

Brand Media

Final Year Project

Session 2015-2019

A project submitted in partial fulfillment of the degree of

BS in Software Engineering



Department of Software Engineering

Faculty of Computer Science & Information Technology

The Superior College, Lahore

Spring 2019

Type (Nature of project)	[<input checked="" type="checkbox"/>] Development [<input type="checkbox"/>] Research [<input type="checkbox"/>] R&D			
Area of specialization	Web & Android			
FYP ID	FYP-BSSE-F18-014			
Project Group Members				
Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	BSEM-F15-110	Muhammad Imtiaz	iwwwtsn@gmail.com	
(ii)	BSEM-F15-077	M Faizan Anwar	fa3577233@gamil.com	
(iii)	BSEM-F15-071	Ammad Amjad	ammadamjad60@gmail.com	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I Muhammad Imtiaz S/D of Malik, group leader of FYP under registration no FYP-BSSE-F18-014 at Software Engineering Department, The Superior College, Lahore. I declare that my FYP report is checked by my supervisor.

Date: 22-08-2019 Name of Group Leader: Muhammad Imtiaz Signature: _____

Name of Supervisor: Ifraseab Afzal

Co-Supervisor: Umar Hameed

Designation: Senior Lecturer

Designation: Professor

Signature: _____

Signature: _____

HoD: Dr. Arfan Jaffar

Signature: _____

Project Report

Brand Media

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
	1.0		<Original Draft>	
			<Changes Based on Feedback from Supervisor>	
			<Changes Based on Feedback From Faculty>	
			<Added Project Plan>	
			<Changes Based on Feedback from Supervisor>	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

Name: _____

Date: _____ Signature: _____

PROJECT MANAGER

Comments: _____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

Date: _____ Signature: _____

Dedication

This work is dedicated to my teachers, and supervisors which help during my degree. I learn many topics related software engineering from my teachers. My supervisor helps in FYP. During FYP much of my topics cleared.

Acknowledgements

I am really thankful to my supervisor who has support in our FYP. We get many knowledge about software documentation from our supervision. I thanking my supervision. My supervisor helps us in developing the system with documentation. I thanking my parents which support us.

Executive Summary

We have Facebook for social media/community (A community where people interact with other peoples) and Amazon for e-commerce that is the largest Internet retailer in the world as measured by revenue and market capitalization and also LinkedIn that is a business and employment-oriented service. All of these platform have different functionalities, services and features. By inspiration all of these we are going to introduce a new online community/platform that is Brand Media. In Brand Media we give services (entertainment, ecommerce and professional networking). Basically we are going to combine (Amazon +Facebook + LinkedIn) in this project. The intention will to bridge the gap between Consumer and Brand and also to help Pakistani Entrepreneurs (start-ups) to promote their products and brands.

Many platforms like daraz.pk, olx.com, goto.pk. They provide shopping services in their website but not any Social services like Facebook. Actually people want to like, comment, share their favorite brand products and also to see the choice of other peoples like friends, family members and colleagues. In Brand Media community we have not allow to any whole sellers to sale products. Only Manufacturer /owner of the product can sale the products to consumer so people can facilitate with original quality and actual price of product. It will become World first brand community, where a consumer can fully engage with their favorite brands.

A community/platform of official brands where all brands can directly communicate with their consumers. A platform for consumers where they can meet with their needs and demands and get high quality products at one click. A community where quality is more important than money. To facilitate 193.2 million Pakistan's people with high quality product at low price. This platform will become very helpful for Pakistani Entrepreneurs and especially helpful for Superior Technopreneurship program. All students who will make innovative products in Technopreneurship event can be easily promote their product at global level and easily get customer attraction. This platform will be going very helpful to make Pakistan economically superior.

Table of Contents

Dedication	v
Acknowledgements	vi
Executive Summary	vii
Table of Contents	viii
List of Figures	xiii
List of Tables	xv
Chapter 1	1
Introduction	1
1.1. Background	2
1.2. Motivations and Challenges	2
1.3. Goals and Objectives	3
1.4. Literature Review/Existing Solutions	4
1.5. Gap Analysis	5
1.6. Proposed Solution	5
1.7. Project Plan	6
1.7.1. Work Breakdown Structure	6
1.7.2. Roles & Responsibility Matrix	8
1.7.3. Gantt Chart	8
1.8. Report Outline	11
Chapter 2	12
Software Requirement Specifications	12
2.1. Introduction	13
2.1.1. Purpose	13
2.1.2. Document Conventions	13
2.1.3. Intended Audience and Reading Suggestions	13
2.1.4. Product Scope	14
2.1.5. References	14
2.2. Overall Description	15
2.2.1. Product Perspective	15
2.2.2. Product Functions	16
2.2.3. User Classes and Characteristics	18
2.2.4. Operating Environment	19
2.2.5. Design and Implementation Constraints	19
2.2.6. User Documentation	19
2.2.7. Assumptions and Dependencies	19
2.3. External Interface Requirements	20
2.3.1. User Interfaces	20
2.3.2. Hardware Interfaces	20
2.3.3. Software Interfaces	21
2.3.4. Communications Interfaces	21
2.4. System Features	21
2.4.1. Login Account	21
2.4.1.1. Description and Priority	21

2.4.1.2	Stimulus/Response Sequences	21
2.4.1.3	Functional Requirements	22
2.4.2	Forget Password	22
2.4.2.1	Description and Priority	22
2.4.2.2	Stimulus/Response Sequences	22
2.4.2.3	Functional Requirements	23
2.4.3	Update Profile	23
2.4.3.1	Description and Priority	23
2.4.3.2	Stimulus/Response Sequences	23
2.4.3.3	Functional Requirements	24
2.4.4	Search Engine	24
2.4.4.1	Description and Priority	24
2.4.4.2	Stimulus/Response Sequences	24
2.4.4.3	Functional Requirements	25
2.4.5	Timeline	25
2.4.5.1	Description and Priority	25
2.4.5.2	Stimulus/Response Sequences	26
2.4.5.3	Functional Requirements	26
2.4.6	News feed	26
2.4.6.1	Description and Priority	26
2.4.6.2	Stimulus/Response Sequences	27
2.4.6.3	Functional Requirements	27
2.4.7	Follow BRAND	28
2.4.7.1	Description and Priority	28
2.4.7.2	Stimulus/Response Sequences <i>user login his account</i>	28
2.4.7.3	Functional Requirements	28
2.4.8	Unfollow user	29
2.4.8.1	Description and Priority	29
2.4.8.2	Stimulus/Response Sequences	29
2.4.8.3	Functional Requirements	29
2.4.9	View follower list	30
2.4.9.1	Description and Priority	30
2.4.9.2	Stimulus/Response Sequences	30
2.4.9.3	Functional Requirements	30
2.4.10	Notifications	31
2.4.10.1	Description and Priority	31
2.4.10.2	Stimulus/Response Sequences	31
2.4.10.3	Functional Requirements	31
2.4.11	Verified address using Map	32
2.4.11.1	Description and Priority	32
2.4.11.2	Stimulus/Response Sequences	32
2.4.11.3	Functional Requirements	32
2.4.12	Create customer account	33
2.4.12.1	Description and Priority	33
2.4.12.2	Stimulus/Response Sequences	33
2.4.12.3	Functional Requirements	33

2.4.13	Shopping Cart	34
2.4.13.1	Description and Priority	34
2.4.13.2	Stimulus/Response Sequences	35
2.4.13.3	Functional Requirements	35
2.4.14	Add to cart	35
2.4.14.1	Description and Priority	35
2.4.14.2	Stimulus/Response Sequences	36
2.4.14.3	Functional Requirements	36
2.4.15	Product rating	36
2.4.15.1	Description and Priority	36
2.4.15.2	Stimulus/Response Sequences	37
2.4.15.3	Functional Requirements	37
2.4.16	Place Order/Checkout	37
2.4.16.1	Description and Priority	37
2.4.16.2	Stimulus/Response Sequences	38
2.4.16.3	Functional Requirements	38
2.4.17	Register Brand	38
2.4.17.1	Description and Priority	38
2.4.17.2	Stimulus/Response Sequences	39
2.4.17.3	Functional Requirements	39
2.4.18	Manage Products	40
2.4.18.1	Description and Priority	40
2.4.18.2	Stimulus/Response Sequences	40
2.4.18.3	Functional Requirements	40
2.4.19	Manage orders	41
2.4.19.1	Description and Priority	41
2.4.19.2	Stimulus/Response Sequences	41
2.4.19.3	Functional Requirements	41
2.4.20	Deliver Products	42
2.4.20.1	Description and Priority	42
2.4.20.2	Stimulus/Response Sequences	42
2.4.20.3	Functional Requirements	43
2.4.20	Dashboard Login	43
2.4.21.1	Description and Priority	43
2.4.21.2	Stimulus/Response Sequences	43
2.4.21.3	Functional Requirements	44
2.4.22	Approve Brand	44
2.4.22.1	Description and Priority	44
2.4.22.2	Stimulus/Response Sequences	45
2.4.22.2.1	Functional Requirements	45
2.4.23	Manage categories	46
2.4.23.1	Description and Priority	46
2.4.23.2	Stimulus/Response Sequences	46
2.4.23.3	Functional Requirements	47
2.4.24	Manage accounts	47
2.4.24.1	Description and Priority	47

2.4.24.2	Stimulus/Response Sequences	47
2.4.24.3	Functional Requirements	47
2.4.25	Delete Business Page.....	48
2.4.25.1	Description and Priority.....	48
2.4.25.2	Stimulus/Response Sequences	48
2.4.25.3	Functional Requirements	48
2.4.26	Manage Reports.....	49
2.4.26.1	Description and Priority.....	49
2.4.26.2	Stimulus/Response Sequences	49
2.4.26.3	Functional Requirements	49
2.4.27	Reset Passwords	50
2.4.27.1	Description and Priority.....	50
2.4.27.2	Stimulus/Response Sequences	50
2.4.27.3	Functional Requirements	50
2.5	Other Nonfunctional Requirements	50
2.5.1	Performance Requirements.....	50
2.5.2	Safety Requirements.....	53
2.5.3	Security Requirements.....	53
2.5.4	Software Quality Attributes.....	53
2.5.5	Business Rules.....	56
2.6	Other Requirements.....	56
Chapter 3	57
Use Case Analysis	57
3.1.	Use Case Model	59
3.2.	Use Case Descriptions.....	60
Chapter 4	91
System Design	91
4.1.	Architecture Diagram.....	93
4.2.	Domain Model.....	94
4.3.	Entity Relationship Diagram with data dictionary	95
4.4.	Class Diagram	104
4.5.	Sequence / Collaboration Diagram	106
4.6.	Operation contracts	123
4.7.	Activity Diagram.....	147
4.8.	State Transition Diagram	173
4.9.	Component Diagram	174
4.10.	Deployment Diagram	175
Chapter 5	176
Implementation	176
5.1.	Important Flow Control/Pseudo codes.....	177
5.2.	Components, Libraries, Web Services and stubs	182
5.3.	Deployment Environment	182
5.4.	Tools and Techniques.....	188
5.5.	Best Practices / Coding Standards.....	189
5.6.	Version Control	190
Chapter 6	191

Testing and Evaluation	191
6.1. Use Case Testing.....	192
6.2. Equivalence partitioning	202
6.3. Boundary value analysis.....	204
6.4. Data flow testing	207
6.5. Unit testing	208
6.6. Integration testing.....	215
6.7. Performance testing.....	220
6.8. Stress Testing	224
Chapter 7.....	226
Summary, Conclusion and Future Enhancements	226
7.1. Project Summary	227
7.2. Achievements and Improvements	228
7.3. Critical Review.....	228
7.4. Lessons Learnt.....	229
7.5. Future Enhancements/Recommendations	229
Appendices.....	231
Appendix A: User Manual	232
Appendix B: Administrator Manual	235
Appendix C: Information / Promotional Material	236
Reference and Bibliography	242

List of Figures

Figure 1 Work Break Down Structure	7
Figure 1.1 Gant Chart	11
Figure 3.1 UseCase	59
Figure 4.1 Architecture Diagram	93
Figure 4.2 Domain Model.....	94
Figure 4.3 ERD Diagram	95
Figure 4.4 Class Diagram	105
Figure 4.5 Register Brand Account	106
Figure 4.6 Approve Brand Request	107
Figure 4.7 Add Product.....	108
Figure 4.8 Brand User Profile.....	109
Figure 4.9 Customer Registration.....	110
Figure 4.10 Follow User	111
Figure 4.11 Unfollow User	111
Figure 4.12 Block User	112
Figure 4.13 Block User	112
Figure 4.14 Login.....	113
Figure 4.15 View New Order.....	114
Figure 4.16 View Delivered Order	114
Figure 4.17 Deliver Order.....	115
Figure 4.18 Remove Orders.....	116
Figure 4.19 Notification.....	116
Figure 4.20 Process Shopping Cart.....	117
Figure 4.21 Rating Product.....	118
Figure 4.22 Rating Product.....	118
Figure 4.23 Search	119
Figure 4.24 Give Order	119
Figure 4.26 Update Profile.....	120
Figure 4.27 Remove Product	121
Figure 4.29 Customer Profile.....	122
Figure 4.30 Update Profile.....	122
Figure 4.31 Login Account.....	148
Figure 4.32 Forget Password	149
Figure 4.33 Update Profile.....	150
Figure 4.34 Search Engine.....	151
Figure 4.35 timeline	152
Figure 4.36 NewsFeed	153
Figure 4.37 Follow.....	154
Figure 4.38 Notification.....	155
Figure 4.39 Verified Address Using Map.....	156
Figure 4.41 Create Customer Account.....	157
Figure 4.42 Shopping Cart.....	158
Figure 4.44 Product Rating.....	159

Figure 4.46 Create Brand Account	160
Figure 4.47 Manage Products	161
Figure 4.48 Manage Orders	162
Figure 4.49 Dashboard Login	164
Figure 4.49 (a) Dashboard Login.....	165
Figure 4.50 Manage Categories	166
Figure 4.51 Manage Accounts	167
Figure 4.52 Delete Business Page.....	168
Figure 4.53 Manage Reports.....	169
Figure 4.53 (b) Reset Password	170
Figure 4.54 State Transition Diagram.....	172
Figure 4.55 Component Diagram	173
Figure 4.55 Deployment Diagram	174
Figure 5.1 Login.....	182
Figure 5.2 Category Diagram	182
Figure 5.3 Admin Dashboard.....	183
Figure 5.4 Add New Product	183
Figure 5.5 New Product Added	184
Figure 5.6 NewsFeeds.....	185
Figure 5.7 Profile	186
Figure 5.8 Search Result	186
Figure 5.9 Registration.....	187
Figure 6.1 Login Account Testing	209
Figure 6.2 Navbar Testing	209
Figure 6.3 Categories Testing.....	210
Figure 6.4 Follow Testing.....	211
Figure 6.5 Upload Product Testing.....	212
Figure 6.6 Cart Testing	213
Figure 6.7 Integration Testing.....	214

List of Tables

Table 1.1 Gant Chart.....	10
Table 4.1 Customer Table.....	96
Table 4.2 Brand Table.....	97
Table 4.3 Product Table.....	98
Table 4.4 Order Table.....	99
Table 4.5 cart Table.....	100
Table 4.6 Likes Table.....	100
Table 4.7 Share Table.....	101
Table 4.8 Comment table.....	102
Table 4.9 Friend Table.....	103
Table 4.10 CNotification Table.....	103
Table 6.1 Use Case Testing.....	201
Table 6.2 Equivalence Partitioning.....	203
Table 6.3 Boundary Value Analysis.....	205
Table 6.4 Data Flow Testing.....	207
Figure 6.1 Login Account Testing.....	209
Figure 6.2 Navbar Testing.....	209
Figure 6.3 Categories Testing.....	210
Figure 6.4 Follow Testing.....	211
Figure 6.5 Upload Product Testing.....	212
Figure 6.6 Cart Testing.....	213
Figure 6.7 Integration Testing.....	214

Chapter 1

Introduction

Chapter 1: Introduction

This chapter describes the introduction of system. This describes the complete motivation and challenges that system face. This chapter describes the complete goals and objectives of system. Also this describes the existing system and what are the gap that people face with existing systems.

1.1. Background

Many platforms like daraz.pk, olx.com, goto.pk. they provide shopping services in their website but not any Social services like Facebook. People want to like, comment, share their favorite brand products and also to see the choice of other peoples like friends, family members and colleagues.

Big problem is that Supply chain from manufacturer to consumer. In this supply chain there many people and whole sellers involve.

For example

1. Manufacturer of product sales product to their customer in 600 Rs
2. Customer sales the same product to local whole seller in 900 Rs
3. Local whole seller sale that product at shops in 1200 Rs
4. And then consumer buy the same product in 1500 Rs from shops

In Brand Media community we have not allow to any whole sellers to sale products. Only Manufacturer /owner of the product can sale the products to consumer, so people can facilitate with original quality and actual price of product.

1.2. Motivations and Challenges

Motivations

By using the “Brand Media” people will be motivated. In this platform users easily see the branded products, easily buy the products, share the products with friends, comments and like the product, easily see the new and upcoming products in home page. In this platform user easily create their profile with no difficulty. This platform will be usable for users. In this platform brands introduce a jobs for buyers. For example if brand will be wanted to hire the employee(s) in his company then it will issue the job(s) with full description. User see this opportunity in his main window and also user apply for the job online by sending CV. This opportunity will also highly motivate the brand media’s users. User will see the other’s profile. In this profile user easily see the products which will be purchased by other user. This feature will increase the motivations of every user in Brand Media. And this feature will motivate the users.

Challenges

The big challenge is that How let we create the positions in facebook.com and amazon.com users. Because their users are strongly linked with facebook.com and amazon.com.

We should that we highly market the “Brand Media”. So that every user know the brand media. And be interact with Brand Media. And also big challenge is that this platform must be developed quickly. And this platform require a much of time to be built. But we will achieve it in time frame.

1.3. Goals and Objectives

Goals

We will introduce a new social media network. We will provide the facility to buy products with some social media features. We will provide the platform in which our users see others likeness about products. We will provide the facility in which manufacturers of the products will deliver the products with the facility of map. We will provide the search engine in which user search the products, another users, and brand pages. We provide a platform to our user in which users will be updated with new information about friends, products, brands on the daily basis.

Objectives

After achieving our goals users will be use our platform (Brand Media). Users can buy the products. Users share the products of brands to others. The objective of our platform is that we create fun in the user's life. Brands can check his product's customers. User buy the products in his/her home with not difficulty. Users easily can see the product of brands with full information. Our users will prefer Brand Media to others. Users can make the friends in Brand Media. When our users will be 5000 then I will earn money from Brand Media.

1.4. Literature Review/Existing Solutions

First of all, there is no this type of community/platform available for people which we are going to provide in Brand Media. We have system/platform (Facebook, Amazon, and LinkedIn) that available in market. All of these system have their own unique features. But there is no system available in market that provide all of these feature at one place.

Amazon

One of the largest online shopping websites in the world. The site is sell electronics, fashion, furniture, and apparel. Users can purchase and also can sell items using Amazon's online marketplace system.

Facebook

Facebook is a social networking website where users can post comments, share photographs and post links to news or other interesting content on the web, chat live, and watch short-form video. Shared content can be made publicly accessible, or it can be shared only among a select group of friends or family, or with a single person.

1.5. Gap Analysis

In the world, there is no any system that provide the social media functionalities to online sellers and buyers. Customers want that they get entertainment by using ecommerce system. Customer cannot view the other customers. Customer wants that they view other customer's choices about products. Famous website amazon.com that have not social media feature in which customer view the other customer's choices. This website just provide the ecommerce features to users. In addition, amazon.com does not provide any news feeds of products to customers. Customer feel problem to buy products.

Facebook.com is a very famous social media system. This system provides the social media features. By using facebook.com, user always connected with another. But users can't buy branded product from facebook.com. In this website many of the users have created their own pages for sell the product. Those users sell the branded products in their own rates. As everyone knows that brands sell the product to other in maximum price 200Rs. And other person sells that product in 2000Rs. This is the big gap that customer buy the branded product in much price.

Both have the good functionalities but people use both and spent much time. And no any system is developed that provide the ecommerce with social media features.

1.6. Proposed Solution

I want to give solution of these problems by making new community/platform that is Brand Media where people can buy product directly from manufacturer of that product. By using this technique there is no confusing about quality and pricing of products because people are buying products directly from manufacturer/ Owner. They can directly meet with their favorites brands

- First of all, our platform provides trusted relationship with our users.

