



Amazon SRS

Object Oriented Programming Using Java (Sardar Patel University)

Software Requirements Specification (SRS)

On

AMAZON

Software Engineering (2160701)

Submitted To:

Proff. Deven Gol

Submitted by:

Ankit Vekariya(170110116061)

Table of Contents

1. Introduction

1.1	Overview	4
1.2	Purpose	4
1.3	Scope	4
1.4	Outline of SRS	4
1.5	References	4
1.6	Definitions, Acronyms and Abbreviations	5

2. Overall Description

2.1	Product Perspective	7
2.2	Product Function	7
2.3	User Classes and Characteristics	7
2.4	User Documentation	7
2.5	Operating Environment	8
2.6	Assumptions and Dependencies	8
2.7	Design Implementation	8

3. Specific Requirements

3.1	Functional Requirements	9
3.2	Non-Functional Requirements	10
3.2.1	Performance	10
3.2.2	Security	10
3.2.3	Reliability	11
3.2.4	Maintainability	11
3.3	Constraints	11

4. External Interfaces

4.1	User Interface	12
4.1.1	Home Page User Interface	12
4.1.2	Login Interface for User	13
4.1.3	Customer's Login Interface	13

4.1.4	Login interface for Seller	14
4.1.5	Seller's Login Interface	14
4.2	Hardware Interface	15
4.3	Communication Interface	15
4.4	Analytics Interface	15
5. Software Design Specification		
5.1	Data Flow Diagram	15
5.2	Unified Modelling Language	21
5.2.1	Use Case Diagram	21
5.2.2	Sequence Diagram	22
5.2.3	Class Diagram	26
5.2.4	Component Diagram	27
5.2.5	Activity Diagram	28
5.2.6	State Chart Diagram	30
5.2.7	Deployment Diagram	31
5.3	Database Design	
5.3.1	Table Design	32
5.3.2	Entity Relationship Diagram for User	35
5.3.3	Entity Relationship Diagram for Seller	36
6. Testing of Software		
6.1	Unit Testing	37
6.2	Integration Testing	37
6.3	Validation Testing	37
6.4	Test Cases	38
6.4.1	Objectives	38
6.4.2	Description	38
6.4.3	Features to be tested	39
6.5	Performance Testing	44
6.6	Suspension and Resumption Requirements	44
6.6.1	Suspension Criteria	44
6.6.2	Resumption Requirements	44

1. Introduction

1.1 Overview:

The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system, its requirements with respect to consumers. The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. Developing an online platform is a job that requires equal share of technological expertise and sound decision making. Principally built on PHP, this website offers human experience which makes the shopping experience as satisfying as in a real store. Programming engineers make it sure that the site is high on responsiveness and low on tech errors, which is the most favorable selling point of this online store.

1.2 Purpose:

Amazon.com is an e-commerce website allows consumers to electronically exchange goods and services with no barriers of time or distance. Electronic commerce has expanded rapidly over the past five years and is predicted to continue at this rate, or even accelerate. India's biggest online store for Mobiles, Fashion(Cloths/Shoes), Electronics, Home Appliances, Books, Jewelry, Home, Furniture, Sporting goods, Beauty & personal care and more! Largest selection from all brands at lowest price. Payment options - COD, EMI, Credit card, Debit card & more.

1.3 Scope:

E-commerce is a term for any type of business, or commercial transaction that involves the transfer of information across the Internet. It covers a range of different types of businesses, from consumer based retail sites, through auction or music sites, to business exchanges trading goods and services between corporations. It is currently one of the most important aspects of the Internet to emerge.

1.4 Outline of SRS:

This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality.

Nonetheless, it helps any designer and developer to assist in software delivery lifecycle (SDLC) processes.

1.5 References:

Reference for geeky operating environments www.w3techs.com

Reference for user interfaces of e-commerce www.uianduxdesign.com

Reference for Flipkart's Vision and moto www.wikipedia.com

Reference for functional diagram

<https://creately.com/diagram/example/hdkb50lr/Amazon+DFD> Reference for <https://programformca.com>

1.6 Abbreviations:

SRS	System Requirements Specification
UI	User Interface
FAQ	Frequently Asked Questions
CRM	Customer Relationship Management
DFD	Data Flow Diagram

1.7 Definitions:

- 1.7.1 **End user/Customer** - The end user will be the one who visits the website and buys products online from the ones added by the Vendors/Sellers.
- 1.7.2 **Vendor/Seller** - Vendors/Sellers will add their products to the database, which will be seen in the website to the customers who can buy the products by selecting them. Vendors will have the special privileges than the end users, and have ability to manage the products added by them.
- 1.7.3 **Product** - A product is the item offered for sale. The price that can be charged depends on the market, the quality, the marketing and the segment that is targeted. Each product has a useful life after which it needs replacement, and a life cycle after which it has to be re-invented.
- 1.7.4 **Administrators** - Administrators are the ones who add or administer the categories for the products, and administer the Vendors.
- 1.7.5 **User ID** - At the time of login or signup user is required to enter our complete details which gets saved at the particular company's database and our unique ID is generated which helps in all correspondence in future.
- 1.7.6 **Seller ID** - The seller must have a seller ID and password for this which is used to verify and find the unique profile of the seller. This allows him to list the products that seller has available to sell.
- 1.7.7 **OrderID** - Each purchased and sold item has a unique Identification no. called OrderID that is used to track and record the details of the purchase done on the platform. It is generated by the system itself and no two products can ever have the same OrderID.
- 1.7.8 **Bank Details** - The transaction amount will be transferred back to the saved account. So the bank details will be synchronized and saved at the backend database securely.
- 1.7.9 **Online Discussion** - The data collected through the online discussion will be spread at various websites for reference purposes and people's review points will be noted.

2. Overall Description

2.1 ProductPerspective:

Flipkart's online website based e-commerce is an outstanding way of bringing Sellers and customers on an online platform to make purchase in a secured and efficient manner irrespective of distance between the two. The product is one stop for 30 million products online. Amazon offers daily recommendations for discounts and offers on products based on their category.

2.2 ProductFunction:

- Provide a simple interface and platform to ease the process of buying as well as selling products online
- Include smooth functionality and efficiency that adds to buyers' confidence.
- Appropriate space is dedicated to its Offers zone and deal of the day that fetches adequate attention of users.
- Amazon keeps a constant focus on new category creation and expansion of products.
- Ads and promotion activities motivate shoppers to buy online.
- Tracking feature of orders and pending deliveries for both sellers and buyers.
- Notifying order and delivery reports to customers and sellers.

2.3 User Classes andCharacteristics:

- 23.1 Customer** - He or she is a verified user of website who is intended to buy a product from the seller via the Amazon platform. The customer must have a username and password to make a purchase. The person is regularly updated and fed with latest offers and discounts according to their interest.
- 23.2 Seller** - He or she is a verified person who is allowed to sell items over the platform. Seller's details are stored on database and all the products are listed under him that he is ready to sell or are available. He is responsible to set products details, price, and quantity.
- 23.3 Administrator** - He or she is responsible for monitoring functions and procedures on platform. Administrator is responsible to provide valid information of a purchase to the concerned authority in case of any dispute between the customer and seller or in case of exchange.

2.4 UserDocumentation:

- Notifications
- Contact Us
- UserAgreement
- Select the operator
- Make A Payment
- Submit Account Details
- Payment Security
- Privacy Policy

2.5 DesignImplementation:

The user interface shall be implemented using any tool or software package like Java Applet, MS Front Page, Enterprise Java-Beans (EJB) etc. on following languages.

Client-side Programming Language	
JavaScript	JavaScript is a lightweight, object-oriented, cross-platform scripting language, mainly used within web pages.
HTML5	HTML5 is the fifth revision of the HTML standard.
CSS	Cascading Style Sheets define style rules in a separate CSS file.

2.6 OperatingEnvironment:

Recommended browsers are Chrome, Firefox, Safari and Internet Explorer 8 or higher.

articulars	Client System	Server System
Operating System	Windows/Linux/Android/iOS	Linux
Processor	Intel or AMD	Intel or AMD
Hard Disk	1 GB	1 TB
RAM	256	8 GB

2.7 Analytics

Analytics are required in order to monitor and analyze the requirements details from customers.

Structured Data Formats	The Open Graph protocol, originally developed by Facebook, is an RDF a-based format that enables any web page to become a rich object in a social graph.
Google Analytics	Google Analytics is a free service to get detailed statistics about the visitors of a website, provided by Google.

2.8 Assumptions andDependencies

- The customer and seller must have basic knowledge of computers and English language.
- Each User must have a User ID and password.
- Each Seller must have Seller ID and password.
- There must be an Administrator.
- Internet connection is a must.
- Proper browsers should be installed in the user's system.

3. Specific Requirements

3.1 Functional requirements:

This subsection contains the requirements for the e-store. These requirements are organized by the features discussed in the product functions. Features from there, they are then refined into use case diagrams and to sequence diagram to best capture the functional requirements of the system.

3.1.1 Provide Search facility-

- User to enter the search text on the screen and display matching products based on the search.
- Enable user to select multiple options on the screen to search.
- Display only 10 matching result on the current screen.
- Enable user to navigate between the search results.
- Notify the user when no matching product is found on the search.

3.1.2 Tailored Content for the User-

- Using the cookies of the user to study the buying pattern of the user.
- Assists him in his purchase and customizes his homepage according to his previous interactions.
- Including additions to his wish-list, page views, previous searches, reviews about the product among other things.
- Making good use of data is imperative to designing great customized user experiences.

3.1.3 Displaying Related Items -

- Including results for related items to the items already bought.
- This can incentivize the customer to purchase more.
- By tracking the buying pattern of the user who is logged in, system recommends similar items for the user
- Display similar products with high ratings newly in-stock or are featured at special discount prices.
- Displaying all related items that other users purchased in addition to that product.
- This exposes the user to a larger variety of items of his interest.

3.1.4 Create Sellable items on website-

- Provide an interface to all the sellers to add product description and price through their account.
- List all the sellers for a common product they offer to sell and prioritize them based on percent of discount.

3.1.5 Easy Return or Exchange-

- The system allows customers to return the item back to the seller in case he or she doesn't like the product and get the feedback of customer
- The system allows customer to exchange the product with seller in case of delivery of any defective item and get feedback from customer.
- Notify seller about the customer's reason and feedback about the product.

3.1.6 Provide personalized profile -

- The seller's and customer's profile contain all details of seller and customer respectively like contact, location, address and seller's license.
- To display both the active and completed order history in the customer profile.
- To allow user to select the order from the order history and detailed information.
- To display the most frequently searched items by the user in the profile.
- To help seller keep list of products and its information in its profile.

3.1.7 Move transaction money and details to Profile-

- Safe transfer of money bank account after successful transaction of money.
- Add details of transfer of money to company's selling and purchasing events logs and database.
- Send an order confirmation to the user and seller about the purchase through email or SMS.

3.1.8 Detailed invoice for Customer and Seller -

- The system shall display detailed invoice for current order once it is confirmed.
- The system shall allow user to print the invoice.
- The system shall display detailed information of seller's detail to customer as well as customer's detail to seller.

3.1.9 Provide shopping cart facility-

- The system shall provide shopping cart during online purchase.
- The system shall allow user to add/remove products in the shopping cart.
- Later customer can confirm orders for purchase.

3.1.10 Online tracking of shipments-

- The system shall allow user to enter the order information for tracking.
- The system shall display the current tracking information about the
- order The system notifies seller about delivery of product to the consumer.

3.1.11 Allow multiple payment methods-

- The system shall display available payment methods for payment.
- The system shall allow user to select the payment method.

3.1.12 Provide Customer Support.-

- The system shall provide online help, FAQ's customer support, and sitemap options for customer support.
- The system shall allow user to select the support type he wants.
- The system shall allow user to enter the customer and product information for the support.
- The system shall display user contact of seller and Amazon support desk
- The system shall display the online help upon request.
- The system shall display the FAQ's upon request.

3.2 Non-Functional Requirements:

3.21 Performance:

- The product shall be based on web and has to be run from a web server.
- The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run.
- The performance shall depend upon hardware components of the client/customer.

3.22 Security:

3.2.2.1 Data Transfer -

- The system shall use secure sockets in all transactions that include any confidential customer information.
- The system shall automatically log out all customers after a period of inactivity. The system shall confirm all transactions with customer's web browser.
- The system shall not leave any cookies on the customer's computer containing the user's password or confidential information.

3.2.2.2 Data Storage-

- The customer's web browser shall never display a customer's password. It shall always be echoed with special characters representing typed characters.
- The customer's web browser shall never display a customer's credit card number after retrieving from the database. It shall always be shown with just the last 4 digits of the credit card number.
- The system's back-end servers shall never display a customer's password. The customer's password may be reset but never shown.
- The system's back-end servers shall only be accessible to authenticated administrators.
- The system's back-end databases shall be encrypted and within company's perimeter.

3.23 Reliability:

- The system provides storage of all databases on redundant computers with automatic switchover.
- The reliability of the overall program depends on the reliability of the separate components.
- The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes.

3.24 Safety:

- It is the state of being "safe", the condition of being protected against physical, social, spiritual, financial, political, emotional, occupational, psychological, educational or other types or consequences of failure, damage, error, accidents, harm or any other event which could be considered non-desirable.
- This can take the form of being protected from the event or from exposure to something that causes health or economical losses.
- It can include protection of people or of possessions.

