



Project Report On
F4F : Freedom for Fashion

By
Dharmesh Patel (16102012)
Bhargav Borse (16102023)
Under Guidance
of
Mr. Akil Surti

Submitted to Bachelor of Computer Applications Program
in partial fulfillment of the requirements
for the award of degree
Bachelor of Computer Applications



Navrachana University
Vadodara
May 2019

ACKNOWLEDGEMENT

The success of any project depends largely on the efforts, encouragement and guidelines of others. With a sense of gratitude and respect, we would like to extend our heartiest thanks to those who provided help and guidance to during our project.

We would like to show our greatest gratitude to Mr. Dhaval Mehta and Ms. Hemal Shah for their valuable suggestions and guidance. Without their encouragement and guidance this project have not been materialized.

We would like to thank Mr. Akil Surti for allowing us to work as an intern in Enlighten InfoSystems. We thank them for their constant support and motivation.

We also express our gratitude to our college who were involved in other projects. We would like to thank Navrachana University and our faculty members for helping us.

ABSTRACT

Now a day's people are having problem in buying new costumes for theatre play, marriages, fancy dress competition and many other occasions. They have to pay high amount for these costumes. They can't afford to give this much of money so we F4F (Freedom for Fashion) are providing costumes (Animal, Food, Historical, Superhero, Uniform, etc.), dresses (Suit, Sherwani, Saree, Kurtis etc.) and accessories (Earing, Bracelet, bow tie, etc.) to customers on rent. We are making every attempt to bring an extensive variety of fashion at a fair price. With prices on rise and quality on a decline, we are taking a surprisingly different approach, we strive to bring the highest in quality at the lowest of prices. We deliver directly to your door anywhere in Vadodara to both residential and business addresses.

COMPANY PROFILE



Enlighten Infosystems is a young and dynamic IT software development, professional services and training & consultancy firm in Vadodara that works with Enterprises, companies and businesses.

As a privately-owned firm, Enlighten Infosystems provides software design, development, services and maintenances to the following sector / verticals:

- Financial Services
- Education
- Retail & Services
- Hospitality & Travel
- Health Care
- Manufacturing & Engineering
- Technology

Enlighten Infosystems follows very simple principal:

- Focus on Quality
- Use of latest tools and technology
- Development updates to our esteemed clients

Mission:

Enlighten Infosystems mission is to develop quality software product and provide extended services and support to their esteemed clients.

Vision:

To be a leading IT Software development and services company for Indian and International Clients.

What We Do:

- Product Development
- Software Development
- Mobile Application Development
- IT Training & Consultancy

Company Portfolio:

- Quick Connect
- Jeevan Udaan Consultancy
- Quick Buy
- PriceBuzz
- MobileDoc
- Mahakali Auto Service
- MobileCare
- Food Desk
- Aryan Music
- Digital Society

LIST OF TABLES

Table 1: User Characteristics	3
Table 2: Admin Table	39
Table 3: Customer table	39
Table 4: Item table	40
Table 5: Cart table	42
Table 6: Payment table	42
Table 7: Order table	43
Table 8: Deliveryman(Rent) Acceptancy	44
Table 9: Deliveryman(Buy) Acceptancy	45
Table 10: Feedback table	46
Table 11: Deliveryman table	47
Table 12: Timeline Chart	71
Table 13: Implementation Environment	72
Table 14: End Users Module specification	73
Table 15: Admin Module specification	74
Table 16: Test Case: User Login	86
Table 17: Test Case: Update Details	85
Table 18: Test Case: Men's Clothing	87
Table 19: Test Case: Women's Clothing	87
Table 20: Test Case: Kid's Clothing	88
Table 21: Test Case: Send Feedback	88
Table 22: Test Case: Checkout.....	89
Table 23: Test Case: Invoice	89
Table 24: Test Case: Admin Login	90
Table 25: Test Case: Add Item (Admin)	90
Table 26: Test Case: View Item (Men's Fashion)	91
Table 27: Test Case: View Item (Women's Fashion)	91
Table 28: Test Case: View Item (Boy's Fashion)	92
Table 29: Test Case: View Item (Girl's Fashion)	92
Table 30: Test Case: Add Deliveryman.....	93

Table 31: Test Case: Deliveryman Details	93
Table 32: Test Case: Current Status	94
Table 33: Test Case: Order Details.....	94
Table 34: Test Case: Previous History	95
Table 35: Test Case: Previous History Details.....	95
Table 36: Test Case: Feedback.....	96
Table 37: Test Case: Deliveryman Login	96
Table 38: Test Case: Delivery Details (Deliveryman)	97
Table 39: Test Case: Order Details (Delivery Order).....	97
Table 40: Test Case: Pickup Details (Deliveryman)	98
Table 41: Test Case: Order Details (Pickup Order)	98

LIST OF FIGURES

Figure 1: Software Development Life Cycle	6
Figure 2: Spiral Model.....	9
Figure 3: Use Case Diagram	13
Figure 4: ER Diagram.....	14
Figure 5: Class Diagram	15
Figure 6: Database	16
Figure 7: Admin_Order(Database).....	17
Figure 8: Admin_Previous_History(Database).....	18
Figure 9: Deliveryman Details(Database)	18
Figure 10: Feedback(Database).....	19
Figure 11: Boy's Item(Database)	19
Figure 12: Girl's Item(Database)	20
Figure 13: Men's Item(Database).....	20
Figure 14: Girl's Item(Database)	21
Figure 15: User(1) (Database).....	21
Figure 16: User(2) (Database).....	22
Figure 17: User Login Activity Diagram.....	23
Figure 18: Dress Selection Option Activity Diagram	24
Figure 19: Add To Cart Activity Diagram.....	25
Figure 20: Checkout Activity Diagram	26
Figure 21: Admin Login Activity Diagram	27
Figure 22: Manage Item Activity Diagram.....	28
Figure 23: Rental Details Activity Diagram	29
Figure 24: Penalty Details Activity Diagram.....	30
Figure 25: Deliveryman Login Activity Diagram	31
Figure 26: Delivery Details Activity Diagram.....	32
Figure 27: Return Item Activity Diagram.....	33
Figure 28: Order Item Sequence Diagram	34
Figure 29: Rent Item Sequence Diagram.....	35

Figure 30: Adding Deliveryman Sequence Diagram	36
Figure 31: Manage Item Sequence Diagram.....	37
Figure 32: Control Flow Diagram	38
Figure 33: Flow Chart.....	48
Figure 34: Main Page User	52
Figure 35: User Login.....	53
Figure 36: Profile page(User).....	54
Figure 37: Deals Of The Day(User)	55
Figure 38: Update Details Page(User)	56
Figure 39: Men/Women/Kids Page(User)	57
Figure 40: Feedback(User).....	58
Figure 41: Checkout(User).....	59
Figure 42: Order Details(User)	60
Figure 43: Admin Login	60
Figure 44: Add Item(Admin)	61
Figure 45: View Item Men(Admin).....	61
Figure 46: View Item Women(Admin)	62
Figure 47: View Item Boy(Admin)	62
Figure 48: View Item Girl(Admin)	63
Figure 49: Update Item	63
Figure 50: Current Orders(Admin).....	64
Figure 51: Order Details(Admin)	64
Figure 52: Previous History(Admin)	65
Figure 53: Previous History Details(Admin)	65
Figure 54: Add Deliveryman	66
Figure 55: Deliveryman Details	66
Figure 56: Feedback	67
Figure 57: Deliveryman Login	67
Figure 58: Delivery Order(Deliveryman)	68
Figure 59: Order Details – Delivery Order(Deliveryman)	69
Figure 60: Pickup Order(Deliveryman)	69

Figure 61: Order Details – Pickup Order(Deliveryman)	70
Figure 62: User Login.....	99
Figure 63: Welcome Details.....	99
Figure 64: After Login	100
Figure 65: Deals Of the Day	100
Figure 66: Men's Clothing.....	101
Figure 67: Women'Clothing	102
Figure 68: Kid's Clothing	103
Figure 69: Checkout Page	104
Figure 70: Invoice.....	105
Figure 71: Admin Login	105
Figure 72: Admin Dashboard.....	106
Figure 73: Add Item	106
Figure 74: View Item(Men's Fashion)	107
Figure 75: View Item(Women's Fashion)	107
Figure 76: View Item(Boy's Fashion)	108
Figure 77: View Item(Girl's Fashion)	108
Figure 78: Update Item	109
Figure 79: Add Deliveryman	109
Figure 80: Deliveryman Details	110
Figure 81: Current Orders	110
Figure 82: Order Details	111
Figure 83: Previous History	111
Figure 84: Previous History Order Details	112
Figure 85: Feedback	112
Figure 86: Deliveryman Login	113
Figure 87: Delivery Order.....	114
Figure 88: Pickup Order.....	114
Figure 89: Delivery/Pickup Order Details	115

Freedom For Fashion

CERTIFICATE OF NON-DISCLOSURE OF CODE



**Navrachana University
Vadodara**

CERTIFICATE OF COMPLETION

This is to certify that report submitted along with the project entitled **Freedom For Fashion(F4F)** has been carried out by **Dharmesh Patel (16102012), Bhargav Borse (16102023)** for partial fulfillment of the degree Bachelor of Computer Applications from Navrachana University, Vadodara during the academic year 2018-19.

Ms. Hemal Shah
(Internal Reviewer)

Mr. Dhaval Mehta
(Program Coordinator)

COMPANY PROJECT COMPLETION CERTIFICATE

Freedom For Fashion

Table of Contents

1.	PROJECT INTRODUCTION.....	1
1.1	Business Problem	1
1.2	Project Summary	1
1.3	Purpose.....	2
1.4	Scope.....	2
2.	SYSTEM REQUIREMENTS STUDY	3
2.1	User Characteristics	3
2.2	Hardware and Software Requirements	5
2.3	Software Development Life Cycle	6
2.4	Process Model	9
3.	SYSTEM ANALYSES.....	11
3.1	Study of Current System.....	11
3.2	Problem and Weaknesses of Current System	11
3.3	Requirements of New System.....	11
3.4	Feasibility Study.....	12
3.5	Requirements Validation	12
3.6	Functions of System	13
3.6.1	Use Case Diagram	13
3.7	Data Modeling.....	14
3.7.1	E-R diagram	14
3.7.2	Class diagram.....	15
3.7.3	Database diagram	16
3.8	Functional and Behavioral Modeling	23
3.8.1	Activity Diagram.....	23
3.8.2	Sequence Diagram.....	34
3.8.3	Control flow diagram	38
4.	SYSTEM DESIGN.....	39
4.1	Tables and Relationship.....	39
4.2	System Procedural Design	48

4.3	Design Pseudo Code	49
4.4	Input/ Output and Interface Design (wireframes)	52
4.5	TimeLine Chart	71
5.	IMPLEMENTATION.....	72
5.1	Implementation Environment.....	72
	TOOLS.....	72
	TECHNOLOGIES.....	72
5.2	Module Specification.....	73
5.3	Security Features	75
	Authentication:	75
	Authorization:	76
	Data validation:	76
5.4	Coding Standards	76
5.5	Coding Sample	77
6.	TESTING.....	82
6.1	Test Plan.....	82
6.2	Test Strategy.....	83
6.3	Test Methods	84
6.4	Test Cases	86
7.	SCREENSHOTS	99
8.	LIMITATIONS AND FUTURE ENHANCEMENTS.....	115
	CONCLUSION	116
	References	117

1. PROJECT INTRODUCTION

1.1 Business Problem

Now a day's people are having problem in buying new costumes for theatre play, marriages, fancy dress competition and many other occasions. For the short-term event they have to pay high amount for these costumes moreover sometimes they can't afford to give this much of money.

1.2 Project Summary

As a **Freedom For Fashion**, we will rent out costumes, different type of dresses and accessories for as many days as customer demands and charge a certain percentage of the retail price as the fee. Customer have to pay 30% of amount of order as deposit and give an id proof copy at the delivery time. We will deliver the order in 48 hours. The rental fee charged covers the service charges, dry cleaning, and other minor expenditures. When the order is placed, the product will be dispatched from company's inventory. After wearing the costume, we give that costume for dry cleaning, iron and then we provide that costume to another customer. There is also the provision of holding an extra amount during the full payment period to cover any sort of accidental damage. After wearing costume for on time the price of that costume will be decreased. There will be another option for customer to buy that costume or accessory.

1.3 Purpose

The main purpose of **Freedom For Fashion** to provide good items on rent to customer that they can't afford to buy for one short-term event. We are selling used products what we give on rent, so if they want to buy that product we will give them that product on affordable price. They have to pay high amount for these costumes, so we are making every attempt to bring an extensive variety of fashion at a fair price. With prices on rise and quality on a decline, we are taking a surprisingly different approach, we strive to bring the highest in quality at the lowest of prices.

1.4 Scope

The scope of the **F4F (Freedom for Fashion)** is providing facilities to users across Vadodara city to give costumes, dresses and accessories on rent. Registered customer can manage and order item. Customer can make payment by cash on delivery. Customer will get notification of order confirmation. Admin will manage the whole system from adding, updating or removing items. It also manages payments, users, order and Deliveryman. It also can view Feedback of customers and answer them. Deliveryman will login through registered email that was given by admin. After that they will deliver the product on respective address of order assign by admin. Guest User can view items and can login, Without login guest user cannot able to access whole system.

2. SYSTEM REQUIREMENTS STUDY

2.1 User Characteristics

There are four types of systems that interact with the system: Guest User, Registered User, Deliveryman and Admin. Each type to user has different use of the system and so the requirements differ as per their role

Table 1: User Characteristics

USER	CHARACTERISTICS
GUEST USER	<ul style="list-style-type: none"> • Can login • Can View Item • Can View Website
REGISTERED USER	<ul style="list-style-type: none"> • Can login • Can View Item • Can Add Item In The Cart • Can Manage Item In Cart • Can Checkout • Can Give Feedback • Can manage their profile • Can view their order history
ADMIN	<ul style="list-style-type: none"> • Can login • Admin will manage the users and maintain the users data. • Admin can also manage the orders. • Can receive the feedbacks • Admin can also add the deliveryman • Admin can assign the deliveryman of order • Admin can change password

Deliveryman	<ul style="list-style-type: none">• Can deliver the product• Can collect deposit and penalty• Can update the delivery status• Can change the password
-------------	--

2.2 Hardware and Software Requirements

2.2.1 Developers Requirements

Table 2: Developers Requirements

Tools	Visual Studio Code
Hardware	32/64-bit machine
OS	windows 7 or above <i>Phone:</i> Android 6.0.1 or above
Ram	4GB recommended

2.2.2 Client-End Requirements

Table 3: Client-End Requirements

Equipment	Android Phone/ PC
OS	Android 6.0.1 and above/ windows 7 or above
System connection	connection to internet

2.3 Software Development Life Cycle

SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.

A typical Software Development Life Cycle consists of the following stages –

Stage 1: Planning and Requirement Analysis

Stage 2: Defining Requirements

Stage 3: Designing the Product Architecture

Stage 4: Building or Developing the Product

Stage 5: Testing the Product

Stage 6: Deployment in the Market and Maintenance

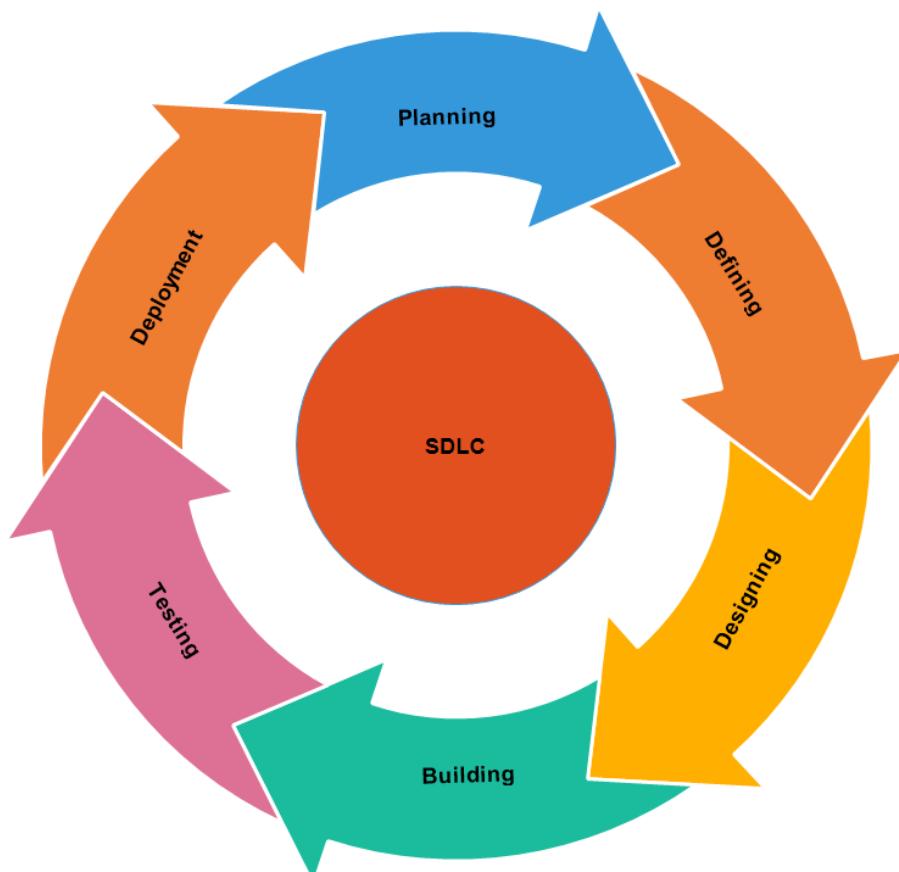


Figure 1: Software Development Life Cycle

STAGE 1: PLANNING AND REQUIREMENT ANALYSIS

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas.

Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

STAGE 2: DEFINING REQUIREMENTS

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

STAGE 3: DESIGNING THE PRODUCT ARCHITECTURE

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS - Design Document Specification.

This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product.

A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

STAGE 4: BUILDING OR DEVELOPING THE PRODUCT

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle.

Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

STAGE 5: TESTING THE PRODUCT

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

STAGE 6: DEPLOYMENT IN THE MARKET AND MAINTENANCE

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing).

Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

(Elliot, 1970)

2.4 Process Model

The process model used is Spiral model. The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral.

This model is best used for large projects which involves continuous enhancements. There are specific activities which are done in one iteration (spiral) where the output is a small prototype of the large software. The same activities are then repeated for all the spirals till the entire software is build.

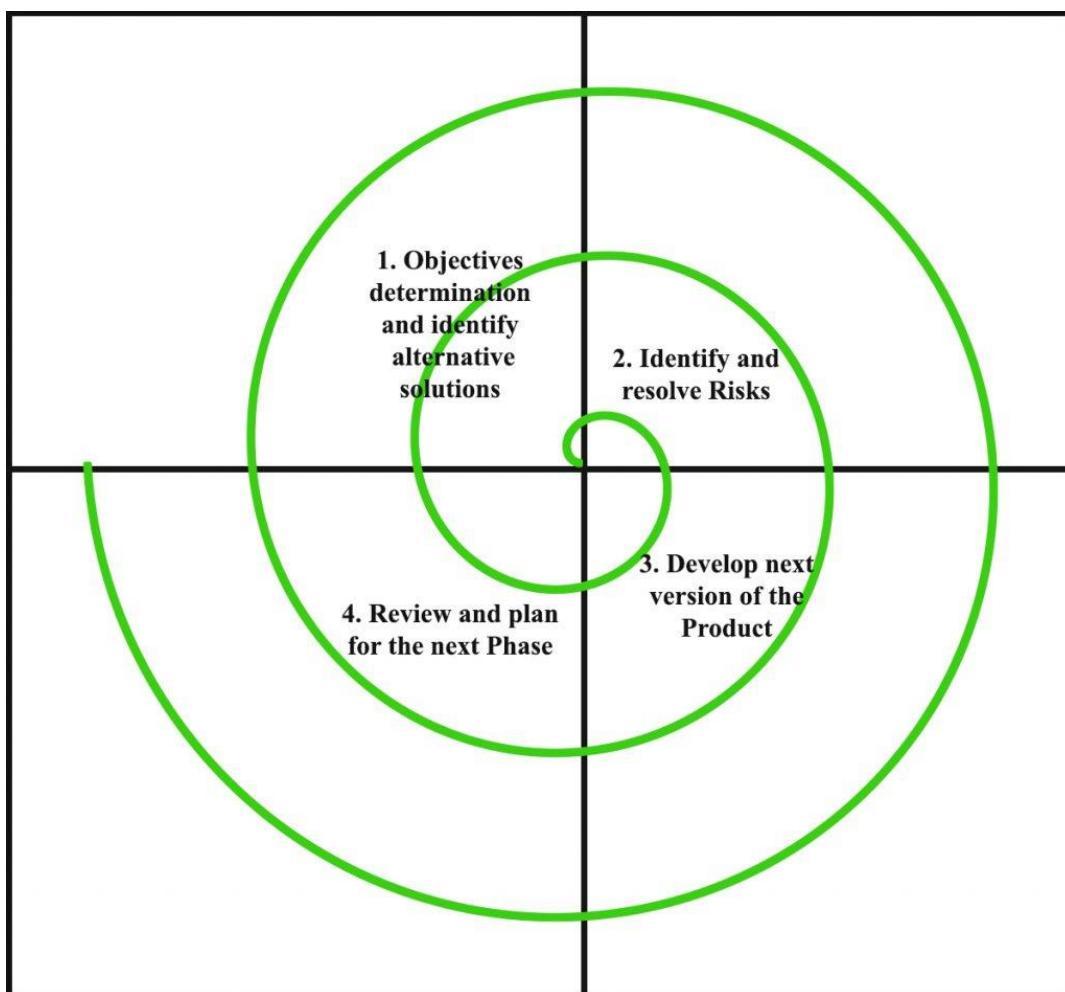


Figure 2: Spiral Model

The Spiral Model is widely used in the software industry as it is in sync with the natural development process of any product, i.e. learning with maturity which involves minimum risk for the customer as well as the development firms.

The following pointers explain the typical uses of a Spiral Model –

- When there is a budget constraint and risk evaluation is important.
- For medium to high-risk projects.
- Long-term project commitment because of potential changes to economic priorities as the requirements change with time.
- Customer is not sure of their requirements which is usually the case.
- Requirements are complex and need evaluation to get clarity.
- New product line which should be released in phases to get enough customer feedback.
- Significant changes are expected in the product during the development cycle.

The advantage of spiral lifecycle model is that it allows elements of the product to be added in, when they become available or known. This assures that there is no conflict with previous requirements and design.

This method is consistent with approaches that have multiple software builds and releases which allows making an orderly transition to a maintenance activity. Another positive aspect of this method is that the spiral model forces an early user involvement in the system development effort.

On the other side, it takes a very strict management to complete such products and there is a risk of running the spiral in an indefinite loop. So, the discipline of change and the extent of taking change requests is very important to develop and deploy the product successfully.

(Strashan, 2004)

3. SYSTEM ANALYSES

3.1 Study of Current System

Currently there are many online shopping sites for shopping different product. The most commonly use online shopping application is Amazon.com, Flipkart.com, etc. This application is having different features like add to cart, add to wish list, Category wise shopping, budget wise shopping, brand wise shopping and many such features.

3.2 Problem and Weaknesses of Current System

Current system doesn't have much problem and weakness but weaknesses that I can observe or I can say is that they don't have rent option on clothes and cheaper price with good quality.

3.3 Requirements of New System

Through our new application, customer can rent items with lower prices and good quality. The main objective of this application is to provide the goods and product all over the city with just one click of a button. In this system user can rent a product by category, rent product by budget, add the product into the cart and later order that product, etc.

3.4 Feasibility Study

When the client approaches the organization for getting the desired product developed, it comes up with rough idea about what all functions the software must perform and which all features are expected from the software.

Referencing to this information, the analysts does a detailed study about whether the desired system and its functionality are feasible to develop.

This feasibility study is focused towards goal of the organization. This study analyzes whether the software product can be practically materialized in terms of implementation, contribution of project to organization, cost constraints and as per values and objectives of the organization. It explores technical aspects of the project and product such as usability, maintainability, productivity and integration ability.

The output of this phase should be a feasibility study report that should contain adequate comments and recommendations for management about whether or not the project should be undertaken.

Feasibility test is conducted by Enlighten InfoSystems and the requirements also provided by company.

(Roy, 2004)

3.5 Requirements Validation

- Password should be of 6 characters or more
- Email Id should be in proper format with “.” And “@” symbols
- Contact number should be of 10 digits
- Contact number should contain numeric values
- Pin-code number should be of 6 numbers
- Customer cannot select previous dates for pick-up date
- Admin cannot select alphabetic values in quantity