- Consumer can buy all their favorite brands product at actual price from Brands
- We have no whole seller
- People can buy original quality product directly from brands
- They can see upcoming stories of product at different events Like Eids, Winter, Summer collections. Or in wedding seasons
- A community for start-up brands, promote their brands (customer attraction)
- Deliver product with same day delivery with new delivery methodology
- Provide entertainment to people who join our community with different games and methodology so people can buy product by enjoying themselves
- (Entertainment) There is no this type of functionality available at currently ecommerce sites even in amazon

In foreign countries large numbers of people prefer to buy product online. that's why amazon an American e-commerce platform ranked in top of the world companies.

1.7. Project Plan

1.7.1. Work Breakdown Structure

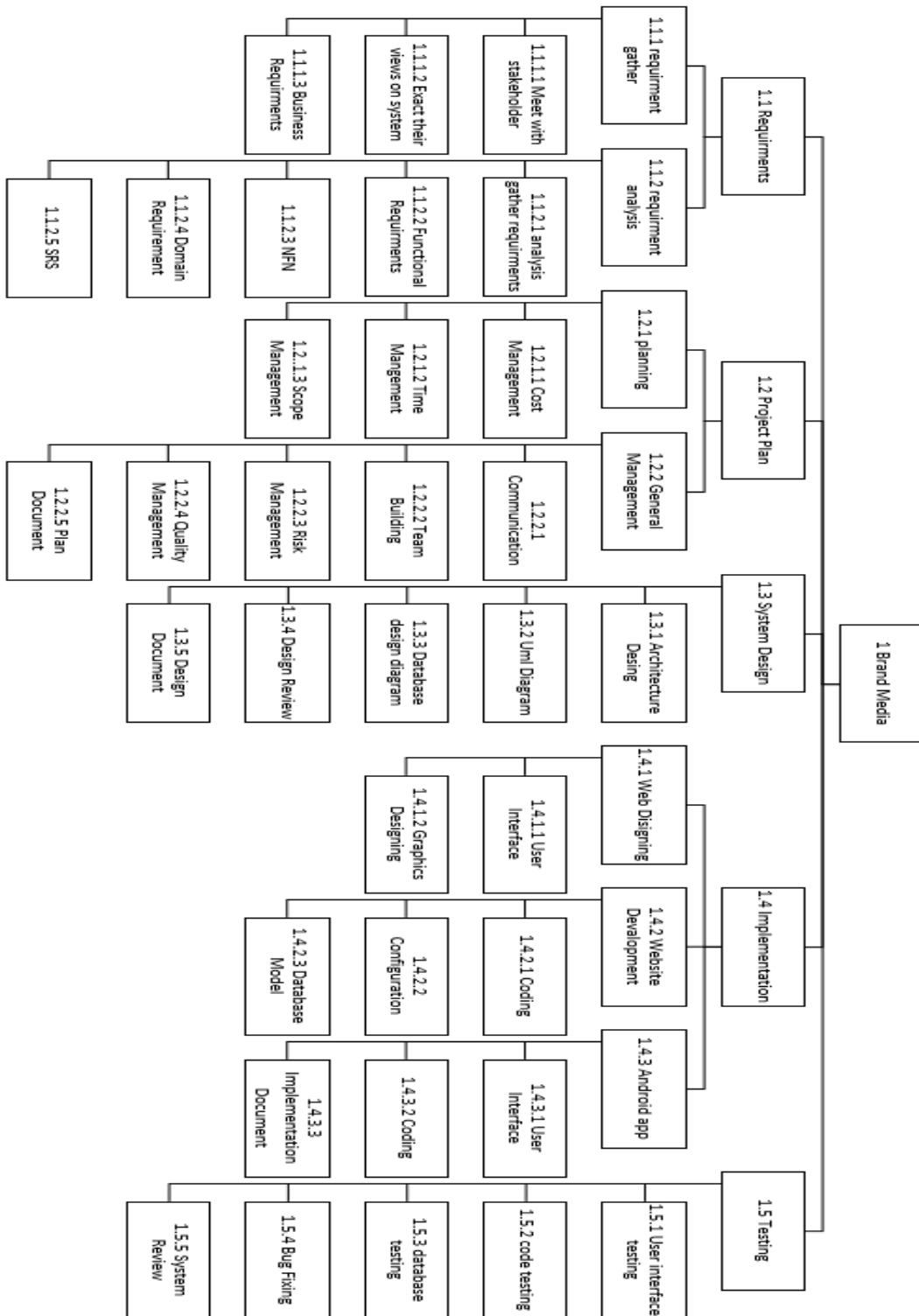


Figure 1.1 Work Break Down Structure

1.7.2. Roles & Responsibility Matrix

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Requirement	1.1	1.1.2.5	16 Days	M Faizan, M.Imtiaz,Ammad
2	Planning	1.2	1.2.2.5	13 Days	M Faizan, M.Imtiaz,Ammad
3	System Design	1.3	1.3.5	15 Days	M Faizan, M.Imtiaz,Ammad
4	Implementation	1.4	1.4.3.3	90 Days	M Faizan, M.Imtiaz,Ammad
5	Testing	1.5	1.5.5	24 Days	M Faizan, M.Imtiaz,Ammad

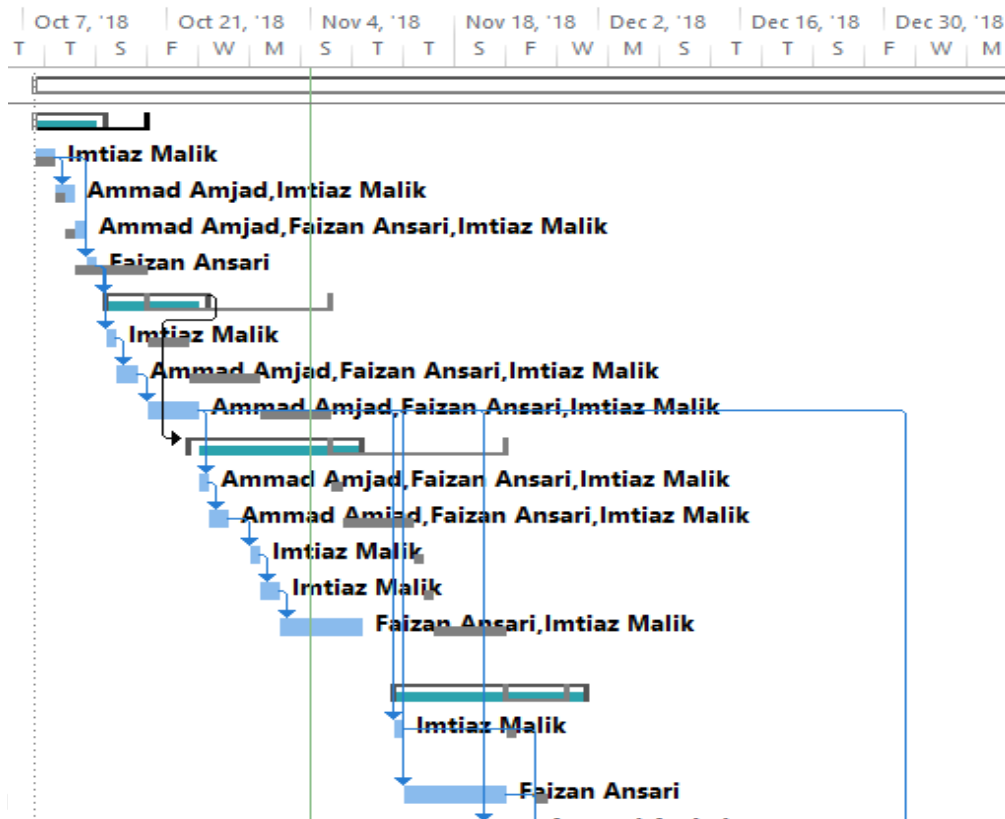
1.7.3. Gantt Chart

Task Name	Duration	Start	Finish	Predecessors	Resource Names
Brand Media	148 days	Mon 10/8/18	Wed 4/24/19		Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	7 days	Mon 10/8/18	Sun 10/14/18		Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	2 days	Mon 10/8/18	Tue 10/9/18		Imtiaz Malik
Brand Media	2 days	Wed 10/10/18	Thu 10/11/18	3	Ammad Amjad,Imtiaz Malik
Brand Media	1 day	Fri 10/12/18	Fri 10/12/18		Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	1 day	Sat 10/13/18	Sat 10/13/18	3	Faizan Ansari

Brand Media	9 days	Mon 10/15/18	Wed 10/24/18	6	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	1 day	Mon 10/15/18	Mon 10/15/18	6	Imtiaz Malik
Brand Media	2 days	Tue 10/16/18	Wed 10/17/18	8	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	4 days	Fri 10/19/18	Tue 10/23/18	9	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	13 days	Tue 10/23/18	Thu 11/8/18	7	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	1 day	Wed 10/24/18	Wed 10/24/18	10	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	2 days	Thu 10/25/18	Fri 10/26/18	12	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Brand Media	1 day	Mon 10/29/18	Mon 10/29/18	13	Imtiaz Malik
Risk management	2 days	Tue 10/30/18	Wed 10/31/18	14	Imtiaz Malik
Quality management	6 days	Thu 11/1/18	Thu 11/8/18	15	Faizan Ansari,Imtiaz Malik
System design	15 days	Mon 11/12/18	Fri 11/30/18		Ammad Amjad,Faizan Ansari,Imtiaz Malik
Architectural design	1 day	Mon 11/12/18	Mon 11/12/18	10	Imtiaz Malik
UML diagram	8 days?	Tue 11/13/18	Thu 11/22/18	10	Faizan Ansari
Database diagram	3 days	Wed 11/21/18	Sun 11/25/18	10	Ammad Amjad
design review \$ document	5 days	Mon 11/26/18	Fri 11/30/18	18,19,20	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Implementation	34 days	Tue 1/1/19	Thu 2/14/19		Ammad Amjad,Faizan Ansari,Imtiaz Malik
web designing	34 days	Tue 1/1/19	Thu 2/14/19	21	Ammad Amjad,Imtiaz Malik,Faizan Ansari
Web user interface	10 days	Thu 1/3/19	Tue 1/15/19	21	Ammad Amjad,Imtiaz Malik
Graphical Designing	5 days	Tue 1/1/19	Sat 1/5/19	10	Ammad Amjad,Imtiaz Malik
Database Model	8 days	Wed 1/16/19	Fri 1/25/19	21	Ammad Amjad,Faizan Ansari,Imtiaz Malik

Backend Coding	14 days	Mon 1/28/19	Thu 2/14/19	21,24	Imtiaz Malik,Faizan Ansari
Android App	32 days	Fri 2/15/19	Sat 3/30/19		Faizan Ansari
User interface	1 wk	Fri 2/15/19	Thu 2/21/19	21	Ammad Amjad,Faizan Ansari
Database model	1 wk	Fri 2/22/19	Thu 2/28/19		Faizan Ansari,Imtiaz Malik
Back End Coding	22 days	Fri 3/1/19	Sun 3/31/19	21	Faizan Ansari,Imtiaz Malik
Testing	18 days	Mon 4/1/19	Wed 4/24/19		Ammad Amjad,Faizan Ansari,Imtiaz Malik
User interface testing	3 days	Mon 4/1/19	Wed 4/3/19	29,24	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Database testing	5 days	Thu 4/4/19	Wed 4/10/19	30,26	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Coding testing	7 days	Tue 4/9/19	Wed 4/17/19	27,31,21	Ammad Amjad,Faizan Ansari,Imtiaz Malik
Bug Fixing	1 wk	Thu 4/18/19	Wed 4/24/19	33,34,35	Ammad Amjad,Faizan Ansari,Imtiaz Malik

Table 1.1 Gant Chart



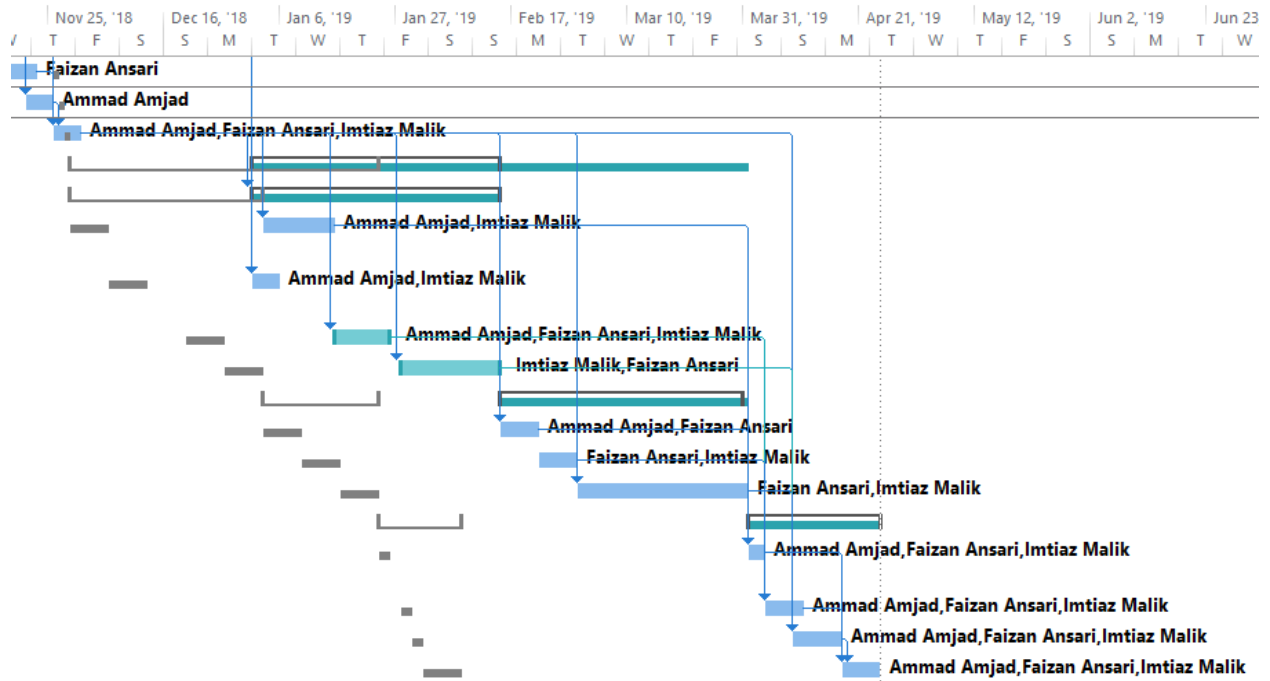


Figure 1.2 Gant Chart

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

Brand Media is one platform which we will provide services of buying and selling of branded products with some social media functions. Brand Media provide the functionality to Brands and customers to make their accounts. Brand Media provide the functionality to brands to add products so that customer buy products. This system provide the facility to customers to follow brands. This system provide the opportunity to customers to like, share and comments on product.

2.1.2. Document Conventions

This document uses the following the requirements:

DB	Database
MYSQL	My Structured Query Language
API	Application Programming Interface

Font size is 12 and font type is Calibri.

2.1.3. Intended Audience and Reading Suggestions

This document is intended for project manager, designer, developer, and tester. This has been implemented under the guidance of senior professor "Sir Ifraseab Afzal". First this document contain the complete overview of the product which describe the Brand Media. Then this document contains the summary of major features of the Brand Media. This feature's summary describe the way that how users will use this product. Then this document contains the how many categories of user for use this product and whose are the features that are used by particular user's category. Then this document describe the operating systems in which this

System will be run. Then this document contains the constraints to developers to develop the system. Then this document contain the external interfaces requirements. In these requirements, user, hardware, software, and communication requirements will be provided. After external interface requirements, all the complete feature's description will be mentioned with complete feature's priority. And then this document contain the complete description of Brand Media's nonfunctional requirements.

2.1.4. Product Scope

Brand Media provide the e-commerce system with some social media features to customer to buy the products of different brands. This system is available on internet in the form of website and also available on android OS. Brand Media provide the feature in both website and android application to order products to brands for buy products. All the features are available in both website and android app. User will give the order to brands with address, this address will be in the form of location of user that show on map so that brand user deliver product through customer's location (manually). This system provide the opportunities and some social media functionalities like, share, and comment on the product, see notifications, view shopping cart, give order for buy the product, follow user, view own profile, view other's profile, view business page, for brands can add, update, or remove product and manage orders. The notifications can be different from both side users (Brand & Customer). Both users have different profiles. The profile contain all the user-related data. Users first create their accounts and then access the feature that provided by brand media. The objective of this system is that users buy and sell products of brands in less price with some social media features. When 1000 users use Brand media daily, then we add ad's in Brand Media. We earn money from ads. These ads will not decrease the zest of users.

2.1.5. References

Website UI design Guide:

[1] <https://www.smashingmagazine.com/2008/01/10-principles-of-effective-web-design/>

[2] <https://visme.co/blog/website-color-schemes/>

Android App UI design Guide:

[3] <https://www.ui.guide/>

Standards:

[4] <https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database>

2.2. Overall Description

2.2.1. Product Perspective

Brand Media is exist on tow systems one is website and other is in the form of android application. Brand Media provide the feature in both website and android application to order products to brands for buy products. At the delivery time, android application show the location of customer on map because the GPS is perfectly work on mobile devices. Brand Media provide the ecommerce functionality as add, remove, update products, buy and sell products, product rrating, shopping cart with some features of Facebook like share, comments, like products, send friend request to other user(s), accept friend request, new feeds, notifications, user profile, follow Brand Business Page (contains the Company's Page with his products). All the data will store on MYSQL which perform as the central database for both android app and website. Brand Media will replace Facebook and amazon in future. Brand Media consist on two domains (Website and Android). Website will be uploaded on internet for which we need a server and its domain. By the help of domain we will take the Brand Media features in android app through APIs (Application Programming Interfaces). In android app all the data will be fetch from server's database, insert into server database, update, and remove from server database with the help of APIs.

2.2.2. Product Functions

Brand Media provide the feature to brands to create their business page so that they can add products to sell to customers. Customer order the product online through the brand media. Then brand media show the notification to brand user. Brand Media provide the feature in the form of map. This map show the location of customer. The feature of rating the product will be performed by customer provided from brand user through brand media. Brand media provide the feature to users to upload CV to get the jobs from Brands.

Provided features are:

Customer/brand – Primary Actor

- Login account
By using this feature customer and brand will login their accounts and get the features to perform that provide by brand media.
- Forget password
By using this feature, users can change their passwords.
- Update Profile
Users can change his/her profile information.
- Search Engine
This feature helps to users to search for other user profile. But condition is that particular user should have profile.
- Timeline
By using this feature users view his/her timeline. This timeline will show products.
- Newsfeed(Like, Share, Comments)
By using this features users will view the products of brand and can like, comment, or share the product.
- Follow user
By using this feature users can follow other users. If customer follow the brand then customer get the newsfeeds of that brand.
- Unfollow user
By using this feature users can unfollow the other user.
- View follower List
This feature show the followers to users.
- Notifications
By using this feature users get the notifications of daily changes.
- Verified Address Using Map
This feature provide the map facility to set the location on giving order time.

Customer

- Create Customer Account
By using this feature customer will create their accounts and get all the brand media's features.
- Shopping Cart
By using this feature, customer will manage added products in shopping cart and can order the products.
- Add to Cart
By using this feature, customer will add products in cart to buy.
- Product Rating
This feature helps to Customer to rate the products.
- Place Order/Checkout
By using this feature customer will order the products to brands.

Brand

- Register Brand
By using this feature brands will create their accounts and get all the brand media's features.
- Manage Products
By using this feature brand can add, delete, or update their products.
- Manage Orders
By using this feature brand can add, delete, or update orders that coming from customers.
- Deliver Products
This feature provide all the customer's order details and customer's location. So that brand can deliver the product.

Admin – Secondary Actor

- Dashboard Login
- Approve Brand
When the brand register his page the first this request come to admin and will approve the request. After approve the request brand can get all the features provided by brand media.
- Manage Categories
By using this feature admin will add the category and also add the sub category of that category.
- Manage Accounts
By using this feature, admin can view/delete the accounts of users.
- Delete Business Page
By using this feature, admin can delete the business pages.
- Manage Reports

By using this feature admin will view the reports coming from brands and customers.

- Reset Passwords
Admin can reset password.

2.2.3. User Classes and Characteristics

Brand Media exist for three types of user first is Brand User, second is Customer User, and third is administrator.

Brand User:

Brand user will create his/her business page. In this page Brand will add products to sell. Also the brand user can update and delete the products from his page. Brand user see the feedback appear from customer for his sold products. Brand User use the map to see the location of customer user for delivery the product. Brand user can also change his Business page information.

Customer User:

Customer will create his/her account to perform his/her functions. Customer view the news feeds. In this news feed they view the products of brands, like the product, comment on product, and share the products to friends. Customer will see to brand's product which will be followed by customer. Customer see the opportunity of job provided from Brand Business page. Customer send the CV to brands users. Customer add products in shopping cart. This shopping cart show the list of products which will be added by user. Users will send request to other users to make friend by follow them. Other Customer will either accept the request or deny the request. Customer rating the product after buy the product. This rating will be provided from Brand User. User search the Brand Business Pages and other Customers.