3.25 Maintainability

- A commercial database is used for maintaining the database and the application server takes care of the site.
- In case of a failure, a re-initialization of the program will be done.
- Also the software design is being done with modularity in mind so that maintainability can be done efficiently

3.3 Constraints

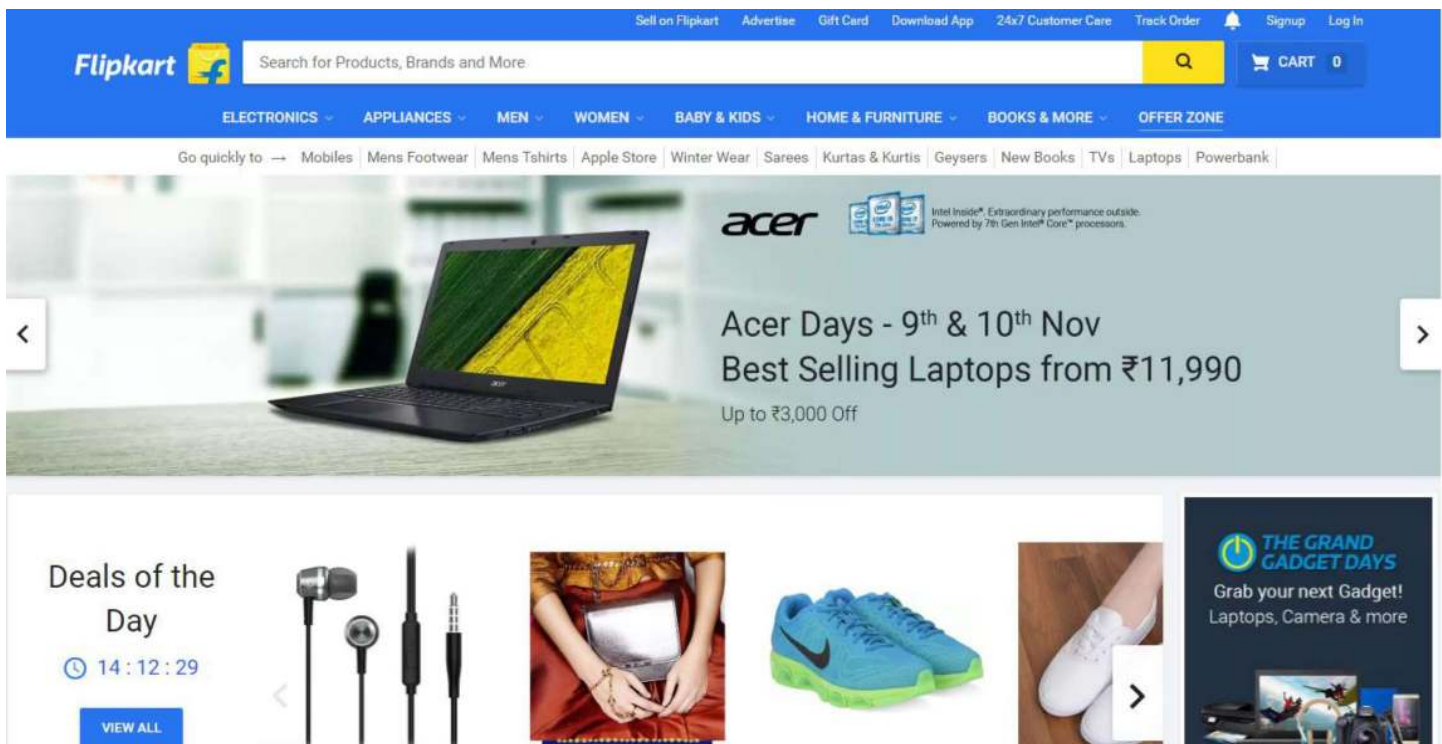
- Limited numbers search queries for the users. Servers incapable of handling high traffic burst
- The actual product might differ from its display image. Uniform quality of service is not ensured.
- It cannot ensure the reliability of the review.
- The User and Seller must have confirmed User ID and Seller ID respectively.

4. Interface Requirements

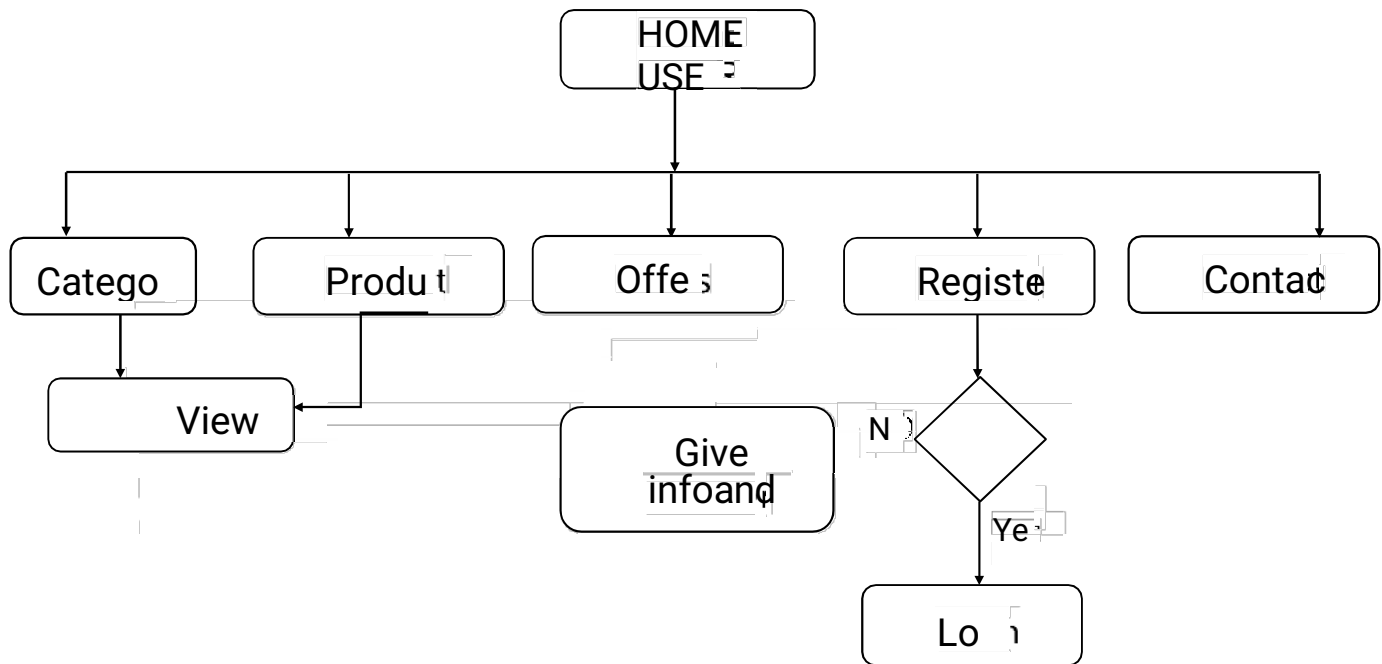
4.1 User Interface

- Login or Signup Page
- Homepage containing products according to customer's interest.
- There will be a screen displaying information about all products that user searches.
- If the customers select any product then it will open another tap about the product information.
- After all transaction the system makes the selling report as portable document file (pdf) and sent to the customer E-mail address.

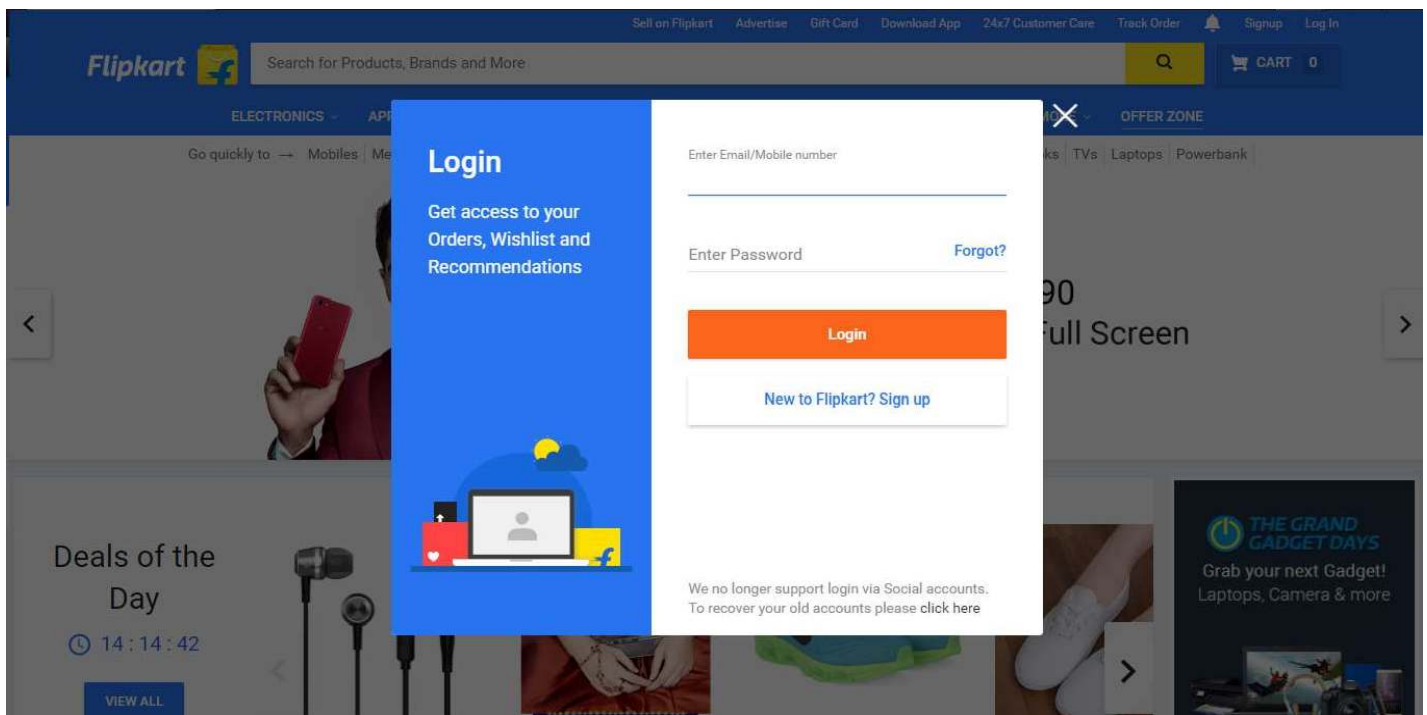
4.1.1 Home Page User Interface:



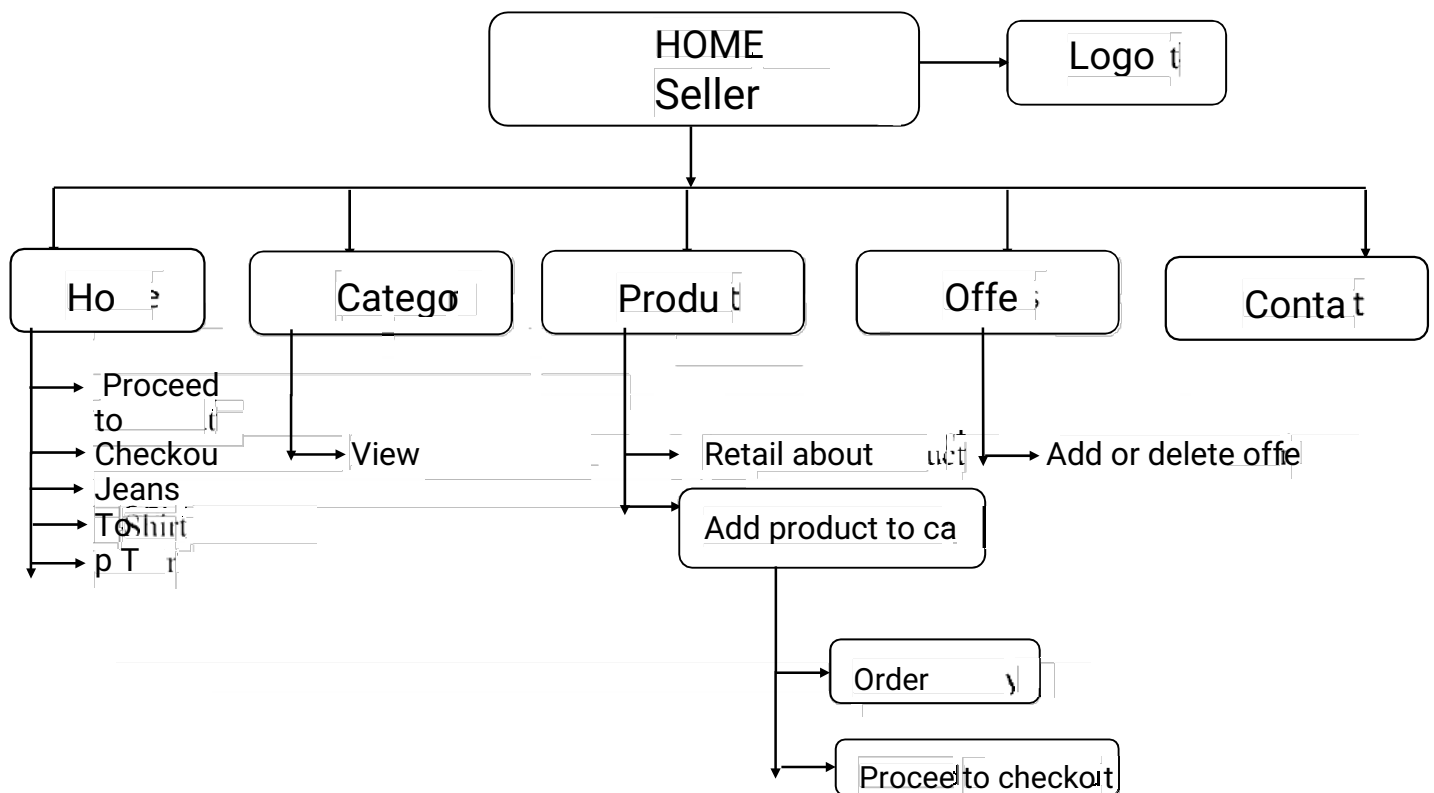
4.1.2 Login Interface for User:



4.1.3 Customer Login Interface:



4.1.4 Login Interface for Seller:



4.1.5 Seller's Login Interface:



Grow your business with the leader in Indian e-commerce



4.2 Hardware Interfaces:

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable. The system require Database also for the store the any transaction of the system like MYSQL etc. system also require DNS (domain name space) for the naming on the internet. At the last user need web browser for interact with the system.

