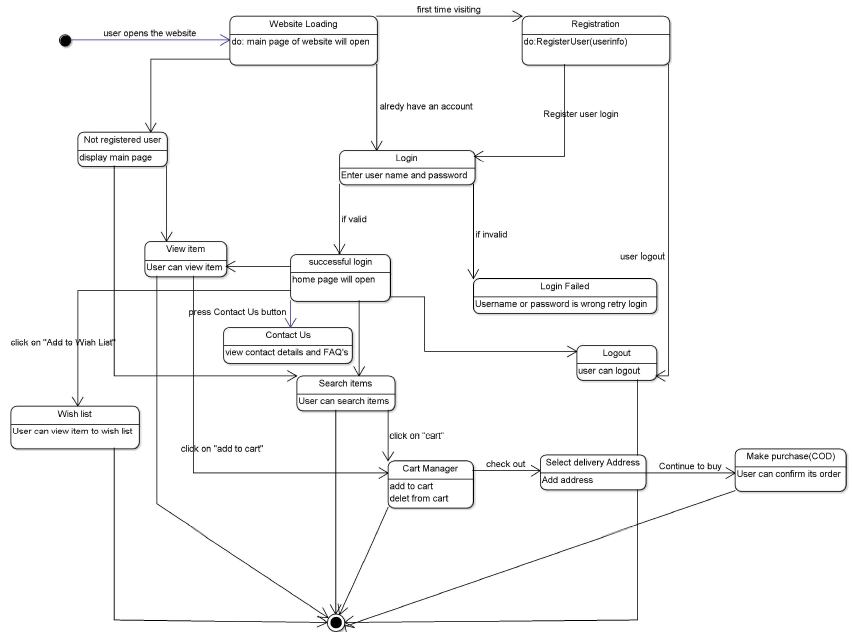
**State Diagram: Make Purchase**

******

**State Diagram: Registration**

******

**State Diagram**

******

***3.1 Logical Architecture Description***

**3.1.1 Class Diagram Explanation**

*Customer class is extending User class which is being shown* ***by Inheritance*** *(IS-A relationship).*

***Associations*** *are relationships between classes and displayed as a solid line connecting two classes. Here, shopping cart and customer share an association between them.*

***Aggregations*** *is a special type of relationship which represents a “part of” relationship and displayed as a solid line with an unfilled diamond. The classes that represent aggregation is between Product and Order class.*

***Composition*** *is a special type of relationship which says “where parts are destroyed when the whole is destroyed”. The classes that represent composition are:*

1. *Order and Shipping Details*
2. *Order and Order details.*
3. *Customer and Order.*
4. *Shopping Cart and Order.*
5. *Shopping Cart and Cart Item.*

***Realization*** *is a special type of relationship between the interface and it’s implementing class. The classes that represent realization is Interface Product Group and Product Group.*

**3.1.2 Sequence Diagram Explanation**

*In the sequence diagram, there are 4 arrow lines which represent communication between objects.*

1. *Synchronous Message using a line with a solid arrowhead pointing from one object to another.*
2. *Asynchronous Message are drawn with an arrow connecting two lifelines.*
3. *Return message is drawn with a dotted line and an open arrowhead pointing back to the original lifeline.*
4. *Self Message shown as U-shaped arrow pointing back to itself.*

**3.1.2.1 About Page**

*This Page will display about the major contributions of the brands and about the Denim Xpress, which later can be viewed by the user.*

**3.1.2.2 Login**

*After Registration, this page allows customers to login with Username and Password to purchase or view items in the GMU. This process is used to login into the Website.*

**3.1.2.3 Registration**

*Users who want to be converted into store owners will use this application form to register and to buy the item user has to be a registered customer. This process is used to register a customer.*

**3.1.2.4 FAQ Page**

*Customer posts question on this page, which later can be viewed by all other customers and can be answered by the Admin. On being answered by Admin, the customer also gets notification for the same.*

**3.1.2.5 Shopping Cart**

*Here, customer can add the items they are interested in and purchase them through COD. Customer can also update and delete the cart and if the user purchases the item the items in the cart will automatically be erased.*

**3.1.2.6 Search**

*Any user (Registered or Unregistered) can search the item. User can search the item by name or brand.*