Administrator:

Administrator login the account. Admin will see the request from Brand Users. Admin will approve the Brand Users. Admin can delete the Customer Account, and Business Page. Admin can view the Customer Profile and Business Page. Customer will view the reports comes from users.

2.2.4. Operating Environment

Brand Media perform in two different ways. First this system will be used by users in the form of website. This website will be perfectly run on google chrome. Second this system will be used by users in the form of android application. Android application will be run on android devices. For android application, the version of android OS must be 5.0. And for website, the latest version of google chrome must be latest 2018. For windows OS, the version should be windows 10 or higher. For Mac OS, the version should be mac OS 13.0 or higher.

2.2.5. Design and Implementation Constraints

The Internet connection is a constraint for the system. Since the application fetches data from the MYSQL database over the Internet, it is critical that there is an Internet connection for the system to function. Both the website and the android application will be constrained by the capacity of the database. Since the database is shared between both application it may be forced to queue incoming requests and therefor increase the time it takes to fetch data.

2.2.6. User Documentation

Brand Media will provide the videos for using the Brand Media. These videos will help the users “How we can use the Brand Media?” If something will not perform perfectly then user send the report to admin. This is very helpful to users for solving the problems.

2.2.7. Assumptions and Dependencies

Data fetching components like view news feeds, notification can be affect in the development in the both website and android application. There is good algorithms will be required for fetching the data from MYSQL. If customer sent the friend request to other then other user will see the notification in the website. If other user run the android application then

he/ she will see the notification of friend request after few seconds depending on the algorithms. Same fetching data algorithms for both android application and website are used, is very difficult. We used the map in brand media. This map can only be get from google API. If this API is crashed then the map in brand media will be crashed.

Intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

2.3. External Interface Requirements

2.3.1. User Interfaces

We use the natural (familiar) elements in the UIs. These UIs are responsive that response to user at every time. If any error will be occurred then system's UIs tell the users about error. We use the standard button, text fields, image views, text views that create the entertainment for the Brand Media's users. In website we use the jQuery, Ajax, html, and css3 for UI. These tools make the UI's elements familiar. But website is not responsive. And android application's UIs are the same as website UIs. But we make the app's UIs by following the mobile application interface standard. We make the app's user interfaces by using the xml language. This is very familiar language.

2.3.2. Hardware Interfaces

Since neither the mobile application nor the web portal have any designated hardware, it does not have any direct hardware interfaces. And the hardware connection to the database server is managed by the underlying operating system on the mobile phone and the web server.

2.3.3. Software Interfaces

We get Map from google system in both android app and website. We make the API to get the Map from google. In android application we make APIs to get all the functionality into android application. These APIs are used to do data transactions with MYSQL.

2.3.4. Communications Interfaces

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for both the mobile application and the web portal. We use the HTTP communication standard.

2.4. System Features

2.4.1 Login Account

2.4.1.1 Description and Priority

This feature provides an interface where any user of the system (customer, brand or admin) needs to authenticate their self before using this system. This is the first page that user see when he/her visit to Brand Media platform. This feature has high priority. If this feature will not exist in Brand Media, then no anyone will use other features of the system.

2.4.1.2 Stimulus/Response Sequences

- 1) First user visits the brand media.
- 2) User must have a register account to login otherwise create a new account.
- 3) User should need to enter credentials to login.
- 4) Click on login Button.

5) After Successfully login user get "Login Success" message.

2.4.1.3 Functional Requirements

REQ-SF1-1:

TAG: LOGIN ACCURACY.

DESC: User should need to input accurate email and password. Without email and password user not be able to login.

REQ-SF1-2:

TAG: EMAIL MATHCH

DESC: If email and password will not be matched with system database then system generate a "Login Success" message. Now user ready to use other feature of the system.

REQ-SF1-3:

TAG: LOGIN ERROR

DESC: If email and password do not match with system database then system generate "Invalid username or password" message. In this case User need to enter correct credentials for authentication.

2.4.2 Forget Password

2.4.2.1 Description and Priority

This feature provide help to users in case if they forget their login password and want to reset password. By using Recovery email user can successfully reset their password or can contact System Admin for further help. Feature has medium priority.

2.4.2.2 Stimulus/Response Sequences

- 1) On Login Page there is link to open forget password page.
- 2) User should need to enter recovery email to get a link to reset password.
- 3) After clicking reset password button an email sent to user email address.
- 4) User need to open that email from Inbox to check link or recovery code.

- 5) By using that link/code user can enter new password and use this password in future.

2.4.2.3 Functional Requirements

REQ-SF2-1:

TAG: RECOVERY EMAIL.

DESC: User must have a recovery email, set in their account registration. When the user forgot password then user get the message from recovery email. This message contain the recovery code.

REQ-SF2-2:

TAG: RECOVERY CODE.

DESC: A Recovery code that user gets from recovery email need to input accurately to reset password. User enter the code and then rest the password.

2.4.3 Update Profile

2.4.3.1 Description and Priority

This feature has the high priority. By using this feature customer/brand can update their profile information. Its very important feature because profile provide identity of every user whether its customer or brand profile. User can change their profile name, bio, profile photo, cover photo, recovery email etc.

2.4.3.2 Stimulus/Response Sequences

- 1) User need login his account.
- 2) User should click to Profile Page.
- 3) On profile page user can click on edit profile to update profile information.
- 4) After click edit profile user now update profile and change username, email, profile photo and cover page etc.
- 5) After changing profile data user need to click on Save Changes.
- 6) System display the message as "Profile Updated Successfully!".

2.4.3.3 Functional Requirements

REQ-SF3-1:

TAG: CHANGE RECOVERY EMAIL.

DESC: User can change recovery email after verifying new recovery email address.

REQ-SF3-2:

TAG: UPDATE ERROR.

DESC: if user enter inaccurate format of email into email field then system generate the message "please enter right email".

REQ-SF3-3:

TAG: ADMIN PERMISSION

DESC: Brand cannot be changing their profile name without admin permission. Brand should need to get permission to change brand name.

REQ-SF3-4:

TAG: USERNAME LIMIT.

DESC: Username should be in character format and in specific length and username changes must have limits in specific period of time.

2.4.4 Search Engine

2.4.4.1 Description and Priority

This feature has the high priority. If this feature will not exist in system then users will not find things easily. Search engine help to user search/find anything that available on Brand Media. User can search product by using product name, brand name and by category name. User can search to find their friend and family profiles. Search Engine provide flexibility to users to find anything easily without wasting a lot of time.

2.4.4.2 Stimulus/Response Sequences

- 1) User need to login his/her account.

- 2) Go to search bar and input text to find anything user want.
- 3) User can get result of their search text characters by hitting Enter button.
- 4) In Search result page user get three kind of information i.e. customers , brands and products name their profile photo and view profile for users and view product detail for products.
- 5) By click on view profile button user can get detail of customer/brand profile.
- 6) There should also have follow button for following users.

2.4.4.3 Functional Requirements

REQ-SF4-1:

TAG: DATA AVAILABILITY.

DESC: Search result should only provide that data which is available on Brand Media system database.

REQ-SF4-2:

TAG: SEARCH ACCURACY.

DESC: User need to enter relevant or accurate (get good results) character to find data. In case of irrelevant search system cannot find any result and display result with “No record found” message.

REQ-SF4-3:

TAG: SEARCH ITEM.

DESC: User can search brand page and customer profile.

2.4.5 Timeline

2.4.5.1 Description and Priority

This feature also has the high priority. Timeline feature is very important and interesting feature of this system. Every user(customer/brand) has their own timeline page where they can share their favorites products of different brands by clicking on share button on

product post available on newsfeed page. Shared post display on timeline according to time when user share the post of product from newsfeed to their own timeline.

2.4.5.2 Stimulus/Response Sequences

- 1) User need login his account.
- 2) Go to profile page.
- 3) On user profile page timeline section will available.
- 4) Users can see timeline of each other by clicking on profile page of the user.

2.4.5.3 Functional Requirements

REQ-SF5-1:

TAG: TIMELINE ACCURACY.

DESC: Customer can share only those products on timeline that will uploaded by brand which will followed by that user.

REQ-SF5-2:

TAG: SHARE BUTTON.

DESC: On news feed page every product post has share button so by clicking on that button user can easily share product on his/her timeline

REQ-SF5-3:

TAG: TIMELINE ON PROFILE.

DESC: Timeline show shared product of each specific user on that profile timeline section in order to time of post shared.

2.4.6 News feed

2.4.6.1 Description and Priority

This feature has the high priority. This is very important module/feature of our system. First time we are introducing a new layout in e-commerce sites, a page for every user in which user saw a constantly updated list of their favorite Brands activities. Newsfeeds highlights information that includes add new product, product price changes, upcoming events, upcoming products and product launching events etc. News Feed feature provide

easiness for users to track activities and updates of their favorite brands and people. Every user has their own News Feed page that give updates and activates to user according to only those brands and people which are following by user. When brand add a new product then on user newsfeed page automatically user get that post then user can comment on that product post can like the product and also can share that product on his/her own timeline page, so their follower can see that product which he/her share.

2.4.6.2 Stimulus/Response Sequences

- 1) User need to login his account.
- 2) By clicking on home page link user can see news feeds.
- 3) Home page provide news feed interface.
- 4) User can share, comment, like and can add to cart products.
- 5) User can check detail of product by clicking on each specific product post.
- 6) Clicking on share button user can share post to his/her timeline.

2.4.6.3 Functional Requirements

REQ-SF6-1:

TAG: FOLLOWING NEED

DESC: To see posts of different brands you should need to follow that brand. Because user can see only those brands posts that he is following.

REQ-SF6-2:

TAG: ESSIANTIAL ITEMS

DESC: Post has Product photo, Price, short description, brands profile photo, brand name, comment section, like button, share button and add to cart button.

REQ-SF6-3:

TAG: CART ICON

DESC: if user interested to buy any product then user can add product to cart by clicking on cart icon button that available on every product post.

2.4.7 Follow BRAND

2.4.7.1 Description and Priority

This feature also has the highest importance and priority. If this feature not exist in brand media then users will not get any update from brand pages. And no any customer will use the brand media. By using this feature user can follow different brands to get notifications and add new product post updates. Customers can also follow each other accounts to share their favorite choices with each other and get feedback before buying the product from friends, family, colleagues and follower circle.

2.4.7.2 Stimulus/Response Sequences *user login his account.*

- 1) User need to login his account
- 2) Open any user (brand) profile.
- 3) On profile page has a follow button.
- 4) Click on follow button to send request to that profile user.
- 5) If the second user accept following request then user successfully can be follower of the second user.
- 6) After following user can get notify and see all updates of posts on news feed that user can add upload in future.

2.4.7.3 Functional Requirements

REQ-SF7-1:

TAG: ACCURACY OF BRAND

DESC: User can follow only those brands that has profile in this system. User can send follow request only those users which one not already follow by that user and only if user can not sent already follow request.

REQ-SF7-2:

TAG: FOLLOW NOTIFICATION

DESC: User can get notify that receive follow request. User get notify, when notification occur from other user that will send the follow request.

REQ-SF7-3:

TAG: SUCCESSFUL FOLLOW.

DESC: After Successfully acceptance of follow request sender user also notify.

2.4.8 Unfollow user

2.4.8.1 Description and Priority

This feature has the high priority. By using this feature user can unfollow that user which he/her not want to follow too longer. If user does not want to see updates from some brands so he simply can unfollow that brand to stop see updates from that brand product posts. Also, if user does not want to see choices of someone from friends and family then simply unfollow that members.

2.4.8.2 Stimulus/Response Sequences

- 1) User need to login his/her account.
- 2) Go to profile page of user.
- 3) Profile has Unfollow button.
- 4) Click on unfollow button to unfollow that profile user.
- 5) User can get again an option to send follow request to user.

2.4.8.3 Functional Requirements

REQ-SF8-1:

TAG: UNFOLLOW ACCURACY.

DESC: User can unfollow only those users which one already follows by that user.

REQ-SF8-2:

TAG: NESWFEED REMOVE.

DESC: User will no longer get notify and receive updates on news feed page from that brand. Not receive choices of that person from his timeline shares.

REQ-SF8-3:

TAG: AGAIN FOLLOW:

DESC: User can follow again of that brand/user to receive notification and posts on news feed page.

2.4.9 View follower list

2.4.9.1 Description and Priority

This feature has the medium priority. By using this feature user can check their followers list and can check following list. By checking this list user can get to know about how much users he/her following and to know quantity and identity of followers. By using this list user can make decision and manage their followers.

2.4.9.2 Stimulus/Response Sequences

- 1) User need to login his account.
- 2) Go to profile page.
- 3) Click on view Followers link.
- 4) By clicking on View Followers link user can see list of followers.
- 5) User can check profiles and also can unfollow the users from this list.

2.4.9.3 Functional Requirements

REQ-SF9-1:

TAG: LIST ACCURACY.

DESC: List show only those brand/users that one follows by user or following to that user.

REQ-SF9-2:

TAG: UNFOLLOW IN LIST

DESC: User Can unfollow any user by clicking on unfollow button that show with every user on list.

REQ-SF9-3:

TAG: BLOCK IN LIST.

DESC: User can also block users by using this list. When user block the other user then other user cannot view his profile.

2.4.10 Notifications

2.4.10.1 Description and Priority

This feature has the high priority. A notification is a message that displays on notification tab UI to provide the user with reminders, communication from other people, or other timely information from app. Users can tap the notification to open that activity or take an action directly from the notification. This system provide notification on major activities like when user receive follow request, accepting follower request, when brand upload a new product and on changing of profile photo of user.

2.4.10.2 Stimulus/Response Sequences

- 1) Brand user need to login his/her account.
- 2) Go to notification bar.
- 3) Click on specific notification to open activity of that notification.
- 4) User can remove notification by click on "X" button.
- 5) Only new notification will alert user by notification msg.

2.4.10.3 Functional Requirements

REQ-SF10-1:

TAG: NOTIFICATION ARRIVAL.

DESC: Notification Alert happen only when user update profile photo, receive follower request to follow, when some accept follow request of that user, when following brand upload a new product and price change on some products.

REQ-SF10-2:

TAG: NOTIFICATION DETAILS

DESC: Notification bar display time of activity, title of notification, username and profile photo of that user that perform activity.

REQ-SF10-3:

TAG: MUST FOLLOWED

DESC: User receive notification only from that brand products which he already following.

2.4.11 Verified address using Map

2.4.11.1 Description and Priority

After clicking on checkout button user need to choose location where he wants to receive package that he is going to order. By using Text field input for Shipment Address we are also going to use Map to get shipment address of order. By getting longitude and latitude of user selected location for order delivery we can deliver product at accurate place without going to any wrong place and to find customer by making phone calls. This feature is very helpful for user to easily tell us delivery location by placing marker in Map and this is very helpful for delivery boy to reach at exact place to deliver order without wasting their and customer time.

2.4.11.2 Stimulus/Response Sequences

- 1) User need to login his/her account.
- 2) Go to Shopping cart page.
- 3) Need to click on submit button after adding product in to cart.
- 4) After Click on Checkout Button user receive a page with Map.
- 5) User can set marker on the place where he/her want to receive deliver product.
- 6) After setting marker on favourite place user need to click on save location button.
- 7) After saving location user receive system generated message" Order Location has been set Successfully".

2.4.11.3 Functional Requirements

REQ-SF11-1:

TAG: SET MARKER.

DESC: User need to set marker on map every time when he places order.

REQ-SF11-2:

TAG: LOCATION SAVING.

DESC: Location save on the base of longitude and latitude on system database.

REQ-SF11-3:

TAG: CHECK MARKER.

DESC: Brand that receive order from user can also check marker on map to deliver product on that marker.

2.4.12 Create customer account

2.4.12.1 Description and Priority

This feature has the high priority. If this feature not exist in brand media then customer users will not create their pages. And no any customer will use the brand media. So, for use of the system user should required an account. If user already have an account then he/she does not require to make a new account he/she can use account by using credentials on login account page. For a new customer should require for making a new account. Customer can easily create account by giving some information. This information is secure by using encryption of data.

2.4.12.2 Stimulus/Response Sequences

- 1) User need to visit on Brand Media platform.
- 2) Go to Sign Up page to create new account.
- 3) Need to provide some information.
- 4) After Click on Sign Up button user can successfully create his account.
- 5) User receive "Account Created Successfully".

2.4.12.3 Functional Requirements

REQ-SF12-1:

TAG: ESSIANTIAL ATTRIBUTES.

DESC: User must have to enter email, password, first name, last name, gender, photo.

REQ-SF12-2:

TAG: CUSTOMER VERIFICATION.

DESC: User can create new account on base of their email address and activate his/her account after verifying email, password, first name, last name, gender, photo that provided during creating account.

REQ-SF12-3:

TAG: ACCOUNT ACCURACY.

DESC: User can create only one account with one email address. Two account does not be create using one email address.

REQ-SF12-4:

TAG: PASSWORD ACCURACY.

DESC: User should need to enter minimum 8 character to enter password.

REQ-SF12-5:

TAG: EMAIL FORMAT.

DESC: Email address should provide in email format i.e. abc@gmail.com. If user will not enter the email then system display message on same time.

2.4.13 Shopping Cart

2.4.13.1 Description and Priority

This feature has the highest priority. If user interested to buy any product, then first of all user need to add that product into cart. User can add more than one product into cart. Shopping cart module have their own separate section in which user can manage all their product which he\her add in shopping cart. User can also drop product items from cart if he changed his decision. Shopping cart calculate the total price and quantity of products that available in cart. Users can use different coupon codes, discount numbers, and promotion codes if they have for getting product at low prices. By clicking on checkout button users can follow next step (chosedelivery location of order by using map) to make a successful order of products.

2.4.13.2 Stimulus/Response Sequences

- 1) User need to login his account.
- 2) User can view his cart by clicking on Cart icon button on navbar.
- 3) User can add, delete and update product into cart.
- 4) Change quantity of products.
- 5) User can view unit price of each product and also sum of total cart.
- 6) There is checkout button to process next step of order.

2.4.13.3 Functional Requirements

REQ-SF13-1:

TAG: SALE ACCURACY.

DESC: Only product can be added into cart that available to sale by brand.

REQ-SF13-2:

TAG: CHANGE QUANTITY.

DESC: User can change quantity of product under limit of given quantity of that product.

REQ-SF13-3:

TAG: DEATIAL DOCUMENT.

DESC: User can view quantity, price and total sum of products and also download the detail report of order in pdf format.

2.4.14 Add to cart

2.4.14.1 Description and Priority

This feature has the high priority. This feature will be performed on customer side. If user interested to buy any product that available on brand media, then first step is to add that product into cart. Without adding product into cart user cannot be able to place order of that product. So, this feature is very important for system to sale product of brands.

2.4.14.2 Stimulus/Response Sequences

- 1) User need to login his account.
- 2) Go to newsfeed, timeline and search bar to find product that user want to buy.
- 3) Click on add to cart button available with every product.
- 4) After clicking on add cart button the product added into user cart.
- 5) User can that product into Shopping Cart page.

2.4.14.3 Functional Requirements

REQ-SF14-1:

TAG: CART ACCURACY.

DESC: Only those products can be added successfully into cart that available for sale and quantity of that product available.

REQ-SF14-3:

TAG: PRODUCT REMOVE

DESC: Before successfully place order product can be automatically removed from your cart in case of end stock of that product which user added into cart for buy into future.

2.4.15 Product rating

2.4.15.1 Description and Priority

This feature has the medium priority. This feature is performed on the customer side. After 15 days of successfully buying product customer receive a form brand that take review and rating from the customer about product that customer purchase. Customer share personal experience about product after using it 2 weeks. Customer given review and rating can be very helpful to new buyers to identify product quality and different aspect of product by reading reviews from actual buyers. Rating and reviews only allow for buyers after 15 days of successful purchase. No one can give review without purchasing and using product.

2.4.15.2 Stimulus/Response Sequences

- 1) User need to login his account.
- 2) Go to profile page.
- 3) There will user rating feedback form section available.
- 4) User can check that section or also user receive notification of rating form after buying product successfully.
- 5) After filling product rating feedback form customer can click on submit form button.
- 6) System store data into database successfully.

2.4.15.3 Functional Requirements

REQ-S15-1: Only customer that buy product can rate the product.

REQ-SF15-2:

TAG: ONE RATTING.

DESC: Only one rating form available for that product which one user successfully buy.

REQ-SF15-3:

TAG: RATTING NOTIFICATION.

DESC: if brand deliver the product then brand generate the ratting notification to customer. After 15 days, then customer view the notification of the ratting. When customer click on ratting notification, then a page will be opened and then customer rate on product.