3.6 Functions of System

3.6.1 Use Case Diagram

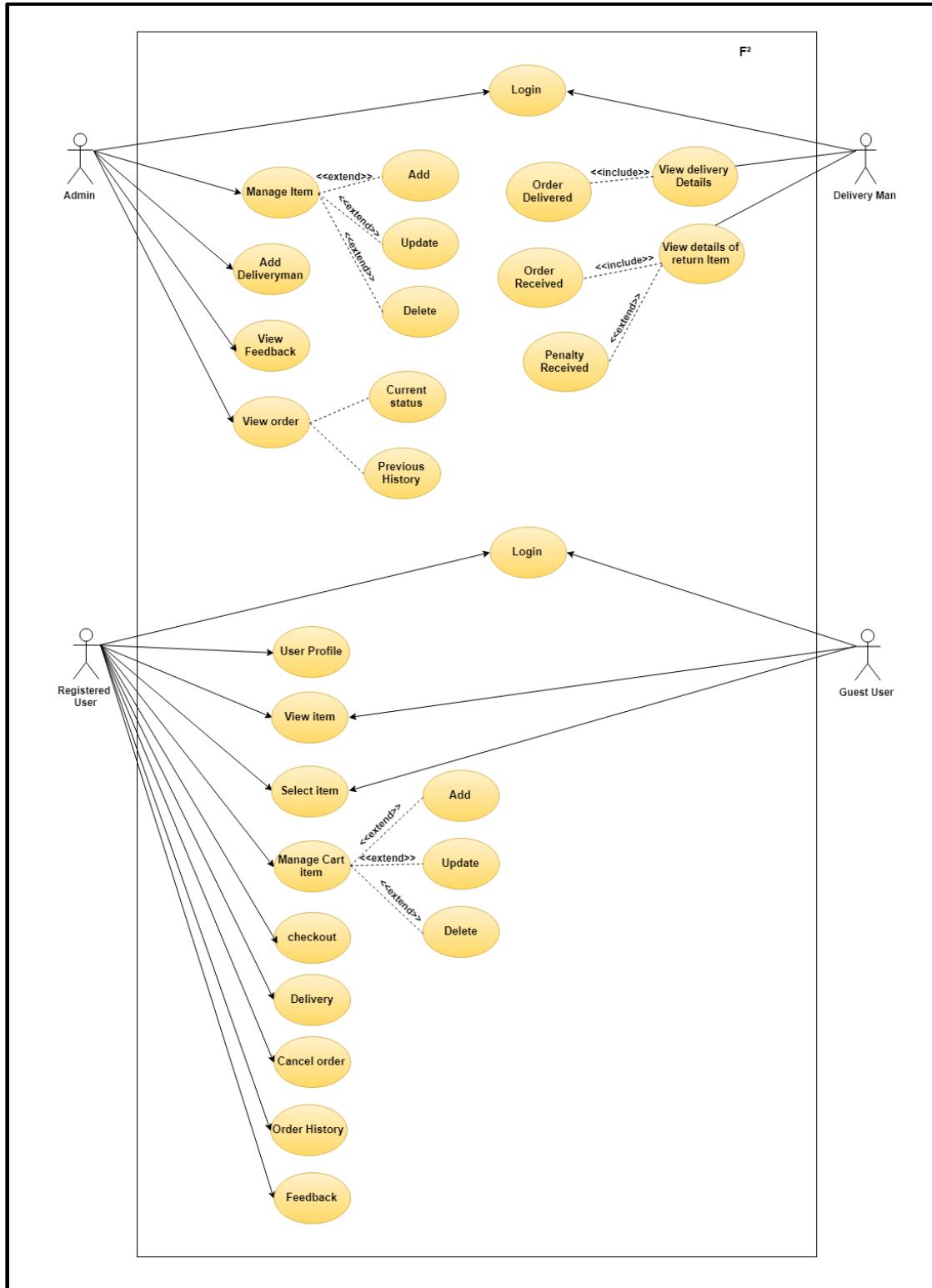


Figure 3: Use Case Diagram

3.7 Data Modeling

3.7.1 E-R diagram

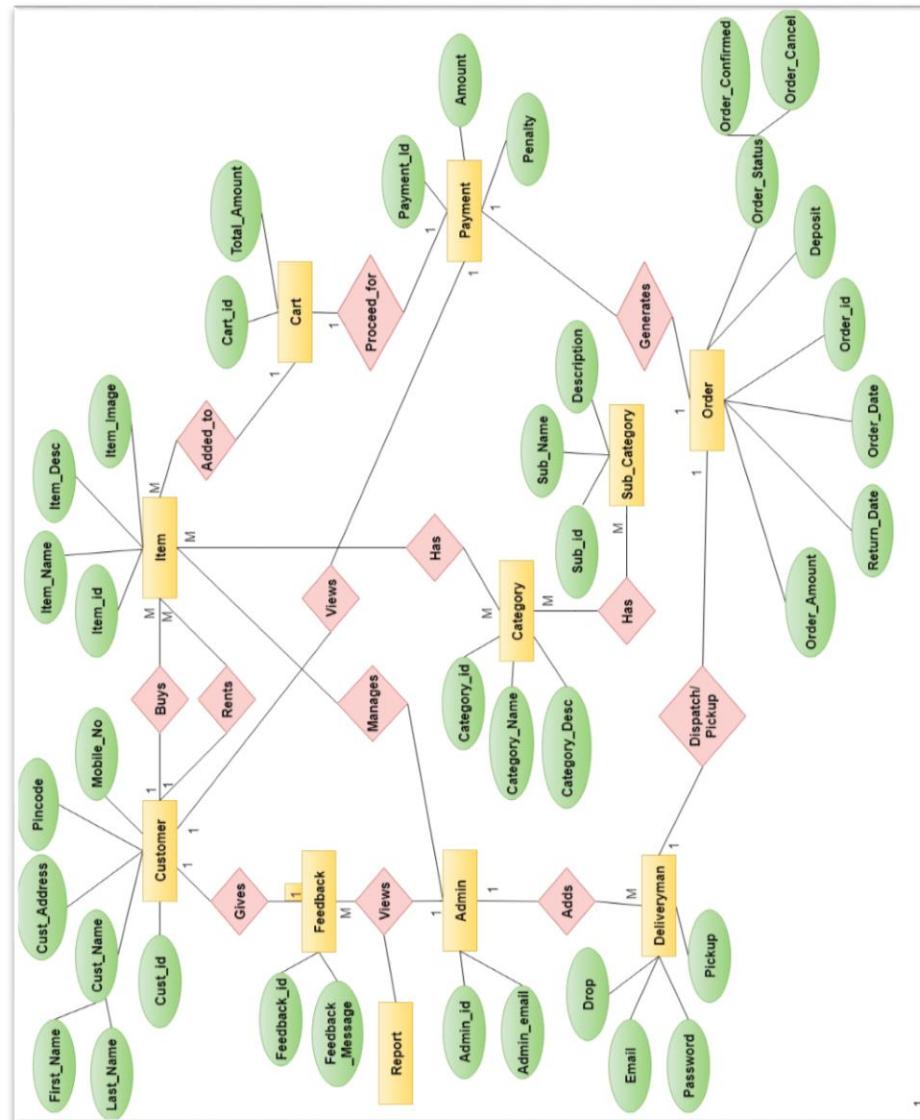


Figure 1: ER Diagram

3.7.2 Class diagram

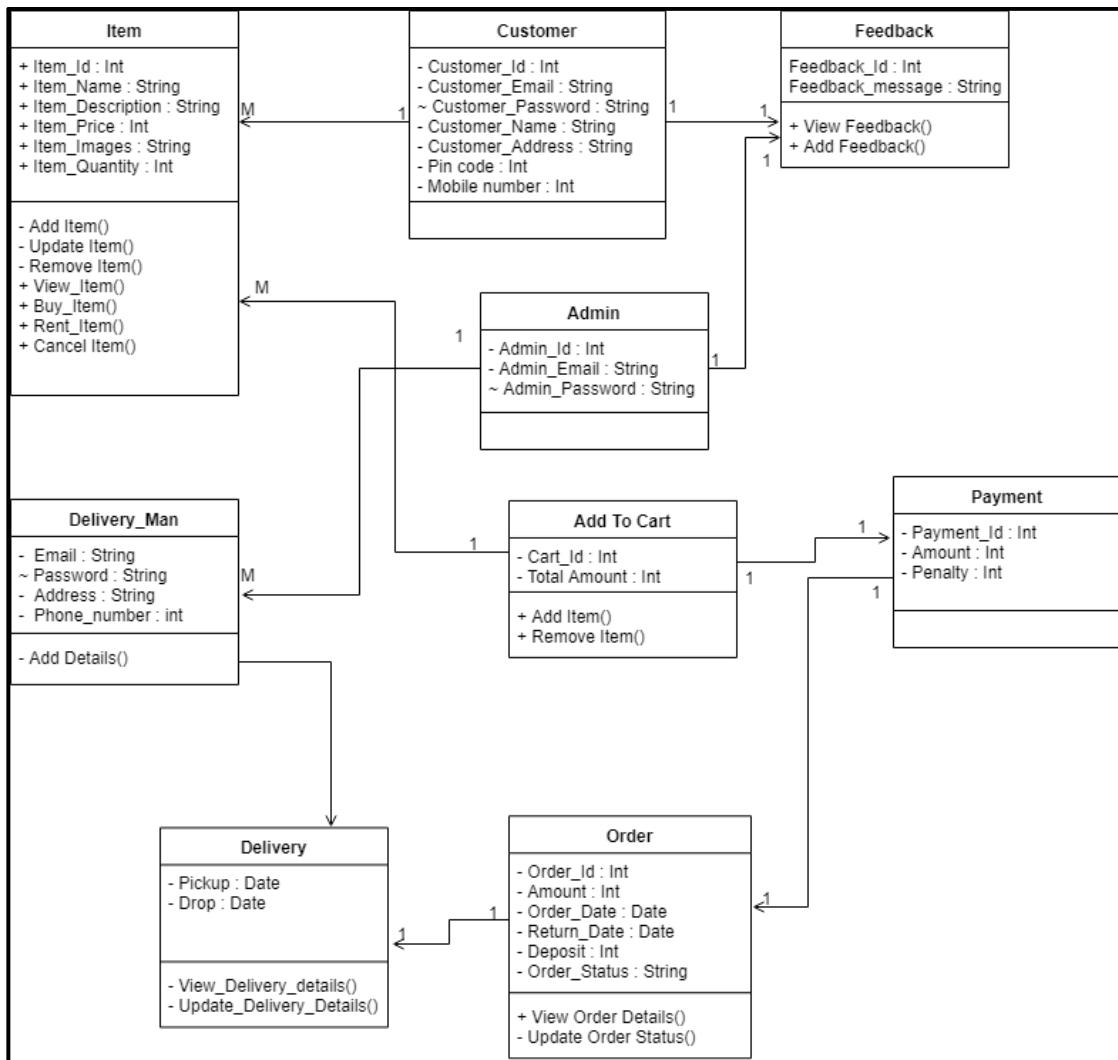


Figure 5: Class Diagram

3.7.3 Database diagram

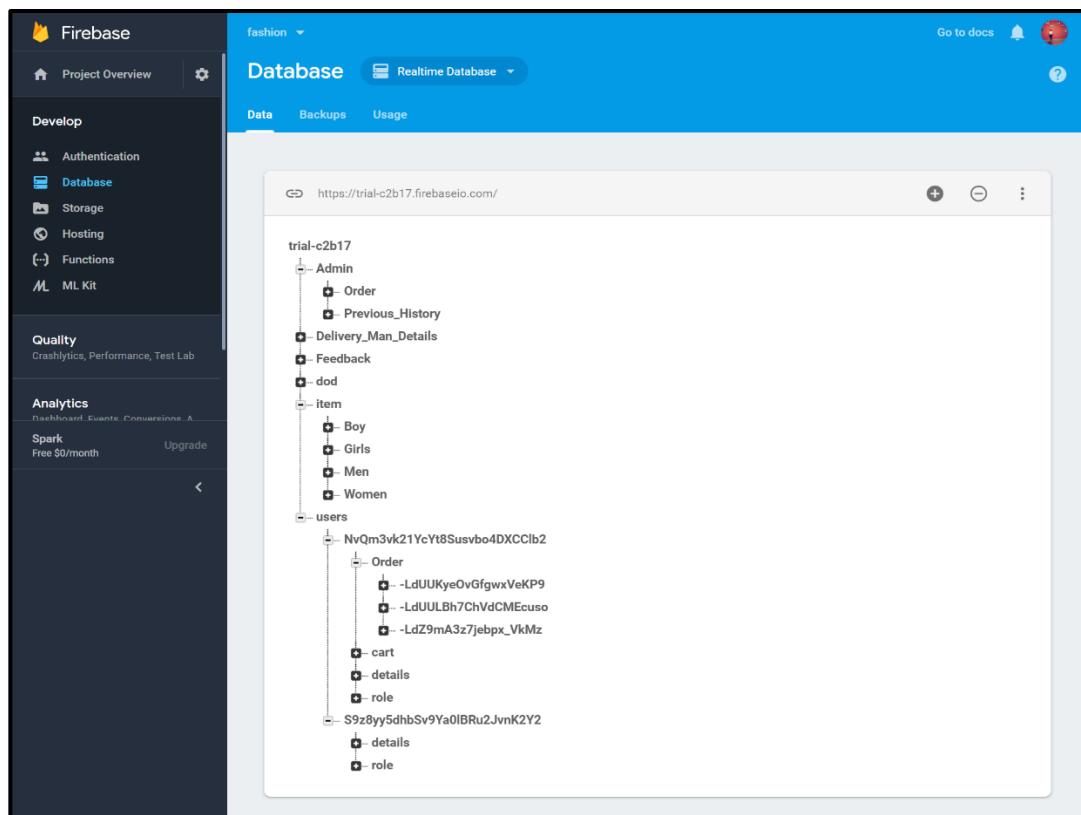


Figure 6: Database

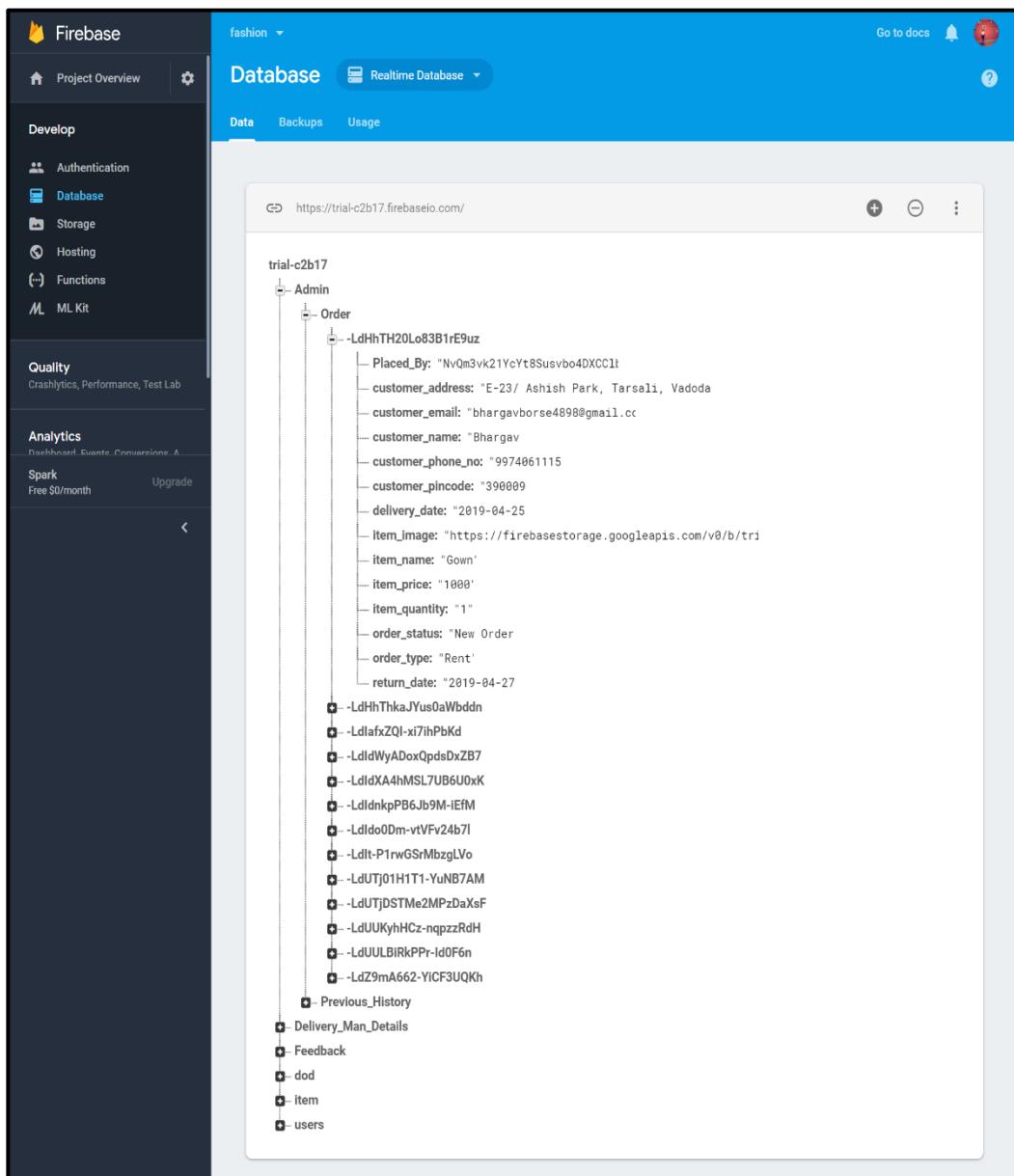


Figure 7: Admin_Order (Database)

Freedom For Fashion

The screenshot shows the Firebase Realtime Database interface for a project named "fashion". The left sidebar shows various development tools like Authentication, Database, Storage, Hosting, Functions, and ML Kit. The main area displays the database structure under the reference "trial-c2b17". The "Order" node contains a child node "Previous_History" which has a child node "-LdGAd-Y2zfyzqyA83g" containing detailed order information. Other nodes include "Delivery_Man_Details", "Feedback", "dod", "item", and "users".

```

trial-c2b17
  - Admin
    - Order
      - Previous_History
        - -LdGAd-Y2zfyzqyA83g
          customer_address: "E-23/ Ashish Park, Tarsali, Vadoda"
          customer_email: "bhargavborse4898@gmail.com"
          customer_name: "Bhargav"
          customer_phone_no: "9974061115"
          customer_pincode: "390009"
          delivery_date: "2019-04-25"
          deliveryman: "bhargavborse4898@yahoo.com"
          item_name: "Saree"
          item_price: "1100"
          item_quantity: "1"
          order_status: "New Order"
          order_type: "Rent"
          return_date: "2019-04-26"
        - -LdGB_GesKnJuW1o-xX1
        - -LdlaownKYna99Bh1c80
      - Delivery_Man_Details
      - Feedback
      - dod
      - item
      - users
  
```

Figure 8: Admin_Recent_History (Database)

The screenshot shows the Firebase Realtime Database interface for a project named "fashion". The left sidebar shows various development tools like Authentication, Database, Storage, Hosting, Functions, and ML Kit. The main area displays the database structure under the reference "trial-c2b17". The "Order" node contains a child node "Delivery_Man_Details" which has a child node "-LdGAd-Y2zfyzqyA83g" containing deliveryman details. Other nodes include "Admin", "Feedback", "dod", "item", and "users".

```

trial-c2b17
  - Admin
  - Delivery_Man_Details
    - -LdGAd-Y2zfyzqyA83g
      Email: "bhargavborse4898@yahoo.com"
      Name: "Bhargav"
  - Order
    - -LdlaownKYna99Bh1c80
      customer_address: "E-23/ Ashish Park, Tarsali, Vadoda"
      customer_email: "bhargavborse4898@gmail.com"
      customer_name: "Bhargav"
      customer_phone_no: "9974061115"
      customer_pincode: "390009"
      delivery_date: "2019-04-25"
      deliveryman: "bhargavborse4898@yahoo.com"
      item_name: "Saree"
      item_price: "1100"
      item_quantity: "1"
      order_status: "New Order"
      order_type: "Rent"
      return_date: "2019-04-26"
      work_order: "delivery"
      address: "hjgvjh"
      id_proof: "https://firebasestorage.googleapis.com/v0/b/trj"
      mobile_number: "9898525252"
    - -LdGB_GesKnJuW1o-xX1
  - Feedback
  - dod
  - item
  - users
  
```

Figure 9 : Deliveryman Details (Database)

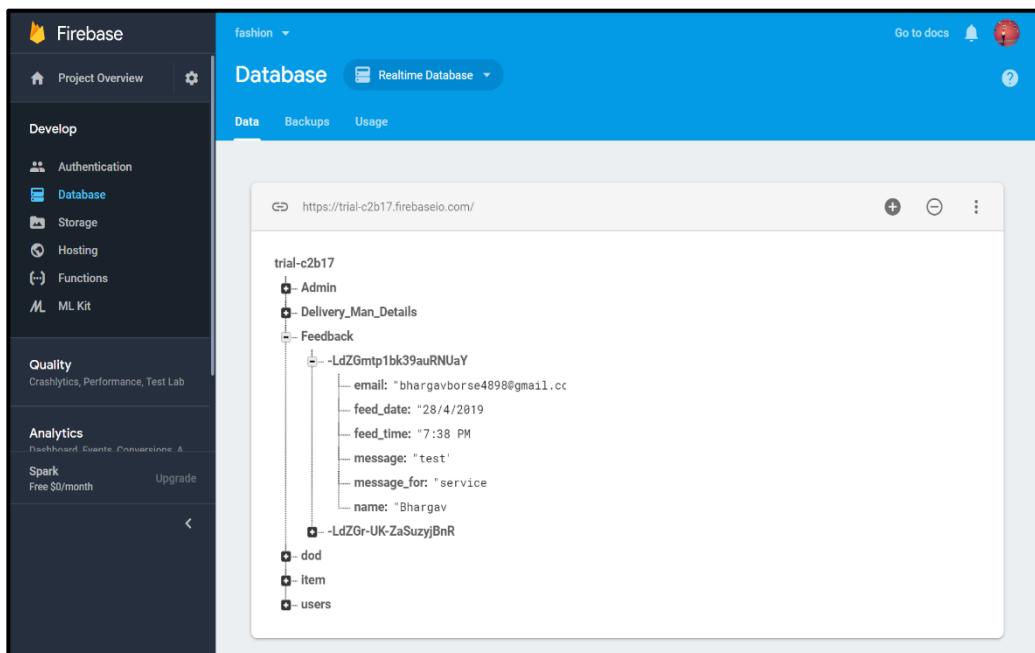


Figure 10: Feedback (Database)

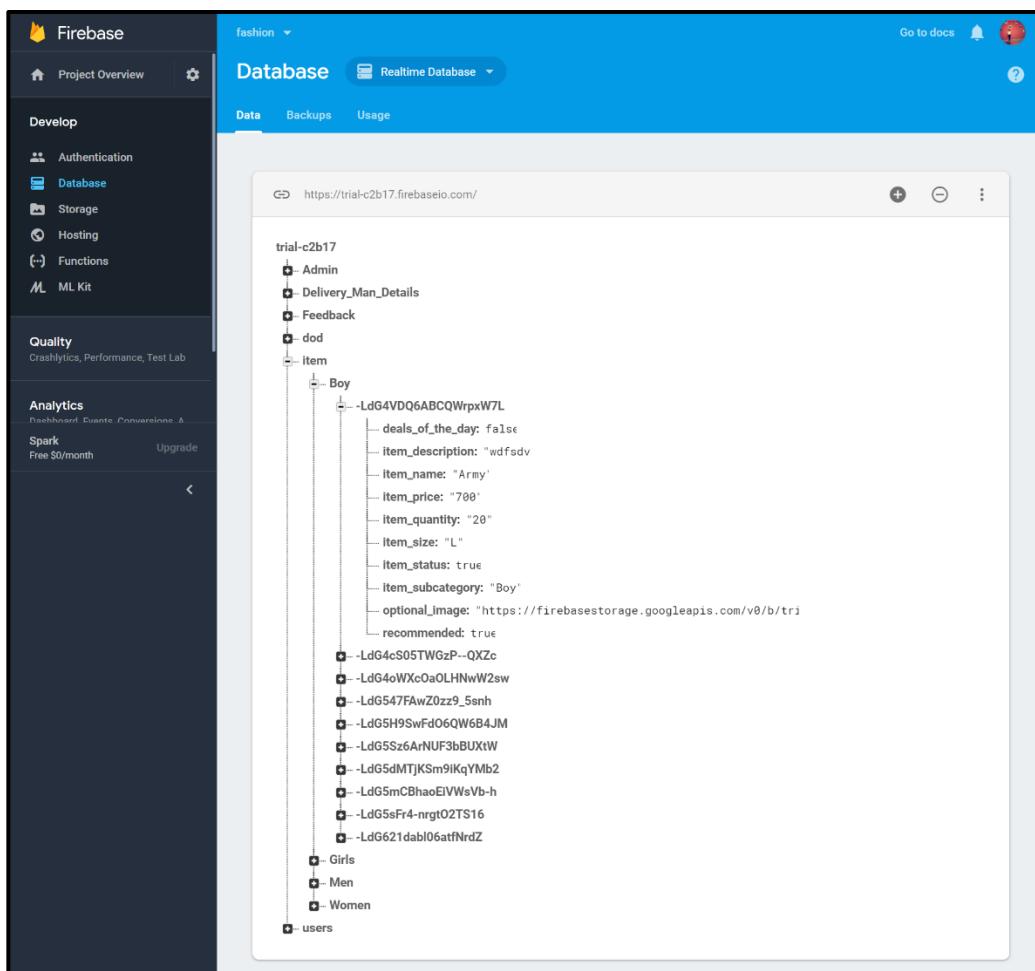


Figure 11: Boy's_Item (Database)

Freedom For Fashion

The screenshot shows the Firebase Realtime Database interface for a project named "fashion". The database structure is as follows:

```

    trial-c2b17
    ├── Admin
    ├── Delivery_Man_Details
    ├── Feedback
    ├── dod
    ├── Item
    │   ├── Boy
    │   ├── Girls
    │   │   └── -Ld6G6GtvI-hOUzzg0bj
    │   │       ├── deals_of_the_day: false
    │   │       ├── item_description: "esfdv"
    │   │       ├── item_name: "Armygirl"
    │   │       ├── item_price: "800"
    │   │       ├── item_quantity: "10"
    │   │       ├── item_size: "S"
    │   │       ├── item_status: true
    │   │       ├── item_subcategory: "Girls"
    │   │       ├── optional_image: "https://firebasestorage.googleapis.com/v0/b/tri"
    │   │       └── recommended: true
    │   ├── Men
    │   └── Women
    └── users
  
```

Figure 12: Girl's_Item (Database)

The screenshot shows the Firebase Realtime Database interface for a project named "fashion". The database structure is as follows:

```

    trial-c2b17
    ├── Admin
    ├── Delivery_Man_Details
    ├── Feedback
    ├── dod
    ├── Item
    │   ├── Boy
    │   ├── Girls
    │   └── Men
    │       └── -LdFxztezsevEcIcWnx8
    │           ├── deals_of_the_day: true
    │           ├── item_description: "fcgh"
    │           ├── item_name: "sherwani"
    │           ├── item_price: "1050"
    │           ├── item_quantity: "10"
    │           ├── item_size: "XL"
    │           ├── item_status: true
    │           ├── item_subcategory: "Sherwani"
    │           ├── optional_image: "https://firebasestorage.googleapis.com/v0/b/tri"
    │           └── recommended: true
    │           ├── -LdfzJEECFnrCIPmGDKI
    │           ├── -Ldfzi4lJ7-W75KHHTUHQq
    │           ├── -LdfzzGL7qduHSummMVa
    │           ├── -Ldg-cDa8v1HDlyQrHnH
    │           ├── -Ldg-zyY-puc0ldla-B6
    │           ├── -Ldg0PzhZimgRWxfBbD9
    │           ├── -Ldg0sJ4n1P8TGm0Z5SA
    │           ├── -LdG14wm5_EbpilAQxIs
    │           └── -LdG1jOrjyILaoqcFFnR
    └── users
  
```

Figure 13: Men's_Item (Database)

Freedom For Fashion

The screenshot shows the Firebase Realtime Database interface. On the left, the project navigation bar includes 'Project Overview', 'Develop' (selected), 'Analytics', 'Quality', and 'Spark'. The main area displays the database structure under the 'fashion' project. The 'Item' node contains sub-nodes for 'Boy', 'Girls', 'Men', and 'Women'. The 'Women' node has several child nodes, each representing a different item. One such node, '-LdG22J9fH96PeluMI2x', is expanded to show its properties:

```

trial-c2b17
  - Admin
  - Delivery_Man_Details
  - Feedback
  - dod
  - Item
    - Boy
    - Girls
    - Men
    - Women
      - -LdG22J9fH96PeluMI2x
        deals_of_the_day: false
        item_description: "hjgf"
        item_name: "Anarkali"
        item_price: "1500"
        item_quantity: "10"
        item_size: "XL"
        item_status: true
        item_subcategory: "Anarkali"
        optional_image: "https://firebasestorage.googleapis.com/v0/b/tri...
        recommended: true
      - -LdG2f6JWH-a04HXC22
      - -LdG2TLUIJxQ1Aox6Dzc
      - -LdG2f1jTh5GtJA5Dcy
      - -LdG2qlqHqqVfOUOFMp
      - -LdG35gyfzbv9PeeNqov
      - -LdG3Dw0ObwBv659-5LF
      - -LdG3fTRwTBbgsyids2B
      - -LdG3wOT4CQKt4-rnGBT
      - -LdG4BIMmczilmWfFys0
  - users

```

Figure 14: Women's_Item (Database)

The screenshot shows the Firebase Realtime Database interface. The 'users' node contains a single child node, 'NvQm3vk21YcYt8Susvbo4DXCCl2'. This node has a child node 'Order' which is further expanded to show a specific order. The order details are as follows:

```

trial-c2b17
  - Admin
  - Delivery_Man_Details
  - Feedback
  - dod
  - Item
  - users
    - NvQm3vk21YcYt8Susvbo4DXCCl2
      - Order
        - -LdUUKyeOvGfgwxVeKP9
          Placed_By: "NvQm3vk21YcYt8Susvbo4DXCCl1"
          customer_address: "E-23/ Ashish Park, Tarsali, Vadoda"
          customer_email: "bhargavborse4898@gmail.cc"
          customer_name: "Bhargav"
          customer_phone_no: "9974061115"
          customer_pincode: "390009"
          days: "1"
          delivery_date: "2019-04-27"
          item_image: "https://firebasestorage.googleapis.com/v0/b/tri...
          item_name: "Saree"
          item_price: "1100"
          item_quantity: "1"
          order_id: "2858NvQm3vk21YcYt8Susvbo4DXCCl1"
          order_status: "Order Sent"
          order_type: "Rent"
          purchase_date: "27/4/2019"
          return_date: "2019-04-28"
        - -LdUULBh7ChVcMEcu0
        - -LdZ9mA3z7jebpx_VkMz

```

Figure 15: User(1) (Database)



Figure 16: User(2)(Database)