Web Server Deployment and Technologies	
Nginx	Nginx (pronounced as "engine X") is a lightweight open source web server developed by Igor Sysoev.
MySQL	MySQL database for storage of Data and user as well as seller information
RESTful API	A RESTful API is an application program interface (API) that uses HTTP requests to GET, PUT, POST and DELETE data.

4.3 Communications Interfaces:



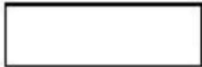

The e-store system shall use the HTTPS protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite. The user must have SSL certificate licensing registered web browser.

5. SYSTEM DESIGN SPECIFICATION

5.1 Data flow Diagram (DFD):

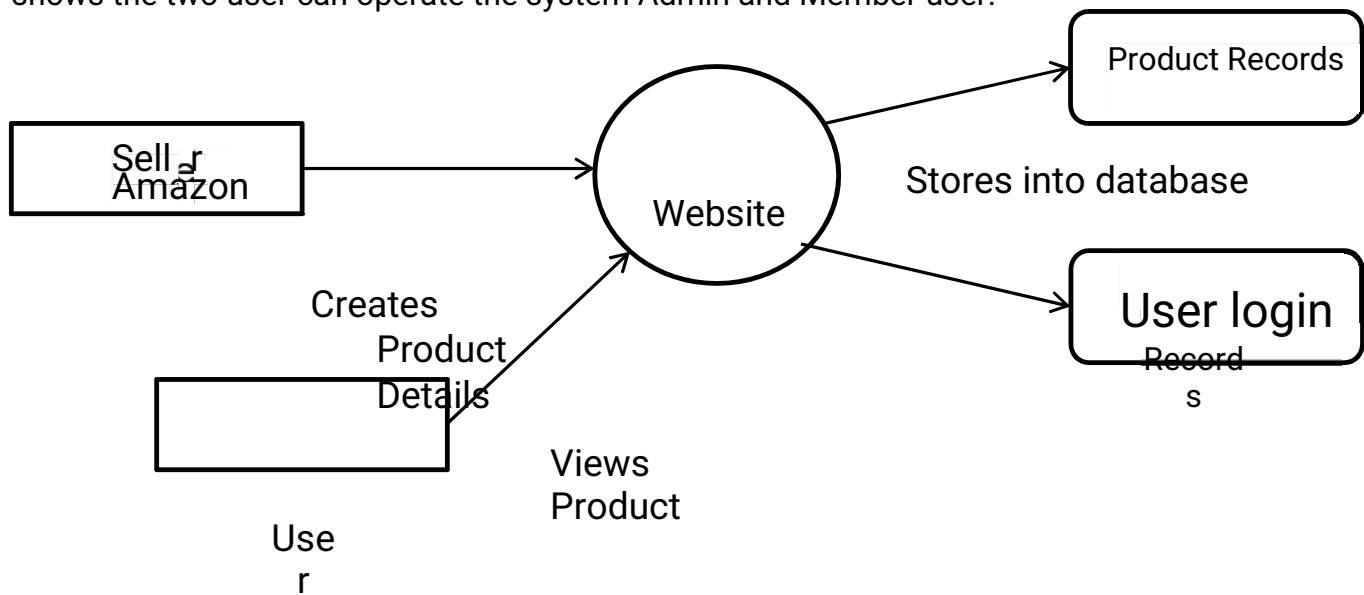
A Data Flow Diagrams is a structured analysis and design tool that can be used for flowcharting in place of, or in association with, information-oriented and process-oriented systems flowcharts. A DFD is a network that describes the flow of data and the processes that change, or transform, data throughout a system. This network is constructed by using a set of symbols that do not imply a physical implementation. It has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design phase that functionality decomposes the requirement specifications down to the lowest level of detail.

5.1.1 Data flow diagrams symbol:

Symbols	Description
	Data Flow: Data flow are pipelines through the packets of information flow.
	Process: A Process or task performed by the system
	Entity: Entity are object of the system. A source or destination data of a system
	Data Store: A place where data to be stored

5.1.2 ContextlevelDFD – 0level:

The context level data flow diagram (DFD) is describe the whole system. The 0-level DFD describe the all user module who operate the system. Below data flow diagram of online shopping site shows the two user can operate the system Admin and Member user.



o– Level DFD for Online shopping websiteproject

The Seller will have the access to create Product ads on the website. Once the seller has added item to sell list in website, administrator will receive the items details to take into consideration. Only after administrator approve, the item will only be allowed to go into sell list of the Amazon Online Shopping Store Website. So, the website is act as a medium between the seller and customer (buyer). Besides that, the website will serve to provide the details of all product listed there, if a user intend to buy, he/she has to contact the seller and make payment through the website

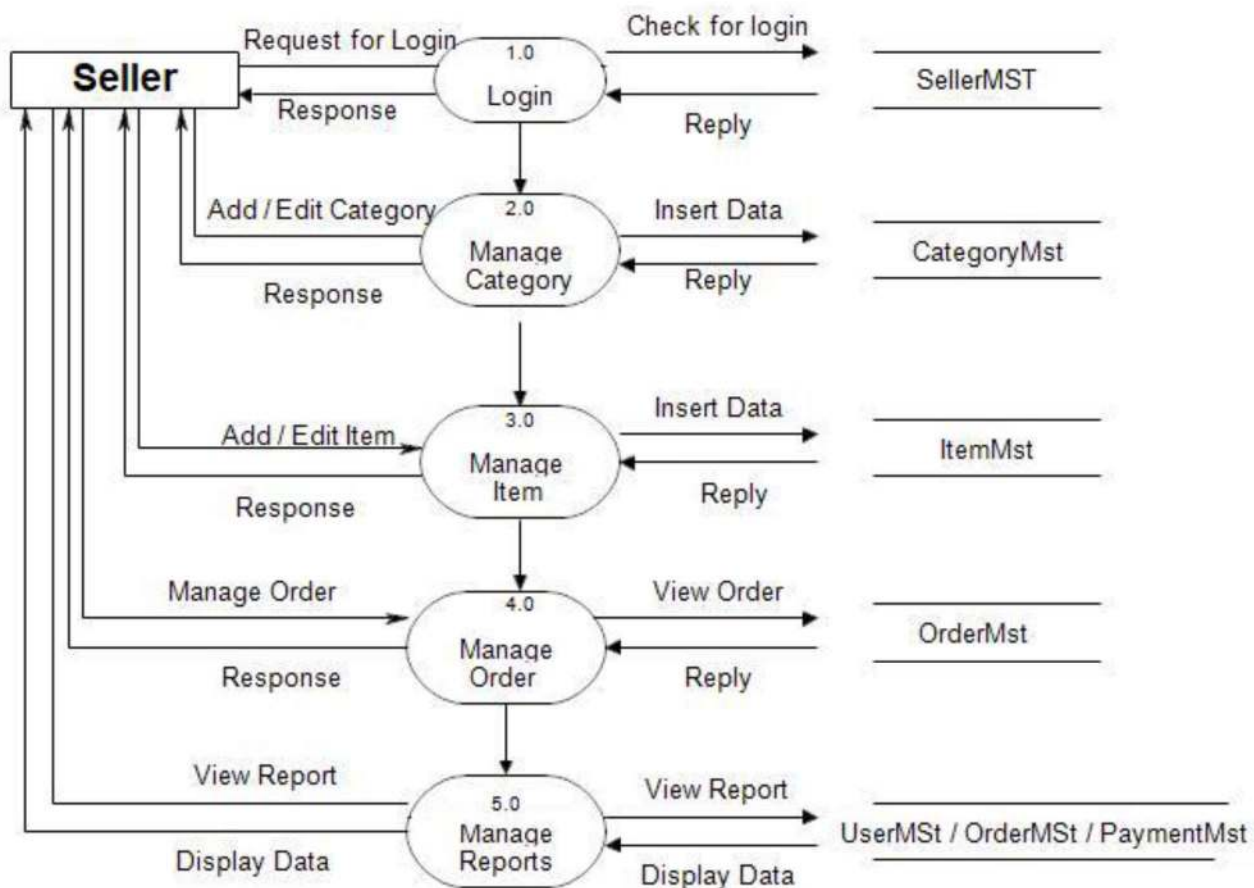
Meanwhile, the user can access the website for buying, selling or to get product information. As only a registered user can buy or sell items in Amazon Online Shopping Online Store Website, he/she must agree to the terms and conditions of the system. Once register, he/she will receive a verification mail to his/her email, which have a link to our website. By clicking on the link then will direct verify the user's registration and then he/she will become an official member in Amazon Online Shopping Store.

A seller can advertise the item he/she willing to sell with the personal detail. If the item advertise is not approve by administer, he will receive a message from administer regarding it. The user still can review the detail about the item he/she willing to sell which is still wait for administrator consideration.

5.1.23 1st Level Seller Side DFD:

The Seller side DFD describe the functionality of Seller. Seller can first add category of item and then add items by category wise and he can manage order and payment detail.

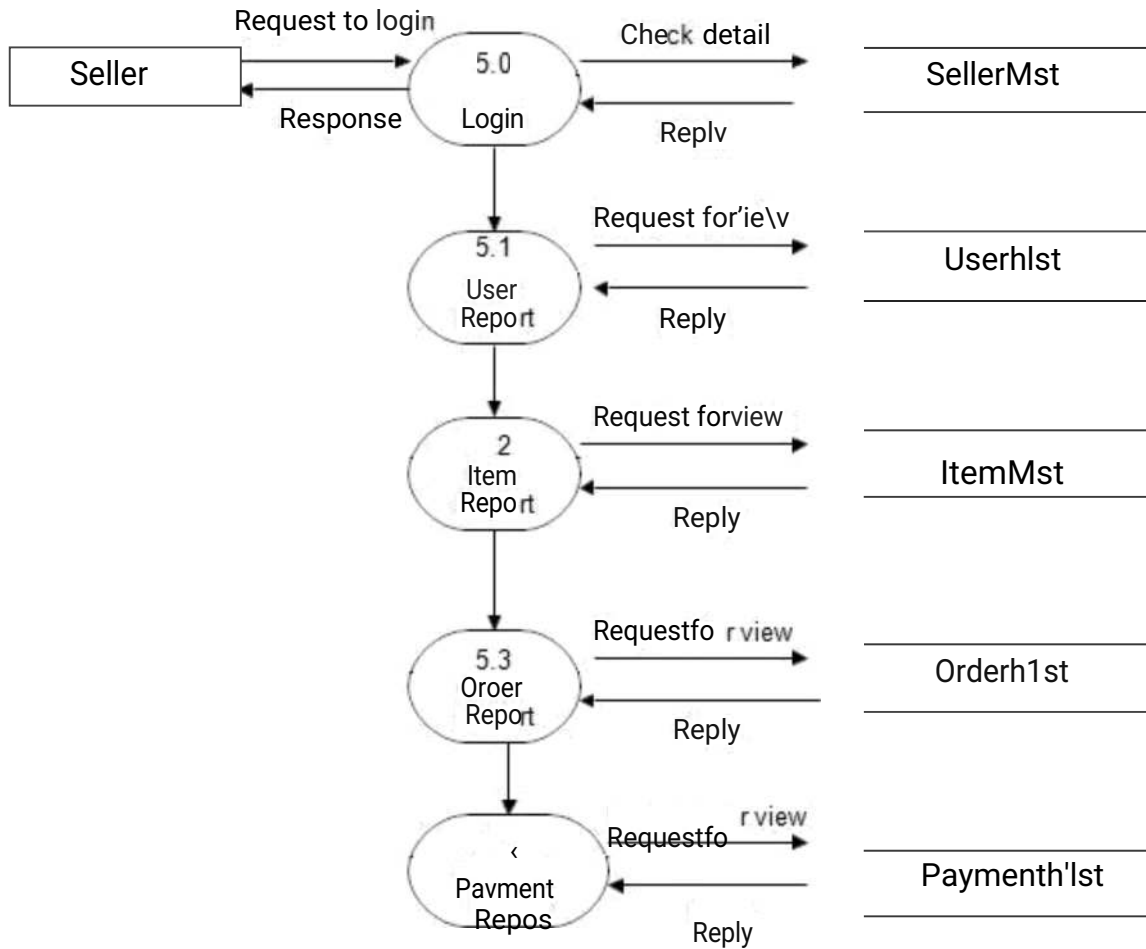
Seller Side DFD - 1st Level



1stLevel – Seller Side Data FlowDiagram

5.1.24 2ndLevel Seller sideDFD:

2nd Level SellerDFD

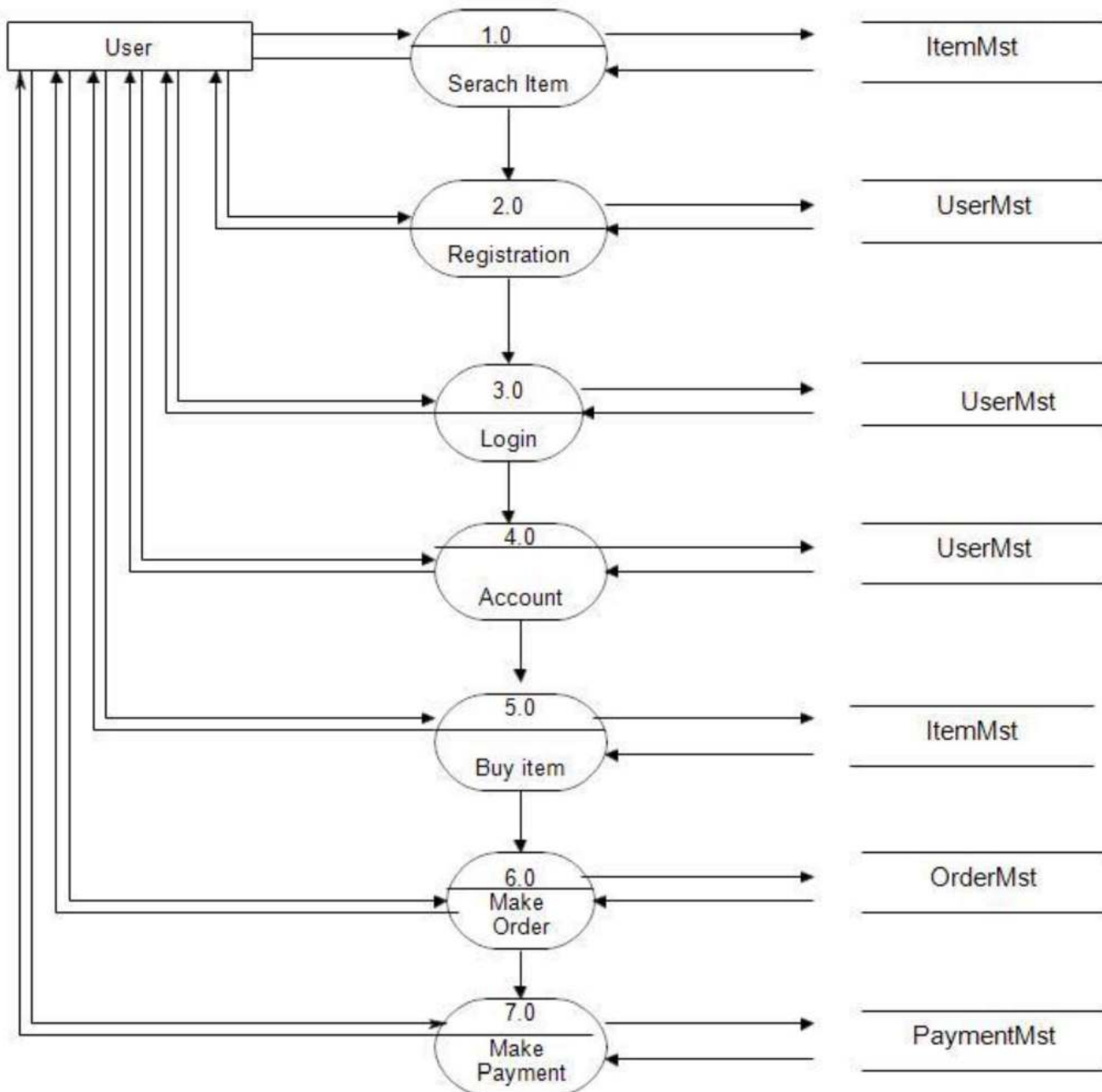


Seller side DFD for online shopping website project

5.1.25 1stlevel – User side Data flowDiagram

The user is all people who operate or visit our website. User is a customer of a website. User can first select product for buy, user must have to register in our system for purchase any item from our website. After register he can login to site and buy item by making online payment through any bank debit card or credit card.

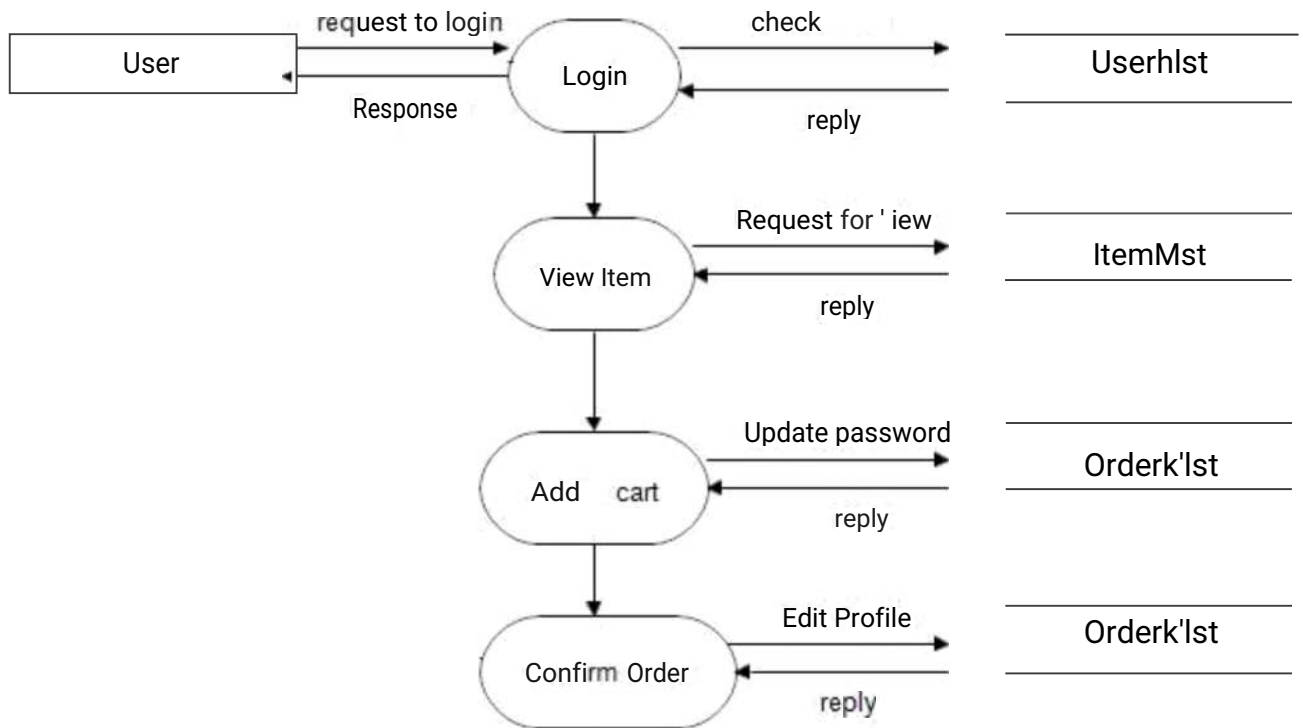
1st Level User side DFD



User side DFD for online shopping website project

5.1.2.6 2ndlevel User sideDFD

2st Level User DFD



DFD for online shopping website project

5.2 UMLDiagrams

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML diagrams are not only made for developers but also for business users, common people, and anybody interested to understand the system. The system can be a software or non-software system. Thus it must be clear that UML is not a development method rather it companies with processes to make it a successful system.

UML is a modeling language used to model software and non-software systems. Although UML is used for non-software systems, the emphasis is on modeling OO software applications. Most of the UML diagrams discussed so far are used to model different aspects such as static, dynamic, etc.

Now whatever be the aspect, the artifacts are nothing but objects. If we look into class diagram, object diagram, collaboration diagram, activity diagrams all would basically be designed based on the objects.

5.2.1 Use CaseDiagram:

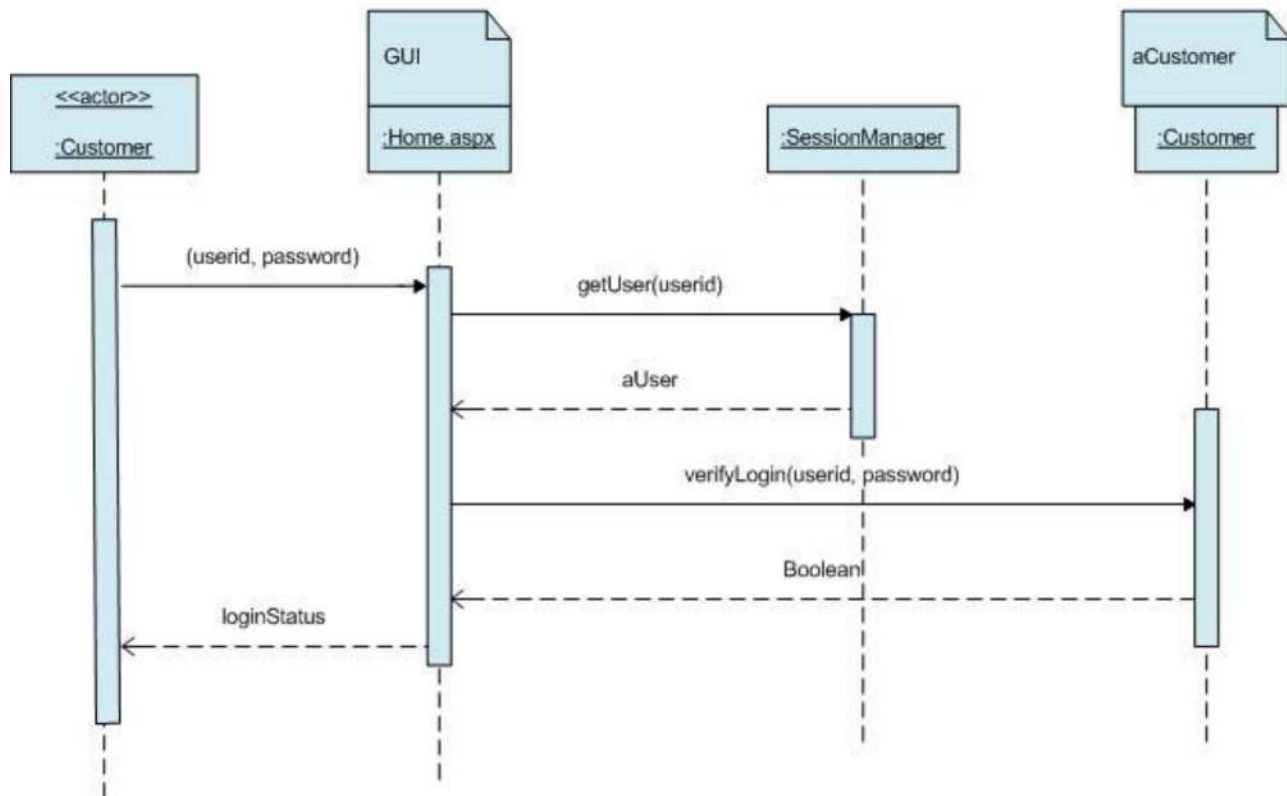
The UML provides the use case diagram notation to illustrate the name of the use case actors and relationship between them. User case diagrams are used to model the functional interaction between users and system.



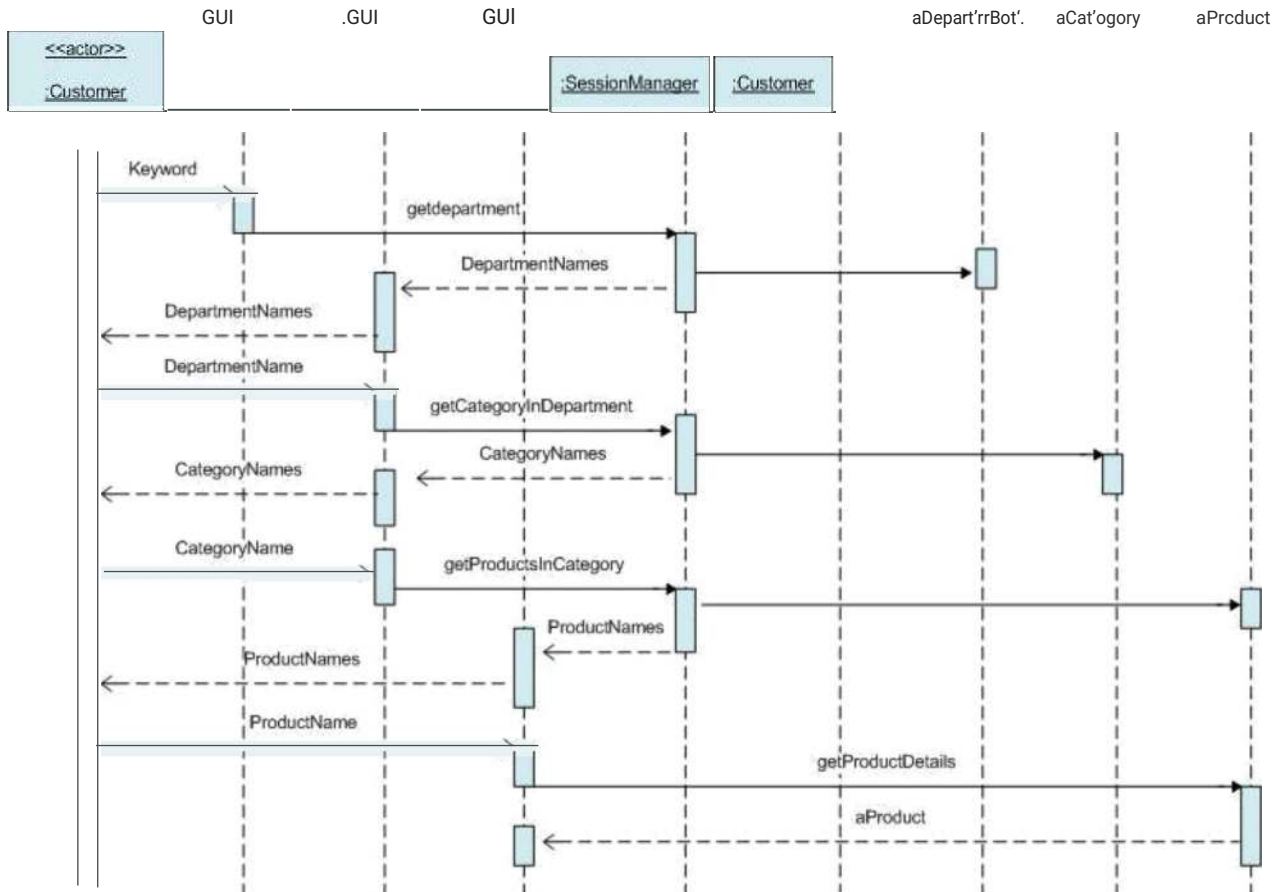
5.2.2 Sequence Diagram:

- A sequence diagram illustrates in a kind of format in which each object interacts via messages. It is generalization between two or more specification diagram.
- Sequence diagram is an interaction over view diagram. It provides a big picture over view of how a set of interaction is related in terms of logic and process flow.