**3.1.2.7 Order**

*A Registered customer can buy an item using this process. This process includes search item, validate customer, add to cart, etc.*

**3.1.2.8 Discounts**

*User will get some offers and discounts on the items on special occasions, on this day user will get the items in a less cost than the present market.*

**3.1.2.9 Cancel Order**

*User can cancel the orders by visiting the Your Orders section in your account. If the user plans to cancel multiple items, select the checkbox next to each item you wish to remove from the order.*

**3.1.3 State Diagram**

***3.1.3.1 Login***

*Allows user to enter credentials, which are being checked for authentication in the back-end. After being authenticated successfully, it lands up on home page. It has various buttons like FAQ, ViewProfile, Add to cart, Contact Us, and About to land up on their respective pages, and click on Back to return to home page. On clicking on Logout, it returns to the parent page.*

***3.1.3.2 Registration***

*Users who want to be converted into store owners will use this application form to register and to buy the item user has to be a registered customer. This process is used to register a customer.*

***3.1.3.3 View Items***

*After successful login, user can view items in the home page and based on his interest he can purchase or store them in a cart or wish list.*

***3.1.3.4 Contact Us***

*For any emergency issues or any queries regarding the products*

*Or return and cancel the items user can view the contact us page in which he can contact through email.*

***3.1.3.5 Wishlist***

*On opening this page user should first create a list to save time or to give great gifts on special occasions to their loved ones or to check price changes.*

***3.1.3.6 Cart Manager***

*Here, user can manage the cart after adding items into it like Adding, updating or deleting the cart.*

***3.1.3.7 Make purchase***

*After adding to the cart user will have a checkout option to purchase the items through COD.*

***3.1.3.8 Search***

*User will have this option in the main page so that when he/she will have some options in the search bar like men/women/kid’s, etc. This improves the efficiency to easily navigate the product.*

**3.2 Class name: User**

**Description:** *The user class is used to store the details of all the users. It is used to create a new user or help an existing user to login.*

*login.*

***3.2.1 Method 1: createNewUser()***

*Input: OnClickListener on createNewUser button.*

*Output: handle the details of all the users.*

***Method Description:***

*This method stores all the credentials of the users to enter into the website and also it helps in creating a new user or helps an existing user.*

*There are two types of users: registered and unregistered users. Every user can place an order or search for a product.*

***3.2.2 Method 2: loginExistingUser()***

*Input: Enter Username and Password*

*Output: handle authentication*

***Method Description:***

*This method helps the existing user to login into the website with proper credentials. When username and password are equal then this method is invoked.*

**3.3 Class name: Customer**

**Description:**

*After being authenticated successfully, it lands up on home page. It has various buttons like FAQ, ViewProfile, Add to cart, Contact Us, and About to land up on their respective pages.*

**3.3.1 Method 1: addToCart()**

*Input: OnClickListener on addToCart button.*

*Output: shows all items of the customer.*

***Method Description:***

*This method allows the user to add items into the cart and purchase items from the cart if he is interested.*

**3.3.2 Method 2: deleteFromCart()**

*Input: OnClickListener on deleteFromCart button.*

*Output: shows the updated cart when some items are deleted.*

***Method Description:***

*This method allows the user to delete the items from the cart when the user purchases or when he is not interested in the product after some days.*

**3.3.3 Method 3: search()**

*Input: OnClickListener on Search button*

*Output: shows the content with respect to user message.*

***Method Description:***

*User will have this option in the main page so that when he/she will have some options in the search bar like men/women/kid’s, etc. This improves the efficiency to easily navigate the product.*

**3.3.4 Method 4: updateProfile()**

*Input: OnClickListener on Update profile button*

*Output: shows the updated profile to the user.*

***Method Description:***

*When the user click on this button then it will invoke setOnClicklistener method which will update the profile of the user.*

**3.3.5 Method 5: viewItems()**

*Input: OnClickListener on View items button*

*Output: shows all the items*

***Method Description:***

*When the user click on view items button then it will invoke setOnClicklistener method which will view all the items that are fetched from the server.*

**3.4 Class name: Shopping Cart**

**Description:**

*After being authenticated user can add items into the cart to purchase products or save it for future purpose.*