3.8 Functional and Behavioral Modeling

3.8.1 Activity Diagram

User Login

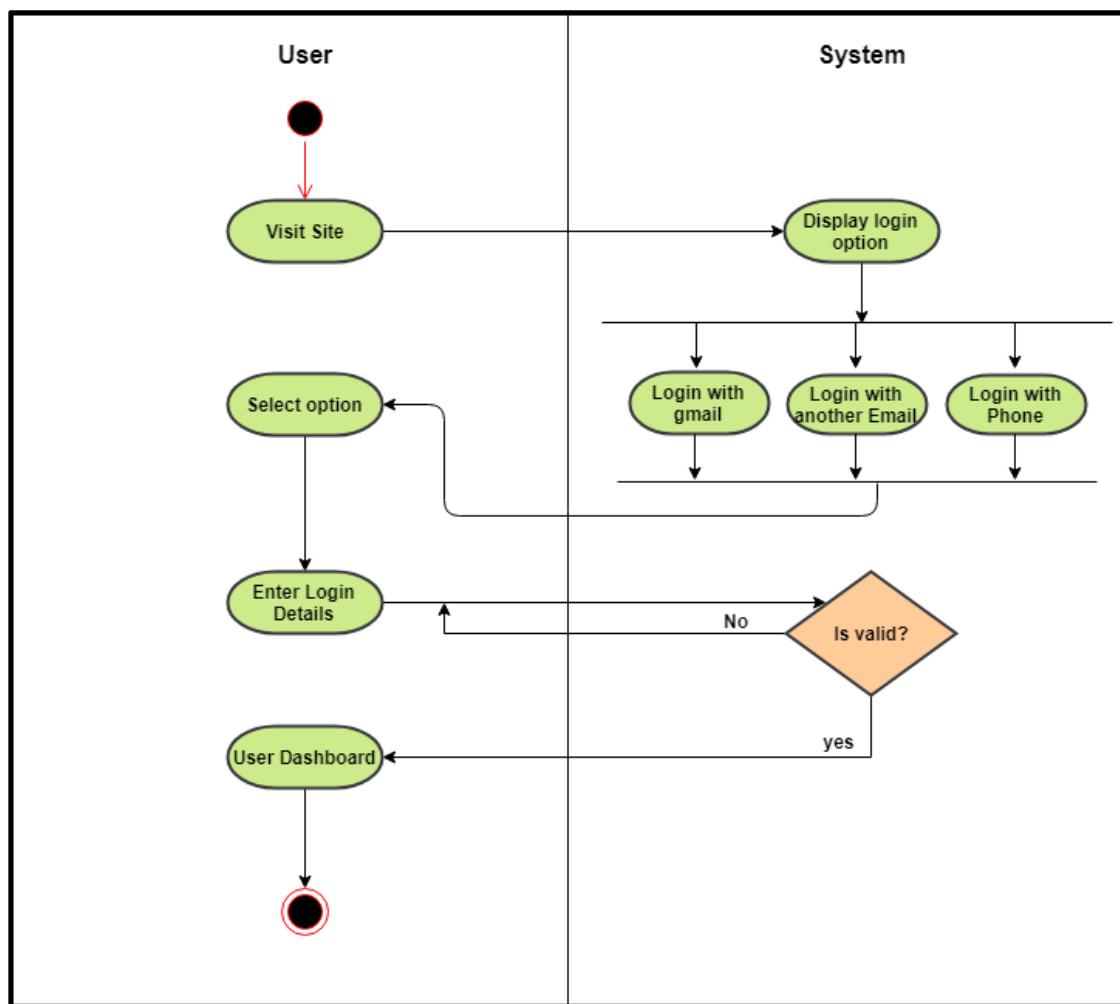


Figure 17: User Login Activity Diagram

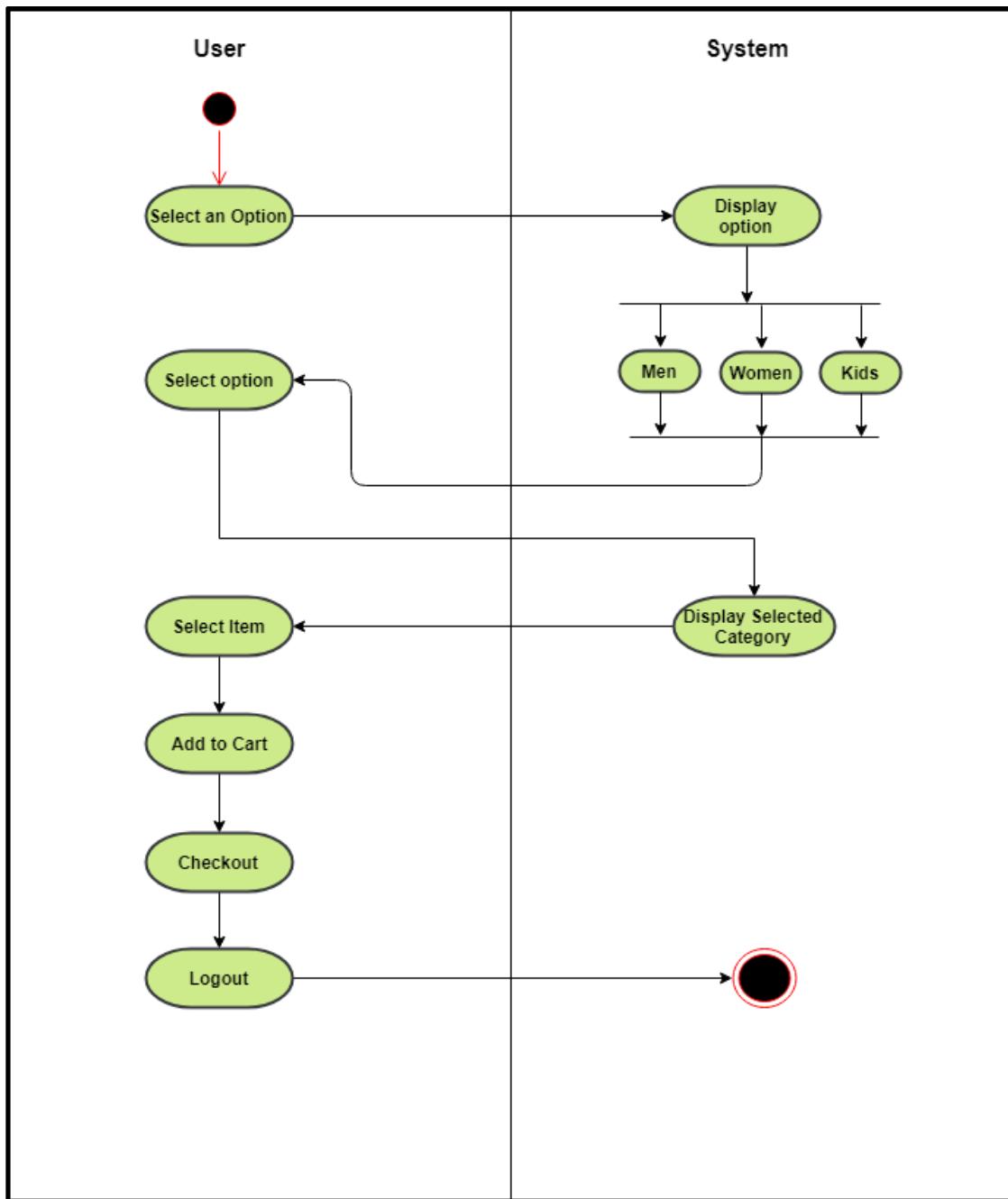
Dress Selection Option(User)

Figure 18: Dress Selection Option Activity Diagram

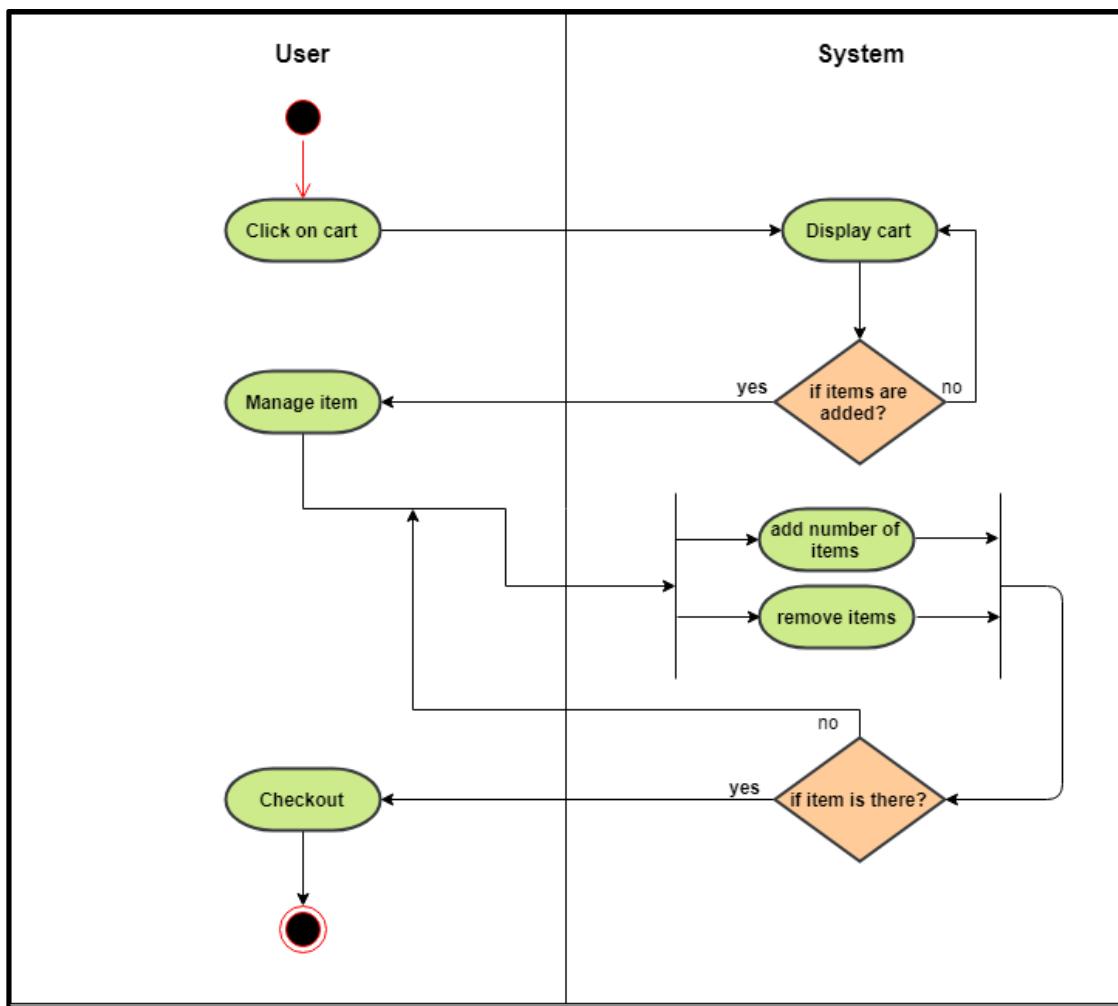
Add To Cart(User)

Figure 19: Add To Cart Activity diagram

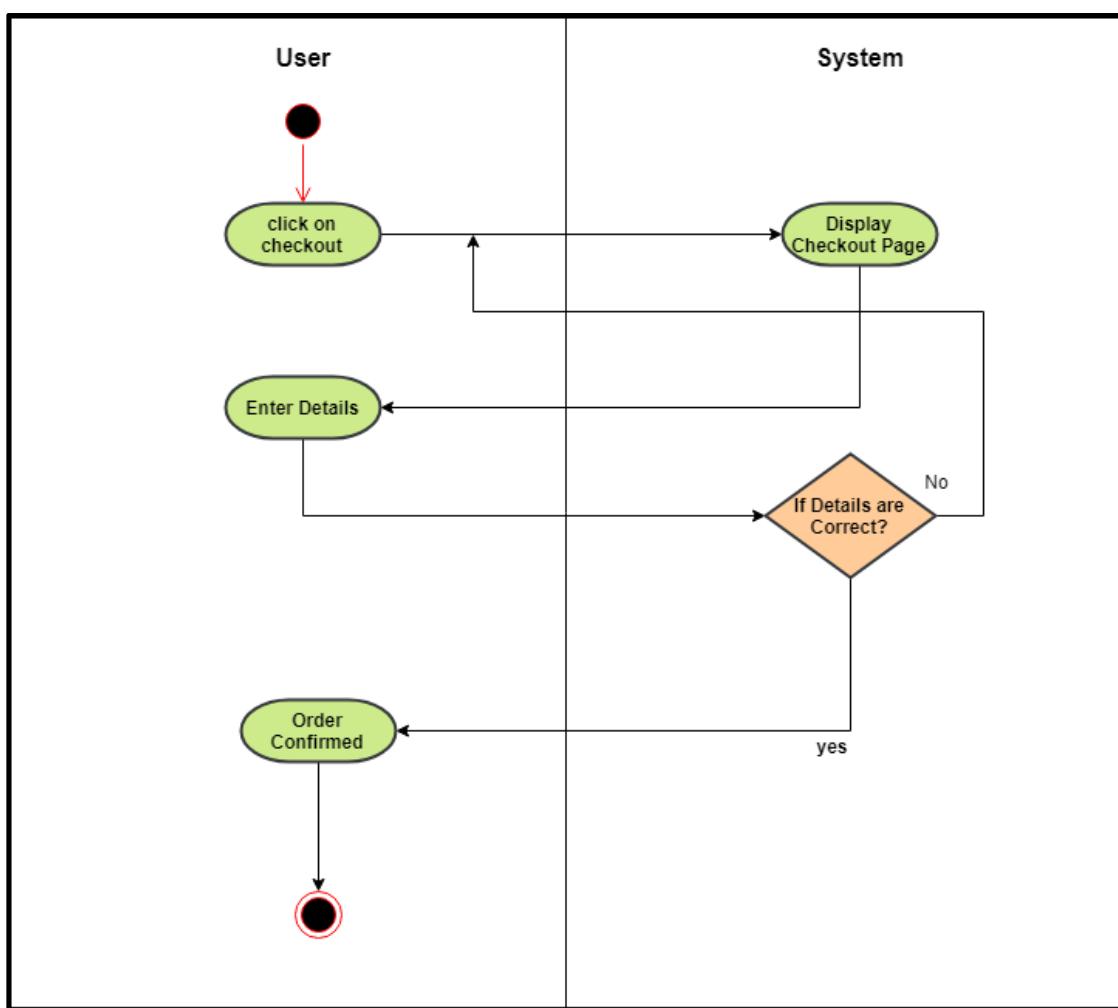
Checkout(User)

Figure 20: Checkout Activity Diagram

Admin Login

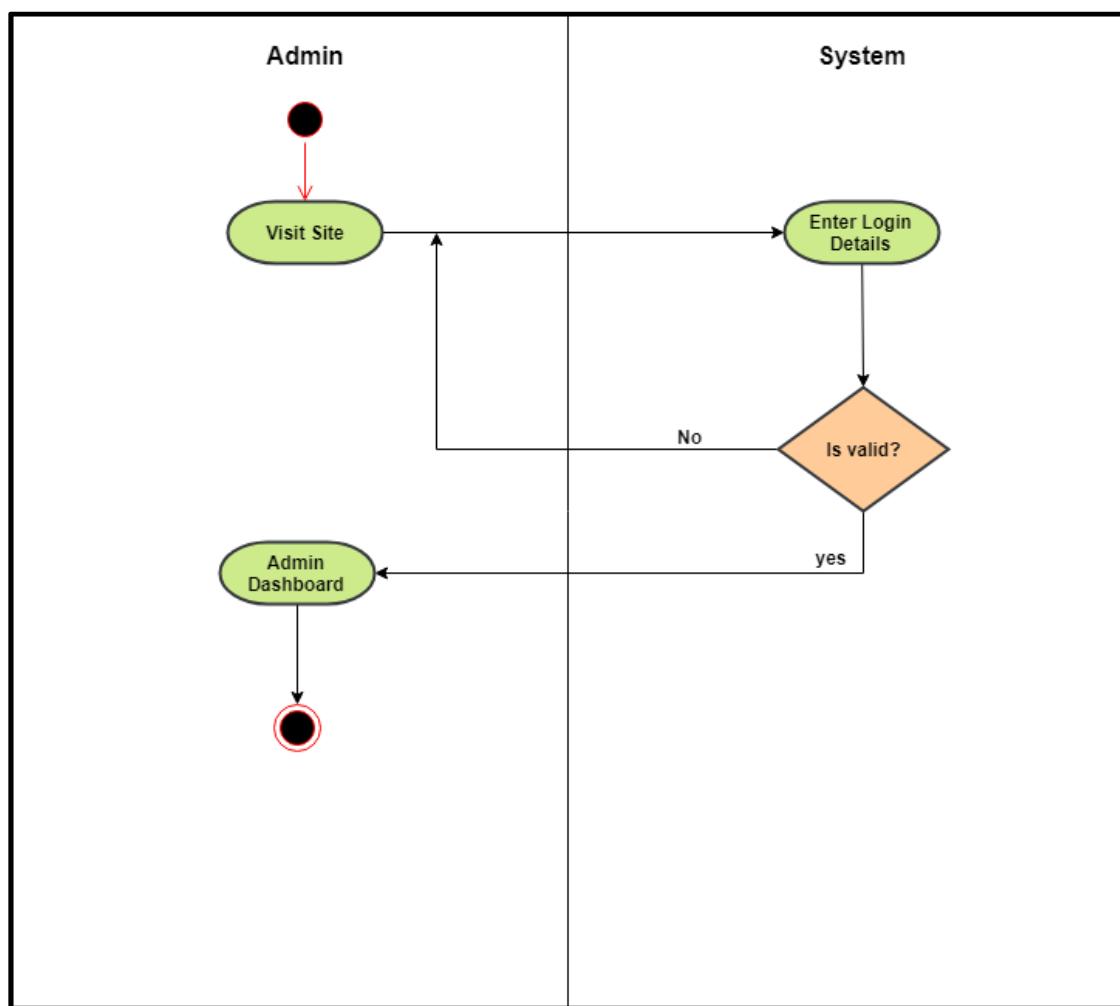


Figure 21: Admin Login Activity Diagram

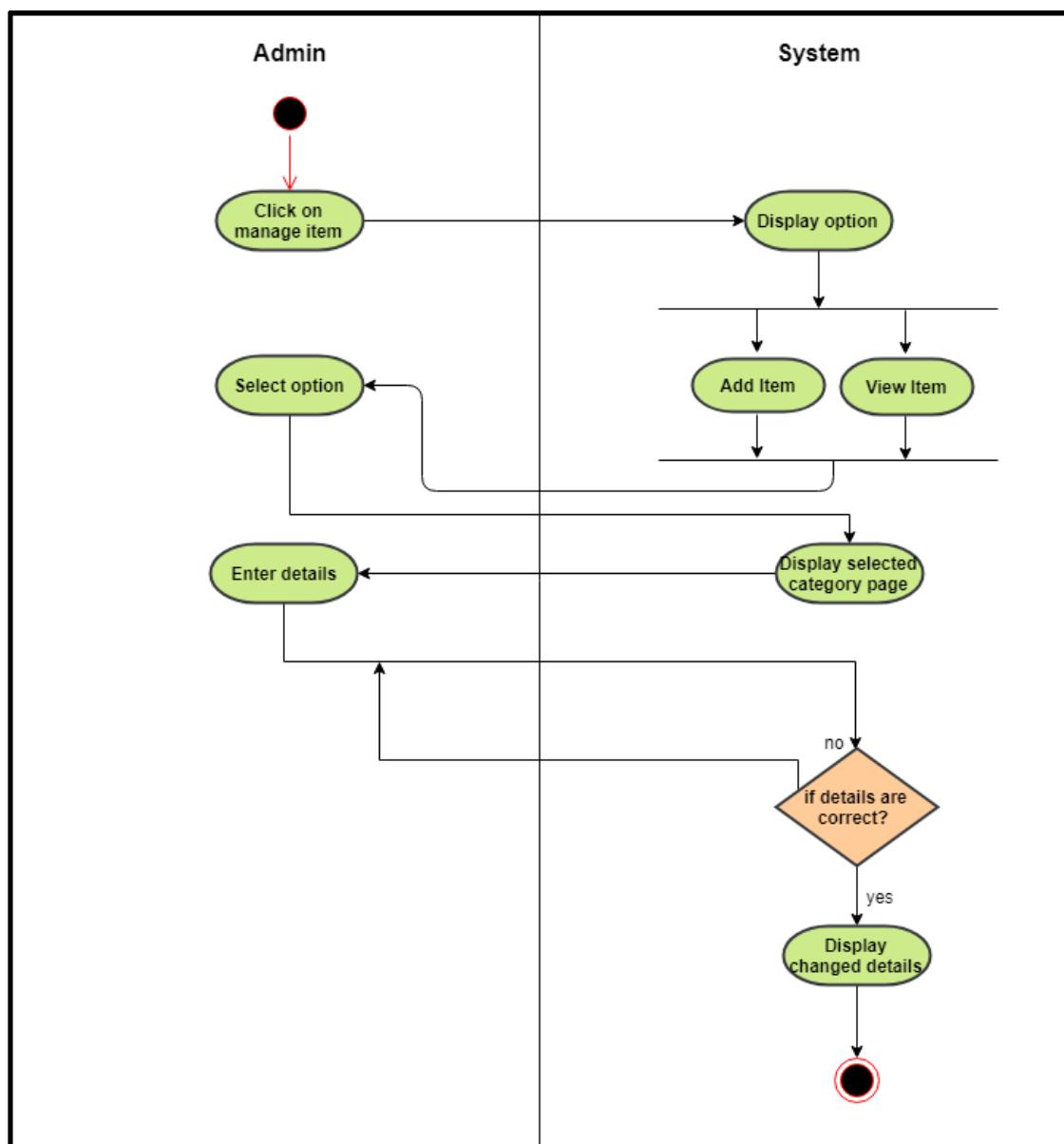
Manage Item(Admin)

Figure 22: Manage Item Activity Diagram

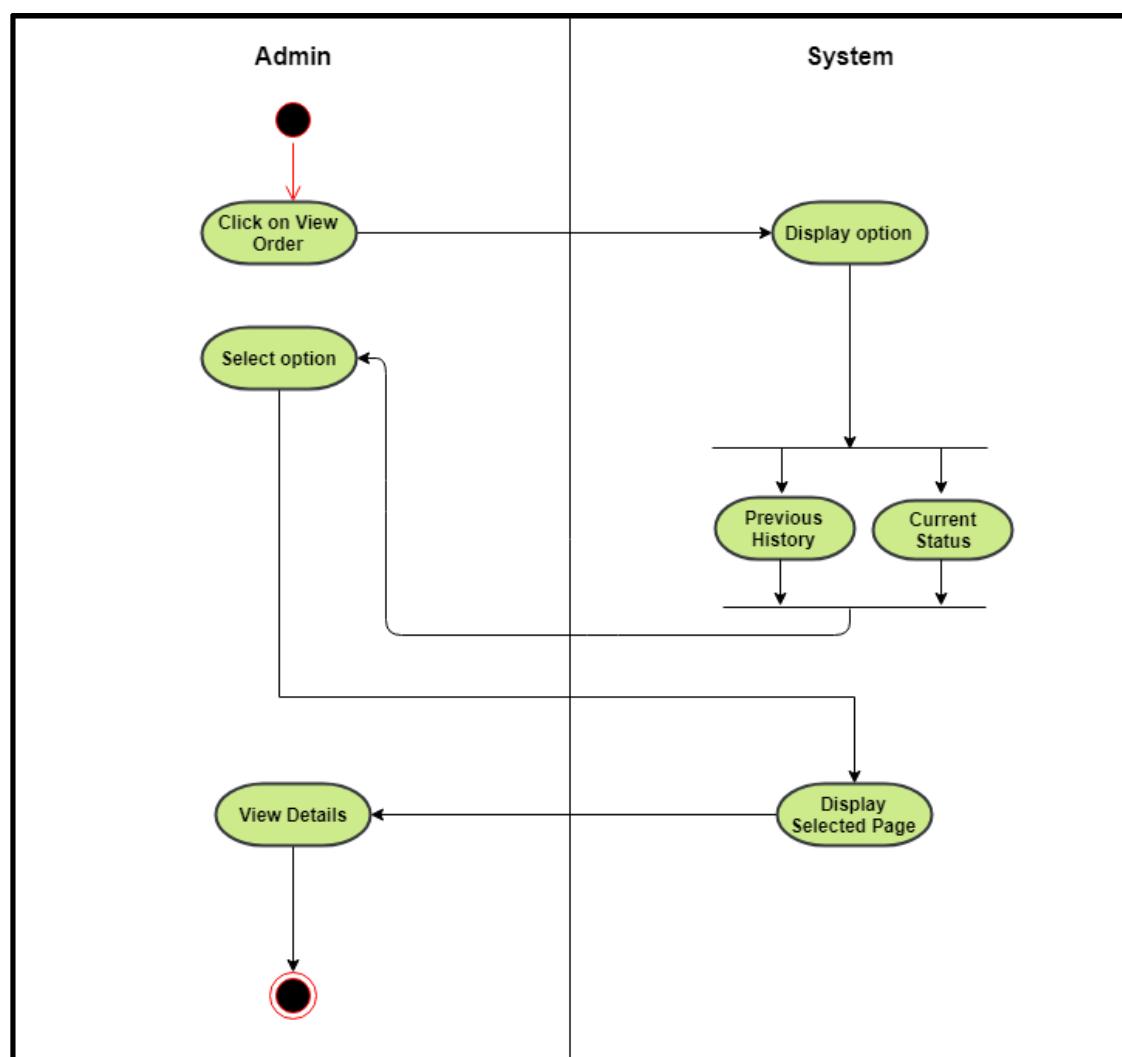
Rental Details(Admin)

Figure 23: Rental Details Activity Diagram

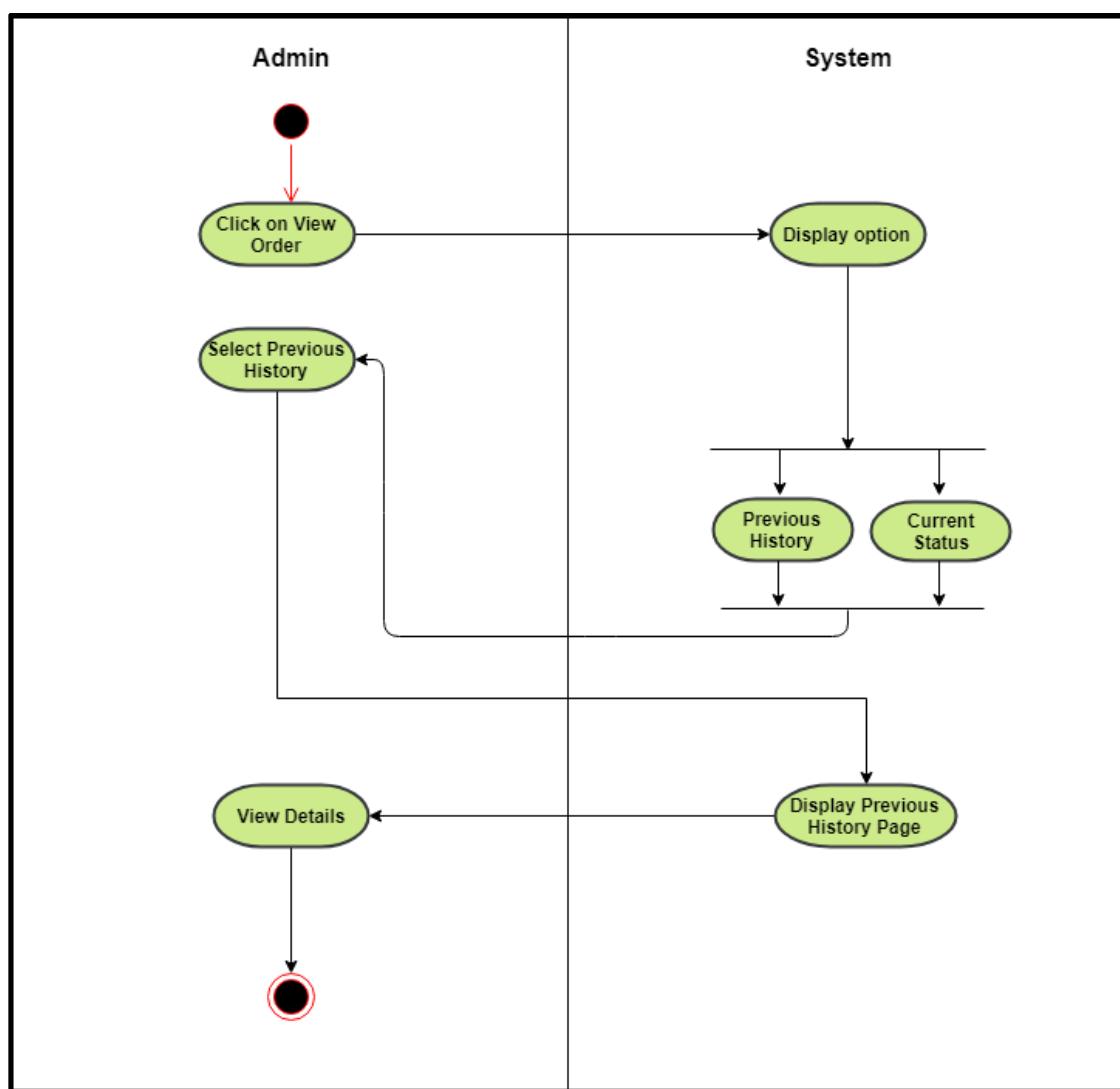
Penalty(Admin)