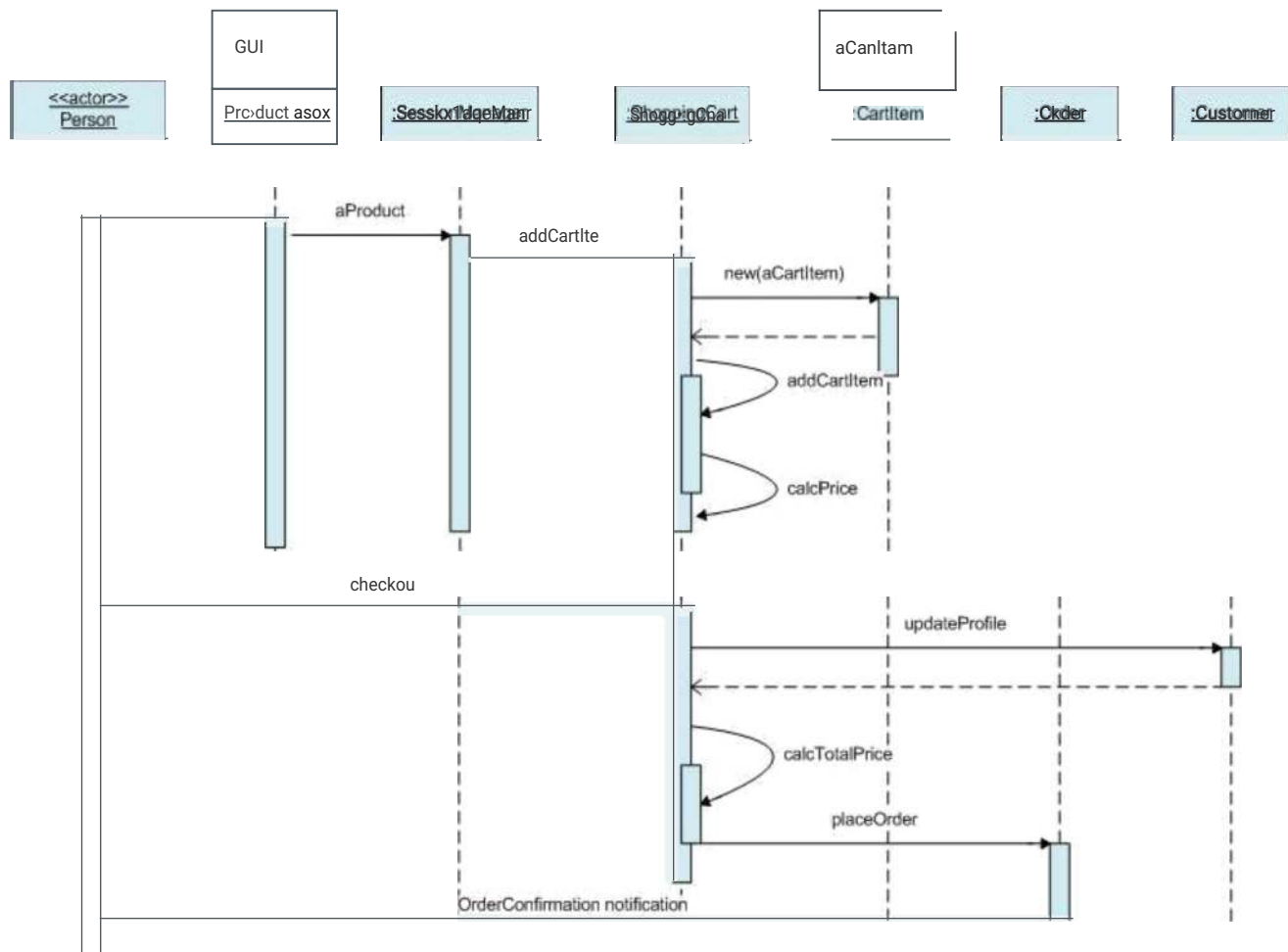
5.2.2.1 User Login SequenceDiagram:



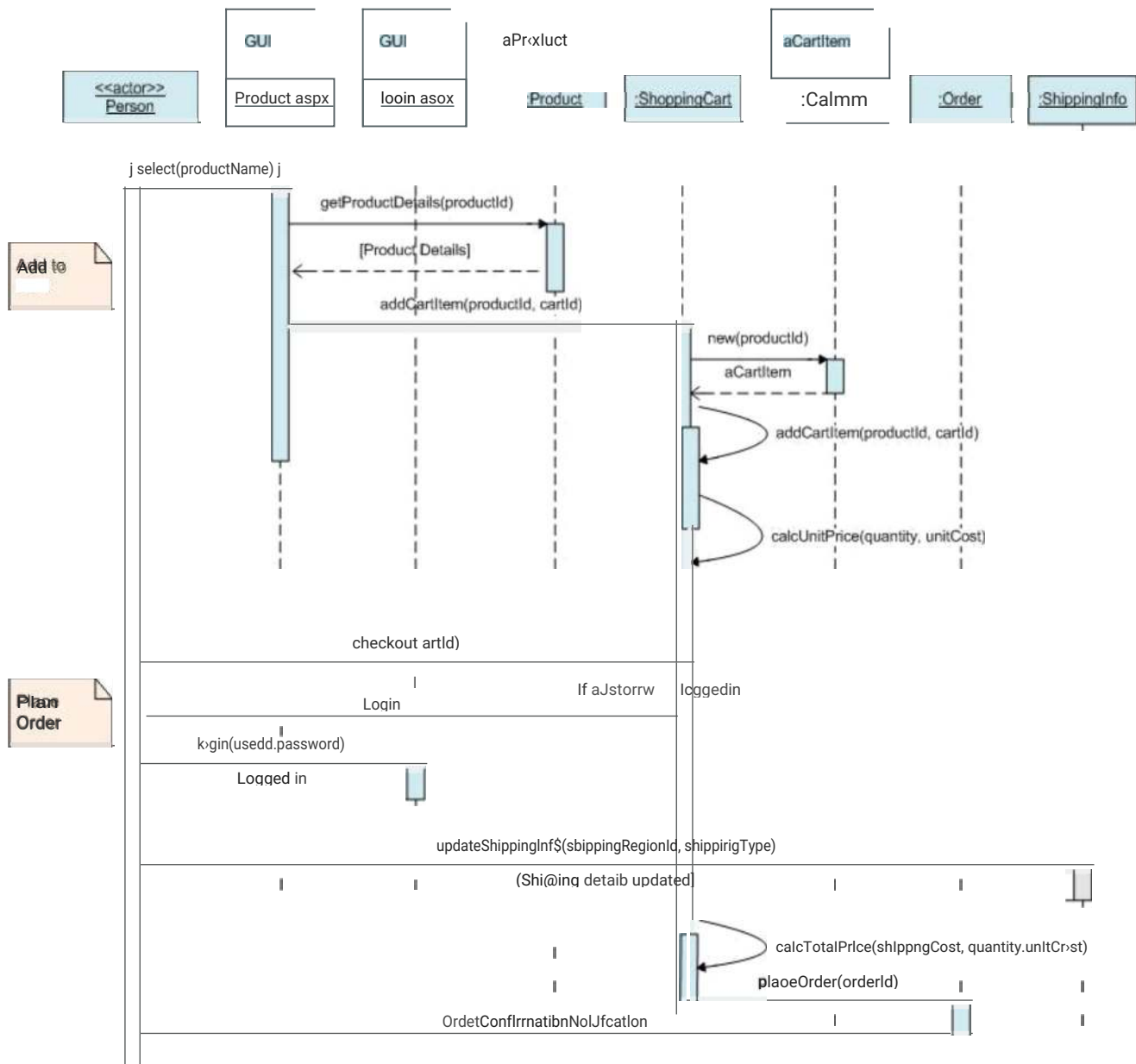
5.2.2.2 Search SequenceDiagram:



5.2.2.3 Buying (Add Product to Shopping Cart) SequenceDiagram:

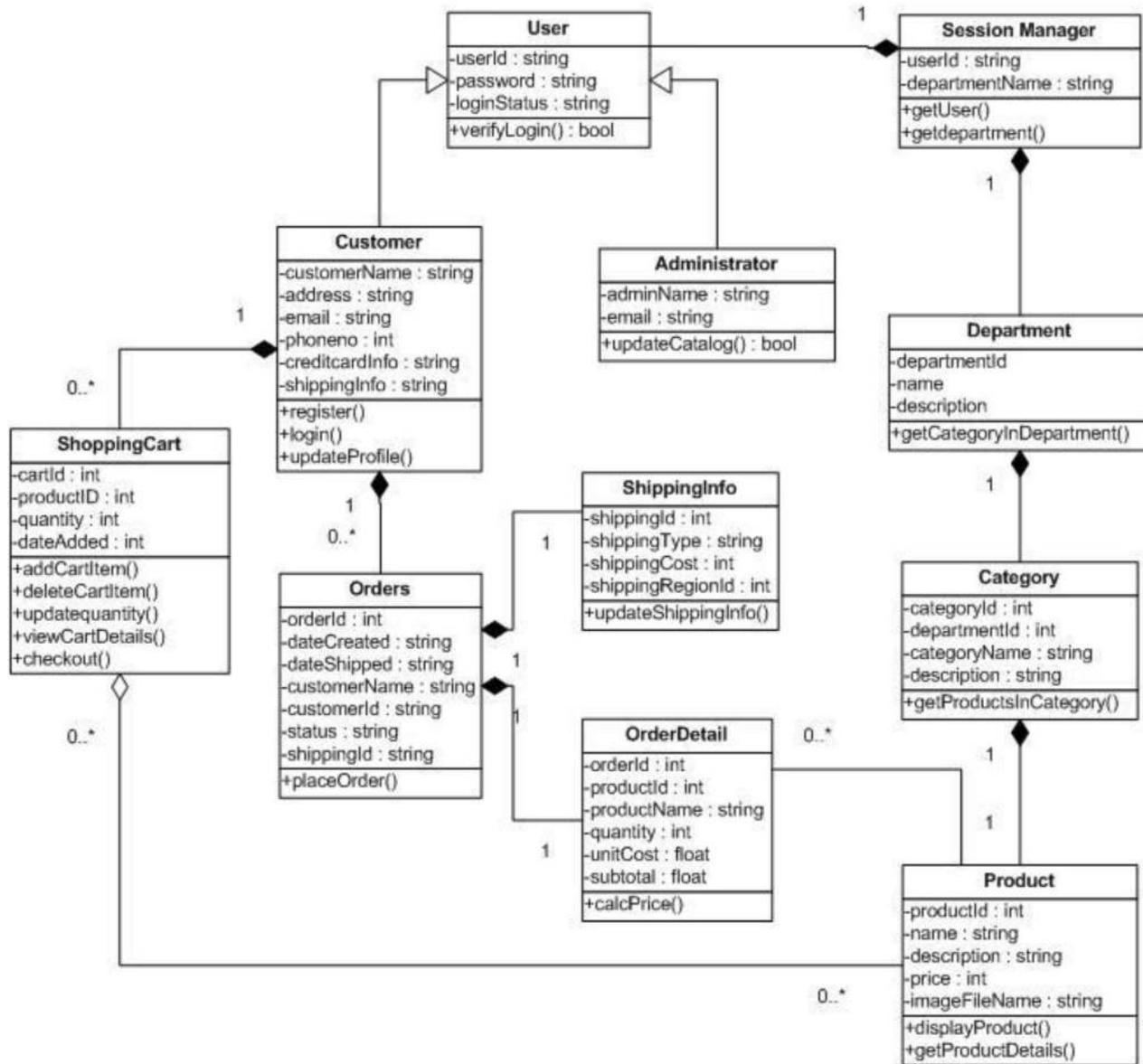


5.2.2.4 Place Order SequenceDiagram:



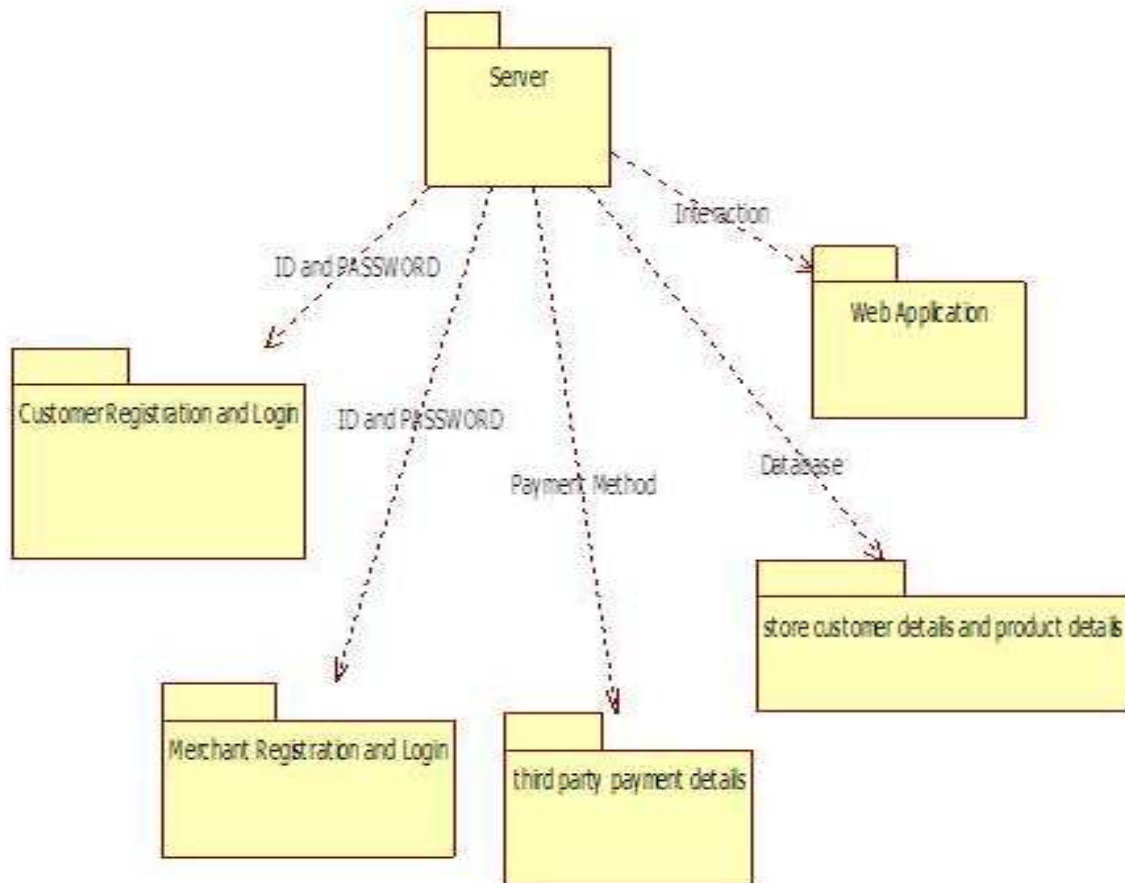
5.2.2 ClassDiagram:

- Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.
- Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.



5.2.2.3 ComponentDiagram:

- A component represents a modular part of a system, that encapsulates its contents and whose manifestation is replaced within its environment. A component defines its behaviors in terms of provide and required interfaces.
- Here the three components are applicant, system admin and authority.
- The interface between people and system admin, from people to authority.



5.2.2.5 Activitydiagram:

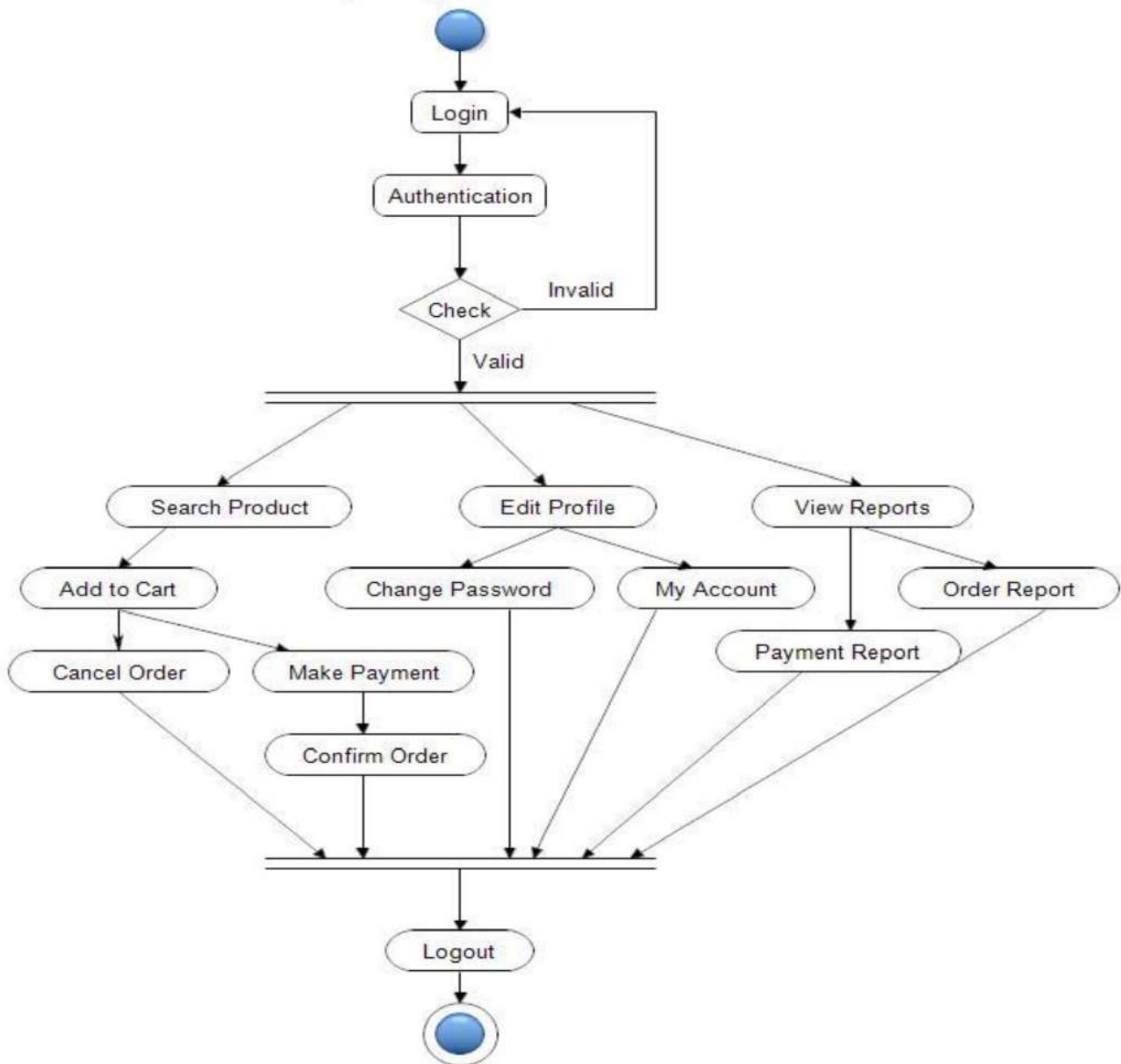
The activity diagram used to describe flow of activity through a series of actions. Activity diagram is an important diagram to describe the system.

The activity described as an action or operation of the system.

- Activity diagram shows sequential and parallel activities in a process. They are useful for modeling business, workflows, the data flows and complex algorithm.
- A UML activity diagram offers rich notation to flows a sequential of activities. It may be including parallel activities. It may be applied to any purpose, but it is popular for visualization of business workflows and use case.