REQ-SF15-4:

TAG: STAR RATTING.

DESC: Form must be on star rating base or one question to answer.

2.4.16 Place Order/Checkout

2.4.16.1 Description and Priority

This feature has the high priority. After successfully adding product into shopping cart the next step is to check out the cart or to place order. User can checkout cart to process

order into next steps by clicking on Checkout button that available on shopping cart page.

2.4.16.2 Stimulus/Response Sequences

- 1) User need to login his account.
- 2) Go to shopping cart page.
- 3) Manage all ordered product quantity and subtotal.
- 4) Click on checkout Button to process order into next step.

2.4.16.3 Functional Requirements

REQ-SF16-1:

TAG: CHECKOUT BUTTON

DESC: Every time user needs to click on Checkout button to place order.

REQ-SF16-2:

TAG: ORDER PROBLE.

DESC: If there is an issue with price and quantity of products in shopping cart then user can't be checkout the cart successfully.

REQ-SF16-3:

TAG: MESSAGE RECEIVING.

DESC: User receive system generated messages in case of any issue happen during checkout of cart.

2.4.17 Register Brand

2.4.17.1 Description and Priority

This feature has the high priority. If this feature not exist in brand media then Brand users will not create their pages. And no any customer will use the brand media. Every brand should need to register their business page before using Brand Media. Brand can

easily create their business page by using signup form. Admin approval required to activate business page.

2.4.17.2 Stimulus/Response Sequences

- 1) First brand user visits the brand media.
- 2) Brand user will create his business page.
- 3) If brand user would have register page then System generate the message to brand
- 4) User as “please wait for approval”.

2.4.17.3 Functional Requirements

REQ-SF17-1:

TAG: REGISTRATION FORM

DESC: When brand register the page then he must provide the Brand Name, Email, Password, Brand Type, Brand Information, Brand Photo, phone no, and Proof.

REQ-SF17-2:

TAG: CONFIRMATION

DESC: When user enter all information and enter register page. Then system will display the confirmation of Brand Name, Email, Password, Photo, Proofs, Phone no, Business Page Category. If brand user confirms the information, then system generate the message to brand “Please Wait for Approve the Request”.

REQ-SF17-3:

TAG: REGISTRATION INVALID INPUT

DESC: When brand enter invalid proofs then the system generates the message to brands please enter right proofs. If user enter characters in phone no field, then system generate message “you must enter digits”. If user enter numeric digit in “Brand Name”

the system should that generate the message to user for invalid input. Register should not be done until user enter right inputs.

2.4.18 Manage Products

2.4.18.1 Description and Priority

This feature has the high priority. This feature available only for brand users. By using this feature brands can manage all their products. A brand can add new product, update and delete their products.

2.4.18.2 Stimulus/Response Sequences

- 1) Brand user login his account.
- 2) Go to product gallery.
- 3) Add new product.
- 4) Provide the valid information for new product.
- 5) System will add data into database.
- 6) System display the message as “New Product Has Been Enter Successfully”.

2.4.18.3 Functional Requirements

REQ-SF18-1:

TAG: PRODUCT GALLERY.

DESC: Product gallery must exist in brand media. Brand user must enter the data into product name, price, quantity, description, and product photo.

REQ-SF18-2:

TAG: ERROR MESSAGE.

DESC: if user enter character into price field then system generate the message “please the digits”.

REQ-SF18-3:

TAG: UPDATE QUANTITY.

DESC: When will be the product enter successful then automatically “Total Products” will be updated. When the brand user will go to his business page then new product must be show on top.

2.4.19 Manage orders

2.4.19.1 Description and Priority

This feature has the high priority. Brand user can use this feature to manage all the orders which they receive from customers when they place orders. Brand user can view all delivered order and new order and all history of the orders. It shows all detail of order like product name, quantity ,username , user profile photo and the address of customer where the order will deliver.

2.4.19.2 Stimulus/Response Sequences

- 1) Brand user login his account.
- 2) Go to Manage Orders page.
- 3) View all orders that was delivered and new orders which one will deliver.
- 4) After deliver order successfully to customer Brand user mark that order as Successfully delivered.
- 5) System will add data into database.
- 6) System display the message as “Product Has Been Delivered Successfully”.

2.4.19.3 Functional Requirements

REQ-SF19-1:

TAG: CHECKKING ORDER.

DESC: Brand user can check all their orders whether it is new, or it was delivered in past.

All type orders display in different lists.

REQ-SF19-2:

TAG: REMOVE ORDER.

DESC: User can view orders, delete fake orders.

REQ-SF19-3:

TAG: ORDER DETAIL.

DESC: Brand user can get pdf file reports list of all their orders.

REQ-SF19-4:

TAG: RECENT LIST.

DESC: Recent orders will display at the top in list so brand can easily see new orders and can take further steps to deliver product as soon as possible.

2.4.20 Deliver Products

2.4.20.1 Description and Priority

This feature has the high priority. Delivery of product is within time very necessary. When customer place order of some specific product then this order will transfer to that specific brand to the product. Brand receive new order from customer which contain one product or bulk quantity of product. Here is the user need to deliver product and receive cash from the home the customer in COD method of payment. So, we are providing a new feature to get address of customer using map. Using this feature a brand can easily transfer the product at customer destination.

2.4.20.2 Stimulus/Response Sequences

- 1) Brand user login his account.
- 2) Go to Manage Orders page.
- 3) View all new orders which will deliver.
- 4) Check deliver address and marker (Latitude, Longitude) on map where the order will deliver.
- 5) After deliver order successfully to customer Brand user mark that order as successfully delivered.
- 6) System will add data into database.
- 7) System display the message as “Product Has Been Delivered Successfully”.

2.4.20.3 Functional Requirements

REQ-SF20-1:

TAG: ESSANTIAL DELIVERT.

DESC: Order must contain product name, price, quantity, description, and product photo with customer name, customer email, customer photo, and customer location.

REQ-SF20-2:

TAG: DELIVERY DONE.

DESC: After successful delivery of order then brand mark order as order marked as delivered order.

REQ-SF20-2:

TAG: DELIVERY DONE.

DESC: After successful delivery of order then brand mark order as order marked as delivered order.

REQ-SF20-2:

TAG: CUSTOMER LOCATION.

DESC: Brand open the map in the order. This map has the location of customer. By seeing the customer location brand easily deliver the product on right address.

2.4.20 Dashboard Login

2.4.21.1 Description and Priority

This feature has the high priority. By using this feature admin of brand can get access and authenticate himself into system. In dashboard admin can manage Brand Media platform from backend. Admin can manage databases, performance and security of the system. So the system work smoothly without interruption from external unauthenticated resources.

2.4.21.2 Stimulus/Response Sequences

- 1) Admin visit brand media platform.
- 2) Go to login page of the system.

- 3) Enter Credentials.
- 4) Click on login button.
- 5) System will add data into database.
- 6) Authenticate admin credentials.
- 7) System display the message as “Login Success”.

2.4.21.3 Functional Requirements

REQ-SF21-1:

TAG: ADMIN ACCOUNT.

DESC: Admin account must exist in system database to for login successfully into system and admin user need to input the valid email address and password to authenticate himself.

REQ-SF21-2:

TAG: LOGIN ERROR.

DESC: if admin user input invalid character into email field then system generate the message “please enter email in valid format”.

REQ-SF21-3:

TAG: RECORD MATCHING.

DESC: Admin user email and password should be match with the email and password that store in system database against admin role for login success.

2.4.22 Approve Brand

2.4.22.1 Description and Priority

This feature has the high priority. When brand user created his brand page for the first time the approval of the business page is very important. Without approval of the brand page user cannot use his account on Brand Media. Here admin user can approve every business page after verifying proof given by brand. Admin check that whether the user

which are going to create brand page that has an actual brand and the products. After verifying all proof and information if the brand is eligible and meet with criteria of brand media to create business page then admin allow user to use his new brand page otherwise admin will reject the request to create a new brand page.

2.4.22.2 Stimulus/Response Sequences

- 1) Admin user login his account.
- 2) Go to approve brand section.
- 3) Check request of new brand pages.
- 4) Admin verify the brand media proof against brand media policy to create new brand page.
- 5) If Brand user meet with criteria and fulfill brand media policy then admin approved the page of brand .
- 6) System display the message as “Brand page has been approved successfully”.

2.4.22.2.1 Functional Requirements

REQ-SF14-1:

TAG: REQUEST GALLERY

DESC: A “request gallery” must be exist on the admin side. When the admin open this request gallery then then all the request will be display in request gallery in the form of list.

REQ-SF14-2:

TAG: VIEW PRODUCT.

DESC: When the admin view the request of brand then a “check proof” option should be exist behind the request and admin will check the proofs. This proofs must be in the forms of photo.

REQ-SF14-3:

TAG: APPROVE PAGE.

DESC: If admin want to approve the request then a “approve button” must be behind with request in request list.

REQ-SF14-4:

TAG: APPROVED LIST

DESC: An “Approved Brands” list must be exist in the request gallery. Approved brand request must be showed in the list of “Approved Brands”.

2.4.23 Manage categories

2.4.23.1 Description and Priority

This feature has the high priority. This module will build to categories products according to the nature of products. By using this section user can easily find different products and brands by choosing those categories of products in which he\she interested and want to buy. i.e. Electronics Category have Cameras, Laptops, and Mobiles and in Fashion Category user find shirts, shoes, trousers etc. Only Admin has rights to manage categories of products.

2.4.23.2 Stimulus/Response Sequences

- 1) Admin user login his account.
- 2) Go to dashboard.
- 3) Open Category section.
- 4) Admin can add, delete, update and modify categories.
- 5) After changing in categories admin need to click on “Save Changes” button.
- 6) System will add data changes into database.
- 7) System display the message as “Categories has been changed Successfully”.

2.4.23.3 Functional Requirements

REQ-SF23-1:

TAG: ADIM MANAGER.

DESC: Only admin user can manage all categories.

REQ-SF23-2:

TAG: CATEGORY MANAGEMENT.

DESC: For adding category admin must need to mention main category and sub category name.

REQ-SF23-3:

TAG: CATEGORY LEVEL.

DESC: Each category has two level 1st one is Main Level(Main Category Name) and 2nd level(Sub Category Name).

2.4.24 Manage accounts

2.4.24.1 Description and Priority

This feature has the high priority. Admin has the right to manage accounts of users(customers/brands). Admin has access to add, update, view and delete information of all other user's accounts.

2.4.24.2 Stimulus/Response Sequences

- 1) Admin user login his account.
- 2) Go to Dashboard.
- 3) Open accounts manage section.
- 4) Search user by his name, email and id to get his account information.
- 5) System will update all data changes into database.

2.4.24.3 Functional Requirements

REQ-SF24-1:

DESC: Only admin user can manage all accounts information.

REQ-SF24-2:

TAG: CHANGING ACCOUNT.

DESC: Admin can change all kind of information related to any account without any permission.

REQ-SF24-3:

TAG: BLOCK ACCOUNT.

DESC: Admin can block any account in case of any illegal activity happened by users. If account blocked, and then user login his/her account, then system display the message to user your account has been block from admin.

2.4.25 Delete Business Page

2.4.25.1 Description and Priority

This feature has the medium priority. For creating new brand page admin approval should needed. So, admin also have rights to delete any business page in future due to some reasons. If brand page can be created and approved by admin in some illegal means so admin must right to delete this business page.

2.4.25.2 Stimulus/Response Sequences

- 1) Admin user login his account.
- 2) Go to Manage Account section.
- 3) Search business page by id, name or email.
- 4) Click on Deactivate account button to block that specific brand page for further usage.
- 5) System will save all changes into database.

2.4.25.3 Functional Requirements

REQ-SF25-1: Only admin has right to delete business page of the brand.

REQ-SF25-3:

TAG: DEACTIVE ACCOUNT.

DESC: By clicking on Deactivate Account button admin can successfully delete the brand page.

2.4.26 Manage Reports

2.4.26.1 Description and Priority

This feature has the low priority. User can manage all reports of activities happened in the system. Reports of all users with account information detail, how much activate users of the months, list of all orders that placed by users to brands ,list of all sold product etc. admin can check reports by daily base format, weekly base format, monthly base format or yearly base format. All reports can be downloaded in pdf format or in excel sheet format.

2.4.26.2 Stimulus/Response Sequences

- 1) Admin user login his account.
- 2) Go to Dashboard.
- 3) Open Reports Section.
- 4) System display all different section of reports with different format of reports.
- 5) Admin can view and download any report.

2.4.26.3 Functional Requirements

REQ-SF26-1:

TAG: PROPER FORMAT.

DESC: Reports can will be display data in ascending order by Date and Time. Reports should be in daily, weekly, monthly and yearly base format to view and download.

REQ-SF26-2:

TAG: DOWNLOAD REPORT.

DESC: Admin can download that report in pdf or excel sheet format.

2.4.27 Reset Passwords

2.4.27.1 Description and Priority

This feature has the high priority. In case of any user forget password and also does not have recovery email access. In this case user can contact admin for reset the password. User must have some proof to verify that the account was registered with user email.

2.4.27.2 Stimulus/Response Sequences

- 1) Admin user login his account.
- 2) Go to dashboard.
- 3) Open account manage section.
- 4) Search user by id, name or email.
- 5) Reset the password of that user.
- 6) System will update data changes into database.
- 7) System display the message as “Password has been reset successfully”.

2.4.27.3 Functional Requirements

REQ-SF27-1: Only admin has right to reset password option.

REQ-SF27-2: Admin must have user name, id or email of that user to identify user account.

REQ-SF27-2: Admin should need to click on reset password button to reset password. New password has some specific character combination according to brand media policy

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

ID: NFR1

FR: Search Item

TITLE: Prominent of search feature

DESC: The search feature must be prominent and easy to find for the user record.

RAT: In order to for a user to find the search feature easily.

ID: NFR2

FR: Search Item

TITLE: Usage of the search feature

DESC: The different search options should be evident, simple and easy to understand.

RAT: In order to for a user to perform a search easily.

ID: NFR3

FR: Managing Request

TITLE: Usage of the result in the list view into request gallery.

DESC: The results displayed in the list view into request Gallery should be user friendly and easy to understand. Selecting a request in the request list should only take one click.

RAT: In order to for a user to use the list view easily.

ID: NFR4

FR: Delivery by using map.

TITLE: Usage of the result in the map view

DESC: The results displayed in the map view should be user friendly and easy to understand.
Marker of customer should be clear for brand.

RAT: In order to for a user to use the map view easily.

ID: NFR5

FR: Search Feature

TAG: Response Time

GIST: The fastness of the search

SCALE: The response time of a search.

MUST: No more than 8 seconds 100% of the time.

WISH: No more than 1 second 100% of the time.

DESC: the search feature has the maximum speed to find the user profile. When user search profile then result should be display quickly.

ID: NFR6

FR: View News Feeds.

TAG: Response Time

GIST: The fastness of the News Feeds

SCALE: The response time of a news feeds.

MUST: No more than 12 seconds 100% of the time.

WISH: No more than 5 second 100% of the time.

DESC: the view newsfeeds should have the maximum speed to display product with their information.

ID: NFR6.1

FR: like product.

TAG: Response Time

GIST: The fastness of the like product.

SCALE: The response time of a like on news feeds.

MUST: No more than 1 seconds 100% of the time.

WISH: No more than 0.5 second 100% of the time.

DESC: the like product feature should have demands minimum time to like on the product.

ID: NFR6.2

FR: comment product.

TAG: Response Time

GIST: The fastness of the comment product.

SCALE: The response time of a comments on news feeds.

MUST: No more than 5 seconds 100% of the time.

WISH: No more than 2 second 100% of the time.

DESC: The comment product feature should have demands minimum time to comment on the product. When user comment the product then this comment should be displayed quickly.

ID: NFR6.3

FR: comment product.

TAG: Response Time

GIST: The fastness of the share product.

SCALE: The response time of a share the news feeds.

MUST: No more than 5 seconds 100% of the time.

WISH: No more than 2 second 100% of the time.

DESC: The share product feature should have demands of minimum time to share the product. When user share the product then total shares should be displayed.

2.5.2 Safety Requirements

2.5.3 Security Requirements

2.5.4 Software Quality Attributes

Usability:

ID: QA1

TITLE: System Usability

DESC: The system should be usable to users so that user easily understand all the operations of the user, learn the system easily, and use the system easily.

RAT: In order for use the system easily.

Reusability:

ID: QA2

TITLE: System Reusability

DESC: The system should be capable of using his components to develop the other system. All the components should be independent.

RAT: In order for system to use his components.

Flexibility:

ID: QA3

TITLE: System Flexibility

DESC: The system be capable for add more features of social media, ecommerce.

RAT: In order for system perform the required features.

Correctness:

ID: QA4

TITLE: System Correctness

DESC: The system must be perform that function for which system is developed.

RAT: In order for system perform the required features.

Portability:

ID: QA5

TITLE: Website Portability

DESC: The website should be easy to operate on other environment. The code should be written in a way that the system should be portable on other OS.

RAT: In order for run the system on other OS.

Maintainability:

ID: QA6

TITLE: System extendibility

DESC: The application should be easy to extend. The code should be written in a way that it favors implementation of new functions.

RAT: In order for future functions to be implemented easily to the application.

Testability:

ID: QA7

TITLE: Application testability

DESC: Test environments should be built for the application to allow testing of the applications different functions.

RAT: In order to test the application

Reliability:

ID: QA8

TAG: System Reliability

GIST: The reliability of the system.

SCALE: The reliability that the system gives the right result on a search.

METER: Measurements obtained from 1000 searches during testing.

MUST: More than 98% of the searches.

PLAN: More than 99% of the searches.

WISH: 100% of the searches.

Availability:

ID: QA9

TAG: System Availability

GIST: The availability of the system when it is used.

SCALE: The average system availability (not considering network failing).

METER: Measurements obtained from 1000 hours of usage during testing.

MUST: More than 98% of the time.

PLAN: More than 99% of the time.

WISH: 100% of the time.

ID: QA10

TITLE: Internet Connection

DESC: The system should be connected to the Internet.

RAT: In order for the system to communicate with the database.

2.5.5 Business Rules

These system is developed for customers and brands there are some principles of Brand Media as:

Customer, and Brands have some same functionalities. They can view notification and newsfeeds, they can share, like, and comment on product. But some difference is that brands can add product while customer can buy that product. Customer can order product while brand view the customer's order and can deliver product. Customer direct create his/her account while the Brand create the account and cannot access the features that brand media provides till admin approve the Brand request.

2.6 Other Requirements

We need a MYSQL database that perform as a central database. We use the APIs to access the database in the android app. Website domain is also need to perform the MYSQL as a central database. Website will be uploaded in server. The database will be provided in database. Then this database will be performed as a central database.

Chapter 3

Use Case Analysis

Chapter 3: System Analysis

This chapter describes all the activities that performed by users with the system. This defines the use case diagram in which Primary users are used system at left side and secondary users are use system at right side. Use case diagram are usually referred to as behavior diagram used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors). Also this chapter include the fully dress use cases. Fully dressed use cases show more detail and are structured; they are useful order to obtain a deep understanding of the goals, tasks, and requirements.