Figure 24: Penalty Details Activity Diagram

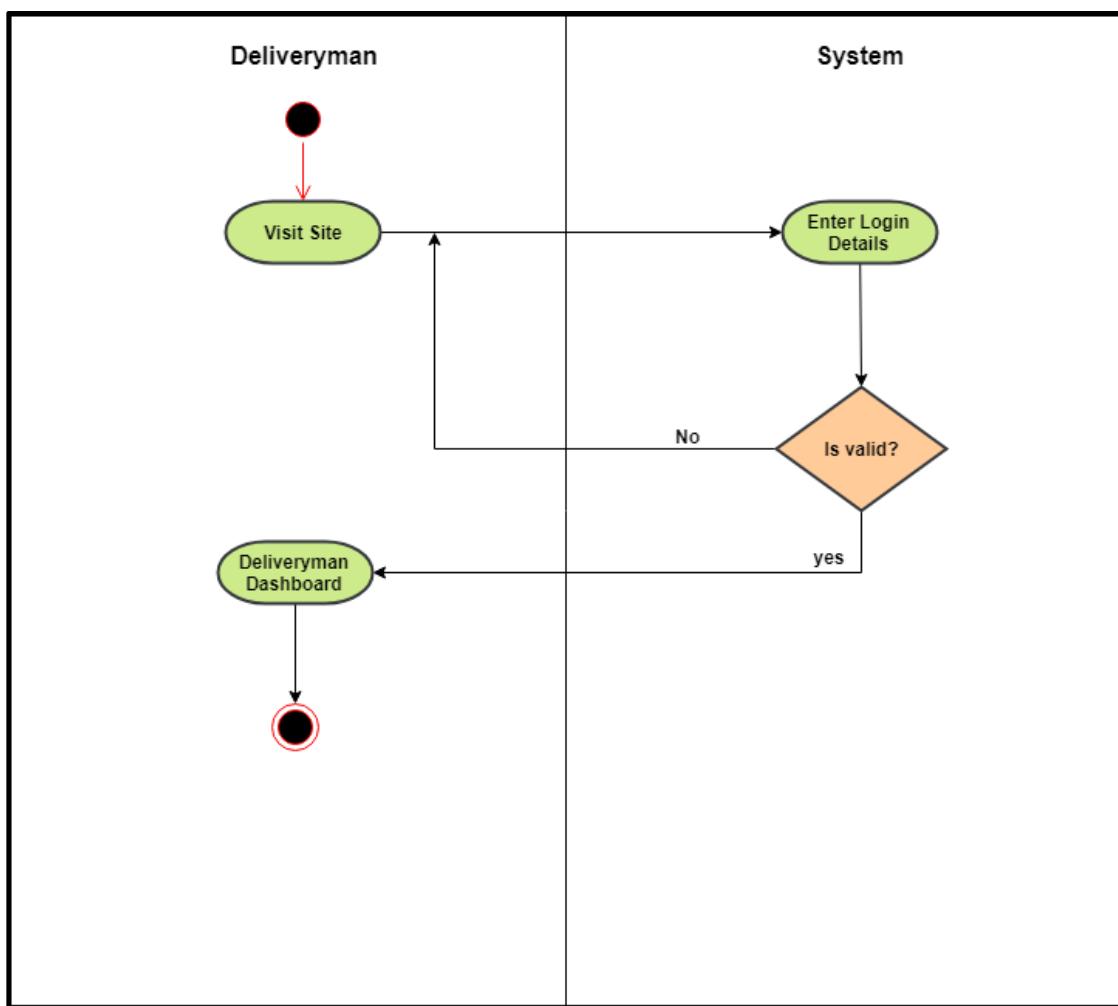
Deliveryman Login:-

Figure 25: Deliveryman Login Activity Diagram

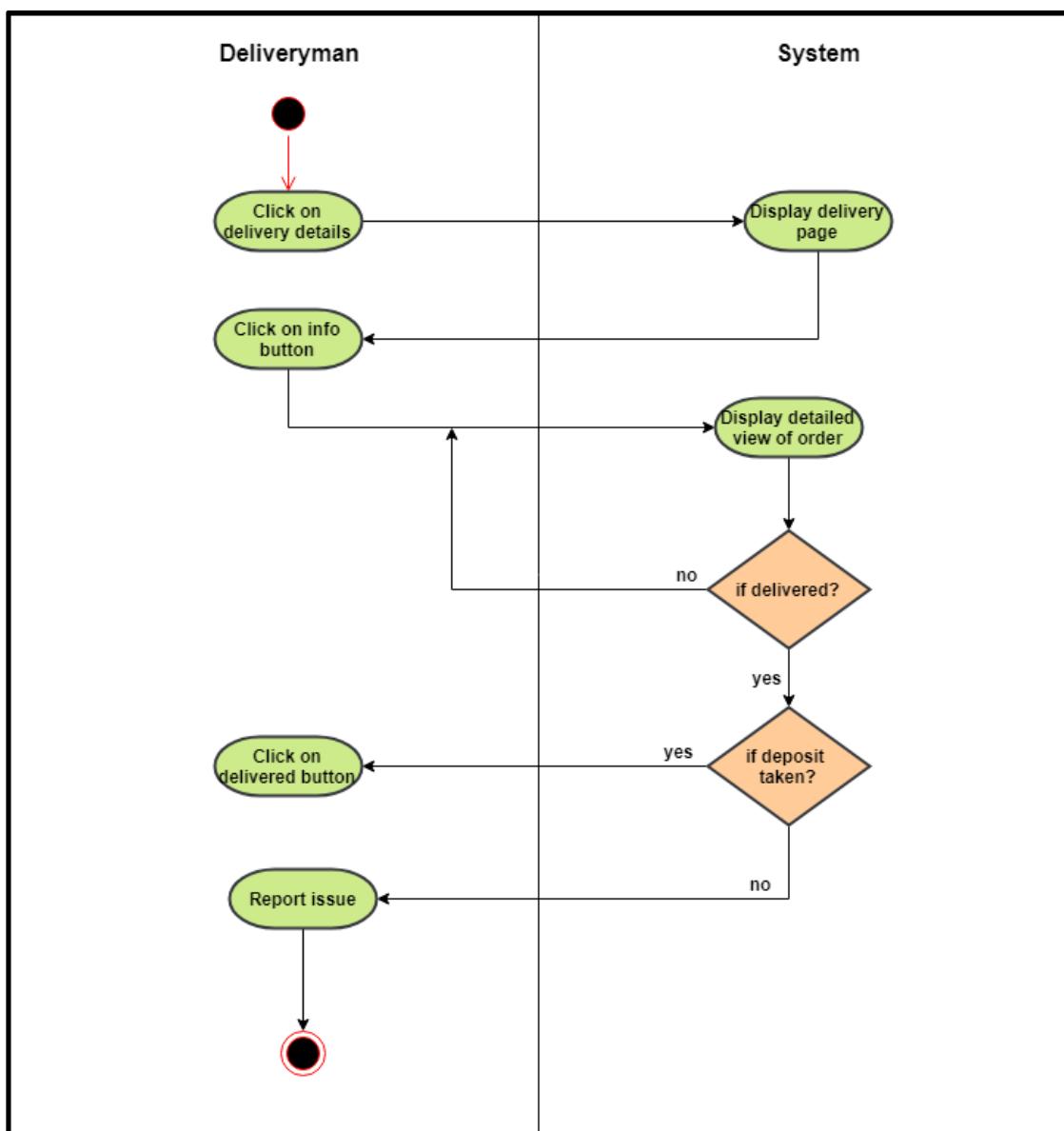
Delivery Details (Deliveryman) :-

Figure 26: Delivery Details Activity Diagram

Return Item (Deliveryman) :-

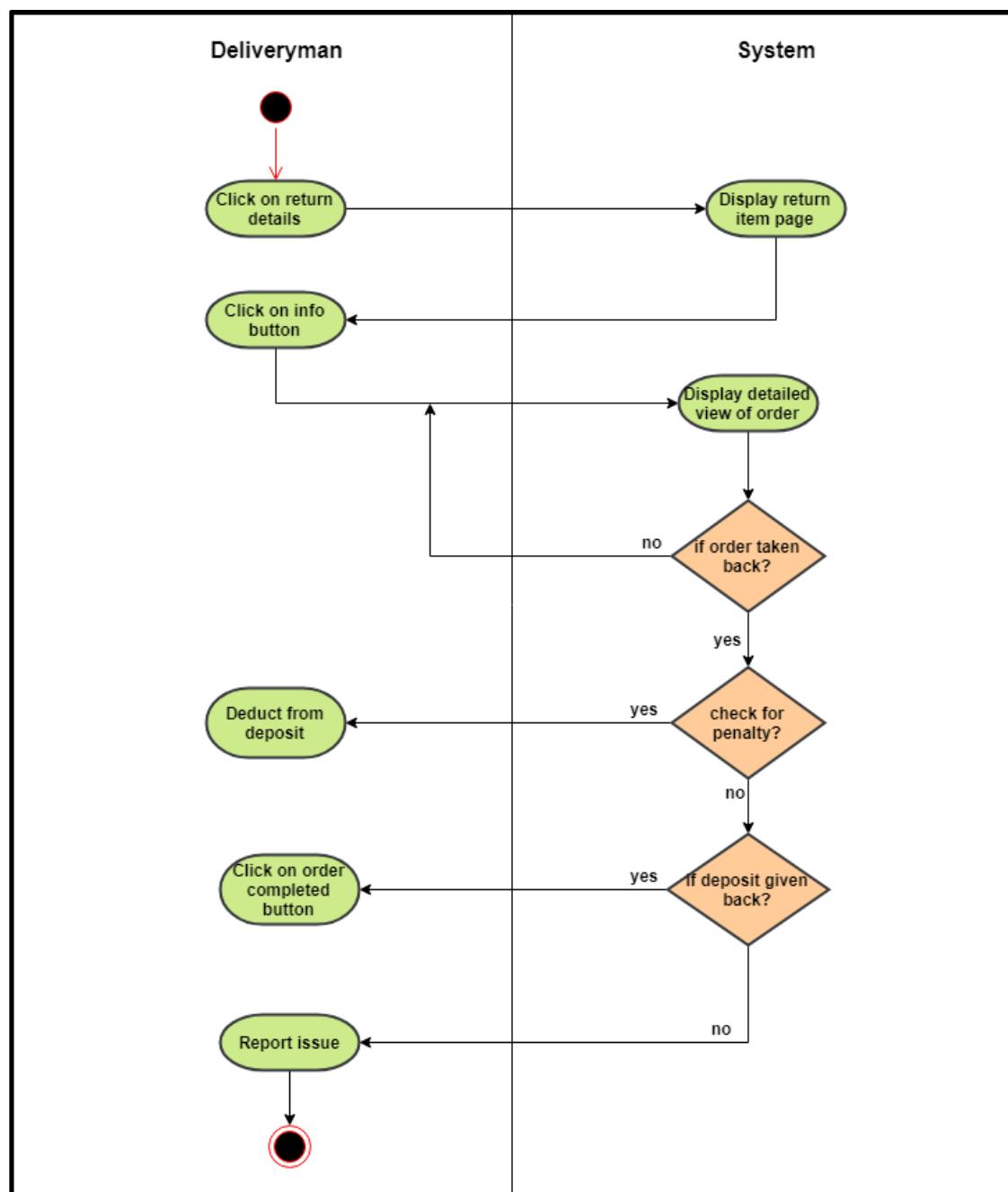


Figure 27: Return Item Activity Diagram

3.8.2 Sequence Diagram

Order Item:

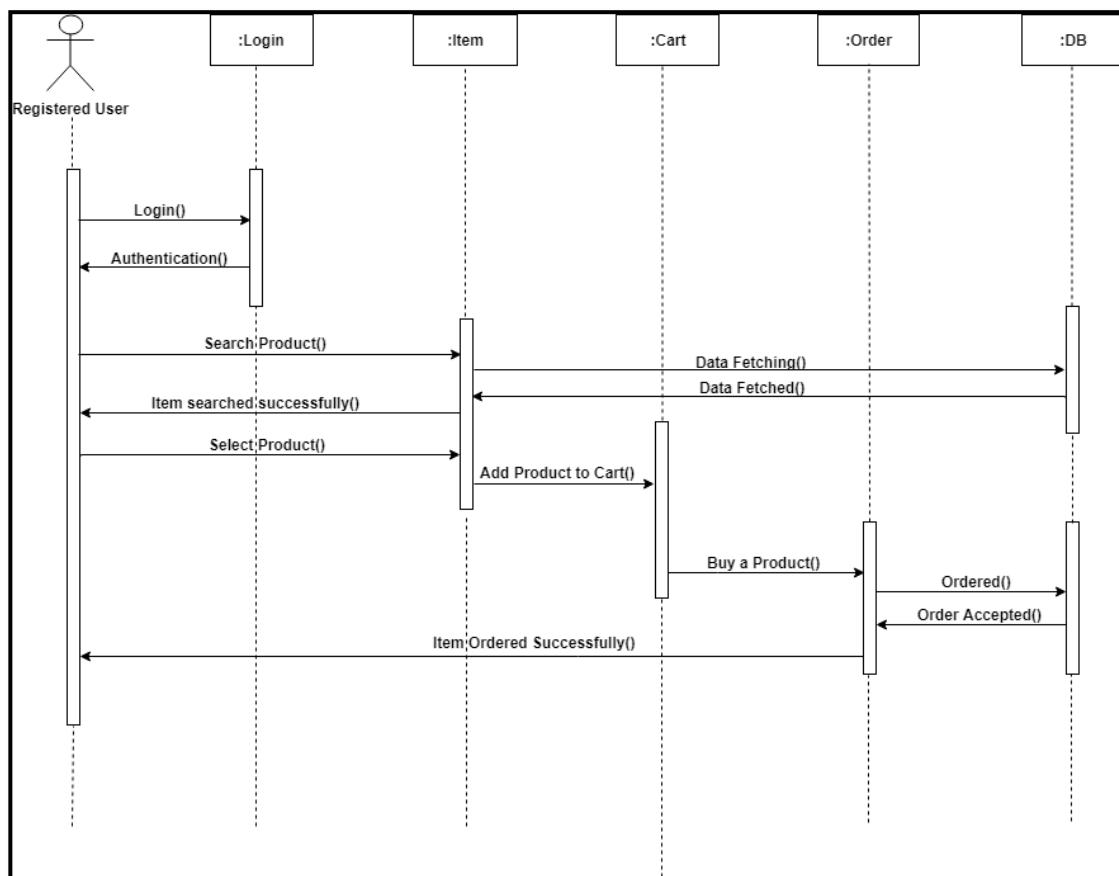


Figure 28: Order Item Sequence diagram

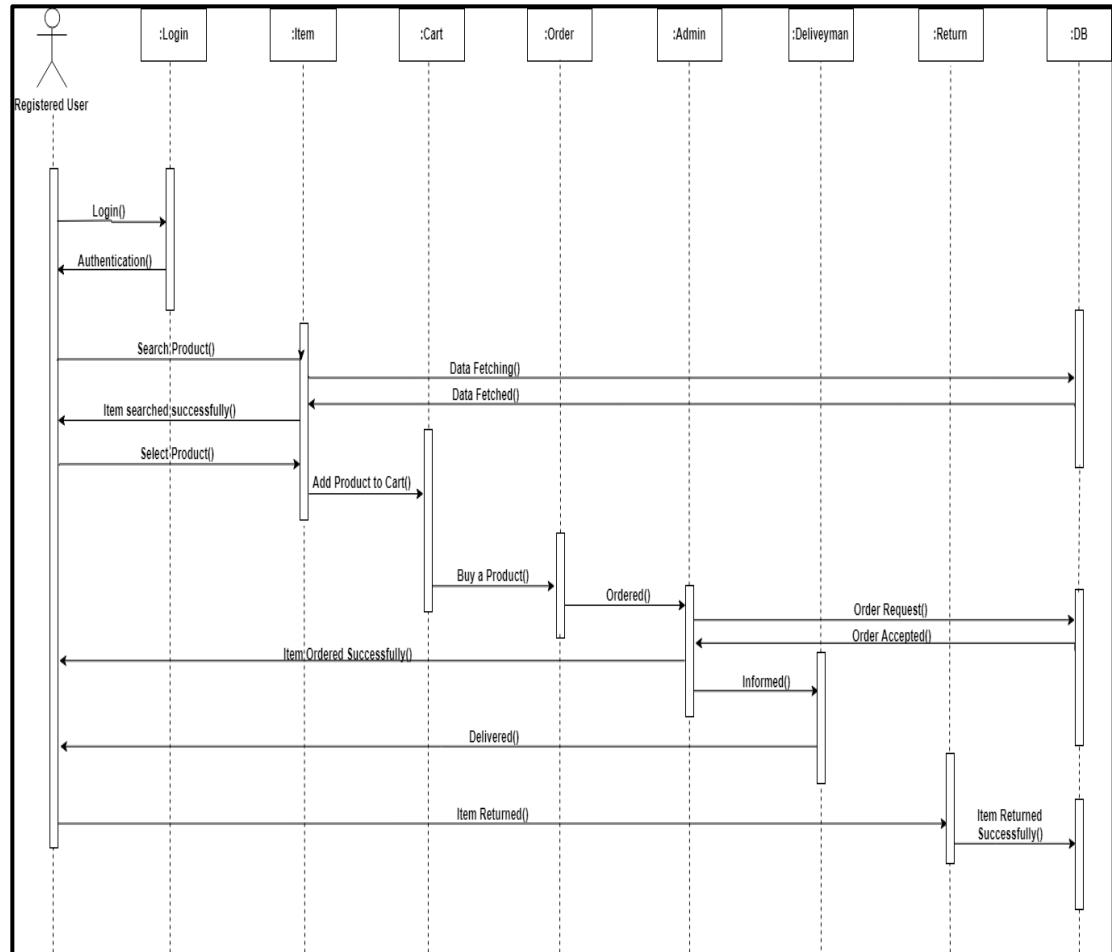
Rent Item:

Figure 29: Rent Item Sequence diagram

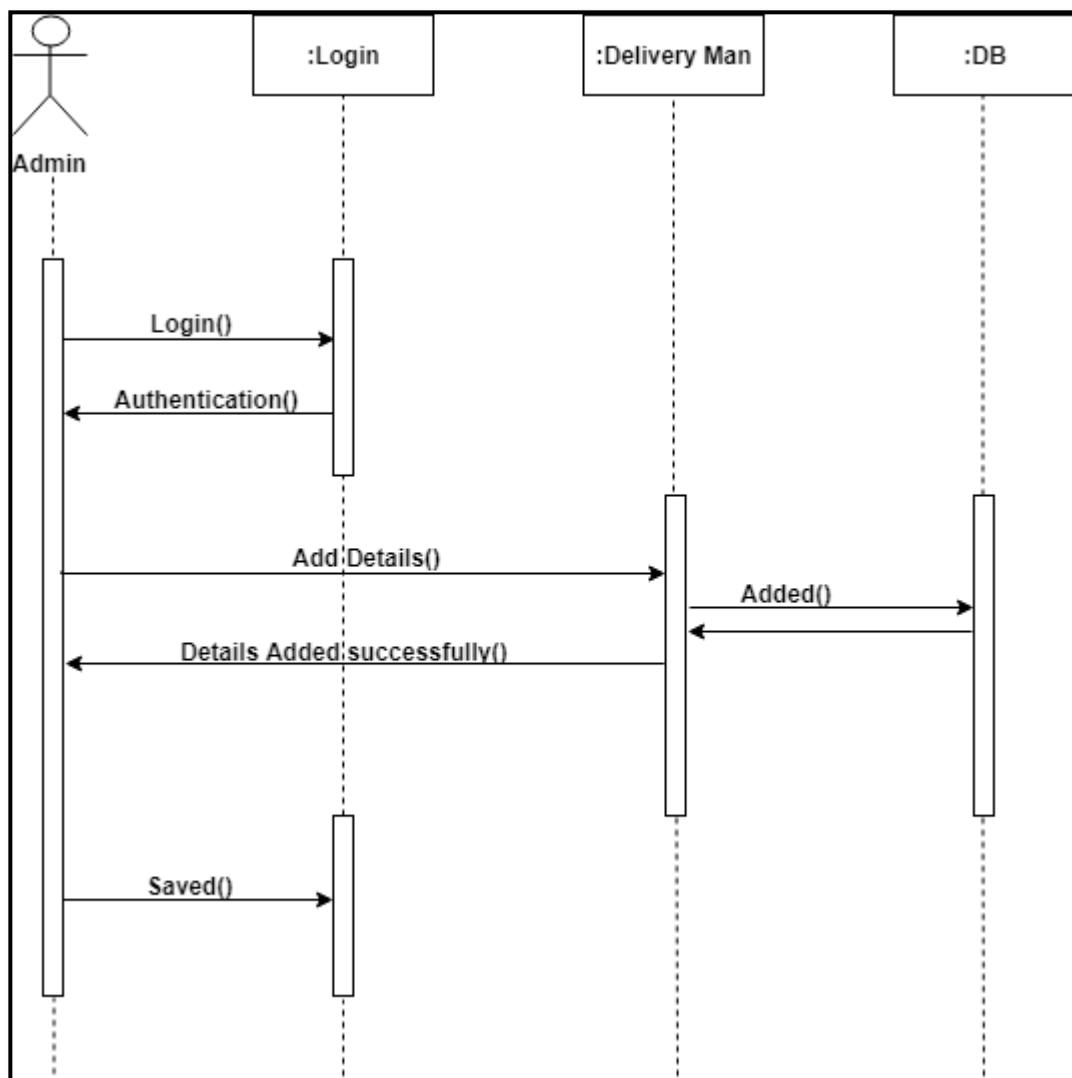
Adding Deliveryman(Admin)

Figure 30: Adding Deliveryman Sequence diagram

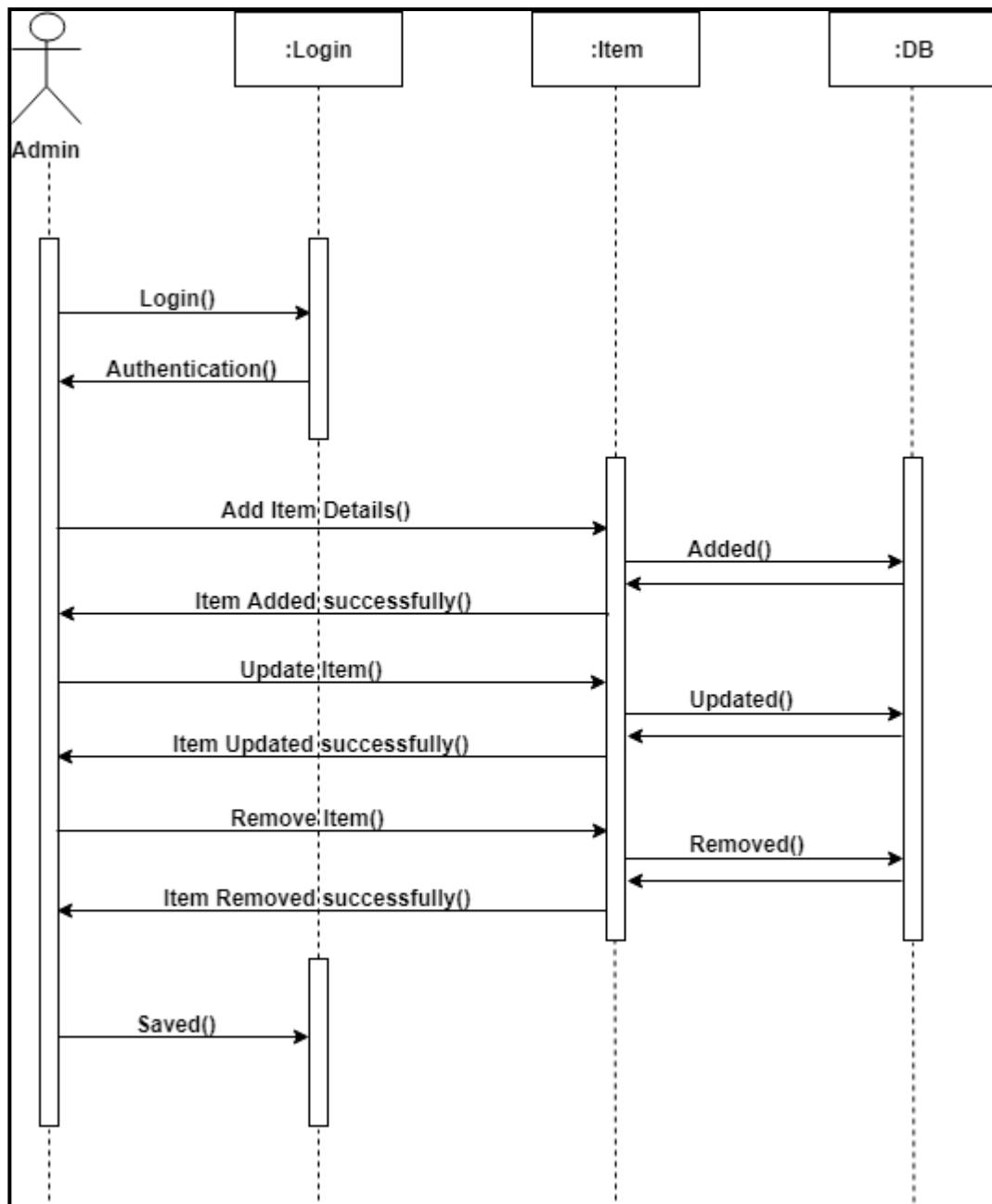
Manage Item(Admin)

Figure 31: Manage Item Sequence diagram

3.8.3 Control flow diagram

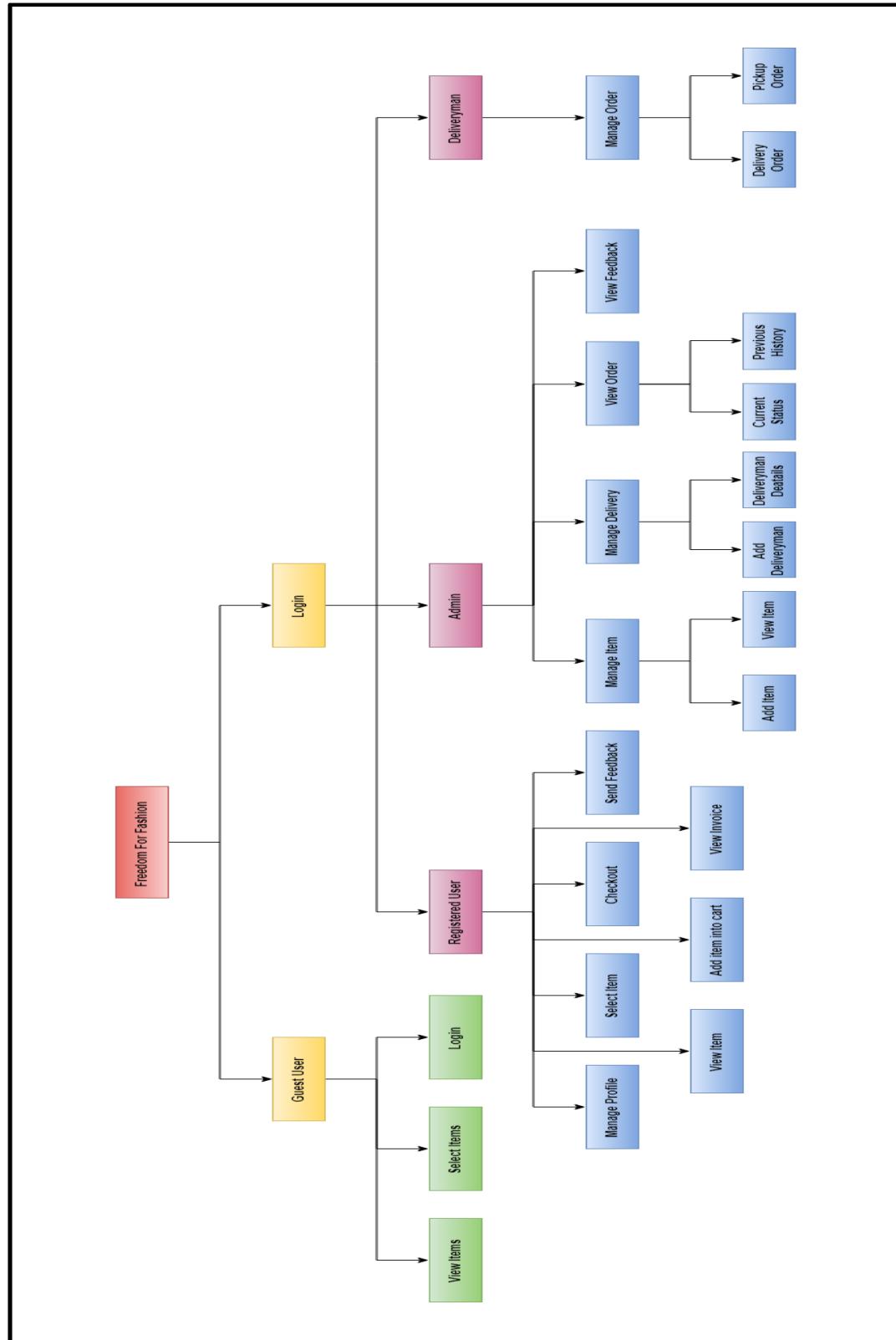


Figure 32 : Control Flow Diagram

4. SYSTEM DESIGN

4.1 Tables and Relationship

Admin				
Field Name	Data Type	Size	Constraint	Description
Admin_Email	Varchar	50	Not Null	Admin_Email is used for Login into admin dashboard.
Password	Varchar	20	Not Null	Password is necessary for login via email.

Table 2: Admin Table

Customer				
Field Name	Data Type	Size	Constraint	Description
Customer_Id	Int	5	Primary Key (Auto Increment)	Customer_Id is unique key for every Customer.
Customer_Name	Varchar	30	Not Null	Customer_Name is required for Customer identification.

Email_Id	Varchar	50	Not Null	Email_id is required for buy or rent a item.
Password	Varchar	20	Not Null	Password is necessary for login via email.
Address	Varchar	100	Not Null	Address is required for delivery purpose
Pincode	Int	6	Not Null	Pincode is used for find a delivery location area
Mobile number	Big Int	10	Not Null	Mobile number is used for contact.

Table 3: Customer table

Item				
Field Name	Data Type	Size	Constraint	Description
Item_Id	Int	5	Primary Key (Auto Increment)	Item_Id is unique key for every item.
Item_Name	Varchar	30	Not Null	Item_Name is a name of particular Item.
Item_Description	Varchar	200	Not Null	Item_Description is Description of any item.

Item_Price	Int	6	Not Null	Item_Price is the Price of particular price of item.
Item_Image	Varchar	200	Not Null	Item Image is image of any particular item.
Item_Quantity	Int	5	Not Null	Item_Quantity is Quantity of any particular item.
Category	Varchar	10	Not Null	There will be four category like Men/Women/Boy/Girls
Subcategory	Varchar	10	Not Null	Subcategory is category of selected Category.
Deals Of The Day	Varchar	10	Not Null	Deals Of the Day is a function that shows some deals of that particular day.
Recommended	Varchar	10	Not Null	Recommended is a function that shows some products that is recommended by admin.

Table 4: Item table

Cart				
Field Name	Data Type	Size	Constraint	Description
Cart_Id	Int	5	Primary Key(Auto Increment)	Cart_Id is a unique key for every new cart.
Item_Id	Int	5	Foreign Key	Item_Id is Key of item added into thr cart.
Amount	Int	6	Not Null	Amount is particular amount of Cart.