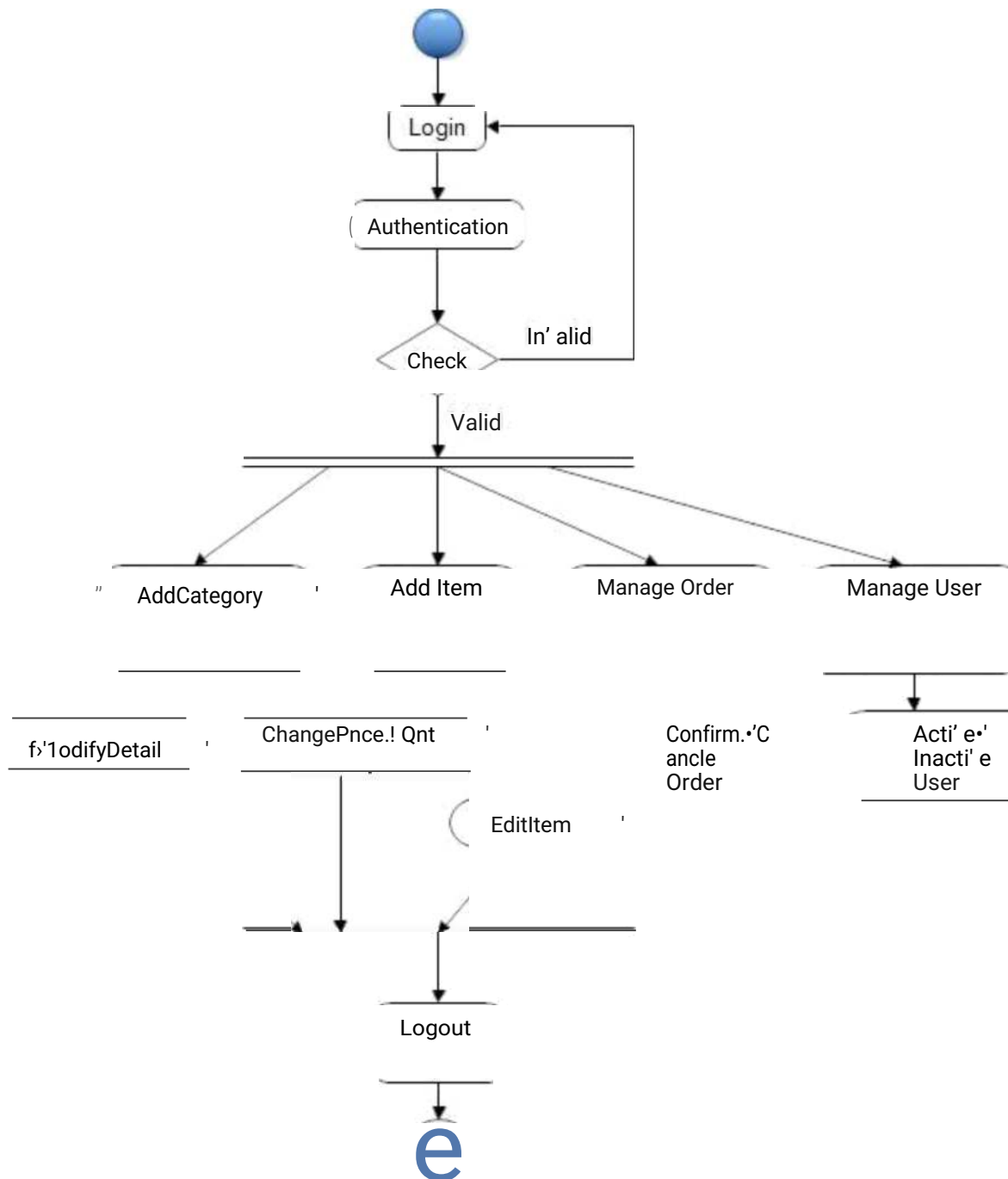
Activity Diagram for User Side

Activity Diagram for User Side



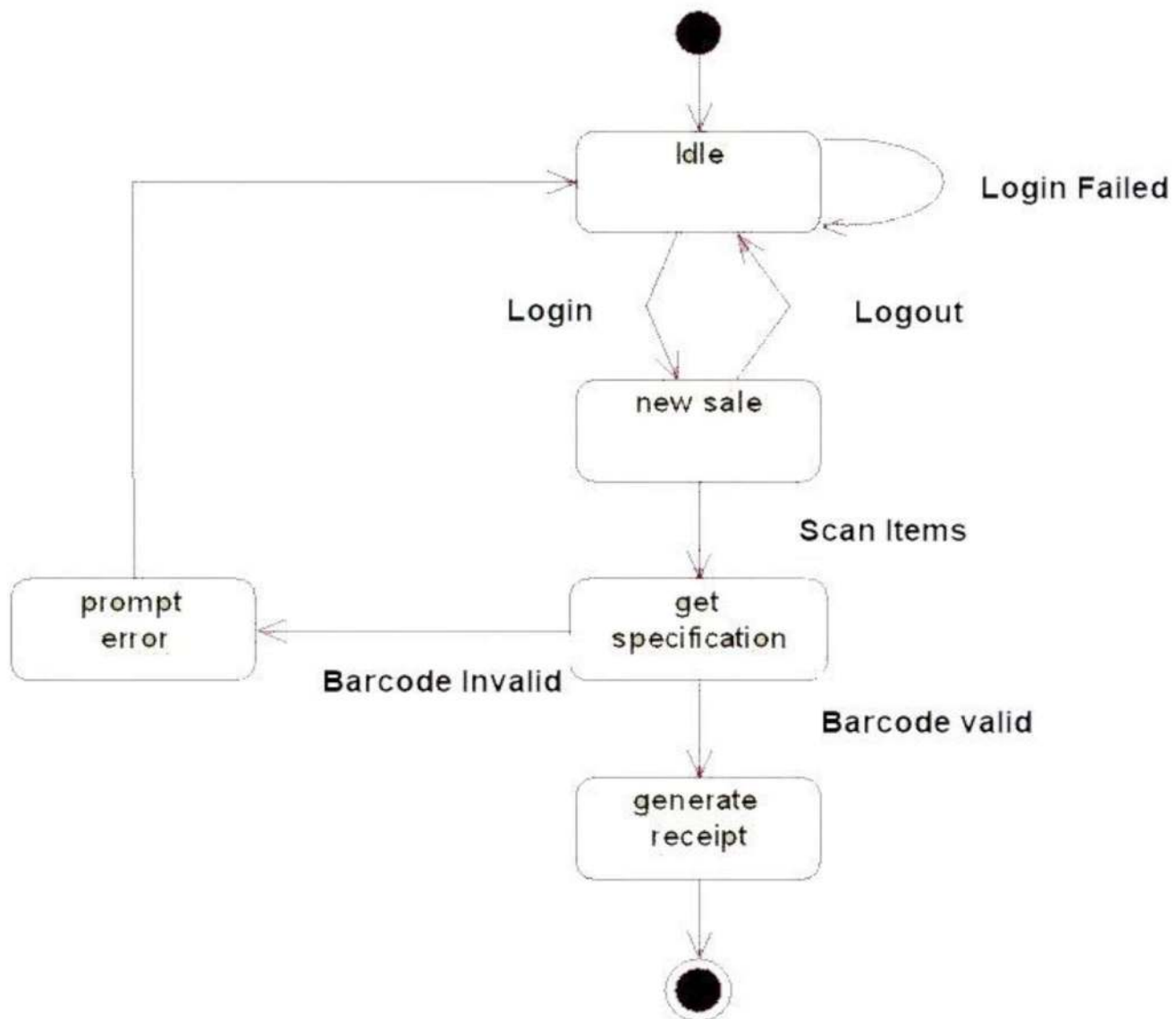
Activity diagram for Admin side

Activity Diagram for AdminSide



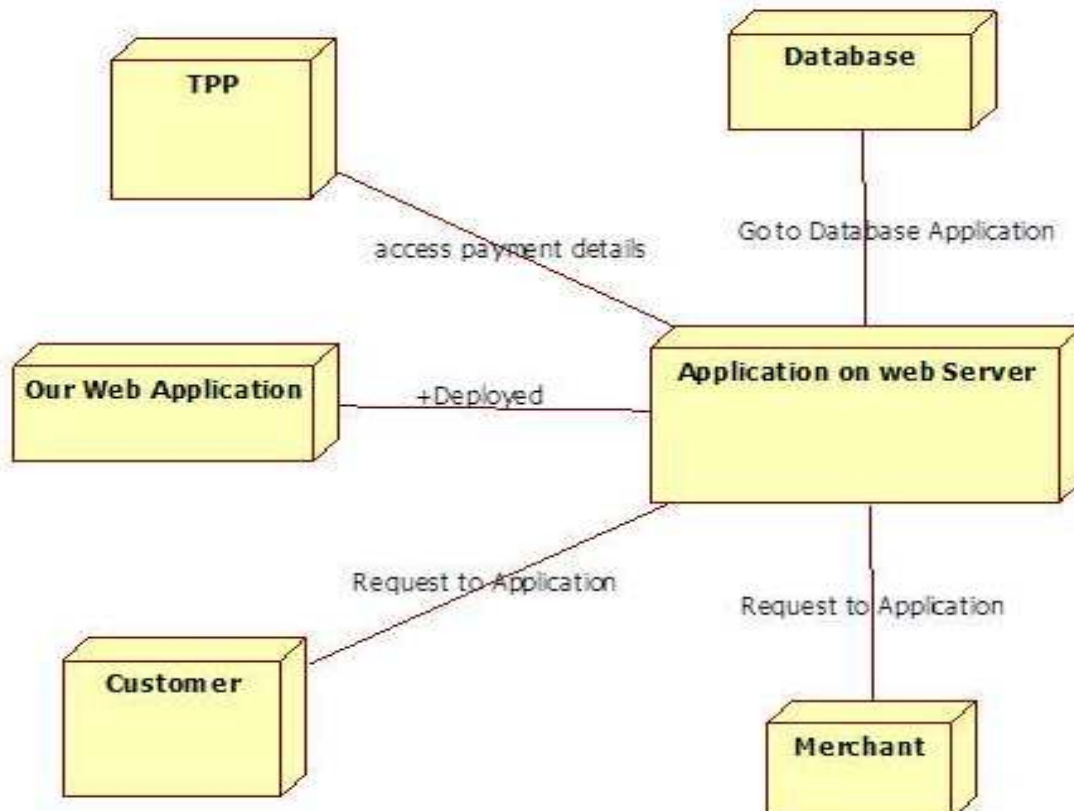
5.2.6 State ChartDiagram:

- It illustrates the in-trusting events and state of an object and behaviors of an object is reaction to an event. Transaction shows as allows labeled with theirs event. It is included with initial pseudo state and final end state.
- The state chart diagram of Online shopping website has 5 stages: Idle, New sale, Get specification, Prompt Error and generate receipt. In each state the system interacts with the user and get desired output based on the given inputs.



5.2.7 DeploymentDiagram:

- A deployment shows the assignment of concrete artifacts to computational nodes. It shows the deployment of software elements to the physical architecture, and the communication. Deployment diagrams are useful to communicate the physical and deployment architecture. In the deployment diagram the object reference in component diagram is also included the deployment diagram. In this authority and system admin, interface through the people.
- It is the process of installing the program.



5.2 DatabaseDesign:

A database design is a collection of stored data organized in such a way that the data requirements are satisfied by the database. The general objective is to make information access easy, quick, inexpensive and flexible for the user. There are also some specific objectives like controlled redundancy from failure, privacy, security and performance.

A collection of relative records make up a table. To design and store data to the needed forms database tables are prepared. Two essential settings for a database are:

1. **Primary key:** - The field that is unique for all the record occurrences.
2. **Foreign key:** - The field used to set relation between tables. Normalization is a technique to avoid redundancy in the tables.

5.2.1 Database Table Design:

1. Categorytable

Column name	Data type	Key constraint
Cat_id	int	Primary key not null
Cat_name	Char(20)	Not null

2. ProductTable

Column name	Data type	Key constraints	extra
Prod_id	Int	Primary key not null	Auto increment
Cat_id	int	Foreign key Not null	
Prod_name	Char(20)	Not null	
Prod_descp	Char(40)	null	
Price	double	Not null	
Available	int	Not null	
Add_date	date	Not null	

3. Seller Logintable

Column name	Data type	Key constraint
Seller_id	int	Primary key not null
password	Char(20)	Not null

4. Logintable

Column name	Data type	Key constraint
User_id	int	Primary key not null
password	Char(20)	Not null

5. Storetable

Column name	Data type	Key constraint
Order_no	int	Primary key not null
report	Char(20)	null

6. Temptable

Column name	Data type	Key constraint
Prod_id	int	Foreign key not null
Prod_name	Char(20)	Not null
Price	Double	Not null
Items	Int	Not null
User_id	Int	Not null
Purchase_date	Date	Not null
Order no	Int	null

7. Usertable

Column name	Data type	Key constraint	extra
User_id	int	primary key not null	Auto increment
Password	Char(20)	Not null	
User_name	Char(20)	Not null	
sex	Char(6)	Not null	

Address	Char(40)	Not null
Date_of_birth	date	Not null
Date_of_register	date	Not null
Phone_no	Char(10)	Not null
email	Char(30)	Not null

8. Feedbacktable

Column name	Data type	Key constraint
User_id	int	not null
type	Char(10)	Not null
feedback	Char(10)	Not null
comment	Varchar(40)	null
Feedback_date	date	Not null

9. AccountTable

Column name	Data type	Key constraint
Bank_name	Char(20)	not null
Account_no	Char(20)	not null
password	Char(20)	Not null
balance	double	Not null

10. OrderTable

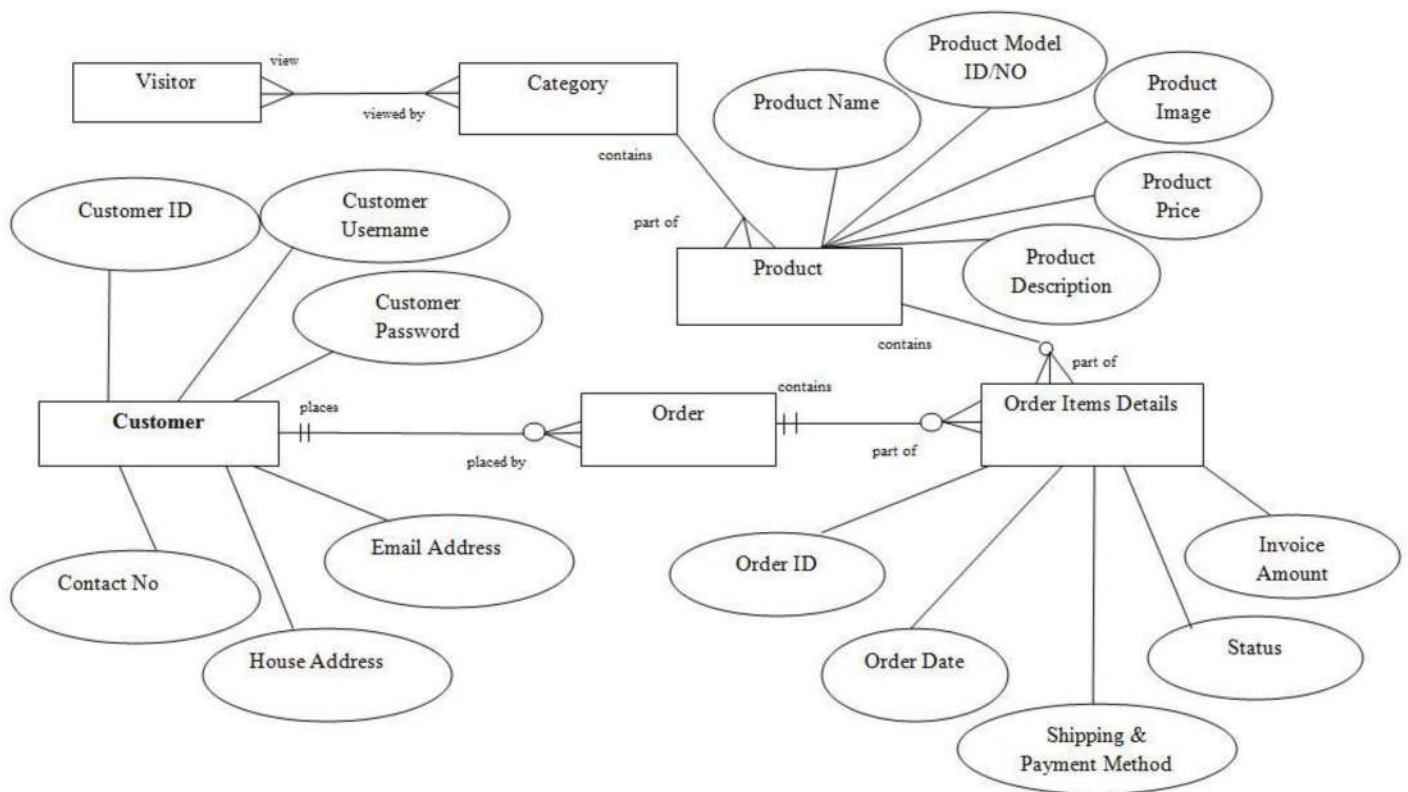
Column name	Data type	Key constraint
Prod_id	Int	not null
Prod_name	Char(20)	not null
price	Double	Not null
items	Int	Not null
User_id	Int	Not null
Purchase_date	Date	Not null
Order_no	int	Not null

5.3.2 ERDiagram:

The purpose of draw this diagram to show the relationship among the objects and personal attributes that belong to them.