***3.4.1 Method 1: viewCartDetails()***

*Input: OnClickListener on View card details button.*

*Output: shows all the details of the items present in the cart.*

***Method Description:***

*This method allows the user to view the cart details fetched from the server and display to the user about the items in the cart.*

***3.4.2 Method 2: checkout()***

*Input: OnClickListener on checkout button*

*Output: takes the webpage to payment page to checkout.*

***Method Description:***

*When user clicks this button it will invoke another method which will display a webpage to purchase the items through Cash On Delivery.*

***3.4.3 Method 3: calculateUnitCost()***

*Input: OnClickListener on calculate Unit cost button*

*Output: shows the price of individual cost of each item.*

***Method Description:***

*This method will display the individual unit price of each item present in the cart. The data is fetched from the server.*

***3.4.4 Method 4: calculateTotalPrice()***

*Input: OnClickListener on calculate total price button*

*Output: shows the price of total cost of each item.*

***Method Description:***

*This method will display the total price of each item present in the cart. The data is fetched from the server as he need to take the input of all the quantities of each item.*

**3.5 Class name: Order**

**Description:**

*The order class contains the details of the placed order. This class is used to place or delete an order. The order class is notified when an order is placed or deleted by a user.*

**3.5.1 Method 1: placeOrder()**

*Input: OnClickListener on place order button.*

*Output: A New webpage will open to place the order.*

***Method Description:***

*This method will place the order from the cart to purchase the item through COD. After placing the order it will ask for shipping details.*

**3.5.2: Method 2:cancelOrder()**

*Input: OnClickListener on cancel order button.*

*Output: A New webpage with terms and conditions to cancel the order.*

***Method Description:***

*User can cancel the orders by visiting the Your Orders section in your account. If the user plans to cancel multiple items, select the checkbox next to each item you wish to remove from the order.*

**3.6 Class name: Shipping Details**

***Description:***

*After checkout user should type the shipping details along with shipping policy and prices which depend upon international Shipping and special shipping policies.*

**3.6.1 Method 1: updateShippingInfo()**

*Input: None.*

*Output: It will take the data from the server and track the shipping information time to time.*

***Method Description:***

*Here, user can track the order through the Track Shipping so that user will get an guaranteed accelerated delivery and delivering photos.*

**3.7 Class name: Product**

***Description:***

*The product class contains the details of the products in*

*the online shopping mall. This class will be notified when*

*the user searches for a product or the admin adds or*

*deletes a product.*

***3.7.1Method1: displayDetails()***

*Input: OnClickListener on display details button.*

*Output: It will take the data from the server and display all the details regarding the product.*

***Method Description:***

*This method will display the information regarding the product with respect to id, name, price and images of the products.*

**3.8 Class Name: StudentFAQ**

**Class Description:**

*This class defines all the methods from the student side to be used to* ***view the questions and answers*** *and* ***post a question****.*

*This class consists of 2 methods:*

***1. private void onCreate()***

***2. private void askQuestion()***

**3.8.1 Method** 1**: private void onCreate()**

***Input: None***

***Output****: This method will display the questions asked till now and the answers in a List View.*

***Method Description*** *: Method responsible for displaying the questions and answers.*

**3.8.2 Method** 2**: private void askQuestion()**

***Input:*** *The question string asked by the student.*

***Output:*** *The question asked is sent to the server.*

***Method Description:*** *to send a question asked by the student to the server at the backend.*

***3.9 Class Name: Notification***

***Description:***

*This class basically handles notification. When the student logs in to the app, the notification tab will display the new activity reports in the form of notification badges.*

*This class also takes care of the Notifications push from the Admins side, i.e. in case Admin require for any document from the student or any important communication needs to be conveyed to the student, admin can push notification and this will notify the concerned person.*

***3.9.1 Method 1: Notification()***

***Input:*** *mAuthListener*

***Output:*** *handle authentication state changes*

***Method Description:***

*When activity start getting visible to user then onStart() will is called. Inside this method parent onStart is invoked. Since it is a Login activity so several state may occur like*

• Right after the listener has been registered

• When a user is signed in

• When the current user is signed out

• When the current user changes

Hence this method listen to the listener and handle authentication state changes

**7.0 Appendices (if any)**

NIL