Table 5: Cart table

Payment				
Field Name	Data Type	Size	Constraint	Description
Payment_Id	Int	5	Primary Key(Auto Increment)	Payment_Id is a unique key for every new Payment.
Amount	Int	6	Foreign Key	Amount is a payable amount of particular Order.
Penalty	Int	6	Not Null	Penalty is a amount that are extra charge of

				customer if he make any fault.
--	--	--	--	--------------------------------

Table 6: Payment table

Order				
Field Name	Data Type	Size	Constraint	Description
Order_Id	Int	5	Primary Key (Auto Increment)	Order_Id is unique key for every new Order.
Item_Id	Int	5	Foreign Key	Item_Id is unique key for every item.
Order_Amount	Int	6	Not Null	Order_Amount is a payable amount of particular Order.
Order_Date	Date	-	Not Null	Order_Date is Date of item Customer buy or rent.
Return_Date	Date	-	Not Null	Return_Date is date when Customer Return Item.
Deposit	Int	6	Not Null	Deposit is a amount that customer has to pay when Item

				delivered at their place.
Order_Status	Varchar	12	Not Null	Order_status Shows that order is in which phase.

Table 7: Order table

Delivery(rent)				
Field Name	Data Type	Size	Constraint	Description
Email	Varchar	50	Primary Key(Auto Increment)	Email is a unique key For every delivery man.
Password	Varchar	10	Not Null	Password is necessary for login .
Order_Id	Int	5	Foreign key	Order_Id is unique key for every new Order.
DM_Id	Int	5	Foreign Key	DM_Id is a Foreign key for delivery man.
Drop	Date	10	Not Null	Drop is a date when delivery

				man has to deliver Item.
--	--	--	--	--------------------------

Table 8: Delivery(Rent)Acceptancy

Delivery(Buy)				
Field Name	Data Type	Size	Constraint	Description
Email	Varchar	50	Primary Key(Auto Increment)	Email is a unique key For every delivery man.
Password	Varchar	10	Not Null	Password is necessary for login .
Order_Id	Int	5	Foreign key	Order_Id is unique key for every new Order.
DM_Id	Int	5	Foreign Key	DM_Id is a Foreign key for delivery man.
Pickup	Date	10	Not Null	Pickup is a date when delivery man has to pick Item.

Table 9: Delivery(Buy) acceptancy

Feedback				
Field Name	Data Type	Size	Constraint	Description
Feedback_Id	Int	5	Primary Key(Auto Increment)	Feedback_Id is a unique key for every new Feedback.
Customer_Name	Varchar	30	Not Null	Customer_Name is required for Customer identification.
Email_id	Varchar	50	Not Null	Email_id is required for Customer identification.
Feedback_Message	Varchar	200	Not Null	Feedback_Messegae is a feedback message Given by Customer.

Table 10: Feedback table

Delivery Man				
Field Name	Data Type	Size	Constraint	Description
DM_Id	Int	5	Primary Key(Auto Increment)	DM_Id is a unique id for every new delivery man.
DM_Name	Varchar	30	Not Null	DM_Name is name of particular name of delivery man.
Address	Varchar	100	Not Null	Address is required for some id proof.
Mobile Number	Big Int	10	Not Null	Mobile number is used for contact.
Deliveryman Image	Varchar	200	Not Null	Deliveryman image for identification of deliveryman.
Id_Proof	Varchar	200	Not Null	Id_proof is for security purpose.

Table 11: Deliveryman table

4.2 System Procedural Design

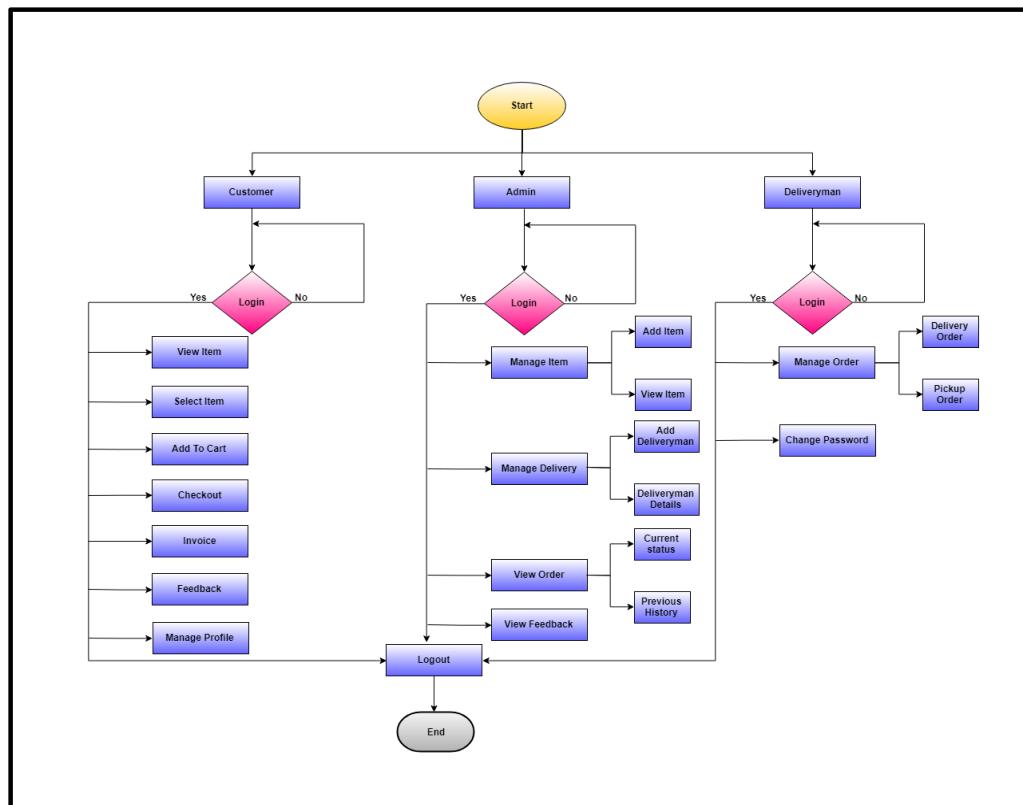


Figure 33: Flowchart

4.3 Design Pseudo Code

1. Select user
 - 1.1 Customer
 - 1.2 Admin
 - 1.3 Deliveryman
2. If user customer
 - 2.1 Login into the system
 - 2.2 If email id and password wrong, display error
 - 2.3 else email and password right
 - 2.4 If forgot password
 - 2.4.1 Display forgot password popup
 - 2.4.2 Enter email id
 - 2.4.3 Link will be sent to that entered email
 - 2.5 Wrong email
 - 2.6 If else welcome details page(if user is already login once)
 - 2.7 Else Display welcome page(First time user)
- 2.8 Select category
 - 2.8.1 If select Men's/Women/kids Clothing(it shows respective pages)
 - a. Display men/women/kids cloths(it shows respective pages)
 - b. View Items
 - c. Add to cart
 - d. Checkout
 - e. Generate invoice
 - 2.8.2 If else select Deals of the day
 - a. Display list of deals
 - b. View item
 - c. Add to cart
 - d. Checkout
 - e. Generate invoice

2.8.3 If else select Recommended

- a. Display list of items
- b. View Item
- c. Add to cart
- d. Checkout
- e. Generate Invoice

2.8.4 If else select send feedback

- a. Display feedback form
- b. Feedback send, display successfully send feedback

2.8.5 if else select profile

- a. update profile details
- b. Display successfully updated profile

2.9 else logout

3. if else user admin

3.1 login into the system

3.2 If email id and password wrong, display error

3.3 else email and password right

3.4 Display welcome page

3.5 Select Module

3.5.1 if select mange item

- a. if select add item
- b. display add item form
- c. fill details and add item
- d. else select view item(men/women/boy/girl)
- e. display items(categorized)
- f. can update or inactive item

3.5.2 if else select manage delivery

- a. if select add deliveryman
- b. display add deliveryman form
- c. fill details and add deliveryman
- d. else select view deliveryman details
- e. can view details of deliveryman

3.5.3 if else select view order

- a. if select current status
 - b. display current orders
 - c. select particular order
 - d. display order details form
 - e. can select deliveryman
 - f. else select previous history
 - g. display previous orders
 - h. select particular order
 - i. can view order details
- 3.5.4 else select view feedback
- a. Display feedback list
- 3.6 else logout
4. else deliveryman
- 4.1 login into the system
 - 4.2 If email id and password wrong, display error
 - 4.3 else email and password right
 - 4.4 Display welcome page
 - 4.5 Select Order Details Module
 - 4.5.1 if select delivery order
 - a. display orders
 - b. select particular order
 - c. display order details form
 - d. can update delivery status
 - 4.5.2 if select pickup order
 - a. display orders
 - b. select particular order
 - c. display order details form
 - d. can update delivery status
 - 4.6 If Select Change Password
 - 4.6.1 Enter registered email
 - 4.6.2 Can change password from link which is sent to their email.
 - 4.7 else logout

4.4 Input/ Output and Interface Design (wireframes)

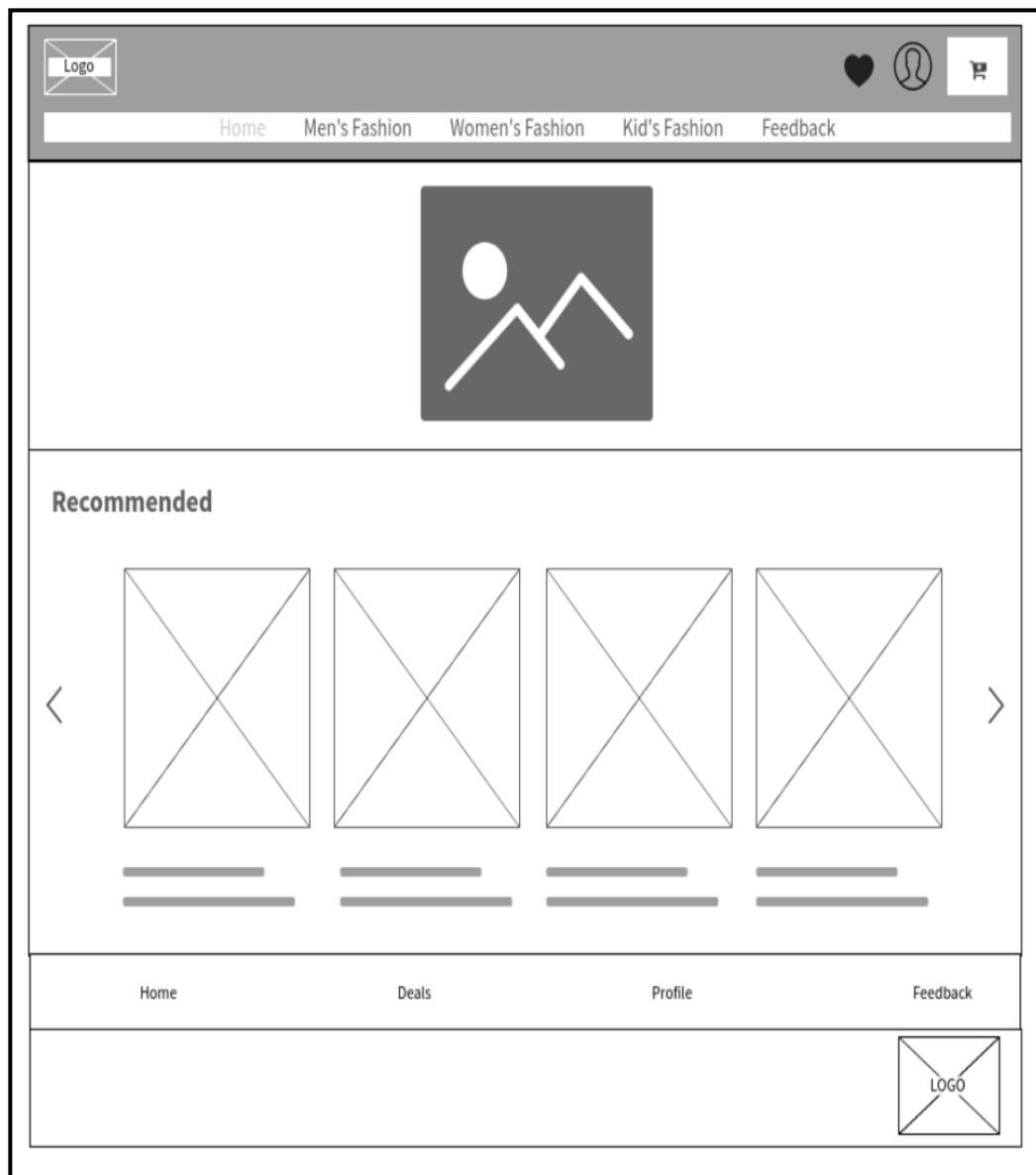


Figure 34: Main Page(User)

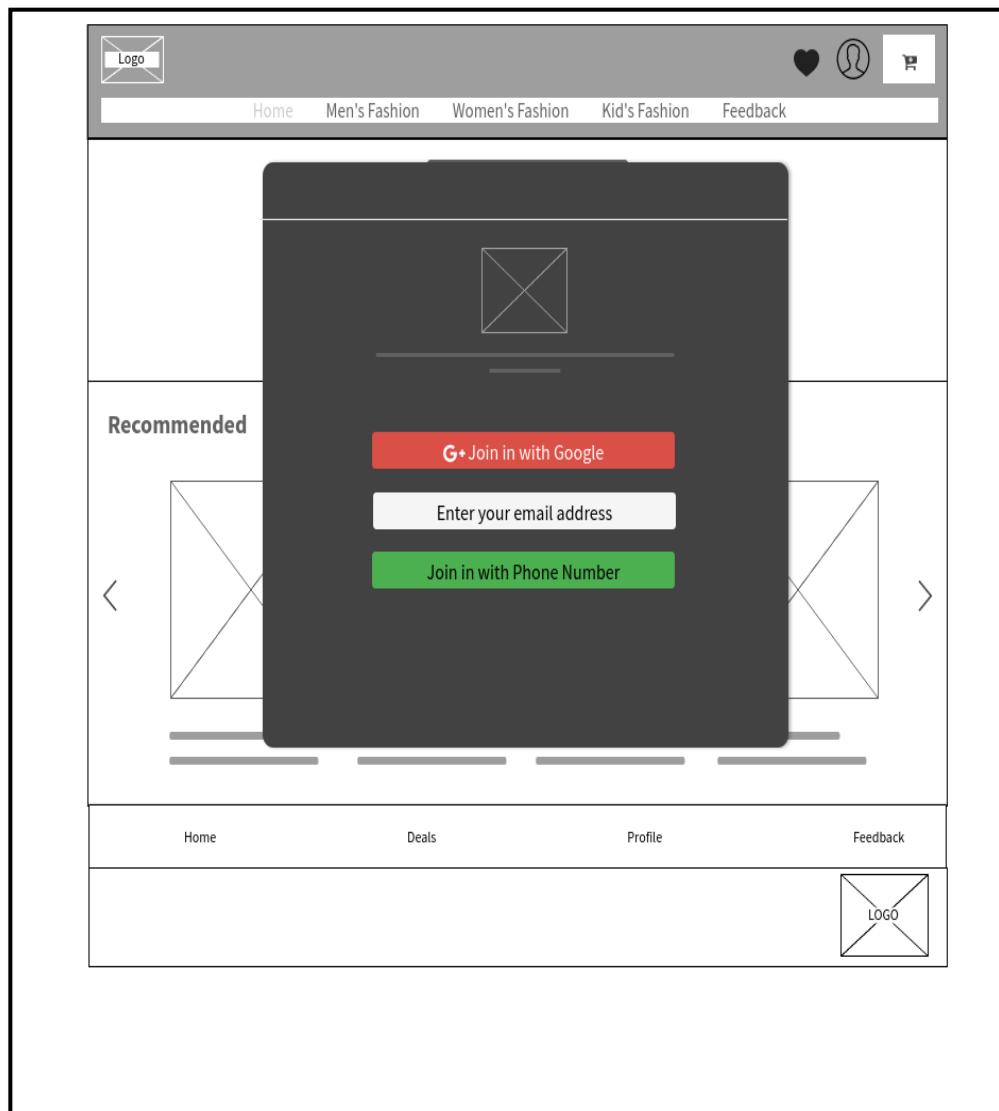


Figure 35: User Login

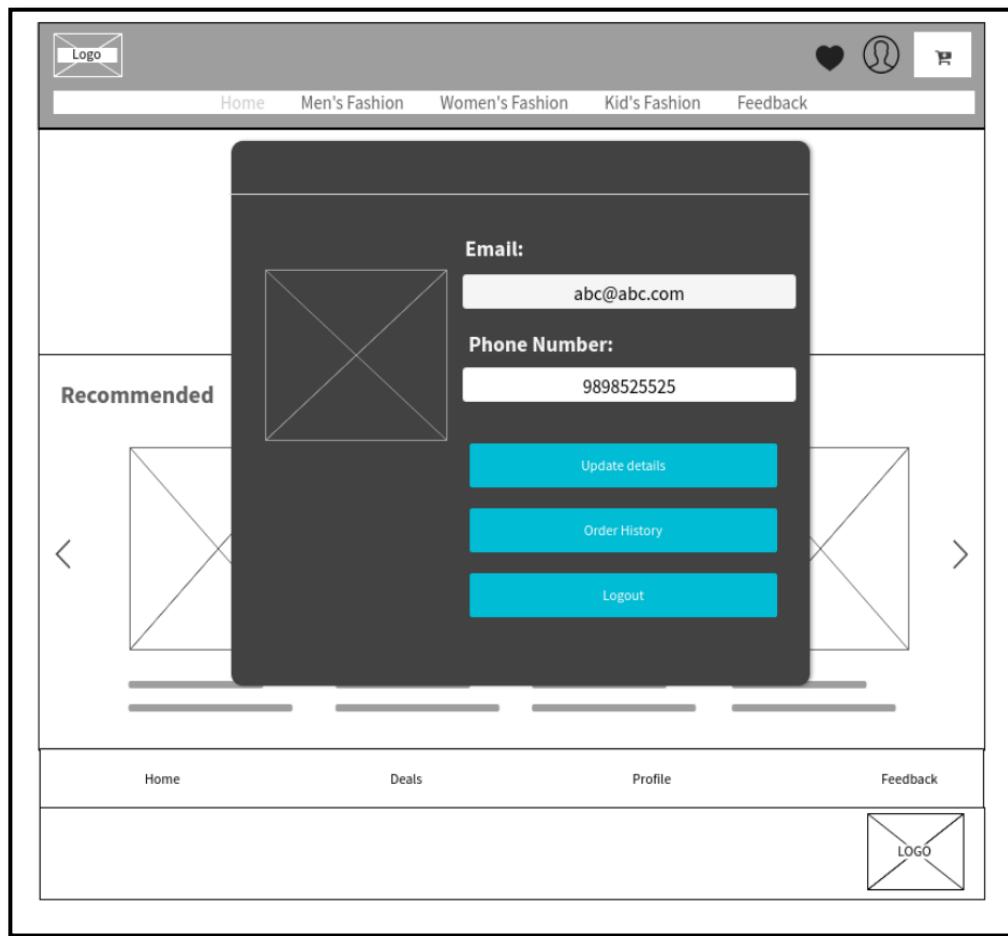


Figure 36 : Profile page(User)

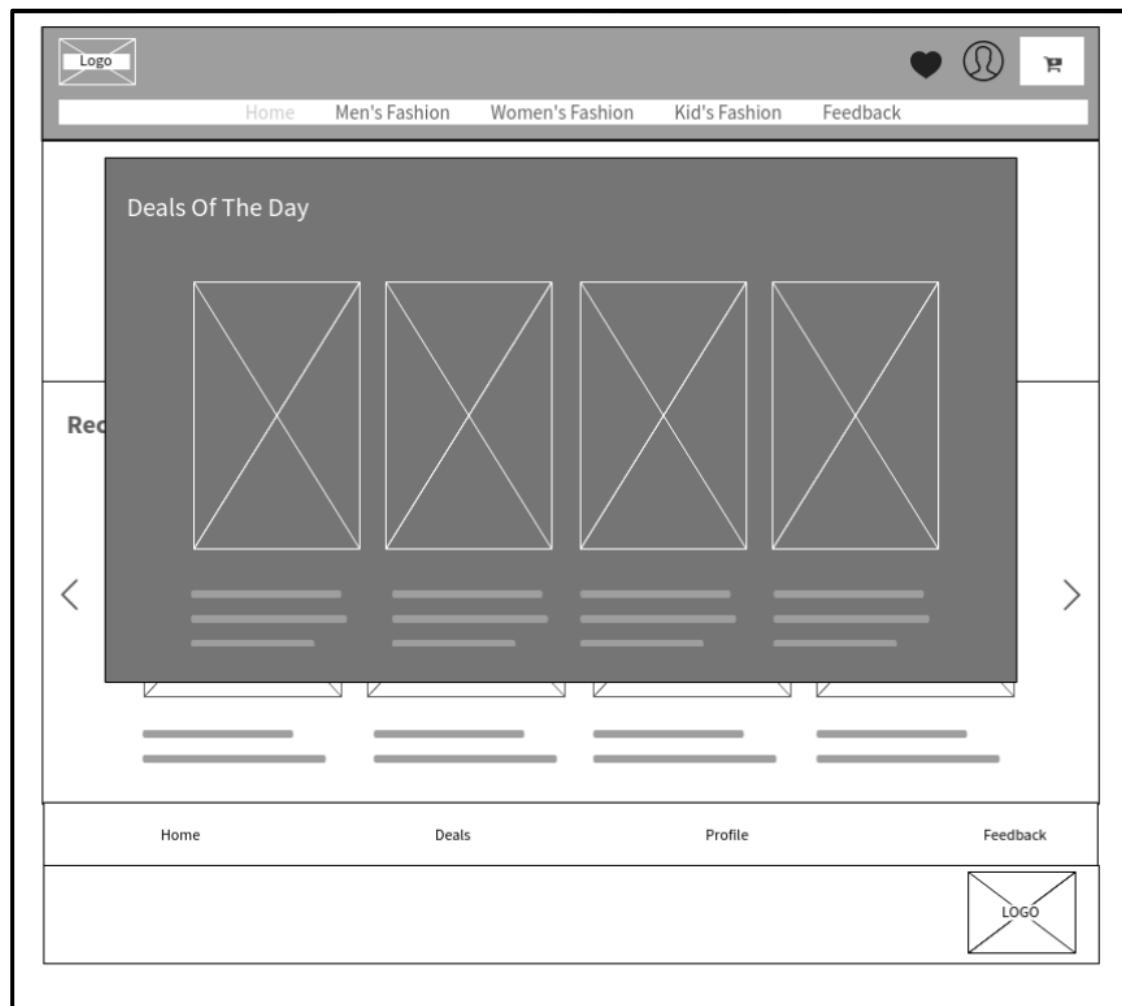


Figure 37: Deals Of The Day(User)

The wireframe shows a mobile application interface for updating personal details. At the top, there is a header bar with a logo icon, navigation links for Home, Men's Fashion, Women's Fashion, Kid's Fashion, and Feedback, and icons for heart, profile, and cart. Below the header, the main content area has a title "Update Details:". It contains fields for First Name, Last Name, Mobile Number, Address, Pincode, and Email, each with an associated input field. To the right of these fields is a placeholder image of a sun and mountains. A "Update Details" button is located at the bottom left of the content area. At the very bottom, there is a footer bar with links for Home, Deals, Profile, and Feedback, and a logo icon labeled "LOGO".

Update Details:

First Name:

Last Name:

Mobile Number:

Address:

Pincode:

Email:

Update Details

Home Deals Profile Feedback

LOGO

Figure 38: Update Details Page(User)

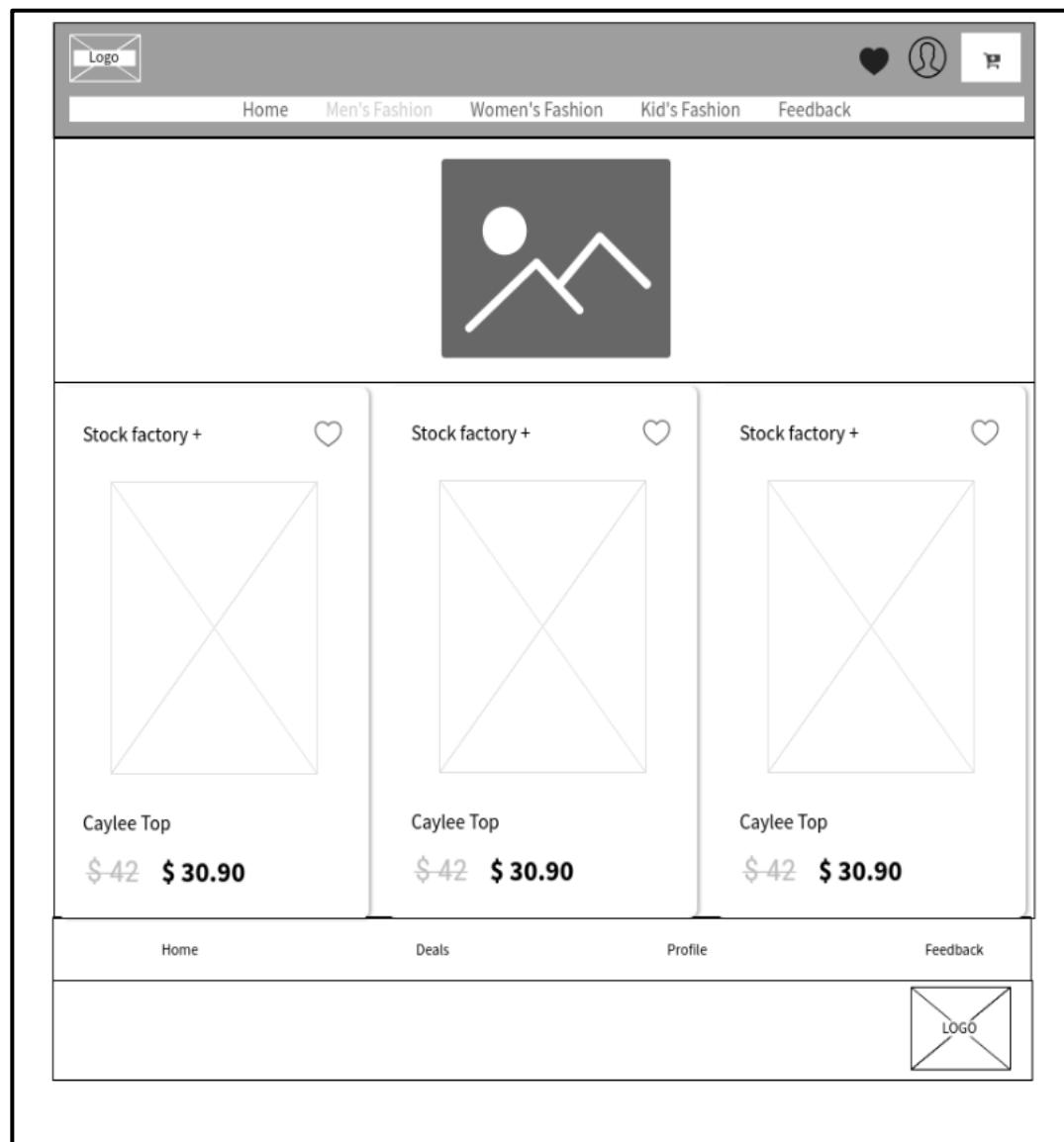


Figure 39: Men/Women/Kids Category Page(User)

The screenshot shows a mobile application interface. At the top, there is a navigation bar with a logo icon, a heart icon, a user profile icon, and a shopping cart icon. Below the navigation bar, a horizontal menu bar contains links for "Home", "Men's Fashion", "Women's Fashion", "Kid's Fashion", and "Feedback". The main content area is titled "Feedback" in large, bold, black font. Below the title, there is a section labeled "Send Us A Note" with three input fields: "Name:", "Email:", and "Message:". To the right of these fields is a dark rectangular button containing a white line-art icon of a person sitting on a mountain. Below the input fields is a dark rectangular "Submit" button. At the bottom of the screen, there is a footer navigation bar with links for "Home", "Deals", "Profile", and "Feedback". On the far right of the footer is a square logo icon with the word "LOGO" inside.

Figure 40: Feedback(User)

The wireframe shows a mobile application interface for a fashion e-commerce platform. At the top, there is a header bar with a logo icon, navigation links for Home, Men's Fashion, Women's Fashion, Kid's Fashion, and Feedback, and icons for a heart, profile, and cart. Below the header is a large title 'Checkout'. A table below the title displays a single item in the cart: SLNo. 1, Product Sherwani, Quantity 5, Price 10000, with a 'Remove' button next to it. There are input fields for 'Check Total:' and 'Billing Address:' followed by five empty lines for address details. At the bottom, there is a footer bar with links for Home, Deals, Profile, and Feedback, and a placeholder area for a logo.