Diagram 1:

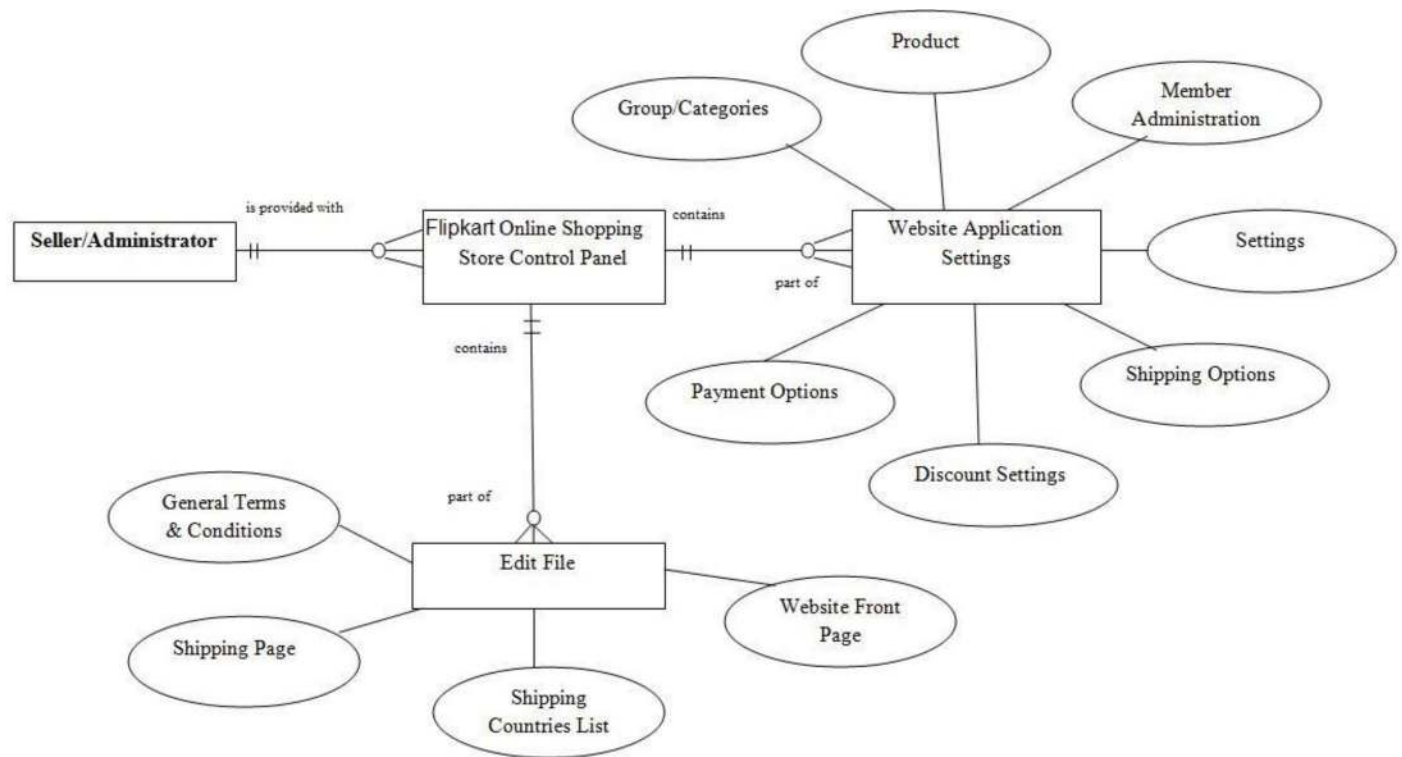
1. Customer can places many order (one to one relationship) as they wish when in Amazon Online Shopping Store and vice versa. In order to have a list of order, we must have at least one customer. There may be situation in which placement order action is not necessary from the customer.
2. Order can contains many order items details in the list and vice versa is part of the order. In order to have an order items details, we must have at least one order taken from the customer.
3. Visitors can view many categories as they wish in Amazon Online Shopping Store.(many to many relationship)
4. Each category will contains many product (one to many relationship).
5. The product will have the order items details.



ER Diagram for User

Diagram 2:

1. Seller is provided with a control panel page at the Amazon Online Shopping Store with series of functions.
2. The Control Panel contains edit file functions and website application functions. (Assumption : In order to have Edit File & Website Application functions, we must have a control panel first)s. There may be a situation in which the functions is not necessary to implement.



ER Diagram for Seller

6. TESTING

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs. It can also be stated as the process of validating and verifying that a software program/application/product meets the business and technical requirements that guided its design and development, so that it works as expected and can be implemented with the same characteristics.

Software Testing, depending on the testing method employed, can be implemented at any time in the development process, however the most test effort is employed after the requirements have been defined and coding process has been completed.

6.1 UnitTesting:

The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect. Each unit is tested separately before integrating them into modules to test the interfaces between modules. Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended. Its implementation can vary from being very manual (pencil and paper) to being formalized as part of build automation.

6.2 IntegrationTesting:

Integration testing, also known as integration and testing (I&T), is a software development process which program units are combined and tested as groups in multiple ways. Integration testing can expose problems with the interfaces among program components before trouble occurs in real-world program execution. There are two major ways of carrying out an integration test, called the bottom-up method and the top-down method. Bottom-up integration testing begins with unit testing, followed by tests of progressively higher-level combinations of units called modules or builds. In top-down integration testing, the highest-level modules are tested first and progressively lower-level modules are tested after that. In a comprehensive software development environment, bottom-up testing is usually done first, followed by top-down testing.

6.3 Validationtesting:

At the validation level, testing focuses on user visible actions and user recognizable output from the system. Validation testing is said to be successful when software functions in a manner that can be reasonably expected by the customer. Two types of validation testing

- **Alpha testing** is simulated or actual operational testing by potential users/customers or an independent test team at the developers' site. Alpha testing is often employed for off-the-shelf software as a form of internal acceptance testing, before the software goes to beta testing.
- **Beta testing** comes after alpha testing. Versions of the software, known as beta version, are released to a limited audience outside of the programming team. The software is released to groups of people so that further testing can ensure the product has few faults or bugs. Sometimes, beta versions are made available to the open public to increase the feedback field to a maximal number of future users.

6.4 Testcases:

This software explains Test cases for amazon Online Shopping Web Application testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. Testing presents an interesting of a system using various test data. Preparation of the test data plays a vital role in the system testing. After preparation, the test data, the system under study is tested those test data. Errors were found and corrected by using the following testing steps and corrections are recorded for future references. Thus, series of testing is performed on the system before it is already for implementation.

The development of software systems involves a series of production activities where opportunities for injection of human errors are enormous. Errors may begin to occur at the very inception of the process where the objectives may be erroneously or imperfectly specified as well as in later design and development stages. Because of human in ability to perform and communicate with perfection, software development is followed by assurance activities ER Diagram for Customer.

6.4.1 Objectives:

The test plan for the system should support following objectives:

- Identify which features of the system will be tested.
- Define the pass/fail criteria for each feature to be tested.
- Specify the testing approaches that will be used during testing.
- Identify the deliverables of the testing process.

6.4.2 Description:

- **Test Steps** - List all the test execution steps in detail. Write test steps in the order in which they should be executed. Make sure to provide as many details as you can.
- **Test Data** - Use of test data as an input for this test case. You can provide different data sets with exact values to be used as an input.
- **Expected Result** - What should be the system output after test execution? Describe the expected result in detail including message/error that should be displayed on the screen.
- **Actual result** - Actual test result should be filled after test execution. Describe system behavior after test execution.
- **Status (Pass/Fail)** - If actual result is not as per the expected result mark this test as **failed**. Otherwise, update it as **passed**.
- **Notes/Comments/Questions** - If there are some special conditions to support the above fields, which can't be described above or if there are any questions related to expected or actual results then mention them here.

6.4.3 Features to be tested

This section outlines all the features that will be tested:

Type of User	Feature Identifier	Description
User	Case-1	System Register
	Case-2	Edit Shopping Cart
	Case-3	Add to Cart
	Case-4	Place Order
Seller	Case-5	Create and Delete product from Category
	Case-6	Manage Orders

CASE 1: System Register

Purpose: Test that users can register with the proper username and password

1. Visit Customer's Login web page
2. Enter First name, Last name, username
3. Enter password, confirm password, e-mail, address
4. Security question and Security answer
5. Click Signup button

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1	Enter empty value for <i>First Name</i>	Display error message to enter some valid text.	Error: "Name is required, Please Enter Correct details".	✓		
2	Enter empty value for <i>Last Name</i>	No error	No error	✓		
3	Enter empty value for <i>User ID</i>	Display error message to enter some valid text.	Error: "This Field is required".	✓		
4.	Enter a username already in use with an existing user	Display error message to the user and he should not be allowed to register for an account with that username.	Error Message: "The input username is already taken please enter some other username".	✓		

5.	Enter a Username not in use with other existing users.	User should be able to register with the website and directed to the secure Web page requested.	No Error	✓		
6.	Enter empty value for <i>Contact Number</i>	Display error message "Enter valid contact number"	Error message: "contact"	✓		
7.	Enter non numeric value in <i>Contact Number</i> field like abcde	Display error message to enter a valid contact number"	Error message: "Invalid contact number, Please enter correct Phone no."	✓		
8.	Enter empty value for <i>Email Address</i>	Display error message "Enter a valid address"	Error message: "Enter a valid address"	✓		
9.	Enter empty value for <i>Confirm Email Address</i>	Display error message "Enter your email address"	Error message: "Enter your email address"	✓		
10	Enter different email format like Yahoo.com.sg	Display error message "Enter a valid email Address"	Error message "Enter a valid email Address"	✓		
11.	Enter empty value for either <i>Password</i> or <i>Confirm Password</i>	Display error message "Password does not match."	Error message: "Password does not match."	✓		
12.	Enter password less than 8 characters long	123456	Error message "Password length must be at least 8 characters"	✓		
13.	Enter empty value for <i>Address</i>	No error	No error	✓		

CASE 2: Edit Shopping Cart

Purpose: Test that clicking Update Quantities will update the cart summary accordingly.

1. After selecting a product, go to Shopping cart Webpage.
2. Check after entering incorrect input, an appropriate message should be displayed.
3. If entered a valid number, check if the total quantity and relative price is updated after clicking update or delete button.

S/N	Input data	Expected output	Actual Result	Pass	Fail	Remarks
1.	Negative input number or input other than integer number in "Quantity" field	Display error message to notify the given input is invalid	Error message: "Please input valid no. of items"	✓		
2.	Enter a Positive integer number in the "Quantity" field	The product quantity should be updated or deleted according to the specified input number	No Error. The item is updated successfully	✓		

CASE 3: Add to Cart

Purpose: Test that clicking add to cart button, product is getting added in the cart

1. Click Add to Cart button
2. Check whether the cart shows the product
3. Check if the quantity count is one, if product is not already in the cart.
4. Check if the quantity count is increased by one, if product is already in the cart

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1.	Click "Add to Cart" Button for product not already in the cart.	The product should be moved to wish list.	No Error. "The product is successfully added to Cart"	✓		
2.	Click "Add to Cart" Button for product already in the cart.	If product already exists then its quantity is updated by 1.	No Error. "The product quantity is successfully increased by one in the Cart"	✓		

CASE 4: Place Order

Purpose: Test that user can place an order with valid profile information

1. Select a product to order.
2. Click Place Order button with one of the fields empty.
3. Check if the message is to enter all required fields.

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1.	An empty required field. (First name and last name, Street address, city, state, zip code, email, credit card information, shipping address)	An appropriate message should be displayed and the user should not be allowed to place the order.	Error message: "Please enter all the required details to place the order"	✓		
2.	The user should be able to place an order and redirected to the order confirmation page.	The product quantity should be updated or deleted according to the specified input number.	No Error. The product is ready to be shipped within the week.	✓		

CASE 5: Edit a Product from Category for Seller

Test that after clicking Delete Product, Update Product, selected product is being removed from the catalog. The carts before and up to that date are removed

1. Click Delete Product button
2. Check for the message- product has been removed from the category.
3. Check that after clicking Show products in category, the deleted product is not shown.

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1.	The seller selects a product to be deleted, the category from which it is to be deleted and clicks delete button	The product should be deleted from the category only for that category.	Error message: Product is successfully removed from that category but not from others, if it belongs to more than one category"	✓		
2.	Negative input number or input other than integer	Display error message to	Error message: "Please input	✓		

	number in "Quantity" field	notify the given input is invalid	valid no. of items"			
3.	Enter a Positive integer number in the "Quantity" field	The product quantity should be updated or deleted according to the specified input number	NoError:"The item is updated successfully"	✓		
4.	Enter a Positive integer number in the "Price" field	The product price should be updated according to the specified input number	NoError:"The item is updated successfully"	✓		

CASE 6: Manage Orders

Purpose: Test that order status is correctly updated

1. Select the order from orderlist
2. Click Edit button
3. Check for the message- Order status successfullyupdated
4. Check that status of other orders in database isunchanged

S/N	Input data	Expected Results	Actual Results	Pass	Fail	Remarks
1.	Enter a Positive integer number in the "Price" field	The product price should be updated according to the specified input number	NoError:"The item is updated successfully"	✓		
2.	Seller will enter the date range, number of orders and type of order, to view the existing orders in database. He/she will then select the order for which the status is to be updated where its details could be edited.	The status of that order is updated, but the status of all other orders remains same	NoError:"The Details of the Product is successfully Updated"	✓		

6.4.4 Approach:

Only functional black box testing will be performed to test the functionality of the system. The features mentioned above describe how the user will interact with the system, so the testing will require the tester to interact with the system in the same way a typical user would. The user actions will be simulated through a set of test scenarios. Each scenario will trace back to a requirement listed in the Vision Document.

6.5 PerformanceTesting:

This will be performed to test the entire system to see whether all driving requirements are satisfied. Allowing multiple users log into the system and perform the operations at the same time using the JMeter testing tool will do this. This test verifies that the components of the systems meet the stated requirements for speed.

The following components of the system would be analyzed for performance:

- Buying: Includes browsing the catalog, selecting a product, add to shopping cart, checkout, enter personal details and place order.
- Searching for a Product.
- Getting Product recommendation

Using JMETER tool, approximately 100 concurrent virtual users with a minimum of 50 requests per user will be inputted to calculate the response times for each of the above components. The above tests would be done in different environments like:

- Local connection of
- 54Mbps LAN connection of
- 100Mbps Wired connection

6.6 Suspension Criteria and ResumptionRequirements:

6.6.1. SuspensionCriteria:

If a test case fails, testing will be suspended for all dependent features. The failed test case will be logged into a test log along with a description of the failure.

6.6.2. ResumptionRequirement:

Test cases, not dependent on the case in which a bug is reported, will continue to be executed in parallel to bug fixing. Testing for the failed test case will resume after the bug has been identified and resolved.