3.1. Use Case Model

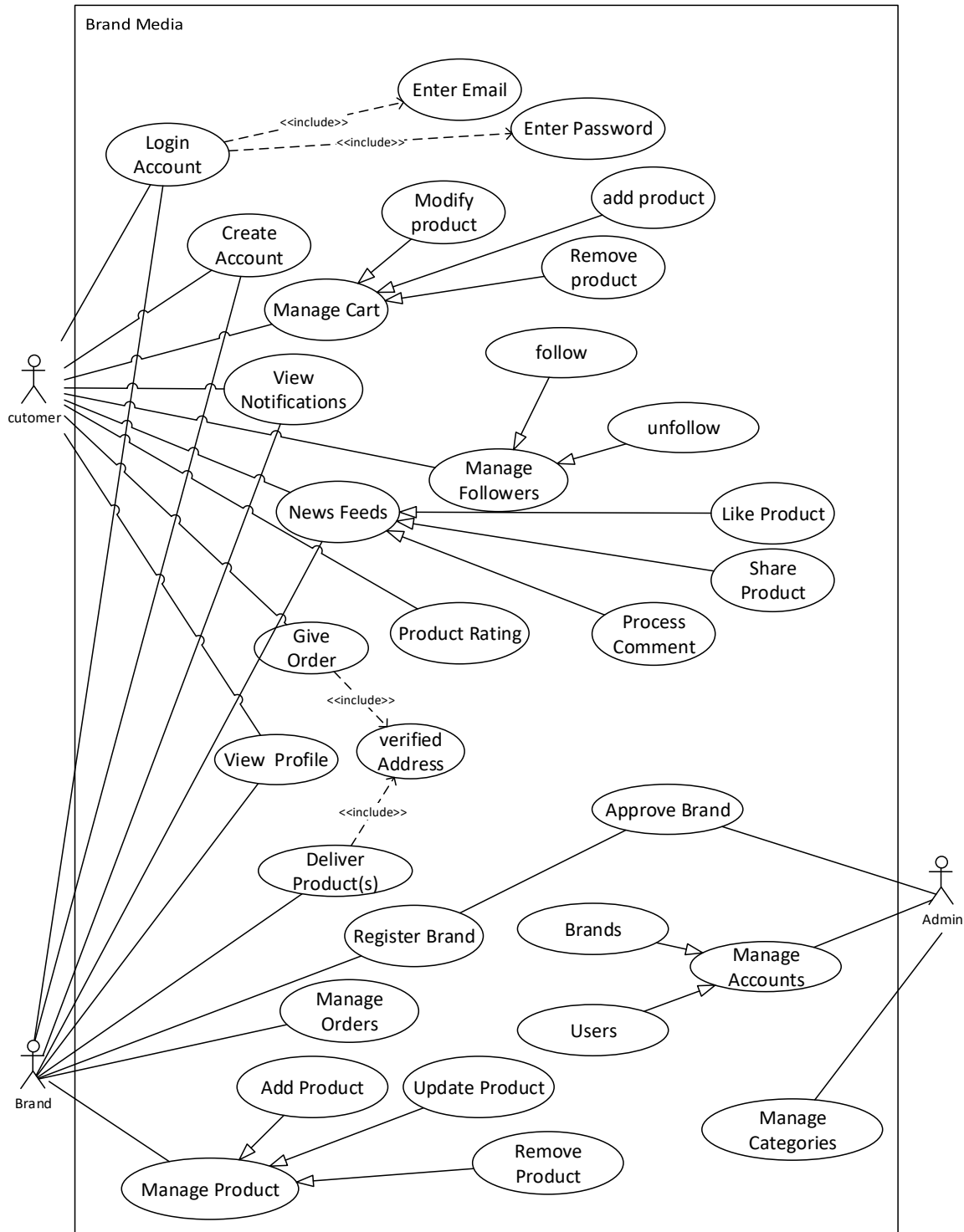


Figure 3.1 UseCase

3.2. Use Case Descriptions

Use case Name: Login Account.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer & Brand.

Stakeholders & Interest:

- Brand Media: wants to accurately user be login with the system and satisfy customer interests. After login user perform his functions with no difficulty.
- Users: want to accurately login with the system. Users want to that their id must be unique and password should be secure so that no anyone access their accounts.

Preconditions: Users is authenticated.

Post conditions:

The message will be generated as “Login Successfully” and users access all the services provided by brand media.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. System display the message to users to login.
3. Users start the login function.
4. User enter the id with password.
5. After enter id with password, if user will be authorized then he/she will be login with the system. And user proceed the user to perform functions.

Alternative flows:

1. User open the Brand Media Website/Android App.
2. User want to tries to login with the system.
3. If user will be fail and receive errors then user send report to admin.
4. Admin will check the problem and solved the problem.

5. If user forget his/her password then the user forget the password.
6. System send the email to user.
7. User check email, and in this email a code will be exist.
8. User will copy the code and paste into forget the password.
9. User enter the code.
10. System show the form.
11. In this form, user change his/her password.
12. And the login the account.

Failure Scenarios

- If internet connection is failed during the registration process then the login process will not be done.

Special Requirements:

- User's accounts must be secured.
- The maximum response time is 1 minute.
- Login feature is usable.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- Email and password will be entered through keyboard.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: None.

Use Case Name: Create Account.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer & Brand.

Stakeholders & Interest:

- Customers: want to register his accounts with no difficulty. User wants to add his account data in database successfully so that they can login with the system with no difficulty and they can buy, search, check products. And wants to get some social media functions.
- Brand Media: want to add record of user accounts into the database. So that users use the Brand Media and perform the brand media functions.

Preconditions: Brand media' registration page must be opened in the website/ android app.

Post Conditions:

The user's account record should be successfully recorded in database. And system take user to main window/activity to perform the functions. After logout, if customer want to login account then he/she will enter the Email with password to login the account.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. Go to the Registration page or activity.
3. System display the message to users to register account.
4. Users start the registration function.
5. User enter all the required information.
6. After enter information, he/she will be able to perform functions on the system.

Alternative Flow:

1. User open the Brand Media Website/Android App.
2. User want to tries to register his account in brand media.
3. If user will be fail and receive errors then user give message to admin.
4. Admin will check the conditions of errors of unsuccessful registration.
5. Admin will solve the problem of unsuccessful registration manually.

Special Requirements:

- After registered account user's accounts will be secured.

- Registration feature has the usability to users.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- User enter all the required information like first name, last name, email, and password etc. using keyboard.
- If user registration has been done then system display the message “you are registered” and then main/home page will be displayed.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If user will not be registered then student will wait for register till the error is removed by the admin.

Use Case Name: View Profile

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer & Brand.

Stakeholders & Interest:

- Users: wants to view his/her profile. User want to view all his/her detail in profile. Users want that he view own or other’s profile.

Preconditions: User must be login with the system.

Post Conditions:

User will view the profile and can manage the profile. All the user related data will be fetched and displayed for that user.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.

4. User go to view profile page/activity.
5. User will view his/her profile.

Alternative Flow:

1. If user open his/her profile then he get any issue in the profile data.
2. User send the report to admin.
3. Admin check the user data from database.
4. If admin find any error from database then admin will solve the problems.
5. Admin will send the email to user either for unsuccessful solving problem or successful solving problem.

Special Requirements:

- Users communicate with the system in Global Language (English).
- User profile is secured.

Technology & Data variation Lists: None.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If user's profile will not be opened then student will wait for view the profile till the error is removed by the admin. User will send report of error to the admin. Admin will remove the error.

Use Case Name: Like Product.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer.

Stakeholders & Interest:

- User: wants to like the products of brand pages. If any type of product that is good for user then user will want to like the product. User wants that if user like the product then he also check the "like" in the product.

- Brand Users: want that users like the products so that they can assess the level of likeness of users. And they add the product on their pages which is good to users.

Preconditions:

User must be login with the system and bran's product is display on the front of the users.

Post Conditions:

The product is "like" by the user. The data of like is stored in database. The "like" is display to the user which he like the product.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to home page/activity.
5. User view the product.
6. User will like the product.
7. Product status will be updated.

Alternative Flow:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to others profile page/activity or brand pages.
5. User view the product.
6. User will like the product.
7. Product status will be updated.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- The product will be shared using touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If user's "like" will not be added due to the poor internet and error then user will wait for add "like" till the internet connection problem/error is removed.

Use Case Name: Process Comment.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer.

Stakeholders & Interest:

- User: wants to "comment" on the products of brand pages. If any type of product that is good for user then user will want to "comment" on the product. User wants that if user comment the product then he also check the "comment" in the product.
- Brand Users: want that users comment on the products so that they can assess the level of likeness of users. And they add the product on their pages which is good to users.

Preconditions:

User must be login with the system and bran's product is display on the front of the users.

Post Conditions:

The user has comment on the product. The data of comment is stored in database. The "comment" is display to the user which he comment the product.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to home page/activity.
5. User view the product.

6. User will comment the product.
7. Product status will be updated.

Alternative Flow:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to others profile page/activity or brand pages.
5. User view the product.
6. User will comment on the product.
7. Product status will be updated.

8. If user comment not enter then user will send the report to admin.
9. Admin will check the error conditions.
10. Admin will solve the problem.
11. Admin will send the email to user.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).
- The comment feature has usability.

Technology & Data variation Lists:

- The product will be shared using touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If user's "comment" will not be added due to the poor internet and error then user will wait for add "comment" on the product till the internet connection problem/ any error is removed.

Use Case Name: Share Product.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer.

Stakeholders & Interest:

- Customer: wants to “share” the products of brand pages. If any type of product that is good for user then user will want to “share” the product to his friends. User wants that other users will share products so that user will check the likeness of other users.
- Brand Users: want that users share the products so that many users like and comments on their products and they can assess the level of likeness of users. And they add the product on their pages which is good to users.

Preconditions: User must be login with the system and bran’s product is display on the front of the users.

Post Conditions:

The share product record will be stored successfully. And the product will be displayed on news feed for his/her friends.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to home page/activity.
5. User view the product.
6. User will share the product.
7. Product status will be updated.

Alternative Flow:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to others profile page/activity or brand pages.
5. User view the product.
6. User will share on the product.

7. Product status will be updated.
8. If user comment not enter then user will send the report to admin.
9. Admin will check the error conditions.
10. Admin will solve the problem.
11. Admin will send the email to user.

Special Requirements:

- Users communicate with the system in Global Language (English).
- The usability feature has usability.

Technology & Data variation Lists:

- The product will be shared using touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: If product is not shared due to the poor internet and error then user will wait for add “share” the product till the internet connection problem/ any error is removed.

Use Case Name: Manage Cart.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer

Stakeholders & Interest:

- User: wants to use the shopping cart so that he/she can buy products. User want to get the total price of products added in shopping cart. This shopping cart will calculate the total price to user of products.
- Brand Users: want that users use the shopping cart for buy the products. So that they sell products to users and earn more.

Preconditions: User must be login with the system and brand’s product is display in the brand’s page on the front of the users.

Post Conditions:

Product(s) will be added in shopping cart and total price of all products will be calculated and showed.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to brand's page/activity.
5. User view the product.
6. User will add product(s) in the shopping cart.
7. Shopping cart will show the all products with their prices and calculate the total prices of all the product(s).
8. User can change the quantity of product.
9. System update the price of that product and display the changed price.
10. User can remove the product from cart.
11. When user will remove the product then automatically product will be removed form cart.
12. If user will not add product then it will view the shopping cart too.

Alternative Flow: None

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).
- Shopping cart is usable.
- Shopping cart will provide efficiency to users.
- The usability feature has usability.

Technology & Data variation Lists:

- The product will be shared using touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If product is not added in shopping cart due to the poor internet and error then user will wait for add the product in shopping cart till the internet connection problem/ any error is removed.

Use Case Name: Give Order.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer.

Stakeholders & Interest:

- Brand Users: want that users give order for buy products. So that they sell products to users and earn more. And get the feedback of their products from users.
- Customer: wants to use the giving order of product for buy to brands so that he/she can get products.

Preconditions: User must be login with the system and shopping cart page/activity is displayed.

Post Conditions:

User get the message of successful completion. Purchase order record will store in database. And if brand user is online then it will receive the purchase order notification from customer. And if brand user is offline so when he/she will login with the system, get the notification of order from users.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to brand's page/activity.
5. User view the product.
6. User will add product(s) in the shopping cart.
7. Shopping cart will show the all products with their prices and calculate the total prices of all the product(s).
8. User will give order to brands.

9. User will view the view the map.
10. User will set the marker into map.
11. User confirm the order.
12. Brand Page users will get the notification of order from user.

Alternative Flow:

1. If user order not enter then user will send the report to admin.
2. Admin will check the error conditions.
3. Admin will solve the problem.
4. Admin will send the email to user.

Special Requirements:

- Users communicate with the system in Global Language (English).
- The usability feature has usability.

Technology & Data variation Lists:

- The product will be shared using touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- If product is not shared due to the poor internet and error then user will wait for give order to the brand users till the internet connection problem/ any error is removed.
- If brand user is offline then user will wait.

Use Case Name: Send Request.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer

Stakeholders & Interest:

- User: wants to make a friends so that he/she share the products to his friends. Friends want to check other friends profile so that they can see the favorite brands of user.

Preconditions: User must be login with the system and other user's profile should be open.

Post Conditions:

The friend request record should be recorded into the database. The other user will receive the request from user.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to other user profile.
5. User will send friend request to other user.

Alternative Flow:

1. If friend request will not be sent request.
2. Then user will send report to admin.
3. Admin check the error conditions.
4. Admin solve the problem.
5. Admin will send the email to user.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- The first user send request to other using touch/mouse.
- The second user accept the request through touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- If second user will not accept the friend request then first user will not make the second user a friend.

Use Case Name: Block User.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer.

Stakeholders & Interest:

- User: wants to block users so that blocked users cannot see his/her profile and cannot follow him.

Preconditions: User must be login with the system and other user's profile should be open.

Post Conditions:

The blocked user record should be recorded. And blocked user cannot be access user profile. The blocked user list be updated.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to other user profile.
5. User will block other user.

Alternative Flow:

1. If any problem will be occur, during blocking user.
2. Then user will send report to admin.
3. Admin check the error conditions.
4. Admin solve the problem.
5. Admin will send the email to user.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- The first user send request to other using touch/mouse.
- The second user accept the request through touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: none.

Use Case Name: Decline Request.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Simple Users (Buyers).

Stakeholders & Interest:

- User: wants to make a friends so that he/she share the products to his friends. Friends want to check other friends profile so that they can see the favorite brands of user.

Preconditions: User must be login with the system and user open the notification.

Post Conditions: Friend request should be canceled of other user and friend request record should be removed.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User view the notification.
5. User will decline the request of other user.

Alternative Flow: None.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- The first user send request to other using touch/mouse.
- The second user accept the request through touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: none.

Use Case Name: Accept Request.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Simple Users (Buyers).

Stakeholders & Interest:

- User: wants to make a friends so that he/she share the products to his friends. Friends want to check other friends profile so that they can see the favorite brands of user.

Preconditions: User must be login with the system and other user profile is opened or friend request notifications must be display on screen.

Post Conditions:

The friend's record will be stored in database. The user friend list will be updated. Both user should get notification of followers.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to other user profile.
5. Or User will see the notification.
6. User accept the friend request.

Alternative Flow: None.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- The first user send request to other using touch/mouse.
- The second user accept the request through touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

- If second user will not accept the friend request then first user will not make the second user a friend.

Use Case Name: Remove Friends.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Simple Users (Buyers).

Stakeholders & Interest:

- User: wants to remove a friends so that he/she cannot share the products in user profile. And other users not see the profile of user.

Preconditions: User must be login with the system and other user profile or friend list is opened.

Post Conditions:

The friend's data will be updated in database. And user friend list will be updated.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to other user profile.
5. User will unfriend the other user.
6. The other user will be removed from user friend's list.

Alternative Flow:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to his/her friend's list.
5. User will unfriend the other user.
6. The other user will be removed from user friend's list.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- The first user unfriend the other through touch/mouse.
- The updated list will be display on screen.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: None.

Use Case Name: Product Rattng.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Customer.

Stakeholders & Interest:

- User: wants to rate the products of brands so that by rate the product(s), user can provide the feedback to brands.
- Brand Users: want that users provide feedback after buy products so that they can take improvements in their products or product feedback is well then they want that they increase the quantity of products.

Preconditions: User must be login with the system and Product is bought.

Post Conditions:

The rattng data will be recorded in database. And brand user get feedback of users by rattng. This rattng status of product should be updated.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Home activity will be opened.
4. User go to brand page.
5. He/she give order of products.
6. Brand user deliver the product to user.
7. Customer will give cash to brand user.
8. After using product, Brand send the feedback to customer and customer get the rattng notification.

9. User provide the feedback by rating the product.
10. Brand user get the rating information from customers.

Alternative Flow: none.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).
- Rating windows is usable for user.

Technology & Data variation Lists:

- User provide the feedback by touch/ left mouse click on question's choice.
- After completing the rating user send feedback by touch/left click of mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: None.

Use Case Name: Register Brand.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Brand

Stakeholders & Interest:

- User: wants that their favorite brands should be available on brand media. So that they can view their favorite products and buy products easily.
- Brand User: wants that their market and promotion increase. They want that they create their own business page and add products in business page so that customer will see their brand information and buy products.
- Owners of Brand Media: want to add record of brand page accounts into the database. So that brand users use the Brand Media and perform the brand media functions.

Preconditions: Brand business's registration page must be opened in the website/ android app.

Post Conditions:

The brand user's account record should be successfully recorded in database. And system take brand user to their created page.

Main Success Scenarios:

1. Brand User open the Brand Media Website/Android App.
2. Go to the Registration page or activity.
3. System display the message to brand users to register account.
4. Brand Users will start the registration.
5. Brands User enter all the required information.
6. After enter information, brand user's record will be store in database.
7. And system will take brand user to their created page and show waiting message till approval of page from admin.

Alternative Flow:

1. Brand User open the Brand Media Website/Android App.
2. Brand User want to tries to register his account in brand media.
3. If user will be fail and receive errors then user give message in the form of report to admin.
4. Admin will check the conditions of errors of registration.
5. Admin will solve the problem of unsuccessful registration.

Failure Scenarios

- If internet connection is failed during the registration process then the registration will not be done.

Special Requirements:

- After registered account user's accounts will be secured.
- The maximum response time is 1 minute.
- Registration feature has the usability to users.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- Brand User enter all the required information like first name, last name, email, and password etc. using keyboard.
- If user registration has been done then system display the message “you are registered” and then main/home page will be displayed.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If user will not be registered then student will wait for register till the error is removed by the admin.

Use Case Name: Manage Order.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Brand

Stakeholders & Interest:

- Brand Users: want that they view the notifications of orders of customers and deliver the product to customers.
- Customer: wants that brand deliver the product on right address.

Preconditions: User must be login with the system.

Post Conditions: All the customer’s order notification will be fetched from database. The product details with details of right customer and their address should be fetched from database.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. User open the all orders weather he delivered or new.
4. User can view/remove the order.
5. Also user check the customer details.

Alternative Flow: none.

Special Requirements:

- Users communicate with the system in Global Language (English).

- The usability feature has usability.

Technology & Data variation Lists:

- The product will be shared using touch/mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: none.

Use Case Name: Add Products.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Brand

Stakeholders & Interest:

- Brand User: wants to add his product in his page. So that customer see and but his product easily.

Preconditions:

Brand User must be login with the system and user must be open his brand business page.

Post Conditions:

The product of the brand will be added in database. And new product will be display on his page.

Main Success Scenarios:

1. User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Business page activity will be opened.
4. User add the product(s).
5. Then his new product will always be show with previous product.

Alternative Flow:

1. If user fined any issue before add product or after add product.
2. Then user send report to admin.
3. Admin will view the report.
4. Admin will solve the problem.

5. Admin will send the mail to user to inform about adding products.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).
- Product record must be secured.
- Add product feature provide the usability.

Technology & Data variation Lists:

- Product will be entered through keyboard and mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If user's status will not be opened due to the poor internet then user will wait for add status till the internet connection problem is removed.

Use Case Name: Update Product.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Brand

Stakeholders & Interest:

- Brand User: wants to update his/her product. Brand User wants to that when customer see his page then customer see his updated product.

Preconditions: Brand User must be login with the system and brand user must be open his page.

Post Conditions: The product of the brand will be updated in database. And updated product will be show in his page.

Main Success Scenarios:

1. Brand User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Business page will be opened.
4. User open the product.
5. User will update the product.

6. Then his updated product will be show in the profile.

Alternative Flow:

- If user try to add irrelevant into photo field then it will be fail.
- And add the right file.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).
- User status is secured.

Technology & Data variation Lists:

- Product will be updated through keyboard and mouse.

Frequency of Occurrence: Could be nearly continuous.

Open Issues:

If product will not be updated due to the poor internet then user will wait for update product till the internet connection problem is removed.

Use Case Name: Remove Product.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Brand

Stakeholders & Interest:

- Brand User: wants to remove his product from page, if he will add product that is idle. So that customer don't see his idle product from his profile.

Preconditions:

Brand User must be login with the system and Business page must be opened.

Post Conditions:

The product of the brand will be removed in database. And page will be reload so that idle status not show in profile.

Main Success Scenarios:

1. Brand User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Business page will be opened.
4. Brand User open the product.
5. Brand User will remove the product.

Alternative Flow: None.

Special Requirements:

- The maximum response time is 1 minute.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists:

- Product will be removed through mouse device or touch.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: If product will not be removed due to the poor internet and error then user will wait for remove product till the internet connection problem/error is removed.

Use Case Name: Deliver Product.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Brand

Stakeholders & Interest:

- Brand User: wants to deliver the product to customer fast. So that customer has no problem.
- Customer: wants that brand deliver the product on relevant address fast.

Preconditions: Customer's order will be opened. All the customer's order details with his/her locations must be opened.

Post Conditions:

A location of the customer will be show on the Map. And the brand's location will be show on the map. A way will be developed between the Brand and customer location.

Main Success Scenarios:

1. Brand User open the Brand Media Website/Android App.
2. First he login his/her account.
3. Business page will be opened.
4. Brand user will view the notification of the customer's order.
5. User open the order.
6. User will check the details of customer and his order
7. Brand user will open the map.
8. By using the map brand will deliver the product on customer's address.