SLNo.	Product	Quantity	Product Name	Price	Remove
1		5	Sherwani	10000	

Check Total:

Billing Address:

Home	Deals	Profile	Feedback

Figure 41: Checkout

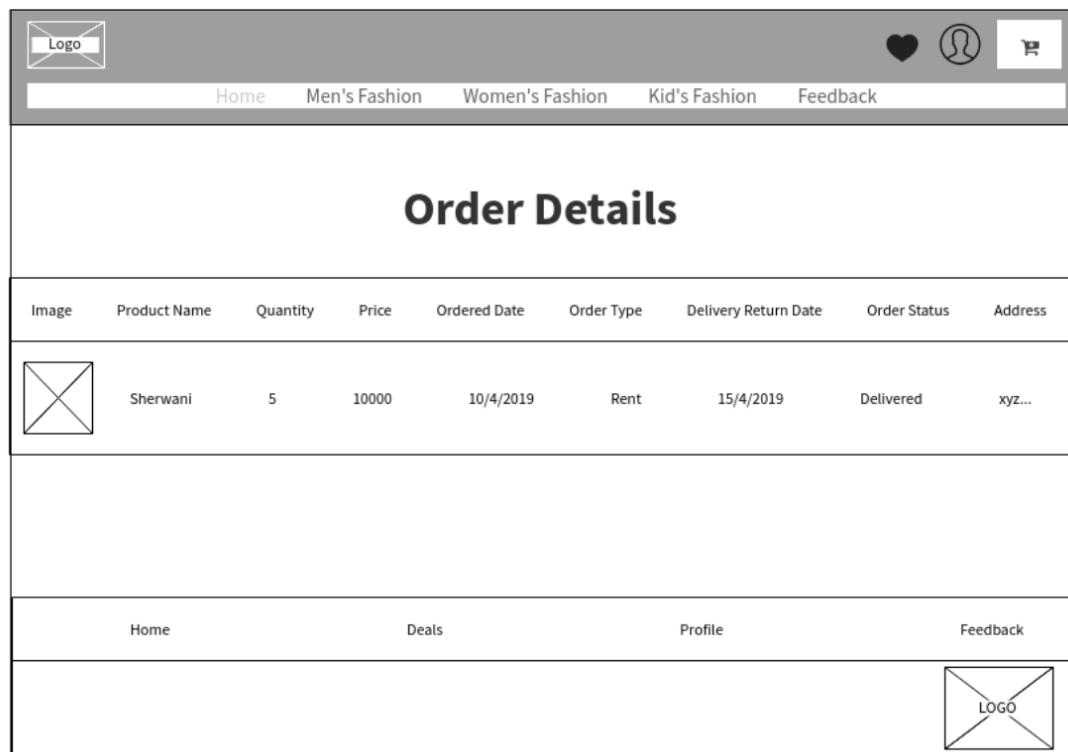


Figure 42: Order History(User)

The screenshot shows the admin login page for the "ADMIN PANEL". The page has a large gray background area. In the center, there are two input fields: one for "Email_Id:" and another for "Password:". Below these fields is a "LOGIN" button. Underneath the "LOGIN" button is a link labeled "Forgot password??".

Figure 43: Admin Login

The screenshot shows a web-based application for adding items. At the top right is a user profile icon and a 'Profile' button. Below the header is a title 'Add Item'. The form contains several input fields and dropdown menus:

- Item Name:** Text input field.
- Quantity:** Text input field.
- Description:** Text input field.
- Price:** Text input field.
- Category:** A dropdown menu with 'Select' as the default option.
- Size:** A dropdown menu with 'Select' as the default option.
- Sub Category:** A dropdown menu with 'Select' as the default option.
- Item Image:** A section with a 'Choose File' button and a message 'No File Chosen'.
- Optional Image:** A section with a 'Choose File' button and a message 'No File Chosen'.
- Categorization Options:** Two checkboxes labeled 'Deals of the day' and 'Recommended'.
- Action Buttons:** A large dark grey 'Add Item' button at the bottom.

Figure 44: Add Item

The screenshot shows a list of items under the heading 'Men's Fashion'. The table has columns for Item_Name, Description, Quantity, Price, Size, and Manage_Item (with an 'Update' button). There are three identical entries for 'Sherwani'.

Item_Name	Description	Quantity	Price	Size	Manage_Item
Sherwani	xyz.....	10	1000	M	<button>Update</button>
Sherwani	xyz.....	10	1000	M	<button>Update</button>
Sherwani	xyz.....	10	1000	M	<button>Update</button>

Figure 45: View Item Men(Admin)

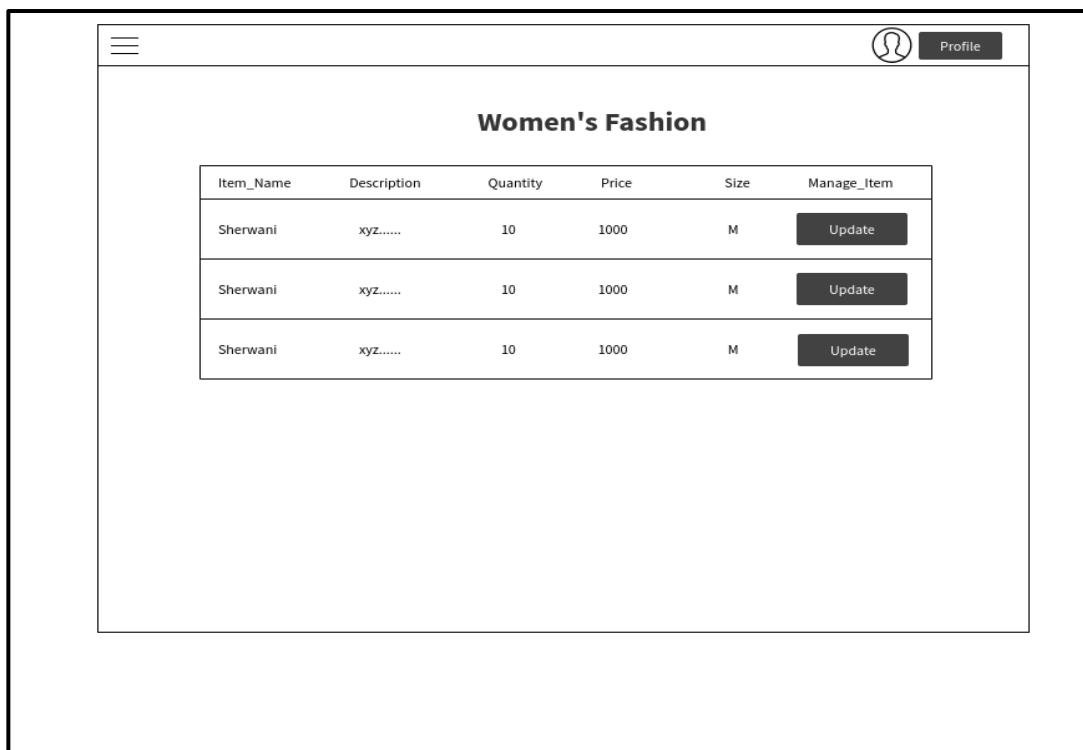


Figure 46: View Item Women(Admin)

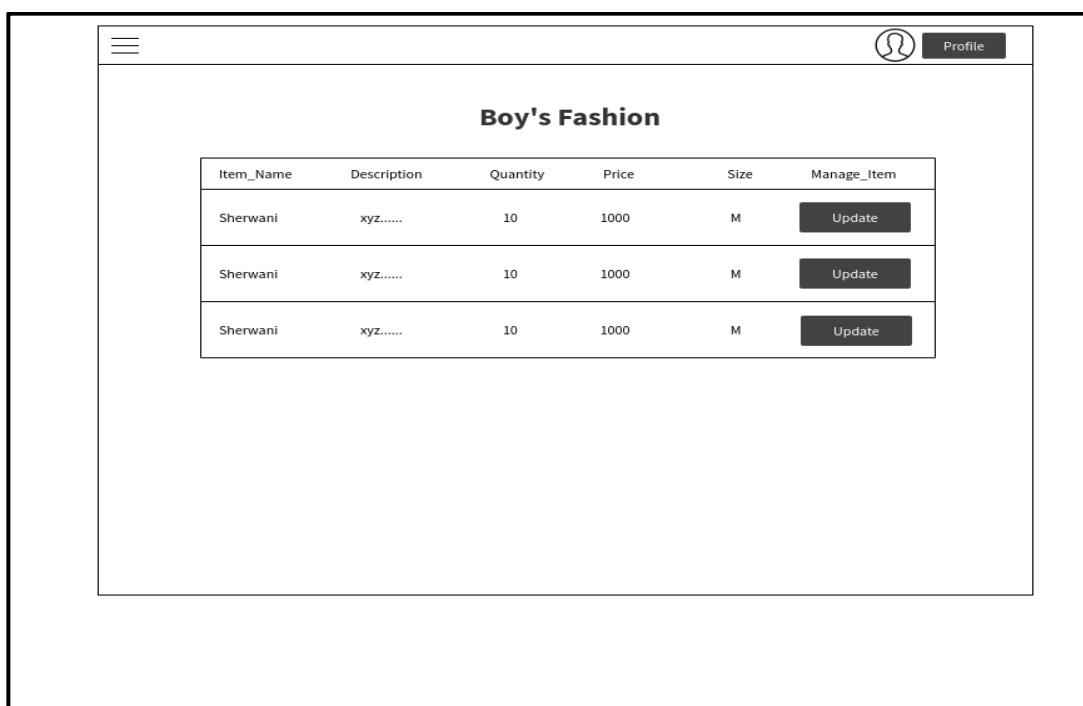


Figure 47: View Item Boy(Admin)

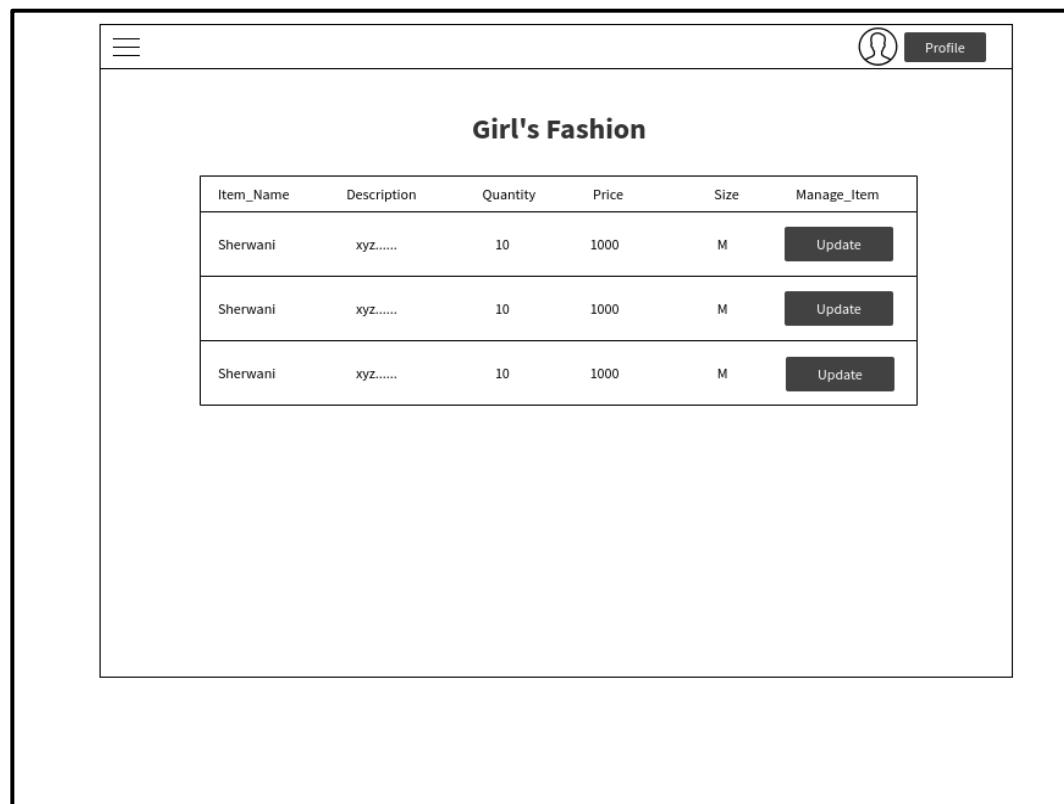


Figure 48: View Item Girl(Admin)

Update Item

Item_Name:

Quantity:

Description:

Price:

Subcategory:

Size:

Deals of the day

Recommended

Item_Image: No File Chosen

Optional Image: No File Chosen

Figure 49: Update Item

The screenshot shows a web application interface titled "Order". At the top right is a user profile icon and a "Profile" link. Below the title is a table with four columns: "Order Status", "Item Name", "Price", and "More Info". There are four rows in the table, each representing a pending order for "Sherwani" at a price of 2000. Each row has an "i" icon in the "More Info" column.

Order Status	Item Name	Price	More Info
Pending	Sherwani	2000	(i)
Pending	Sherwani	2000	(i)
Pending	Sherwani	2000	(i)
Pending	Sherwani	2000	(i)

Figure 50: Current Orders(Admin)

The screenshot shows a web application interface titled "Order Details". At the top right is a user profile icon and a "Profile" link. The main area contains a form with various input fields. The fields are arranged in pairs: Email (left), Quantity (right); Item_Name (left), Price (right); Customer Name (left), Ordered Date (right); Mobile Number (left), Delivery return date (right); Address (left), Order Status (right); Pincode (left), Order Type (right); and a dropdown menu labeled "Deliveryman" (right). At the bottom is a "Submit" button.

Figure 51: Order Details(Admin)

The screenshot shows a mobile application interface titled "Previous History". At the top right is a user icon and a "Profile" button. Below the title is a table with four columns: Customer Name, Item Name, Price, and More Info. The table contains five rows, each representing a purchase by a customer named "Johnny" for an item "Sherwani" at a price of "2000". Each row has an "i" icon in the "More Info" column.

Customer Name	Item Name	Price	More Info
Johnny	Sherwani	2000	(i)
Johnny	Sherwani	2000	(i)
Johnny	Sherwani	2000	(i)
Johnny	Sherwani	2000	(i)

Figure 52: Previous History(Admin)

The screenshot shows a mobile application interface titled "Previous History Details". At the top right is a user icon and a "Profile" button. The main area is a large gray box containing a form with various input fields. The fields are labeled: Email, Quantity, Item_Name, Price, Customer Name, Ordered Date, Mobile Number, Delivery return date, Address, Order Status, Pincode, Order Type, and Deliveryman. Each label is followed by a text input field.

Figure 53: Previous History Details(Admin)

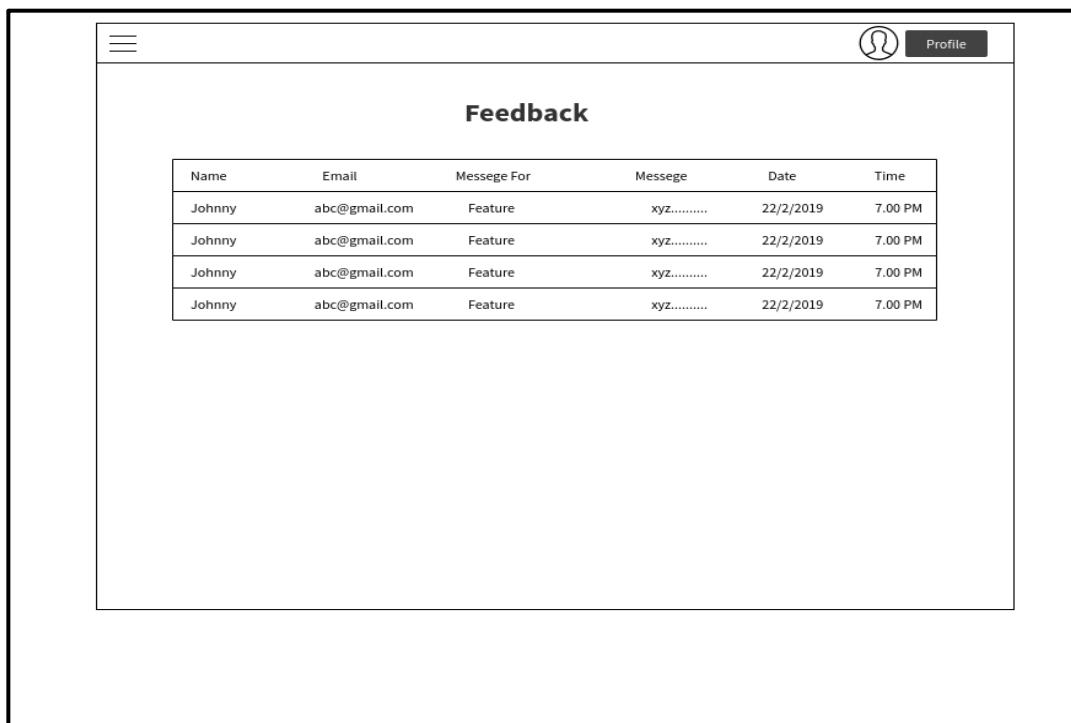
The screenshot shows a mobile application interface for adding a deliveryman. At the top right is a profile icon and a 'Profile' button. Below the header is the title 'Add Deliveryman'. The form contains several input fields: 'Email Id:' with an input box, 'Password:' with an input box, 'Name:' with an input box, 'Mobile Number:' with an input box, 'Address:' with an input box, 'Id Proof:' with a file selection button ('Choose File') and a message 'No File Chosen', and 'Image:' with a file selection button ('Choose File') and a message 'No File Chosen'. At the bottom is a large dark grey button labeled 'Create Account'.

Figure 54: Add Deliveryman

The screenshot shows a mobile application interface displaying 'Deliveryman Details'. At the top right is a profile icon and a 'Profile' button. Below the header is the title 'Deliveryman Details'. A table lists four deliverymen, each with a placeholder image (a square with an 'X'). The columns are labeled 'Image', 'Name', 'Address', 'Email', 'Mobile_Number', and 'Id_Proof'. The data for all four rows is identical:

Image	Name	Address	Email	Mobile_Number	Id_Proof
	Johnny	xyz.....	abc@abc.com	9985252552	
	Johnny	xyz.....	abc@abc.com	9985252552	
	Johnny	xyz.....	abc@abc.com	9985252552	
	Johnny	xyz.....	abc@abc.com	9985252552	

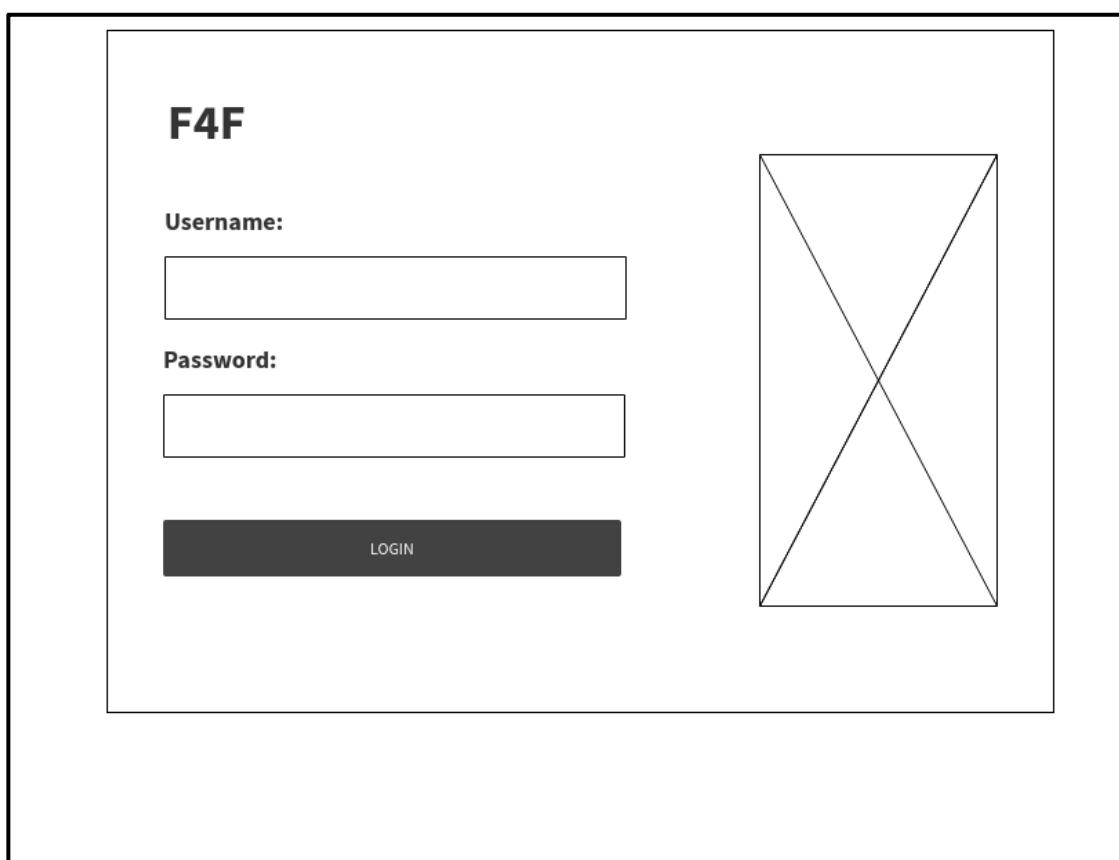
Figure 55: Deliveryman Details



A wireframe of a feedback page. At the top right is a user icon and a 'Profile' button. Below the header is a section titled 'Feedback' containing a table with four rows of data. The table has columns for Name, Email, Message For, Message, Date, and Time.

Name	Email	Mesage For	Mesage	Date	Time
Johnny	abc@gmail.com	Feature	xyz.....	22/2/2019	7.00 PM
Johnny	abc@gmail.com	Feature	xyz.....	22/2/2019	7.00 PM
Johnny	abc@gmail.com	Feature	xyz.....	22/2/2019	7.00 PM
Johnny	abc@gmail.com	Feature	xyz.....	22/2/2019	7.00 PM

Figure 56: Feedback



A wireframe of a deliveryman login page. At the top left is the text 'F4F'. The page features two input fields for 'Username' and 'Password', and a dark grey 'LOGIN' button below them. To the right of the inputs is a large, light gray square containing a large black 'X' mark.

Figure 57: Deliveryman Login

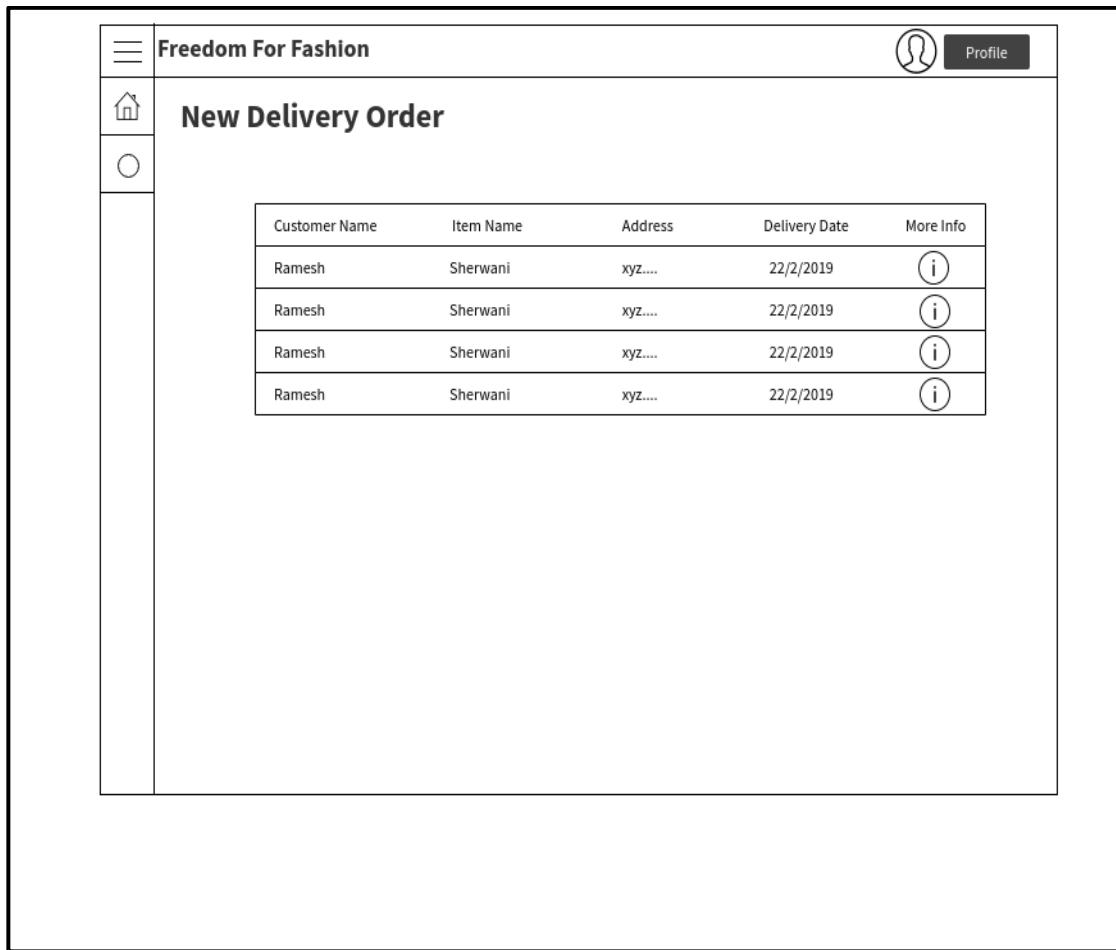
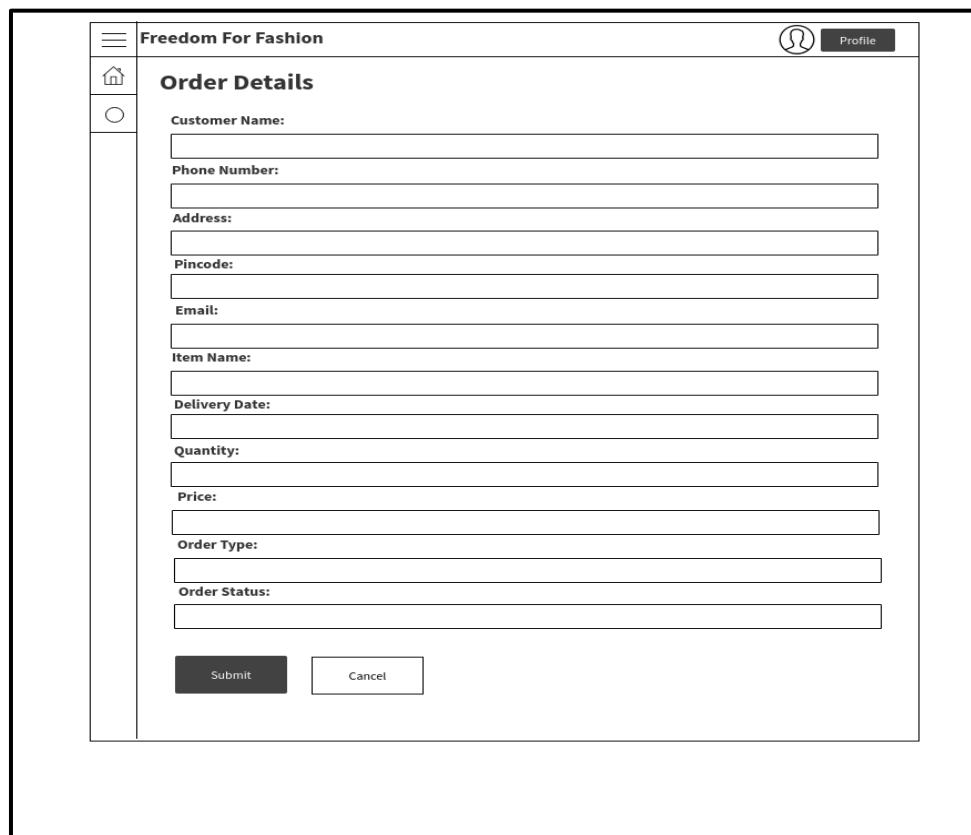


Figure 58: Delivery Order (Deliveryman)

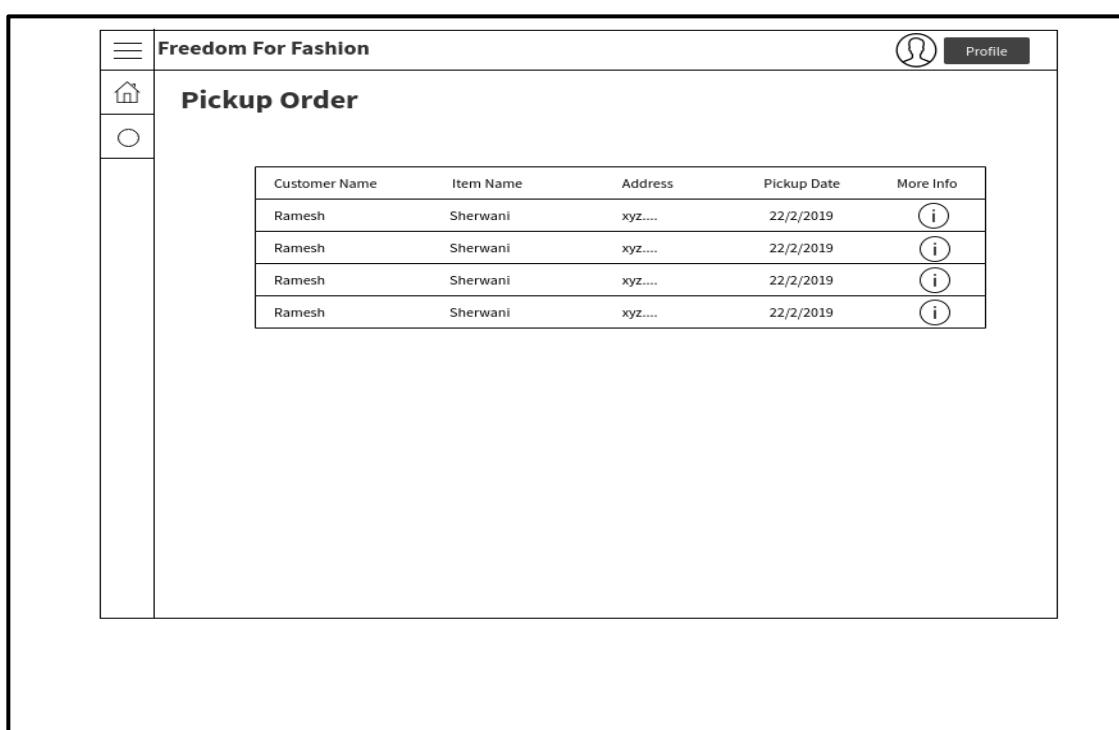


The screenshot shows a mobile application interface for 'Freedom For Fashion'. At the top, there is a navigation bar with three horizontal lines, a house icon, a circle icon, and a profile icon labeled 'Profile'. The main title 'Freedom For Fashion' is centered above a form titled 'Order Details'. The form contains the following fields with placeholder text:

- Customer Name:** _____
- Phone Number:** _____
- Address:** _____
- Pincode:** _____
- Email:** _____
- Item Name:** _____
- Delivery Date:** _____
- Quantity:** _____
- Price:** _____
- Order Type:** _____
- Order Status:** _____

At the bottom of the form are two buttons: 'Submit' (dark grey) and 'Cancel' (light grey).

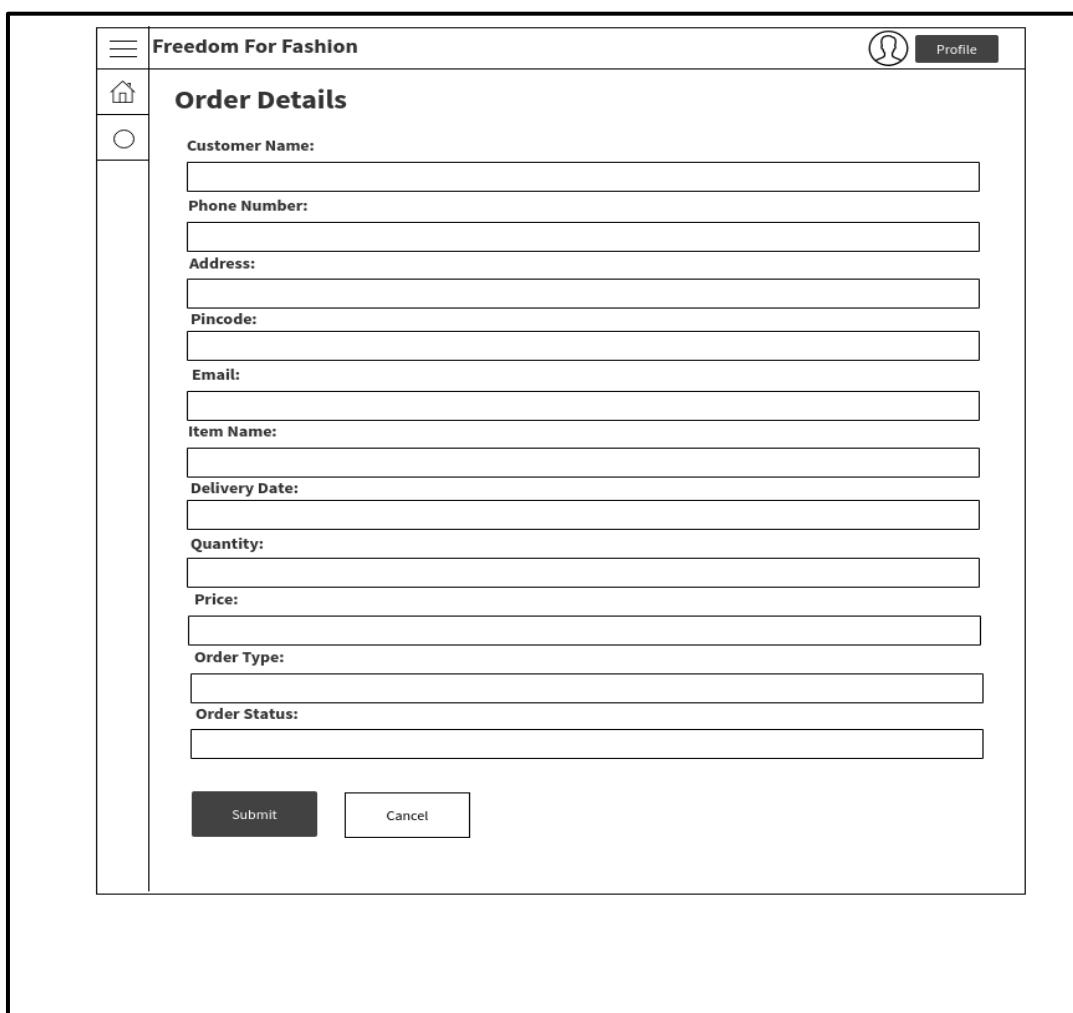
Figure 59: Order Details – Delivery Order (Deliveryman)



The screenshot shows a mobile application interface for 'Freedom For Fashion'. At the top, there is a navigation bar with three horizontal lines, a house icon, a circle icon, and a profile icon labeled 'Profile'. The main title 'Freedom For Fashion' is centered above a section titled 'Pickup Order'. Below this, there is a table listing four pickup orders:

Customer Name	Item Name	Address	Pickup Date	More Info
Ramesh	Sherwani	xyz....	22/2/2019	(i)
Ramesh	Sherwani	xyz....	22/2/2019	(i)
Ramesh	Sherwani	xyz....	22/2/2019	(i)
Ramesh	Sherwani	xyz....	22/2/2019	(i)

Figure 60 : Pickup Order (Deliveryman)



The image shows a mobile application interface for 'Freedom For Fashion'. At the top, there is a navigation bar with three icons: a menu icon (three horizontal lines), a home icon (a house), and a profile icon (a person). The title 'Freedom For Fashion' is displayed in the center of the bar. On the right side of the bar is a 'Profile' button with a user icon. Below the navigation bar, the main content area is titled 'Order Details'. It contains several input fields for customer information and order details, each preceded by a label. At the bottom of the form are two buttons: 'Submit' and 'Cancel'.

Label	Type
Customer Name:	Text Input
Phone Number:	Text Input
Address:	Text Input
Pincode:	Text Input
Email:	Text Input
Item Name:	Text Input
Delivery Date:	Text Input
Quantity:	Text Input
Price:	Text Input
Order Type:	Text Input
Order Status:	Text Input

Submit Cancel

Figure 61: Order Details – Pickup Order (Deliveryman)

4.5 TimeLine Chart

Category	Weeks	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th	15th	16th	17th	18th
Analysis																			
Design																			
Implementation																			
Testing																			
Doc/Finalize																			

Table 12: TimeLine Chart

5. IMPLEMENTATION

5.1 Implementation Environment

TOOLS

- Visual Studio Code
- Node.js
- Browsers (Chrome, Mozilla, Opera)
- Draw.io
- Lighthouse
- Microsoft Office

TECHNOLOGIES

Main Technology	:	PWA
Language	:	HTML, JavaScript
Database	:	NO SQL
Query Language	:	Real-time Database API/ Firebase
Others	:	CSS, jQuery

Table 13: Implementation Environment

5.2 Module Specification

End-users Modules

Below is the modules list of the end users i.e. Customers or users.

Modules	Description
Login	The user has to login into the system through gmail or mobile number.
Update Details	In this User can fill his personal details and update it.
Deals Of The Day	In this User will get deals of the days which was sort-listed by admin.
Recommended	Some items will be displayed by admin to user according admin's choice.
Men/Women/Kids Fashion	In this customer can rent items according these categories.
Send Feedback	Here customer can give feedback .
Checkout	After adding items into the cart customer can check personal details and total amount and order details.
Invoice	After completing order user will get whole details in this page.
Order History	In this user can check their previous order history.

Table 14: End Users Module specification

Admin Modules

Below is the modules list for Admin:

Modules	Description
Login	Admin has to login into the application to use it further.
Manage Item	Admin can add, update and remove items.
Manage Delivery	Admin can add deliveryman and assign deliveryman for delivery.
View Orders	Admin can view current and previous order.
View Feedback	Can view list of feedbacks given by users.

Table 15: Admin Module specification

Deliveryman Module specification

Modules	Description
Delivery Orders	Can view delivery details and can update delivery status.
Pickup Orders	Can view Pickup details and can update delivery status.

5.3 Security Features

Firebase Security Features

Firebase Realtime Database Rules determine who has read and write access to your database, how your data is structured, and what indexes exist. These rules live on the Firebase servers and are enforced automatically at all times. Every read and write request will only be completed if your rules allow it. By default, your rules do not allow anyone access to your database. This is to protect your database from abuse until you have time to customize your rules or set up authentication.

The Firebase Realtime Database provides a full set of tools for managing the security of your app. These tools make it easy to authenticate your users, enforce user permissions, and validate inputs.

Firebase-powered apps run more client-side code than those with many other technology stacks. Therefore, the way we approach security may be a bit different than you're used to.

Authentication:

A common first step in securing your app is identifying your users. This process is called authentication. You can use Firebase Authentication to have users sign in to your app. Firebase Authentication includes drop-in support for common authentication methods like Google and Facebook, as well as email and password login, anonymous login, and more.

User identity is an important security concept. Different users have different data, and sometimes they have different capabilities. For example, in a chat application, each message is associated with the user that created it. Users may also be able to delete their own messages, but not messages posted by other users.

Authorization:

Identifying your user is only part of security. Once you know who they are, you need a way to control their access to data in your database. Firebase Database Rules allow you to control access for each user.

The Firebase Database Rules include built-in variables and functions that allow you to refer to other paths, server-side timestamps, authentication information, and more.

Data validation:

The Firebase Realtime Database is schemaless. This makes it easy to change things as you develop, but once your app is ready to distribute, it's important for data to stay consistent. The rules language includes a .validate rule which allows you to apply validation logic using the same expressions used for .read and .write rules. The only difference is that validation rules do not cascade, so all relevant validation rules must evaluate to true in order for the write to be allowed.

(Security Features, 2002)

5.4 Coding Standards

Coding standards define a programming style. A coding standard does not usually concern itself with wrong or right in a more abstract sense. It is simply a set of rules and guidelines for the formatting of source code.

The Coding standards are used to make sure that the names used in the project are used in a most relevant form. All the attributes have unique name which are self-explanatory. The Ids and names used for different methods and functions are different resembling their work. Methods are stored in different files for procedural structure with true concepts of Object-Oriented to a certain standard

(Steave, 2007)

5.5 Coding Sample

View Men's Fashion Data list (js file)

```

var databaseRef = firebase.database().ref('item');

document.getElementById('fileButton').onchange = function(event){
    selectedFile = event.target.files[0];
}

// var databaseRef_women = firebase.database().ref();

firebase.auth().onAuthStateChanged(function(user) {

    var url_string = window.location.href;
    var url = new URL(url_string);
    var id = url.searchParams.get('id');
    var uid = url.searchParams.get('uid');

    databaseRef.child('Men').child(id).on('value',function(update_snapshot){

        document.getElementById('item_name').value = update_snapshot.child('item_name').val();
        document.getElementById('item_description').value = update_snapshot.child('item_description').val();
        document.getElementById('item_subcategory').value = update_snapshot.child('item_subcategory').val();
        document.getElementById('item_quantity').value = update_snapshot.child('item_quantity').val();
        document.getElementById('item_price').value = update_snapshot.child('item_price').val();
    })
})
}

```

```
document.getElementById('item_size').value  
update_snapshot.child('item_size').val();  
=
```

```
});
```

```
databaseRef.child('Men').child(id).child('deals_of_the_day').on('value',function(update  
_checkbox_snapshot){
```

```
var update_checkbox = update_checkbox_snapshot.val();
```

```
console.log(update_checkbox);
```

```
if(update_checkbox == true)
```

```
{
```

```
var check=document.getElementsByTagName('input');
```

```
for(var i=0;i<check.length;i++){
```

```
if(check[i].type=='checkbox')
```

```
{
```

```
document.getElementById('dod').checked=true;
```

```
}
```

```
else
```

```
{
```

```
document.getElementById('dod').removeAttribute('checked');
```

```
}
```

```
        }
    }
else
{
}
});

firebase.auth().onAuthStateChanged(function(user) {

    databaseRef.child('Men').child(id).child('recommended').on('value',function(update_checkbox_snapshot_rec){

        var update_checkbox_rec = update_checkbox_snapshot_rec.val();

        console.log(update_checkbox_rec);
        if(update_checkbox_rec == true)
        {

            var check_rec=document.getElementsByTagName('input');
            for(var i=0;i<check_rec.length;i++){
                if(check_rec[i].type=='checkbox')
                {
                    document.getElementById('rec').checked=true;
                }
                else
                {
                    document.getElementById('rec').removeAttribute('checked');
                }
            }
        }
        else
    }
});
```

```

    {
    }
});

});

document.getElementById('btn_insertname').onclick = function(){

    var item_name = document.getElementById('item_name').value;
    var item_description = document.getElementById('item_description').value;
    var item_subcategory = document.getElementById('item_subcategory').value;
    var deals_of_the_day = document.getElementById('dod').checked;
    var recommended = document.getElementById('rec').checked;
    var item_quantity = document.getElementById('item_quantity').value;
    var item_price = document.getElementById('item_price').value;
    var item_size = document.getElementById('item_size').value;
    // console.log(email_wd);

    var filename = selectedFile.name;
    var storageRef = firebase.storage().ref('ClothingPictures/' + selectedFile.name);
    var uploadTask = storageRef.put(selectedFile);
    uploadTask.on('state_changed',
        function(snapshot){

            var progress = (snapshot.bytesTransferred / snapshot.totalBytes) * 100;
            alert('Upload Progress : '+progress+'%');

        },function(error){

    });

},function(){

    uploadTask.snapshot.ref.getDownloadURL().then(function(downloadURL){

        databaseRef.child('Men').child(id).update({

            item_name : item_name,
            item_description : item_description,
            item_subcategory : item_subcategory,
        })
    });
}

```

```
deals_of_the_day : deals_of_the_day,  
recommended : recommended,  
item_quantity : item_quantity,  
item_price : item_price,  
item_size : item_size,  
optional_image : downloadURL  
});  
alert('Data Updated Successfully');  
});  
});  
};  
});
```

6. TESTING

Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of A software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process.

(Testing, 1996)

6.1 Test Plan

The test plan is to check all functionality according to SRS defined requirements. Apart from that, the current market needs to be attained at every point of time. We need to make necessary changes if any are required. We need to make sure that all the objectives for which the software was build is properly attained while the system is secure and easy to use. The system also has to be sustainable. Test planning, the most important activity to ensure that there is initially a list of tasks and milestones in a baseline plan to track the progress of the project. It also defines the size of the test effort.

Test Planning Activities:

- To determine the scope and the risks that need to be tested and that are NOT to be tested.
- Documenting Test Strategy.
- Making sure that the testing activities have been included.
- Deciding Entry and Exit criteria.
- Evaluating the test estimate.
- Planning when and how to test and deciding how the test results will be evaluated, and defining test exit criterion.
- The Test artefacts delivered as part of test execution.
- Defining the management information, including the metrics required and defect resolution and risk issues.

- Ensuring that the test documentation generates repeatable test assets.
- (H.W.L, 1996)

6.2 Test Strategy

To perform testing in a planned and systematic manner, software testing strategy is developed. A testing strategy is used to identify the levels of testing which are to be applied along with the methods, techniques, and tools to be used during testing. Developing a test strategy, which efficiently meets the requirements of an organization, is critical to the success of software development in that organization. Therefore, a software testing strategy should contain complete information about the procedure to perform testing and the purpose and requirements of testing. The choice of software testing strategy is highly dependent on the nature of the developed software. For example, if the software is highly data intensive then a strategy that checks structures and values properly to ensure that all inputs given to the software are correct and complete should be developed. Similarly, if it is transaction intensive then the strategy should be such that it is able to check the flow of all the transactions. The design and architecture of the software are also useful in choosing testing strategy. A number of software testing strategies are developed in the testing process. All these strategies provide the tester a template, which is used for testing. (test strategy, 1993)

Below are the types of testing strategy:

Graphical User Interface

The graphical user interface (GUI), is a type of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation. GUIs were introduced in reaction to the perceived steep learning curve of command-line interfaces (CLIs), which require commands to be typed on a computer keyboard. (W3C, 1993)

Usability Testing

Usability testing is a method used in user-centered interaction design to estimate a product by testing it on users. This can be seen as a unique usability practice, since it

offers direct input on how actual users use the system. This is in contrast with usability inspection methods where experts use different methods to evaluate a user interface without concerning users. (ut, 2018)

Functional Testing

Functional testing is a quality assurance (QA) process and a type of black-box testing that bases its test cases on the specifications of the software component under test. Functions are tested by giving them input and examining the output, and internal program structure is rarely considered (unlike white-box testing). Functional testing frequently defines what the system actually does. (bradford, 2009)

6.3 Test Methods

There are two basics of software testing: blackbox testing and whitebox testing.

Blackbox Testing

Black box testing is a testing technique that overlooks the internal mechanism of the system and focuses on the output generated. It is also called functional testing.

Whitebox Testing

White box testing is a technique that takes into account the internal mechanism of a system. It is also called structural testing and glass box testing.

Black box testing is mainly used for validation while white box testing is used for verification.

Types of testing strategies are:

Unit testing

Here in the project we have performed Unit testing. We have tested the most basic parts of codes by dividing them into small units

Component testing

Component testing also known as module or program testing. It's similar to unit testing but contains a higher level of integration. Here modules of the software were tested for defects to verify their individual function.

Integration testing

After performing the unit and component testing, integration testing was performed. Here the modules were integrated together and were tested to make sure that their methods and functions perform properly

Functional Testing

Functional testing was done to ensure that the specified functionality required in the system requirements works properly.

System testing

Components of a project were tested as a whole in different mobile phones with this method. It is one of the final tests in the process. It determines whether the system functions as per the requirements.

Acceptance testing

In this test the client tests software to find out if the developer has created the program to desired specifications. This testing was conducted by our external guide from Enlighten Infosystems.

Security testing

This measures the software's ability to protect users' security. This means authorization functions, authentication, confidentiality, integrity, availability, and non-repudiation.

(test methods, 1970)

6.4 Test Cases

ID	FFTC1
Module	User Login
Dependency	User should have Gmail account or any other email account and Mobile Number
Variables	User Email, Password or Mobile number
Precondition	User should have Gmail account or any other email account and Mobile Number
Action	User should enter email id and password
Expected Result	1.If user is logging in first time it will redirect to the welcome details page 2.If user already login once it will be redirected to home page
Actual Result	1.User is logging in first time redirected to the welcome details page 2.User already login once has redirected to home page
Failure Condition	If Incorrect Email and Password or Mobile Number show message of incorrect Email Id and Password or Mobile Number

Table 16: Test Case: User Login

ID	FFTC2
Module	Update Details
Dependency	User should Login
Variables	First Name, Last Name, Mobile Number, Address, Pin code, E-Mail
Precondition	User should Login
Action	User should update enter Name, Address, Phone No., Shop Name, Email, Password
Expected Result	If user change any details, it will update detail
Actual Result	Updated Details

Table 17: Test Case: Update Details

ID	FFTC3
Module	Men's Clothing
Dependency	User should Login
Variables	Add to cart
Precondition	User should Login
Action	User should click on item which he/she wants to rent
Expected Result	Item added in cart successfully
Actual Result	Item added in cart successfully
Failure Condition	-

Table 18: Test Case: Men's Clothing

ID	FFTC4
Module	Women's Clothing
Dependency	User should Login
Variables	Add to cart
Precondition	User should Login
Action	User should click on item which he/she wants to rent
Expected Result	Item added in cart successfully
Actual Result	Item added in cart successfully
Failure Condition	-

Table 19: Test Case: Women's Clothing

ID	FFTC5
Module	Kids Clothing
Dependency	User should Login
Variables	Add to cart
Precondition	User should Login
Action	User should click on item which he/she wants to rent
Expected Result	Item added in cart successfully
Actual Result	Item added in cart successfully
Failure Condition	-

Table 20: Test Case: Kid's Clothing

ID	FFTC6
Module	Send Feedback
Dependency	User should Login
Variables	Name, Email, Messege
Precondition	User should Login
Action	User should enter Name , Email And Messege
Expected Result	Feedback Sent Successfully
Actual Result	Feedback Sent Successfully
Failure Condition	If Incorrect or Empty any detail show message of incorrect detail information or please fill this.

Table 21: Test Case: Send Feedback

ID	FFTC7
Module	Checkout
Dependency	User should Login
Variables	Quantity and Days(User data will be already auto filled)
Precondition	User should Login
Action	User should enter quantity and days(User can update that data if they want)
Expected Result	Checkout
Actual Result	Checkout
Failure Condition	If cannot enter number of days, it show message of please select days.

Table 22: Test Case: Checkout

ID	FFTC8
Module	Invoice
Dependency	User should Login and Checkout
Variables	-
Precondition	User should Login and Checkout
Action	-
Expected Result	Invoice
Actual Result	Invoice
Failure Condition	-

Table 23: Test Case: Invoice

ID	FFTC9
Module	Admin Login
Dependency	Admin should have particular email and password for login
Variables	Admin Email and Password
Precondition	Admin should have particular email and password for login
Action	Admin should enter email and password
Expected Result	Home Page
Actual Result	Home Page
Failure Condition	-

Table 24: Test Case: Admin Login

ID	FFTC10
Module	Add Item
Dependency	Admin should login
Variables	Item name, Description, category, quantity, price, size, image
Precondition	Admin should login
Action	Admin should enter Item name, Description, category, price, size,image
Expected Result	Item Added Successfully
Actual Result	Item Added Successfully
Failure Condition	If admin forgot to enter any details, it will show message of please fill this detail

Table 25: Test Case: Add Item(Admin)

ID	FFTC12
Module	View Item(Men's Fashion)
Dependency	Admin should login
Variables	Update And Inactive
Precondition	Admin should login
Action	Admin can update details of item or inactive item
Expected Result	Item updated successfully or item inactivated
Actual Result	Item updated successfully or item inactivated
Failure Condition	If admin forgot to enter any details, it will show message of please fill this detail

Table 26: Test Case: View Item(Men's Fashion)

ID	FFTC13
Module	View Item(Women's Fashion)
Dependency	Admin should login
Variables	Update And Inactive
Precondition	Admin should login
Action	Admin can update details of item or inactive item
Expected Result	Item updated successfully or item inactivated
Actual Result	Item updated successfully or item inactivated
Failure Condition	If admin forgot to enter any details, it will show message of please fill this detail

Table 27: Test Case: View Item(Women's Fashion)

ID	FFTC14
Module	View Item(Boy's Fashion)
Dependency	Admin should login
Variables	Update And Inactive
Precondition	Admin should login
Action	Admin can update details of item or inactive item
Expected Result	Item updated successfully or item inactivated
Actual Result	Item updated successfully or item inactivated
Failure Condition	If admin forgot to enter any details, it will show message of please fill this detail

Table 28: Test Case: View Item(Boy's Fashion)

ID	FFTC15
Module	View Item(Men's Fashion)
Dependency	Admin should login
Variables	Update And Inactive
Precondition	Admin should login
Action	Admin can update details of item or inactive item
Expected Result	Item updated successfully or item inactivated
Actual Result	Item updated successfully or item inactivated
Failure Condition	If admin forgot to enter any details, it will show message of please fill this detail

Table 29: Test Case: View Item(Girl's Fashion)

ID	FFTC16
Module	Add Deliveryman
Dependency	Admin should login
Variables	Email id, Name, Address, Password, Mobile number, Idproof, Image
Precondition	Admin should login
Action	Admin should enter email , address, password, mobile number, id proof and image
Expected Result	Deliveryman Added Successfully
Actual Result	Deliveryman Added Successfully
Failure Condition	If admin forgot to enter any details, it will show message of please fill this detail

Table 30: Test Case: Add Deliveryman

ID	FFTC17
Module	Deliveryman Details
Dependency	Admin should Login
Variables	-
Precondition	Admin should Login
Action	-
Expected Result	Deliveryman Details
Actual Result	Deliveryman Details
Failure Condition	-

Table 31 : Test Case: Deliveryman Details

ID	FFTC18
Module	Current Status
Dependency	Admin should Login
Variables	Info Button
Precondition	Admin should Login
Action	Admin should click on info button
Expected Result	Order Details
Actual Result	Order Details
Failure Condition	-

Table 32 : Test Case: Current Status

ID	FFTC19
Module	Order details
Dependency	Admin should Login
Variables	Set Deliveryman
Precondition	Admin should Login
Action	Admin should Select Deliveryman
Expected Result	Order Sent successfully
Actual Result	Order Sent successfully
Failure Condition	If Admin forgot to set deliveryman it will show message of please select deliveryman.

Table 33 : Test Case: Order Details

ID	FFTC20
Module	Previous History
Dependency	Admin should Login
Variables	More info button
Precondition	Admin should Login
Action	Admin should click on info button
Expected Result	Previous Order Details
Actual Result	Previous Order Details
Failure Condition	-

Table 34 : Test Case: Previous History

ID	FFTC21
Module	Previous Order Details
Dependency	Admin should Login
Variables	-
Precondition	Admin should Login
Action	-
Expected Result	Previous Order Details
Actual Result	Previous Order Details
Failure Condition	-

Table 35 : Test Case: Previous Order Details

ID	FFTC22
Module	Feedback
Dependency	Admin should Login
Variables	-
Precondition	Admin should Login
Action	-
Expected Result	Feedback
Actual Result	Feedback
Failure Condition	-

Table 36 : Test Case: Feedback

ID	FFTC23
Module	Deliveryman Login
Dependency	Deliveryman should have particular email and password for login
Variables	Deliveryman Email and Password
Precondition	Deliveryman should have particular email and password for login
Action	Deliveryman should enter email and password
Expected Result	Home Page
Actual Result	Home Page
Failure Condition	Login Page

Table 37: Test Case: Deliveryman Login(Deliveryman)

ID	FFTC24
Module	Delivery Order
Dependency	Deliveryman should login
Variables	More Info Button
Precondition	Deliveryman should login
Action	Deliveryman should click on more info button
Expected Result	Order Details
Actual Result	Order Details
Failure Condition	-

Table 38: Test Case: Delivery Order (Deliveryman)

ID	FFTC25
Module	Order Details
Dependency	Deliveryman should login
Variables	Order Status, Deposit Taken, Any Damage
Precondition	Deliveryman should login
Action	Deliveryman Should Order Status and should submit
Expected Result	Delivery Order
Actual Result	Delivery Order
Failure Condition	-

Table 39: Test Case: Order Details(Delivery Order)

ID	FFTC26
Module	Pickup Order
Dependency	Deliveryman should login
Variables	More Info Button
Precondition	Deliveryman should login
Action	Deliveryman should click on more info button
Expected Result	Order Details
Actual Result	Order Details
Failure Condition	-

Table 40: Test Case: Pickup Order (Deliveryman)

ID	FFTC27
Module	Order Details
Dependency	Deliveryman should login
Variables	Order Status, Deposit Taken, Any Damage
Precondition	Deliveryman should login
Action	Deliveryman Should Order Status and should submit
Expected Result	Pickup Order
Actual Result	Pickup Order
Failure Condition	-

Table 41: Test Case: Order Details(Pickup Order)

7. SCREENSHOTS

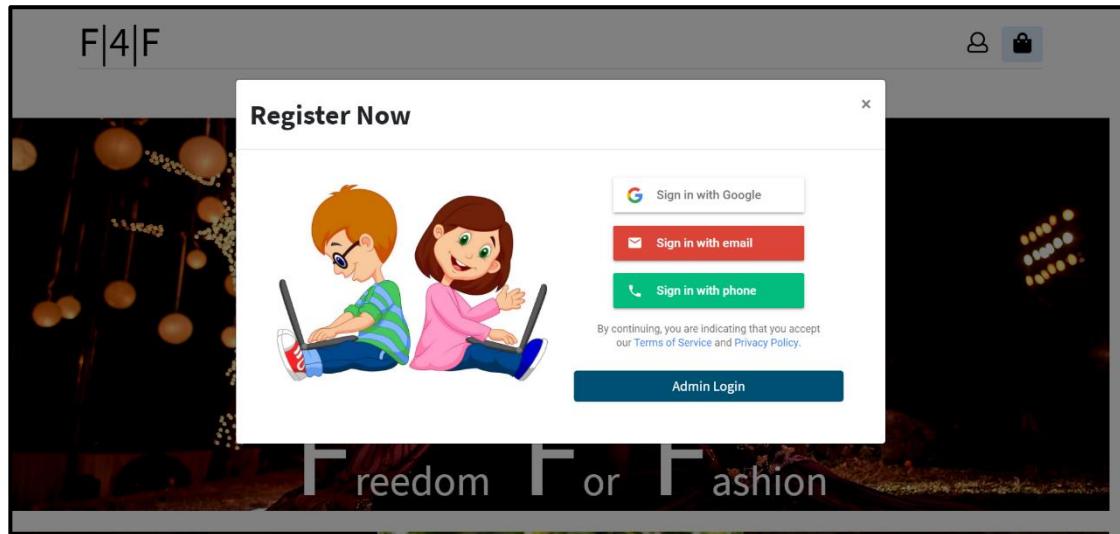


Figure 62 : User Login

A screenshot of a 'Welcome Details' form. The title 'Welcome to Freedom for Fashion' is at the top in blue. Below it, 'Details' is written in large blue letters. A sub-instruction 'Fill the details to proceed.' is in small gray text. The form itself has a dark background with white input fields. It includes fields for 'First Name', 'Last Name', 'Email' (with a note about email ID being non-changeable), 'Phone Number (xxxx-xxx-xxx)', 'Address', and 'Pin Code (000-000)'. At the bottom is a teal 'Submit' button.