Alternative Flow:

1. If map will not display any error.
2. User will send the report to admin.
3. Admin will see the report.
4. Admin will solve the problem.
5. After solving problem admin will send the mail to user.

Special Requirements:

- The maximum response time is 20 seconds.
- Users communicate with the system in Global Language (English).

Technology & Data variation Lists: locations will be show on map and way will be created between both.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: If product will not be removed due to the poor internet and error then user will wait for remove product till the internet connection problem/error is removed.

Use Case Name: Approve Request.**Scope:** Next Generation of Ecommerce with some Social Media Features.**Level:** User Goal.**Primary Actor:** Admin**Stakeholders & Interest:**

- Admin: wants to approve the request of brand. So that brand upload their products to sell. If request is fake then it will not be approved by admin.
- Brand: wants that their page will be registered successfully.

Preconditions: Admin must be login with system.**Post Conditions:** Brand's data will be stored successfully. Brand Page will be registered successfully.**Main Success Scenarios:**

1. Admin login his account.
2. Then it will open the request gallery.
3. Admin will view the list of request coming from brands.
4. Admin will approve the request.
5. Brand page will be registered.

Alternative Flow:

1. If admin get any error or any problem.
2. Then admin will approve the request direct go to database.
3. Update the record of brand.
4. Then request will be approved.

Special Requirements:

- Users communicate with the system in Global Language (English).

Technology & Data variation Lists: Status will be removed through mouse device or touch.**Frequency of Occurrence:** Could be nearly continuous.**Open Issues:** If admin will not view the request then brand user will wait till admin approve the request.

Use Case Name: Manage Reports.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Admin

Stakeholders & Interest:

- Admin: wants to view the reports of brands and customer. And solved their problems.
- Brand: wants that admin will solve their problems.
- Customer: wants that admin will solve their problems.

Preconditions: Admin must be login with system.

Post Conditions: A reports related data should be fetched in the form of list.

Main Success Scenarios:

1. Admin login his account.
2. Then it will open the reports.
3. Admin will view the list of reports coming from brands.
4. Admin view the reports coming from users.
5. Admin solve all the user's problems.

Alternative Flow: None.

Special Requirements:

- Users communicate with the system in Global Language (English).

Technology & Data variation Lists: none.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: If admin will not view the reports then users will wait till admin solve requests.

Use Case Name: Manage Customers.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Admin

Stakeholders & Interest:

- Admin: wants to view and remove the reports of brands and customer.

Preconditions: Admin must be login with system.

Post Conditions: customer/brand information should be fetched from database and details should be displayed in the form of list.

Main Success Scenarios:

1. Admin login his account.
2. Admin will view the customer profile.
3. Admin check misbehaves of users.
4. Admin can send the email.
5. Admin will remove the non-related data.

Alternative Flow: None.

Special Requirements:

- Users communicate with the system in Global Language (English).

Technology & Data variation Lists: none.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: none.

Use Case Name: Manage Brand.

Scope: Next Generation of Ecommerce with some Social Media Features.

Level: User Goal.

Primary Actor: Admin

Stakeholders & Interest:

- Admin: wants to view and remove the reports of brands and customer.

Preconditions: Admin must be login with system.

Post Conditions: customer/brand information should be fetched from database and details should be displayed in the form of list.

Main Success Scenarios:

1. Admin login his account.
2. Admin will view the brand profile.
3. Also admin will view the business page.

4. In page, admin will visit the products.
5. Admin check misbehaves of users.
6. Admin can send the email.
7. Admin will remove the non-related data.

Alternative Flow: None.

Special Requirements:

- Users communicate with the system in Global Language (English). (Readability).

Technology & Data variation Lists: none.

Frequency of Occurrence: Could be nearly continuous.

Open Issues: none.

Chapter 4

System Design

Chapter 4: System Design

This chapter is describe the design behavior of the system. This chapter defines the design characteristics. This chapter defines the activities that how system perform the activities. This chapter show the coding behavior of the system that develop generate the code by following design document. This chapter describe the all system components, database design that what is the database behavior of system, coding behavior and activity behavior.

4.1. Architecture Diagram

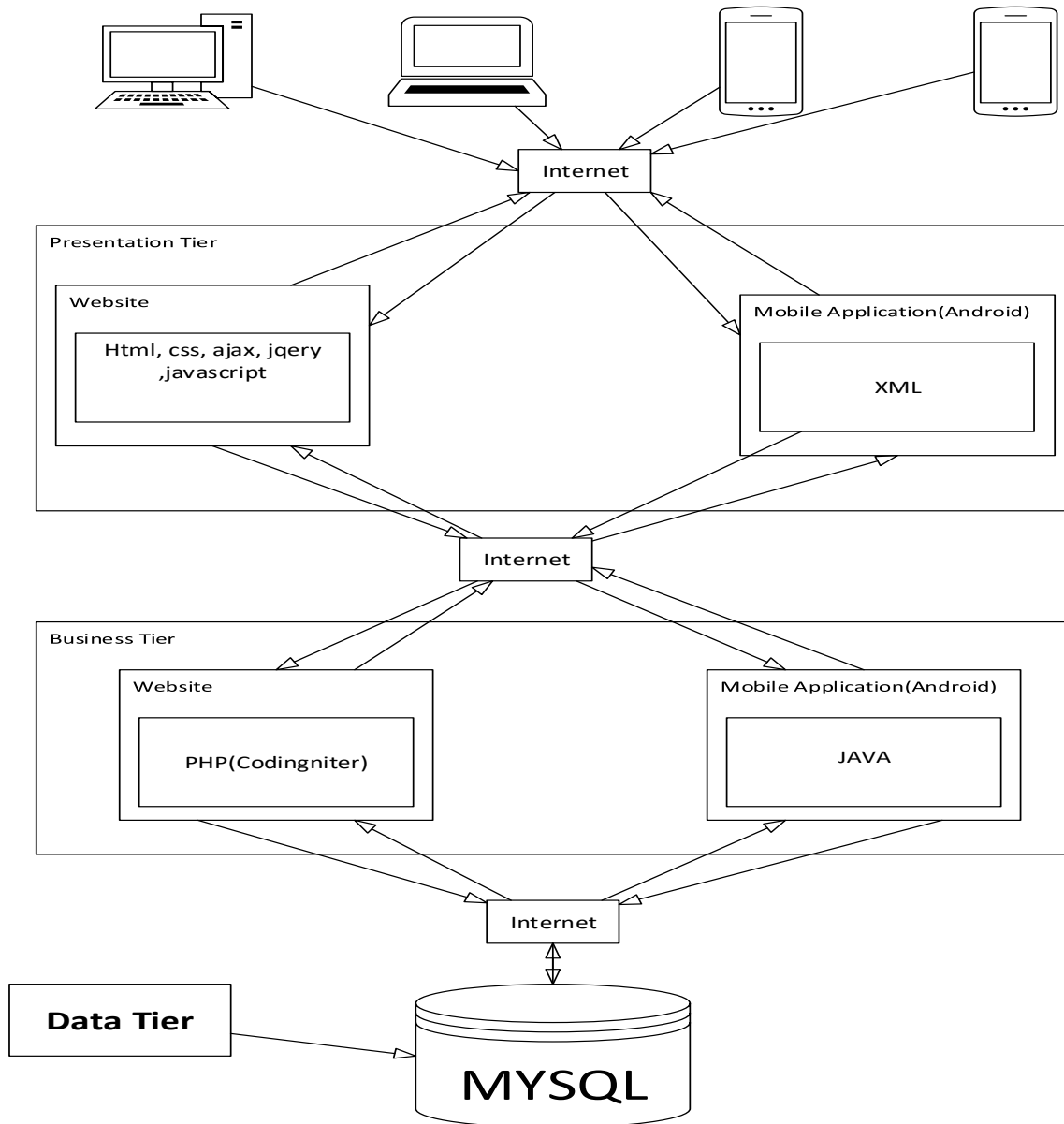


Figure 4.1 Architecture Diagram

4.2. Domain Model

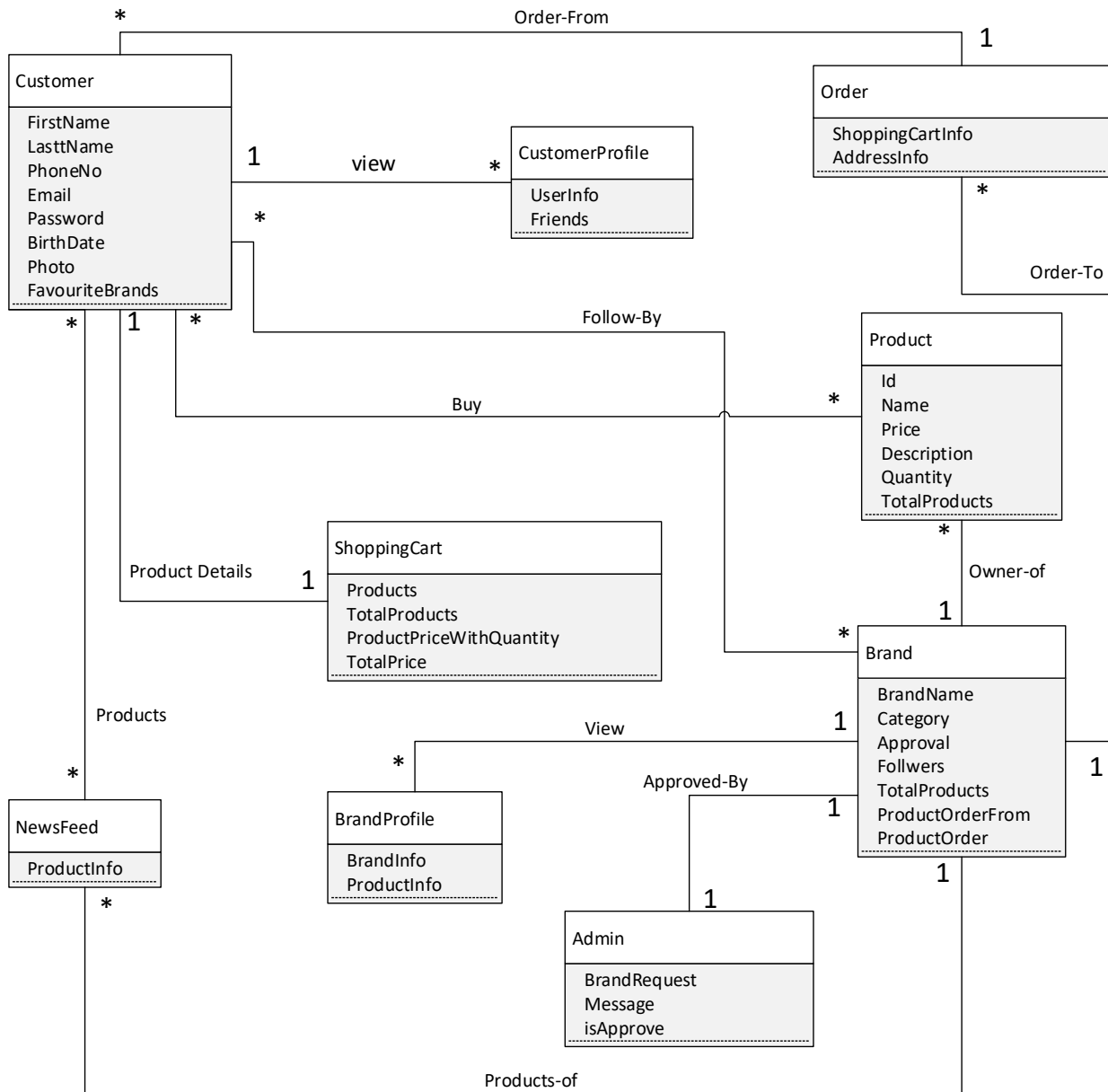


Figure 4.2 Domain Model

Data Dictionary:

Customer

Entity Name	Customer_Data			
Entity Description	It uses to store complete customer account information.			
Columns	Column Description	Data Type	Constraints	Reference
Email	It store email of customer	varchar(30)	Primary Key, Not null	
Password	It store Password of customer	varchar(8)	Not null	
Name	It store Name of customer	varchar(30)	Not null	
Phone No	It store Phone no of customer	Int(15)	Not null	
Photo	It store photo name of customer	varchar(30)	Not null	
Address	It store address of customer	Int(10)	Not null	Address

Table 4.1 Customer Table

Brand:

Entity Name	Brand_Data			
Entity Description	It uses to store complete Brand account information.			
Columns	Column Description	Data Type	Constraints	Reference
Email	It store email of customer	varchar(30)	Primary Key, Not null	
Password	It store Password of customer	varchar(8)	Not null	
Name	It store Name of customer	varchar(30)	Not null	
Phone No	It store Phone no of customer	Int(15)	Not null	
Photo	It store photo name of customer	varchar(30)	Not null	
Address	It store address of customer	Int(10)	Not null	Address
Alternative Email	It store alternative email of customer	varchar(30)	Not null	
Activation	It store true/false of customer	boolean	Not Null	
Proof	It store the name of proof	varchar(30)	Not Null	

Table 4.2 Brand Table

Product:

Entity Name	Product			
Entity Description	It uses to store complete Product information.			
Columns	Column Description	Data Type	Constraints	Reference
ID	It store email of customer	varchar(30)	Primary Key, Not null	
BEmail	It store Password of customer	varchar(8)	Not null, Foreingn key	Brand
Name	It store Name of customer	varchar(30)	Not null	
Quantity	It store Phone no of customer	Int(15)	Not null	
Photo	It store photo name of customer	varchar(30)	Not null	
Description	It store address of customer	Int(10)	Not null	
Date	It store alternative email of customer	Date	Not null	
Time	It store true/false of customer	Time	Not Null	

Table 4.3 Product Table

Order:

Entity Name	Order			
Entity Description	It uses to store complete Order information.			
Columns	Column Description	Data Type	Constraints	Reference
ID	It store id of customer	varchar(30)	Not null	
CEmail	It store email of customer	varchar(8)	Not null, Foreingn key, Primary Key	Customer
CartID	It store id of cart	varchar(30)	Not null, Foreingn key, Primary Key	Cart
Date	It store date Of order	Date	Not null	
Time	It store time Of order	Time	Not Null	

Table 4.4 Order Table

Cart:

Entity Name	Cart			
Entity Description	It uses to store complete customer's shopping cart information.			
Columns	Column Description	Data Type	Constraints	Reference
ID	It store email of customer	varchar(30)	Not null	

CEmail	It store Password of customer	varchar(8)	Not null, Primary Key, Foreingn key	Customer
PID	It store Name of customer	varchar(30)	Not null, Primary Key, Foreingn key	Product
Quantity	It store Phone no of customer	Int(15)	Not null	

Table 4.5 cart Table

Likes:

Entity Name	Pro_Likes			
Entity Description	It uses to store complete information of likes that done form customers on product.			
Columns	Column Description	Data Type	Constraints	Reference
PID	It store product id of like on product by customer	varchar(30)	Primary Key, Not null, Foreingn key	Product
BEmail	It store email of brand	varchar(30)	Not null, Primary Key, Foreingn key	Customer
Date	It store date of job started	Date	Not null,	
Time	It store date of job started	Time	Not null	

Table 4.6 Likes Table

Share:

Entity Name	Timeline			
Entity Description	It uses to store complete information of share that done form customers on product.			
Columns	Column Description	Data Type	Constraints	Reference
PID	It store product id of like on product by customer	varchar(30)	Primary Key, Not null, Foreingn key	Product
Userid	It store email of brand	varchar(30)	Not null, Primary Key, Foreingn key	Customer
Date	It store date of job started	Date	Not null,	
Time	It store date of job started	Time	Not null	

Table 4.7 Share Table

Comments:

Entity Name	Pro_Comments			
Entity Description	It uses to store complete information of comments that done form customers on product.			
Columns	Column Description	Data Type	Constraints	Reference
PID	It store product id of like on product by customer	varchar(30)	Primary Key, Not null, Foreingn key	Product
BEmail	It store email of brand	varchar(30)	Not null, Primary Key, Foreingn key	Customer
Date	It store date of job started	Date	Not null,	
Time	It store date of job started	Time	Not null	
Comment	It store comment of customer from product	Varchar(50)	Not Null	

Table 4.8 Comment table

Follow:

Entity Name	Follow			
Entity Description	It uses to store complete information of friends of customer.			
Columns	Column Description	Data Type	Constraints	Reference
id	It store customer email that accept request	varchar(30)	Primary Key, Not null, Foreingn key	Customer
userid	It store customer email that send request	varchar(30)	Not null, Primary Key, Foreingn key	Customer

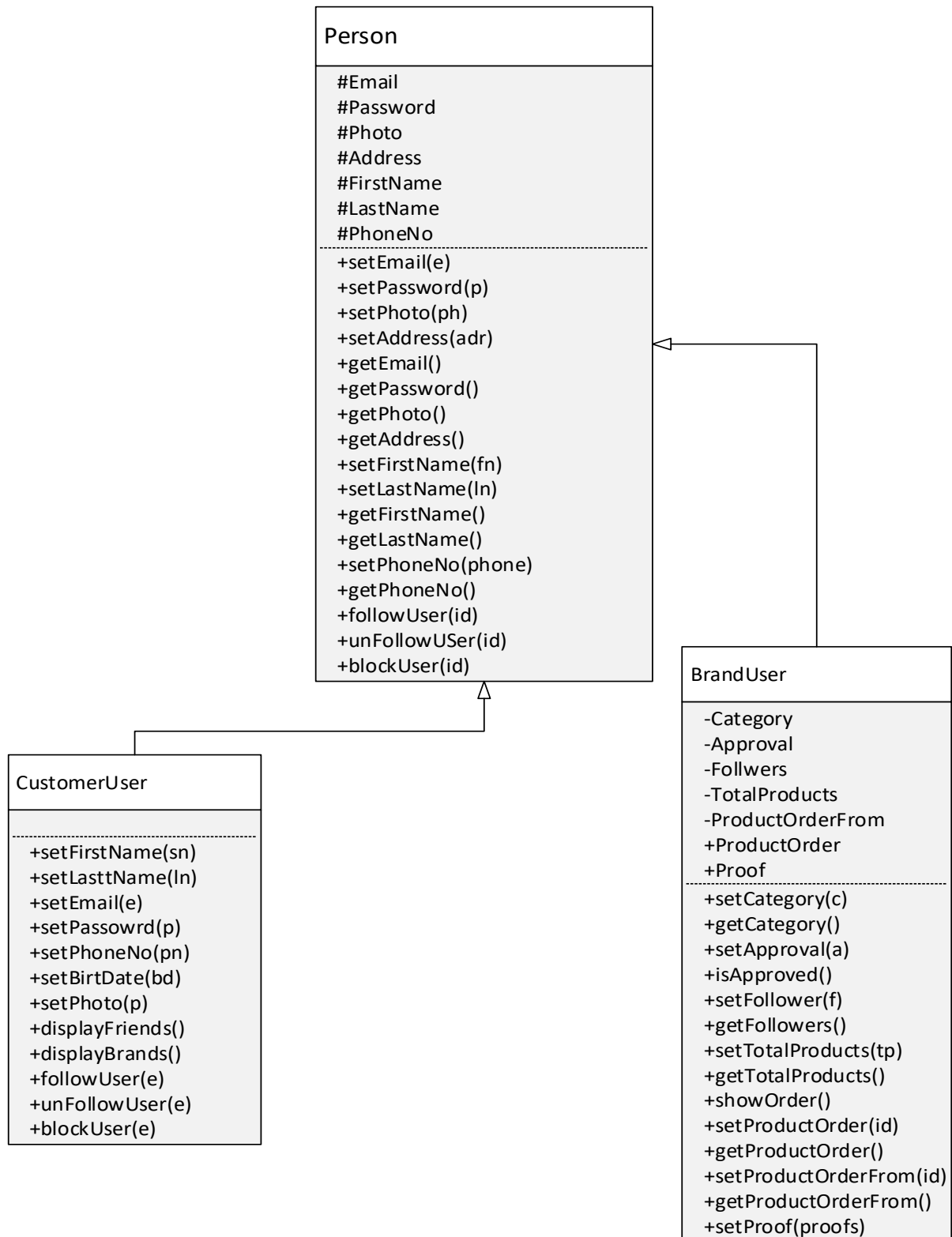
Table 4.9 Friend Table

Notification:

Entity Name	CNotification			
Entity Description	It uses to store complete information of total notifications come to customer			
Columns	Column Description	Data Type	Constraints	Reference
Userid	It store email of customer	varchar(30)	Primary Key, Not null, Foreingn key	Customer
Date	It store date of job started	Date	Not null,	
Time	It store date of job started	Time	Not null	