Figure 63: Welcome Details

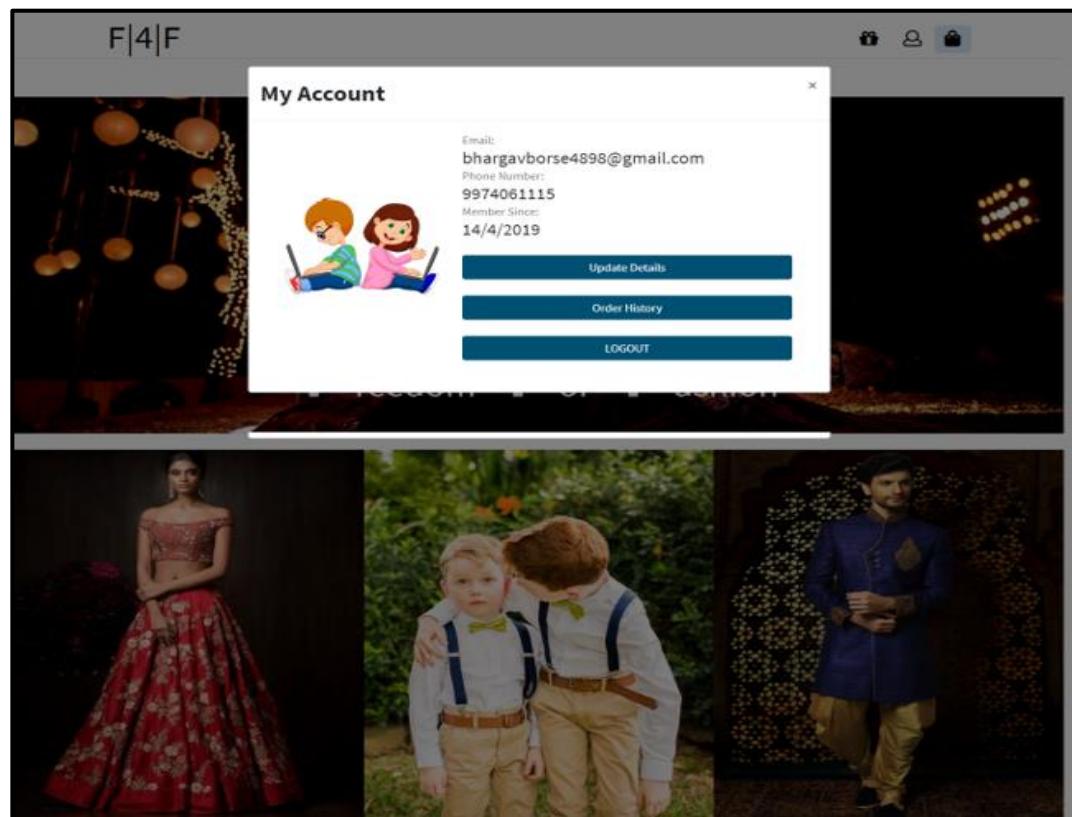


Figure 64: After Login

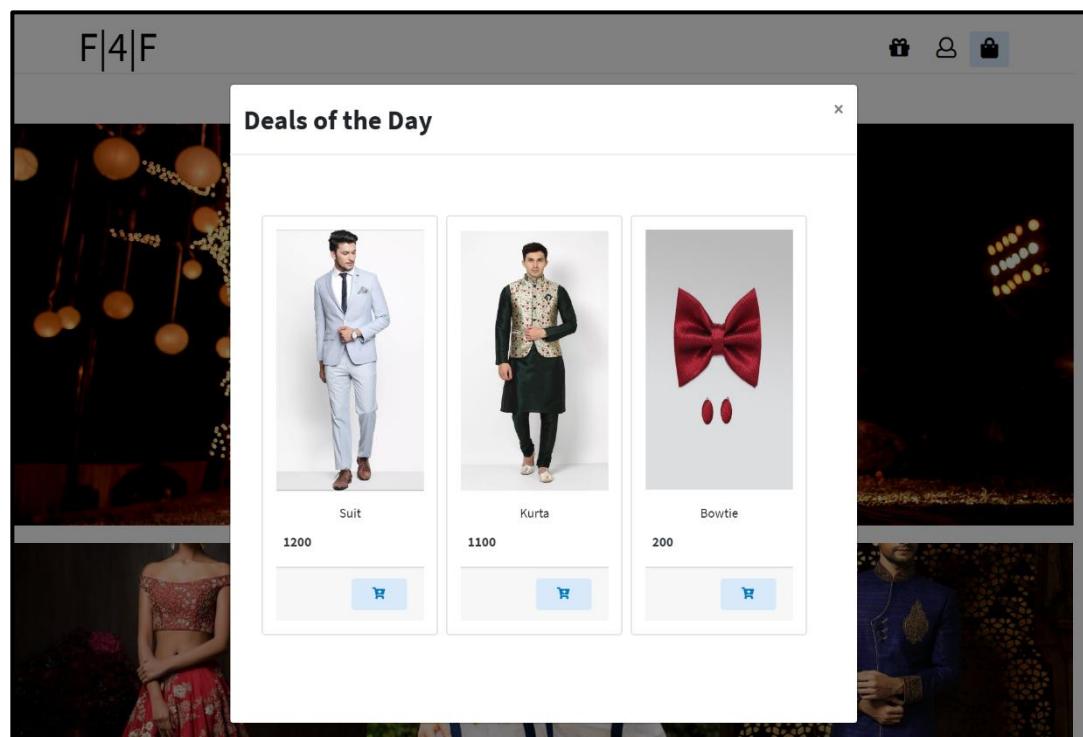


Figure 65: Deals of the Day

F|4|F

Home Men's clothing ▾ Women's clothing ▾ Kids Clothing ▾ Feedback

Item Type	Price (₹)
Sherwani	1050
Blazer	1100
Koti	950
Suit	1200
Kurta	1100
Bowtie	200
Bowtie	200
Sherwani	1200
Blazer	1150
Lapelpin	150

HOME DEALS PROFILE FEEDBACK

FREEDOM FOR FASHION
© 2015 Freedom for Fashion. All rights reserved.

Figure 66: Men's Clothing

F|4|F

Home Men's clothing Women's clothing Kids Clothing Feedback

Anarkali 1500	Anarkali 1500	Gown 1000	Lehenga 1050
Saree 1100	Earring 150	Earring 150	Topskirt 1120
Lehenga 1450	Gown 1300		

HOME DEALS PROFILE FEEDBACK

FREEDOM FOR FASHION

© 2019 Freedom For Fashion. All rights reserved

Figure 67: Women's Clothing

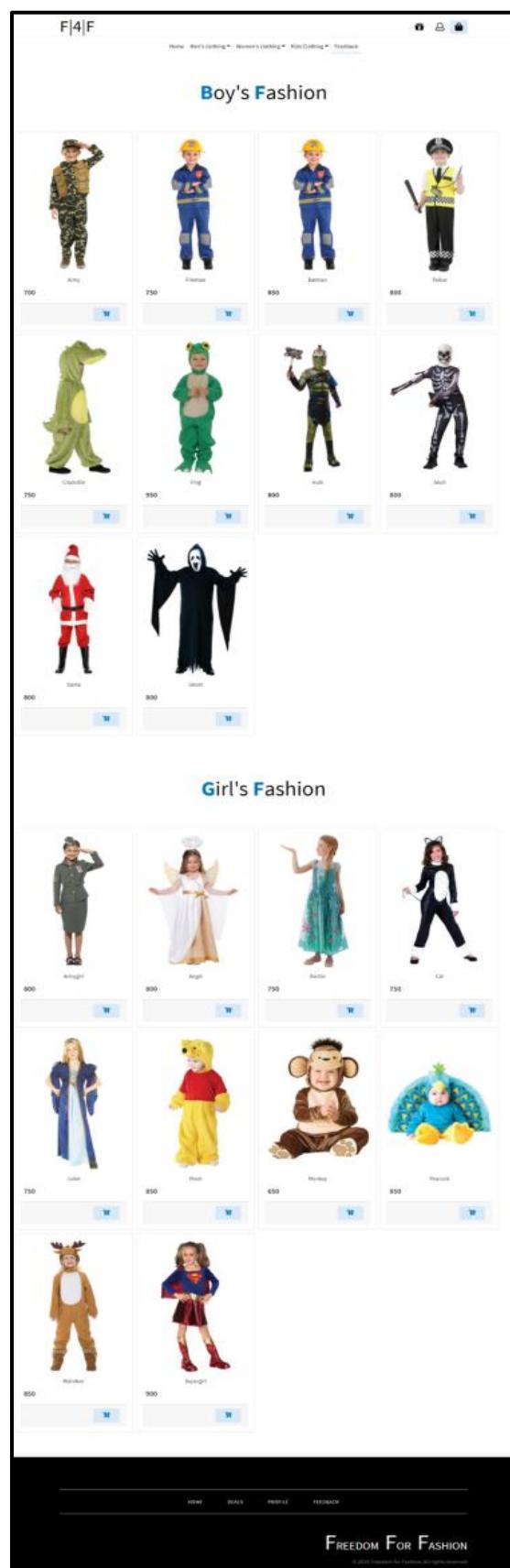


Figure 68: Kid's Clothing

F|4|F

[Home](#) [Men's clothing](#) [Women's clothing](#) [Kids Clothing](#) [Feedback](#)

Check Out

Review Your Order

SL No.	Product	Quantity	Product Name	Price	Remove
1		<input type="button" value="-"/> <input type="text" value="1"/> <input type="button" value="+"/>	Kurta	₹1100	<input type="button" value="X"/>

[Add more items](#)

Your Shopping Cart Contains: 1 Products

Billing Address

First Name:

Last Name:

Mobile number:

Address:

Pin Code:

E-mail:

Delivery Date:

Delivery Time:

Return Date:

Return Time:

[Next](#)

HOME

DEALS

PROFILE

FEEDBACK

FREEDOM FOR FASHION

© 2019 Freedom for Fashion. All rights reserved

Figure 69: Checkout Page

104

The screenshot shows the 'Invoice' page for an order from 'Freedom For Fashion'. At the top, there are three sections: 'Order Owner' (Dharmesh, 7016977902, pateldharmesh007.dp@gmail.com), 'Shipping Address' (b-67 mangaldeep society, makrand desai road, vasna, 390007), and 'Order Date & Timings' (Delivery Date: 2019-04-29, Delivery Time: 13:00, Return Date: 2019-05-01, Return Time: 18:00, Total Days: 2). Below these is a note: 'Purchase Date: 29/4/2019'. The 'Order summary' table details the purchase of a 'Kurta' at 1100 for 1 quantity, with a rent period of 2 days, resulting in a total of 3003. The table includes subtotals for Subtotal (2200), GST(5%) (110), and Deposit (693). At the bottom are buttons for 'Back to Checkout', 'Add More Item', 'Print Invoice', and 'Finish'. The footer features the brand name 'FREEDOM FOR FASHION' and a copyright notice: '© 2019 Freedom for Fashion. All rights reserved'.

Figure 70: Invoice

The screenshot shows the 'ADMIN PANEL' login page. It features a dark background with white text. The title 'ADMIN PANEL' is centered at the top. Below it is a 'EMAIL ID' field containing 'pateldharmesh007.dp@gmail.com'. Underneath is a 'PASSWORD' field with several asterisks. At the bottom are two buttons: 'Forgot Password' on the left and 'Login to Account' on the right.

Figure 71: Admin Login

Freedom For Fashion

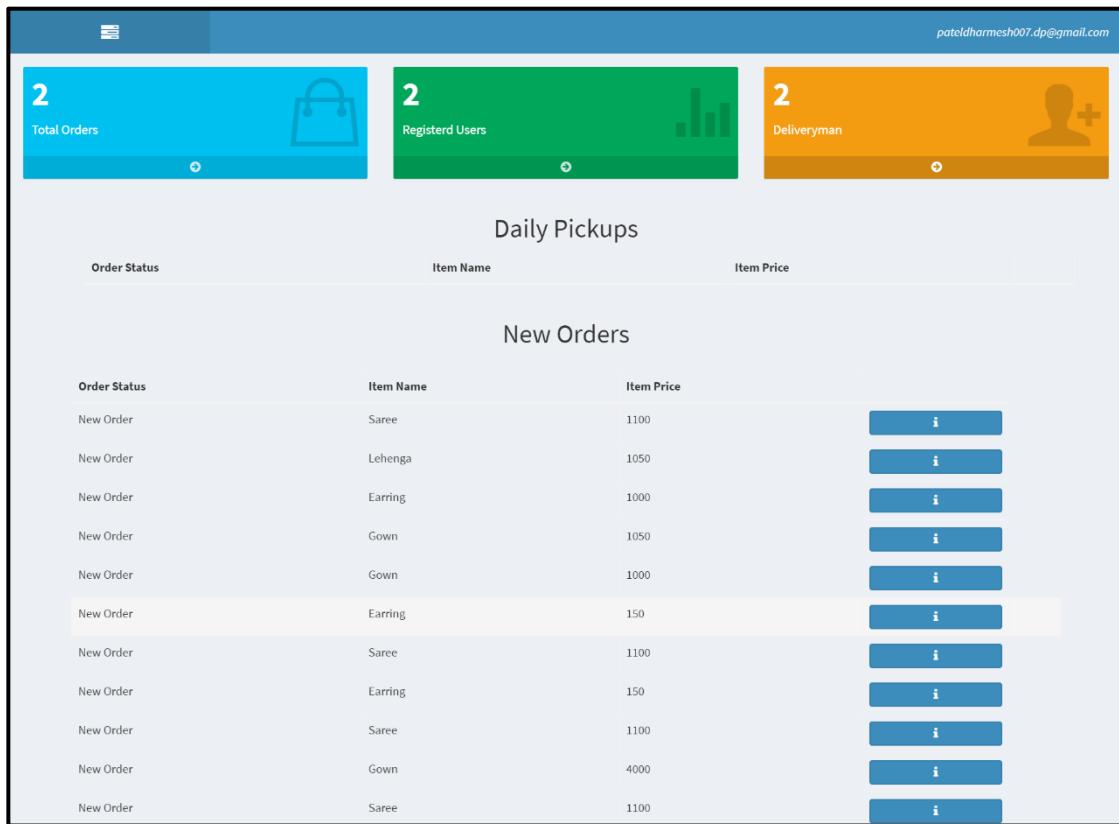


Figure 72: Admin Dashboard

The "Add Item" form is titled "Add Item" and contains fields for item details, categories, deals, and optional image upload.

Fields:

- Item Name:** Sherwani
- Quantity:** 0-100
- Description:** it is
- Price:** 500rs
- Category:** Men
- Size:** -Select Size--
- Deals Of The Day:**
- Recommended:**
- Optional Image:** Choose File | No file chosen

Buttons:

- A large blue "Add Item" button at the bottom.

Figure 73: Add Item

Item Name	Description	Quantity	Price	Size	Item Status	Manage Item
Lablepin	hjgf	25	150	M	true	<button>Update</button>
Blazer	wfegfgn	10	1150	M	true	<button>Update</button>
Sherwani	wfegfgn	10	1200	XL	true	<button>Update</button>
Bowtie	adoiuytre	20	200	XL	true	<button>Update</button>
Bowtie	adoiuytre	30	200	XL	true	<button>Update</button>
Kurta	adsfgdsv	20	1100	S	true	<button>Update</button>
Suit	adsfg	12	1200	XL	true	<button>Update</button>
Koti	fdfv	15	950	M	true	<button>Update</button>
Blazer	tfg	15	1100	L	true	<button>Update</button>
sherwani	fcgh	10	1050	XL	true	<button>Update</button>

Figure 74: View Item(Men's Fashion)

Item Name	Description	Quantity	Price	Size	Item Status	Manage Item
Gown	wdsv	20	1300	XL	true	<button>Update</button>
Lehenga	wdsv	20	1450	XL	true	<button>Update</button>
Topskirt	wdsv	20	1120	XL	true	<button>Update</button>
Earring	dsfv	20	150	L	true	<button>Update</button>
Earring	dsfv	19	150	L	true	<button>Update</button>
Saree	htdf	15	1100	L	true	<button>Update</button>
Lehenga	htdf	10	1050	XL	true	<button>Update</button>
Gown	htdf	15	1000	XL	true	<button>Update</button>
Anarkali	ergsds	10	1500	XL	true	<button>Update</button>
Anarkali	hjgf	10	1500	XL	true	<button>Update</button>

Figure 75: View Item(Women's Fashion)

The screenshot shows a mobile application interface with a blue header bar. On the right side of the header is the email address `pateldharmesh007.dp@gmail.com`. Below the header, the title "Boy's Fashion" is centered in bold black font. A table follows, with columns: Item Name, Description, Quantity, Price, Size, Item_Status, and Manage Item. Each row contains an "Update" button in a blue box. The data in the table is as follows:

Item Name	Description	Quantity	Price	Size	Item_Status	Manage Item
Ghost	esfdv	10	800	S	true	<button>Update</button>
Santa	esfdv	10	800	S	true	<button>Update</button>
Skull	esfdv	10	800	S	true	<button>Update</button>
Hulk	esfdv	10	800	S	true	<button>Update</button>
Frog	esfdv	12	950	S	true	<button>Update</button>
Crocodile	esfdv	12	750	S	true	<button>Update</button>
Police	esfdv	12	800	S	true	<button>Update</button>
Batman	wdfsdv	12	850	S	true	<button>Update</button>
Fireman	wdfsdv	10	750	L	true	<button>Update</button>
Army	wdfsdv	20	700	L	true	<button>Update</button>

Figure 76: View Item(Boy's Fashion)

The screenshot shows a mobile application interface with a blue header bar. On the right side of the header is the email address `pateldharmesh007.dp@gmail.com`. Below the header, the title "Girl's Fashion" is centered in bold black font. A table follows, with columns: Item Name, Description, Quantity, Price, Size, Item Status, and Manage Item. Each row contains an "Update" button in a blue box. The data in the table is as follows:

Item Name	Description	Quantity	Price	Size	Item Status	Manage Item
Supergirl	esfdv	10	900	M	true	<button>Update</button>
Reindeer	esfdv	10	850	M	true	<button>Update</button>
Peacock	esfdv	10	850	M	true	<button>Update</button>
Monkey	esfdv	10	650	M	true	<button>Update</button>
Pooh	esfdv	10	850	XXL	true	<button>Update</button>
Juliet	esfdv	10	750	XXL	true	<button>Update</button>
Cat	esfdv	10	750	XXL	true	<button>Update</button>
Barbie	esfdv	10	750	S	true	<button>Update</button>
Angel	esfdv	10	800	S	true	<button>Update</button>
Armygirl	esfdv	10	800	S	true	<button>Update</button>

Figure 77: View Item(Girl's Fashion)

Update Item

Item Name	Quantity
Lablepin	25
Description	Price
hjgf	150
Subcategory	Size
Men-Accessories	M
<input type="checkbox"/> Deals Of The Day	
<input type="checkbox"/> Recommended	
Item Image	
<input type="button" value="Choose File"/> No file chosen	
Optional Image	
<input type="button" value="Choose File"/> No file chosen	
<input type="button" value="Update Item"/>	
<input type="button" value="Inactive Item"/>	

Figure 78: Update Item

Add Deliveryman

Email Id	Password
1	
Name	Mobile Number
Suresh	9998295252
Address	Id Proof
b-2...	<input type="button" value="Choose File"/> No file chosen
<input type="button" value="Create Account"/>	

Figure 79: Add Deliveryman

Name	Address	Email	Mobile_Number	Id Proof
Dharmesh	hrgfds	dharmesh@gmail.com	8140448041	
Bhargav	hjgvhjb	bhargavborse4898@yahoo.com	9898525252	

Figure 80: Deliveryman Details

Order Status	Item Name	Item Price	More Info
New Order	Gown	1000	
New Order	Earring	150	
New Order	Saree	1100	
New Order	Earring	150	
New Order	Saree	1100	
New Order	Gown	4000	
New Order	Saree	1100	
New Order	Gown	4000	
New Order	Saree	1100	
New Order	Gown	4000	
New Order	Gown	4000	
New Order	Earring	150	
New Order	Gown	1000	

Figure 81: Current Orders

Order Details

Email Id	Delivery Date
bhargavborse4898@gmail.com	2019-04-29
Item Name	Return Date
Saree	2019-04-30
Customer Name	Order Status
Bhargav	New Order
Mobile Number	Order Type
9974061115	Rent
Address	Quantity
E-23/ Ashish Park, Tarsali, Vadodara	1
Pincode	Deliveryman
390009	Bhargav
Price	
1100	
submit	

Figure 82: Order Details

Previous History

Email	Item Name	Item Price	Deliveryman Email	More Info
bhargavborse4898@gmail.com	Saree	1100	dharmesh@gmail.com	
bhargavborse4898@gmail.com	Saree	1100	bhargavborse4898@yahoo.com	

Figure 83: Previous History

Previous History Details

Email Id	Delivery Date
bhargavborse4898@gmail.com	2019-04-25
Item Name	Return Date
Saree	2019-04-27
Customer Name	Order Status
Bhargav	New Order
Mobile Number	Order Type
9974061115	Rent
Address	Quantity
E-23/ Ashish Park, Tarsali, Vadodara	1
Pincode	Deliveryman
390009	dharmesh@gmail.com
Price	
1100	

Figure 84: Previous History Order Details

Feedback

Name	Email	Messege For	Message	Date	Time
Bhargav	bhargavborse4898@gmail.com	features	ghagsa	28/4/2019	7:39 PM
Bhargav	bhargavborse4898@gmail.com	service	test	28/4/2019	7:38 PM

Figure 85: Feedback

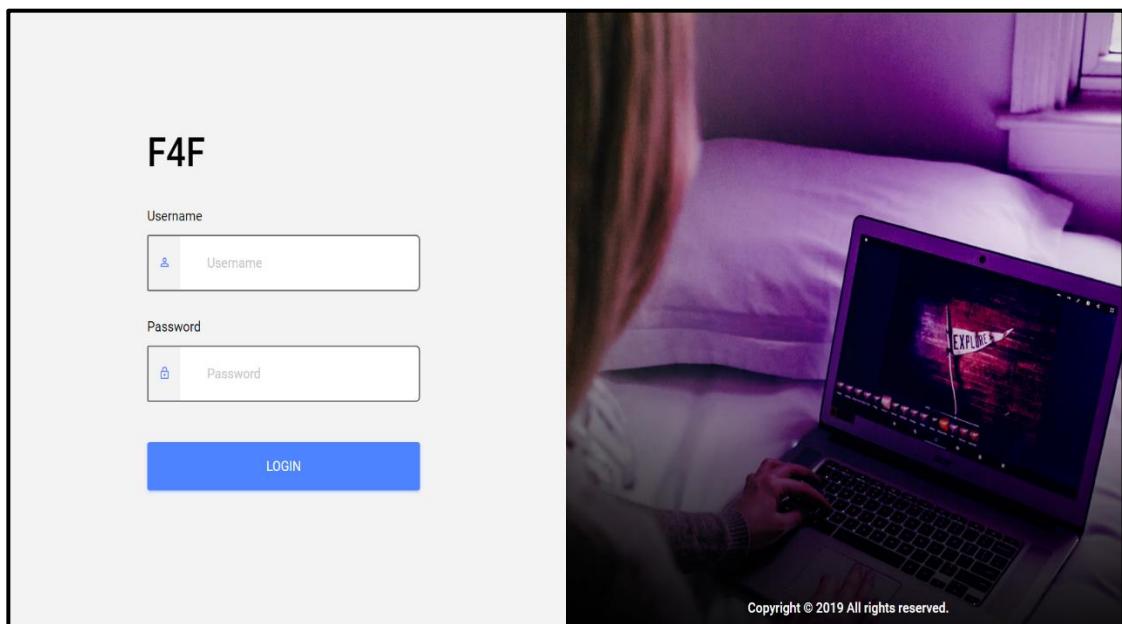


Figure 86: Deliveryman Login

A screenshot of a web-based delivery management system. The left sidebar shows navigation links for 'Dashboard' and 'Order Details' (with sub-options 'Delivery Order' and 'PickUp Order'). The main content area is titled 'NEW DELIVERY ORDER' and contains a table with five columns: Customer Name, Product Name, Delivery Address, Delivery Date, and More Details. The table entry is: Customer Name - Bhargav, Product Name - Saree, Delivery Address - E-23/ Ashish Park, Tarsali, Vadodara, Delivery Date - 2019-04-25, and More Details - a blue info icon. The footer of the page includes the copyright notice 'Copyright © 2019 Freedom for Fashion. All rights reserved.' and a welcome message 'Welcome, dharmesh@gmail.com'.

Figure 87: Delivery Order

Freedom For Fashion

The screenshot shows the 'PICKUP ORDER' section of the website. On the left, there is a sidebar with a 'Dashboard' link and a 'Order Details' dropdown menu containing 'Delivery Order' and 'PickUp Order' options. The main content area has a header 'Freedom for Fashion' and a welcome message 'Welcome, dharmesh@gmail.com'. Below this is a section titled 'PICKUP ORDER' with a sub-section 'No Order'. A table with columns 'Customer Name', 'Product Name', 'Delivery Address', 'PickUp Date', and 'More Details' is shown, but it is currently empty. At the bottom of the page, there is a copyright notice: 'Copyright © 2019 Freedom for Fashion. All rights reserved.'

Figure 88: Pickup Order

The screenshot shows the 'ORDER DETAILS' form. The sidebar remains the same as in Figure 88. The main content area has a header 'Freedom for Fashion' and a welcome message 'Welcome, dharmesh@gmail.com'. The 'ORDER DETAILS' form contains the following fields with sample data:

- Customer Name: Bhargav
- Phone Number: 9974061115
- Address: E-23/ Ashish Park, Tarsali, Vadodara
- Pin Code: 390009
- Email: bhargavborse4898@gmail.com
- Item Name: Saree
- Delivery Date: 2019-04-25
- Quantity: 1
- Price: 1100
- Order Type: Rent
- Order Status: New Order

At the bottom of the form are two buttons: 'Submit' and 'Cancel'. At the very bottom of the page, there is a copyright notice: 'Copyright © 2019 Freedom for Fashion. All rights reserved.'

Figure 89: Delivery/Pickup Order Details

8. LIMITATIONS AND FUTURE ENHANCEMENTS

Though our web-application is ready with basic requirements, there are certain limitations. Our system is only one city based. It also don't have wish-list feature, filter by category feature. We have only cash on delivery system.

For future enhancement, we have to make the web-application more user friendly and also make it multi city/ state based application. Add to wish-list and filter feature should also be added for the users. In our current system we are showing joining date of customer on that basis in future we will give loyalty points for discount. We will add online payment system.

CONCLUSION

This internship provided an experience of learning and skill development. It helped us to extend our sense of understanding and knowledge. It had us open to new languages and functions, made us learn its different features and made us understand the process of developing a product from scratch to end.

REFRENCES

1. bradford. (2009, September 20). functional. Retrieved from software testing:
<http://softwaretestingfundamentals.com/functional-testing/>
2. Elliot. (1970). SDLC. Retrieved from Tutorials Point:
https://www.tutorialspoint.com/sdlc/sdlc_overview.htm
3. H.W.L. (1996, December 17). test plannig activities. Retrieved from tutorials point:
https://www.tutorialspoint.com/software_testing_dictionary/test_plan.htm
4. Roy, R. (2004, 02 18). Feasibility Test. Retrieved from Tutorials Point:
https://www.tutorialspoint.com/software_engineering/software_requirements.htm
5. Security Features. (2002, January 16). Retrieved from Firebase:
<https://firebase.google.com/docs/database/security>
6. Steave, T. (2007, 5 8). Coding standars. Retrieved from Codeahay:
<https://codeahoy.com/2016/05/22/effective-coding-standards/>
7. Strashan. (2004). Spiral. Retrieved from Tutorials Point:
https://www.tutorialspoint.com/sdlc/sdlc_spiral_model.htm
8. test methods. (1970). Retrieved from tutorial point:
https://www.tutorialspoint.com/software_testing/software_testing_methods.htm
9. test strategy. (1993). Retrieved from ecomupternotes:
<http://ecomupternotes.com/software-engineering/software-testing-strategies>
10. Testing. (1996, july 15). Retrieved from Tutorials Point:
https://www.tutorialspoint.com/software_testing/
11. ut. (2018, August 19). Retrieved from tutorials point:
https://www.tutorialspoint.com/software_testing_dictionary/usability_testing.htm
12. W3C. (1993). gui. Retrieved from tutorials point:
<https://www.tutorialspoint.com/graphical-user-interface-gui>