Table 4.10 CNotification Table

4.4. Class Diagram



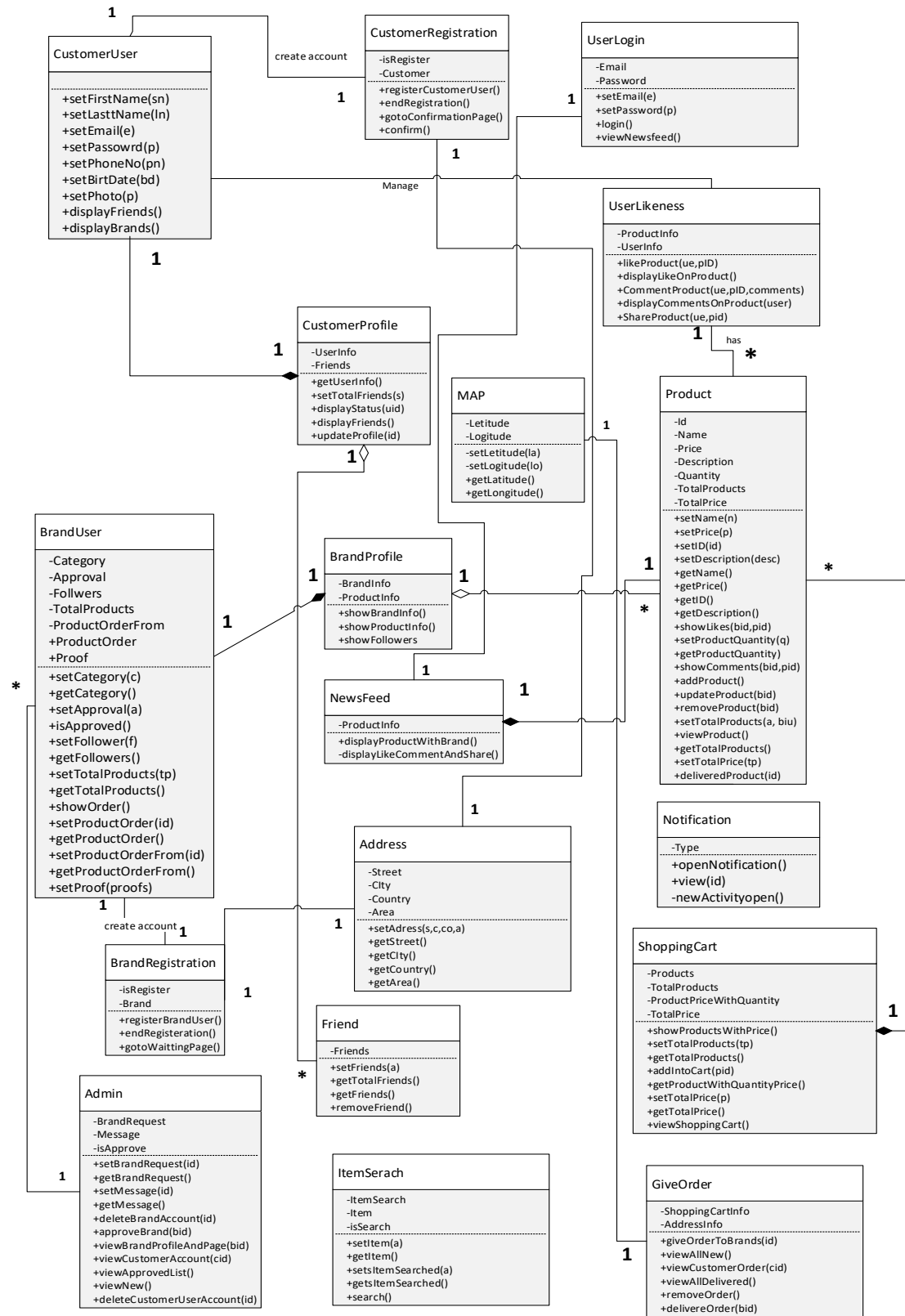


Figure 4.4 Class Diagram

4.5. Sequence / Collaboration Diagram

Register Brand Account:

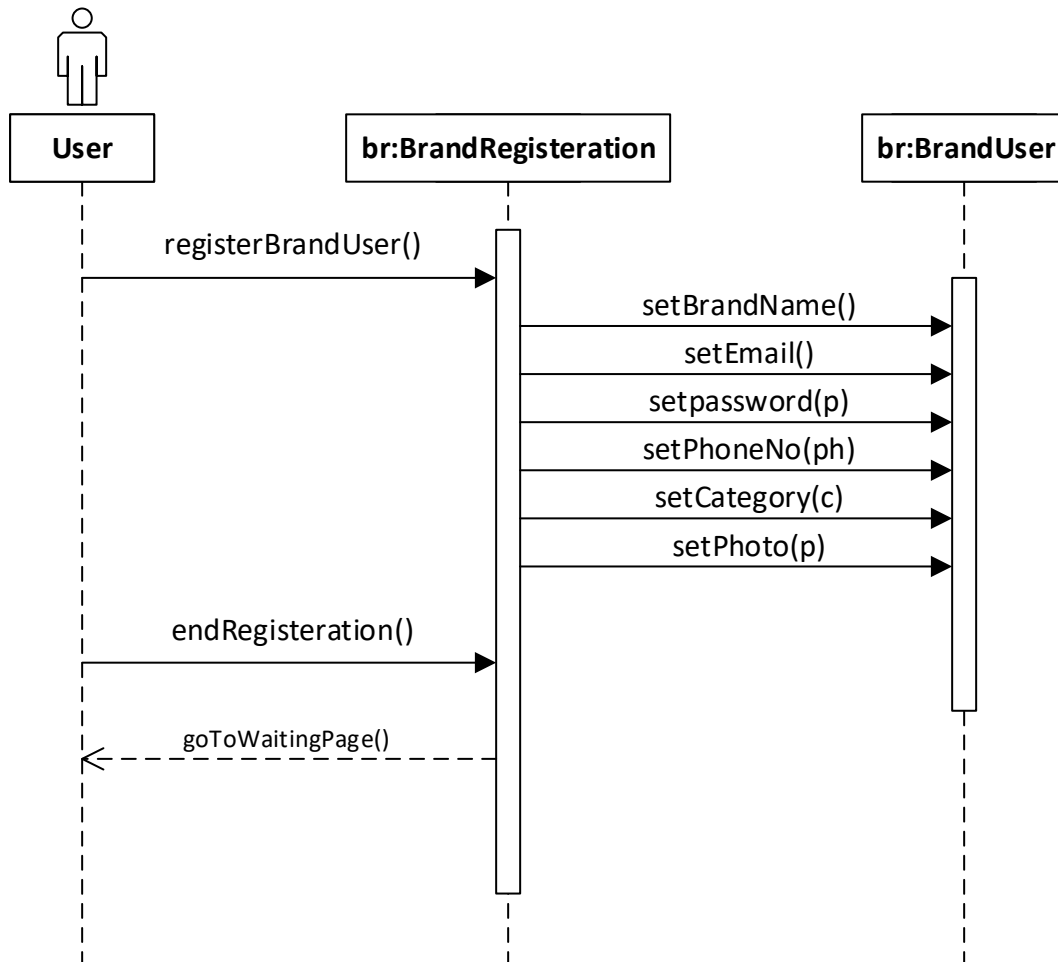


Figure 4.5 Register Brand Account

Approve Brand Request

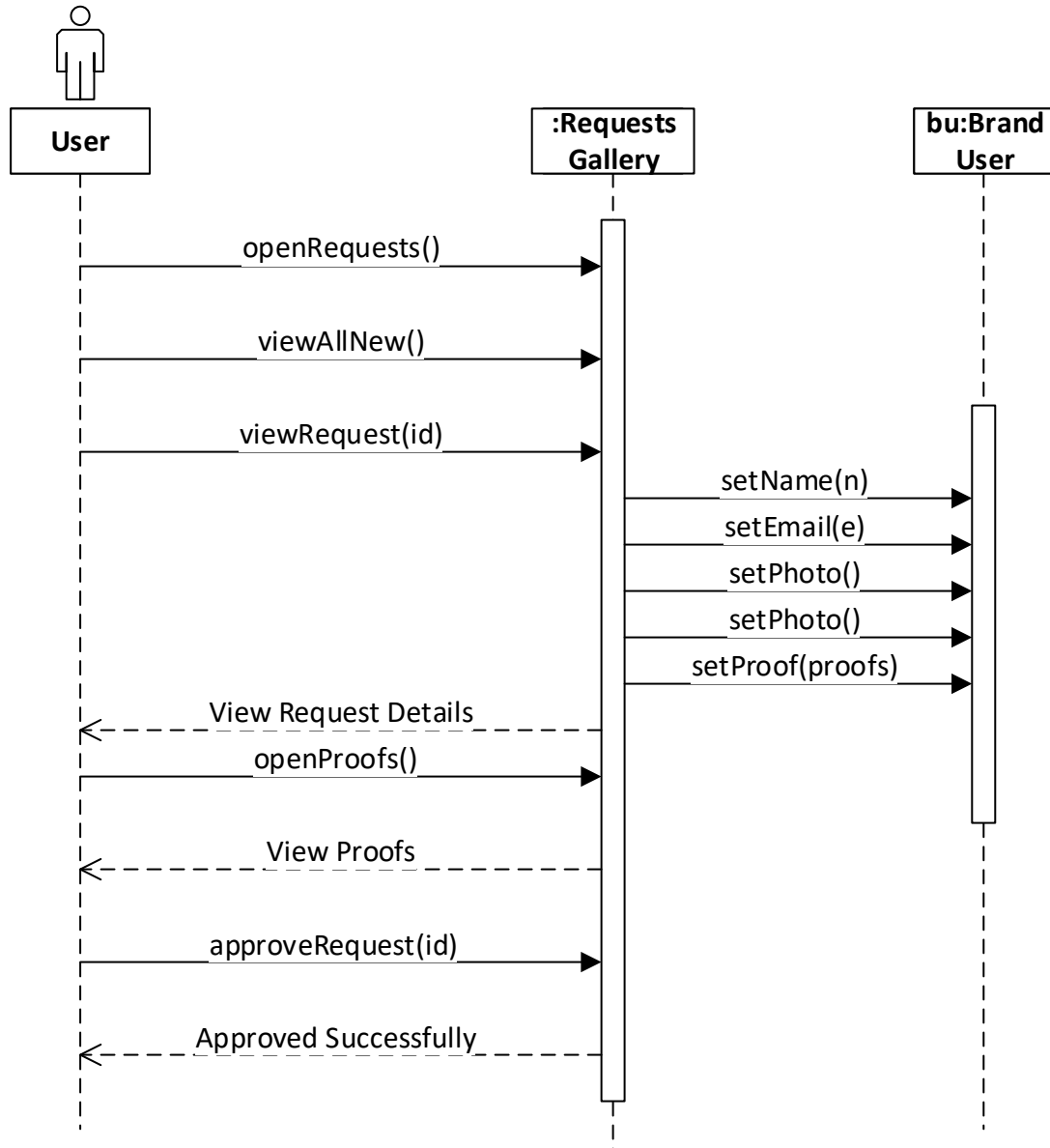


Figure 4.6 Approve Brand Request

Add Product

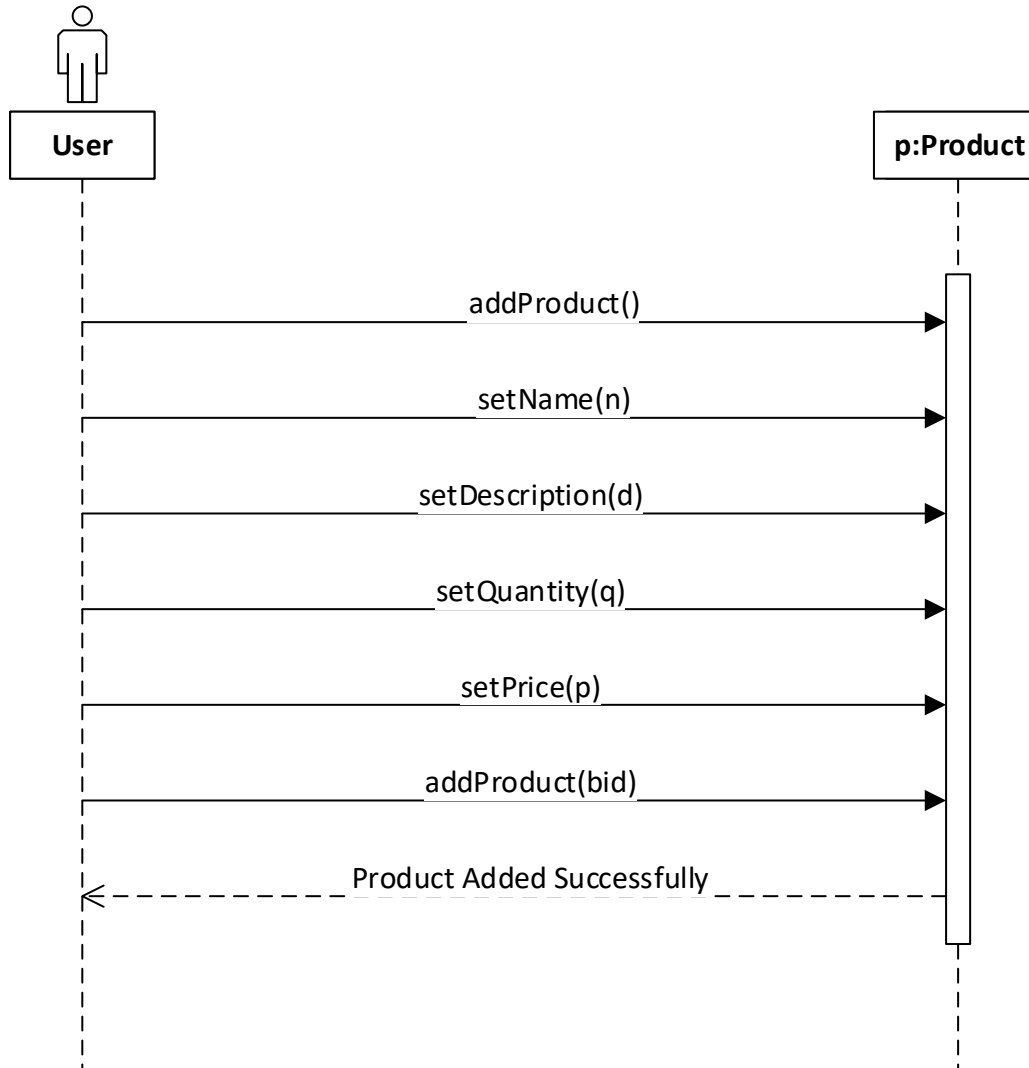


Figure 4.7 Add Product

Brand User Profile

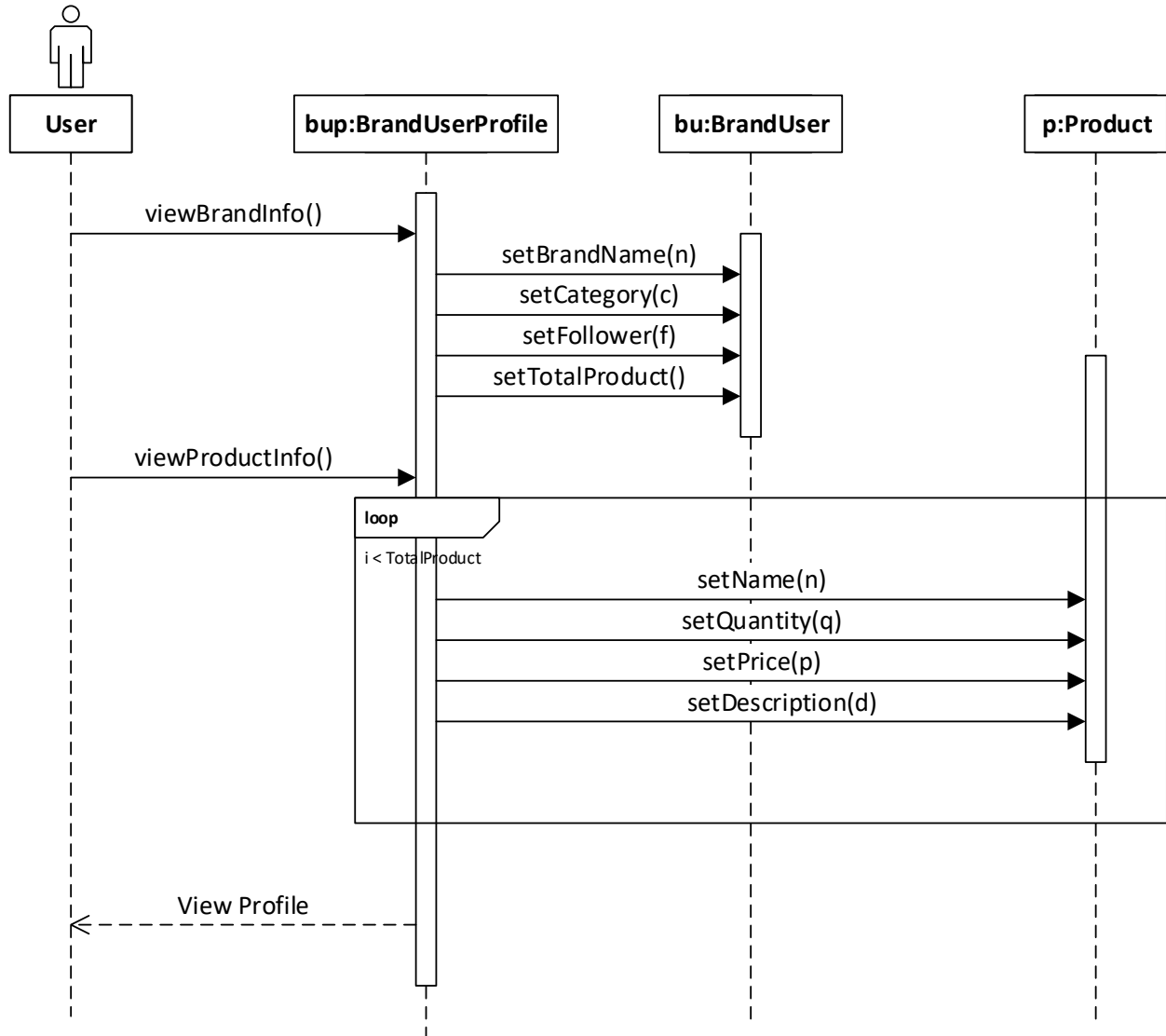


Figure 4.8 Brand User Profile

Customer Registration

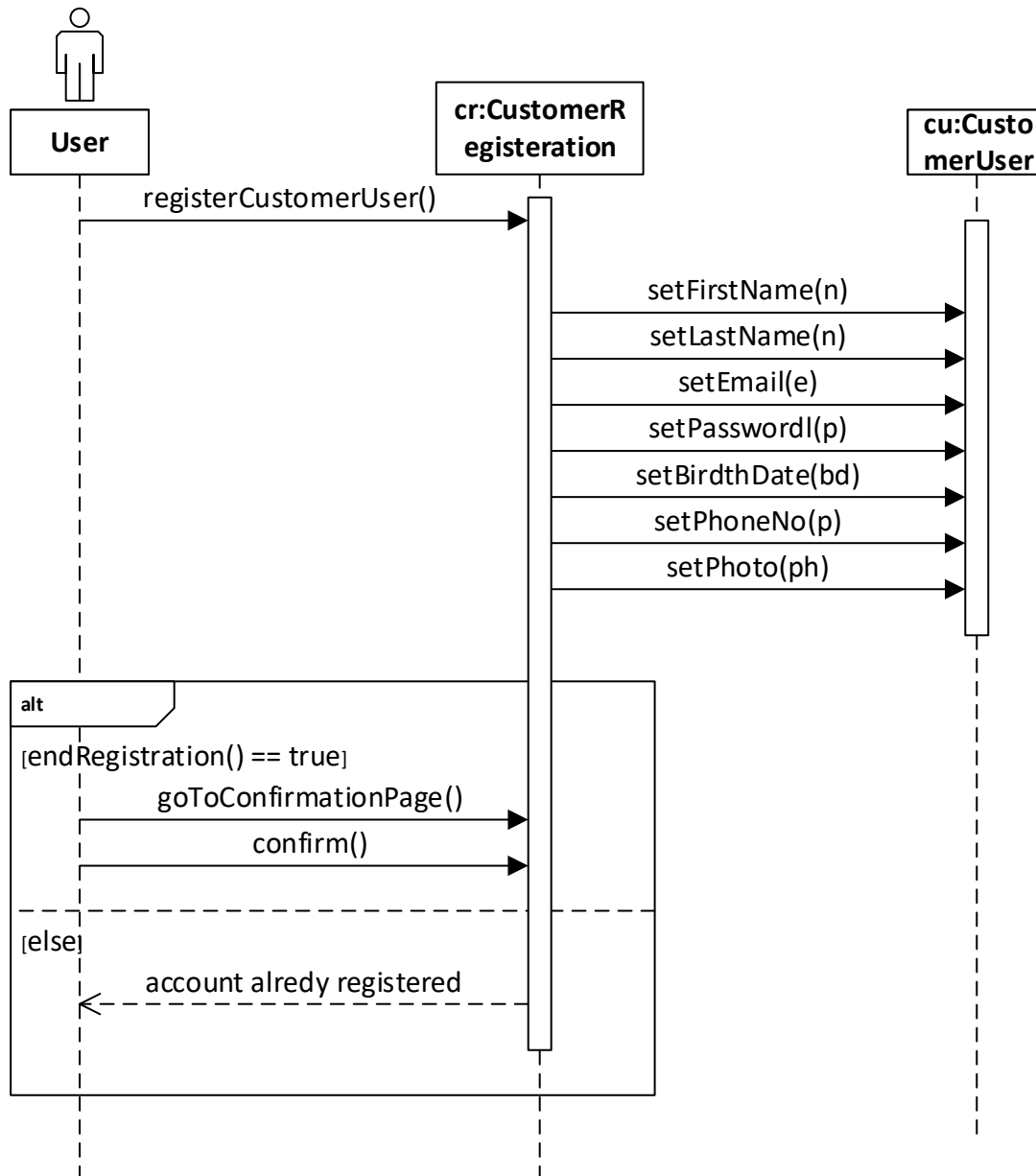


Figure 4.9 Customer Registration

Follow User

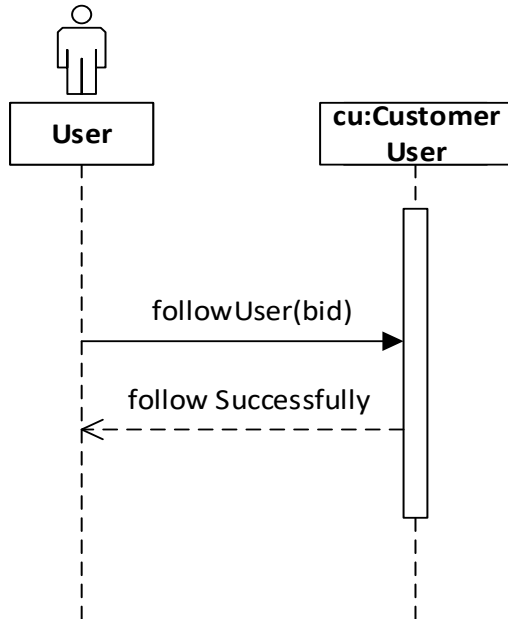


Figure 4.10 Follow User

Unfollow User

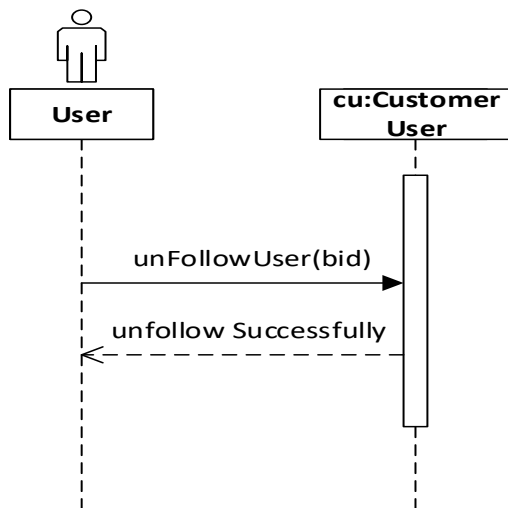


Figure 4.11 Unfollow User

Block User

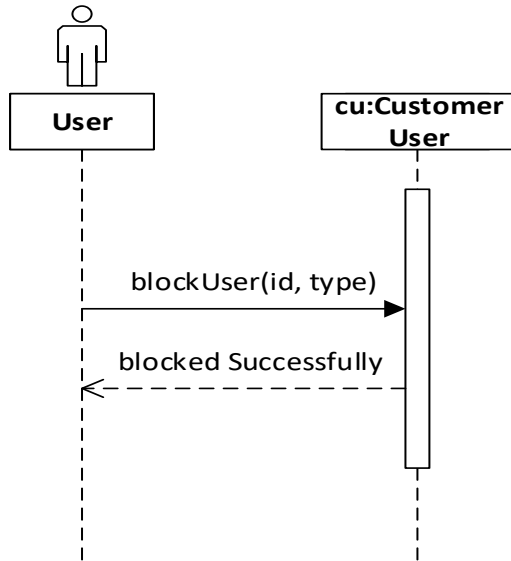


Figure 4.12 Block User

Like, Share, and Comment on Product

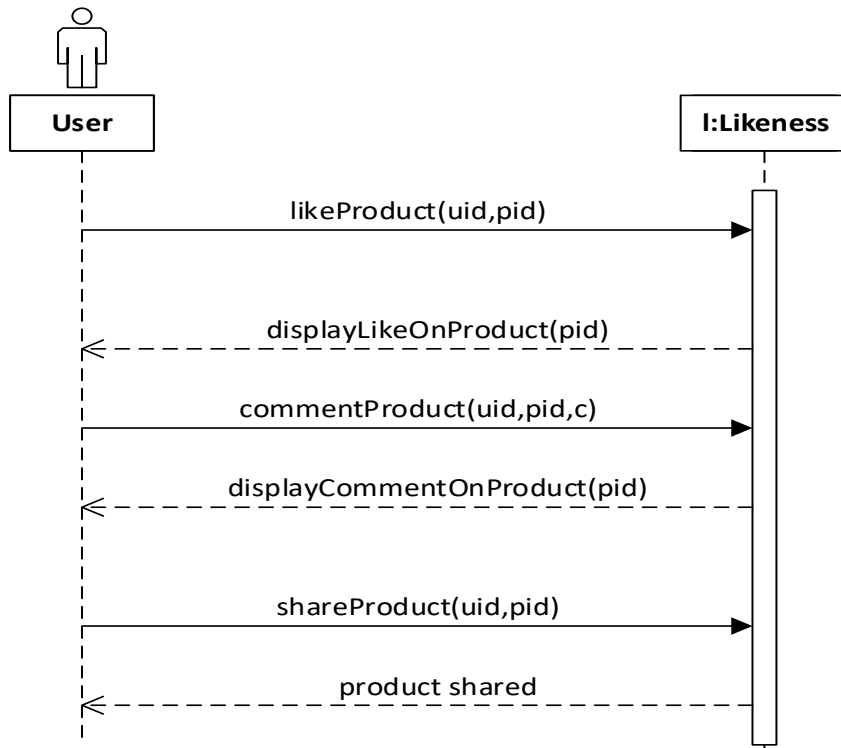


Figure 4.13 Block User

Login

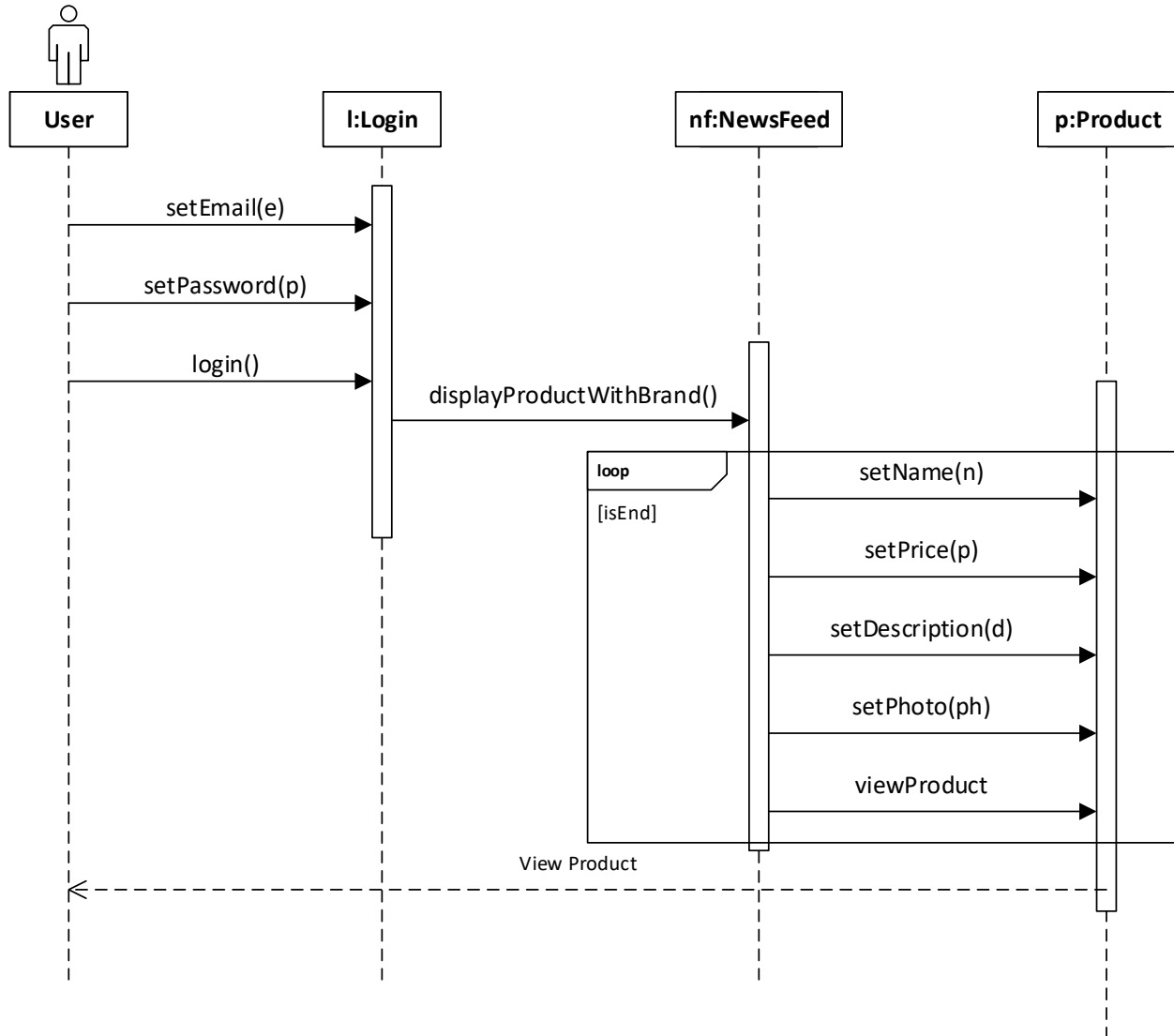


Figure 4.14 Login

View New Order:

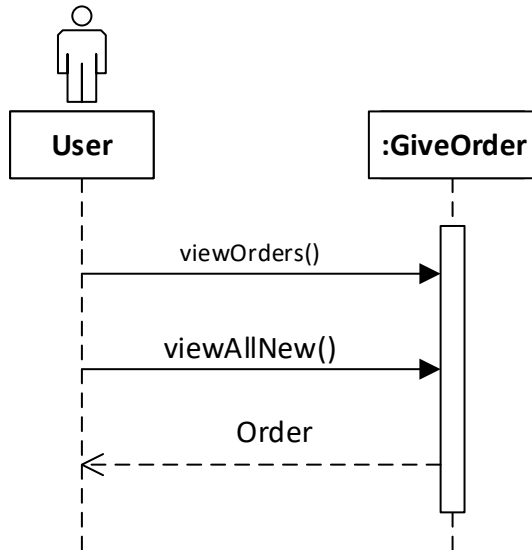


Figure 4.15 View New Order

View Delivered Order:

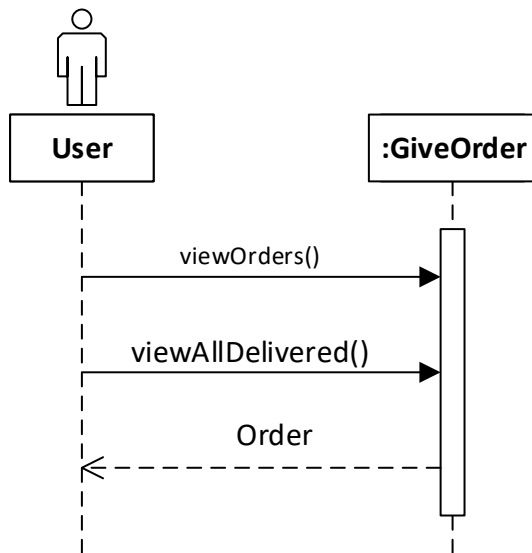


Figure 4.16 View Delivered Order

Deliver Order

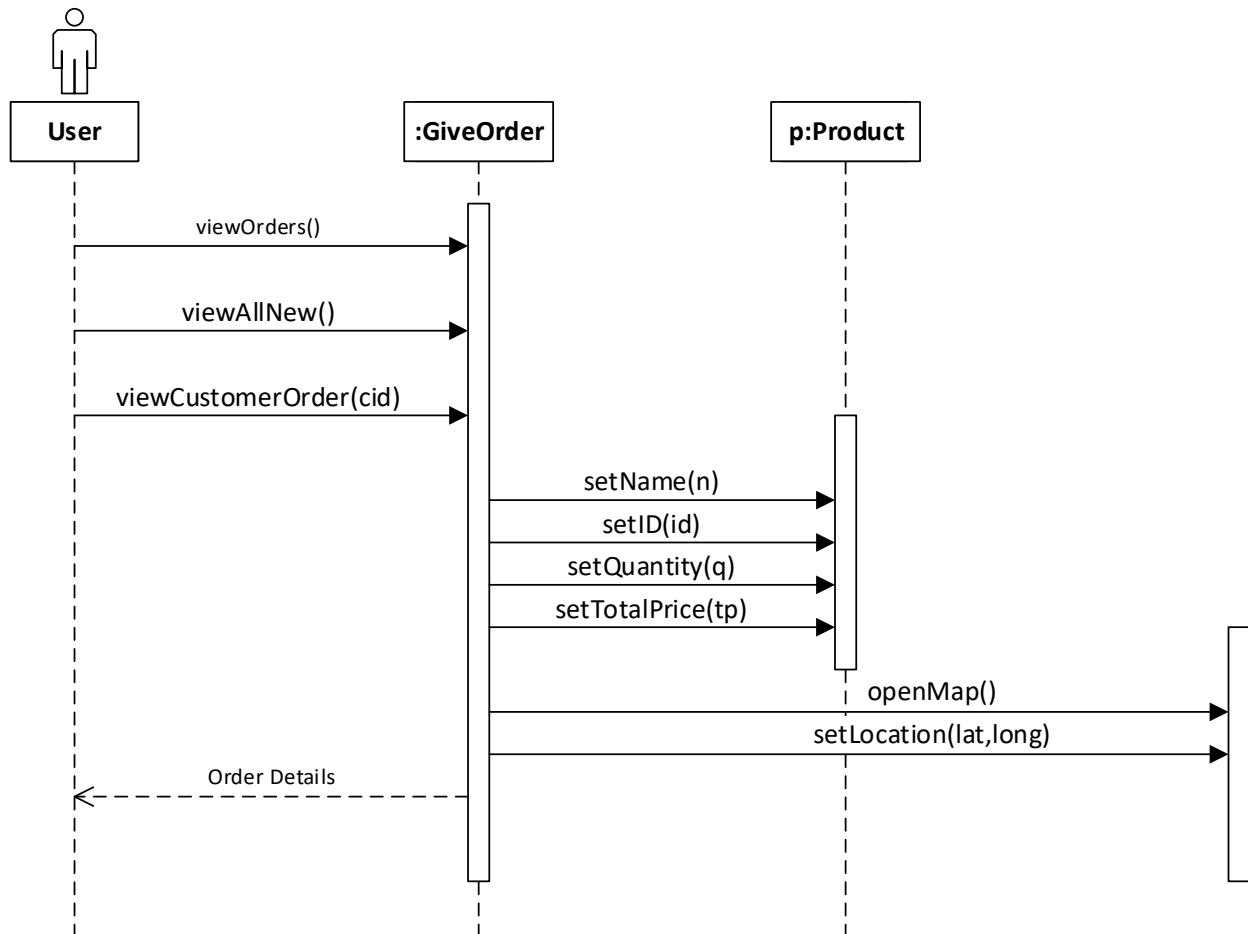


Figure 4.17 Deliver Order

Remove Order

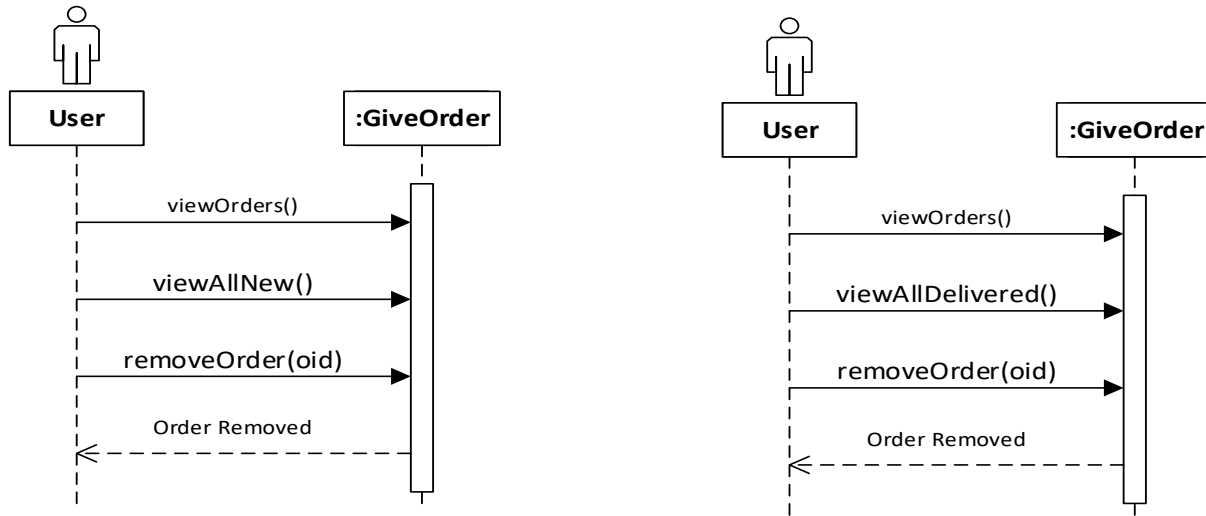


Figure 4.18 Remove Orders

Notification

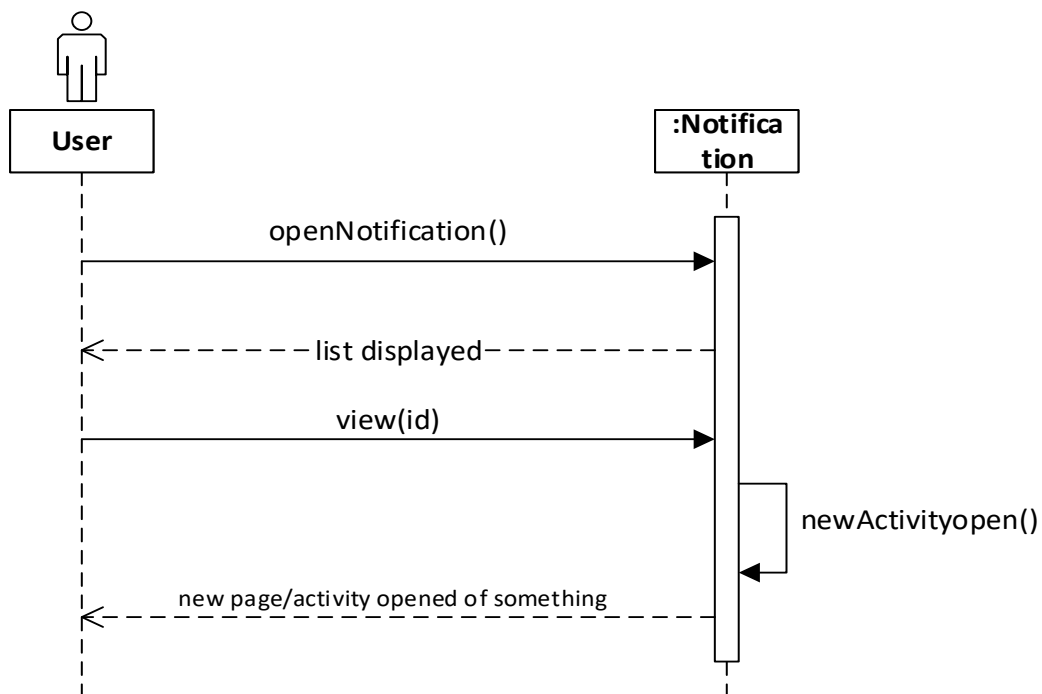


Figure 4.19 Notification

Process Shopping Cart

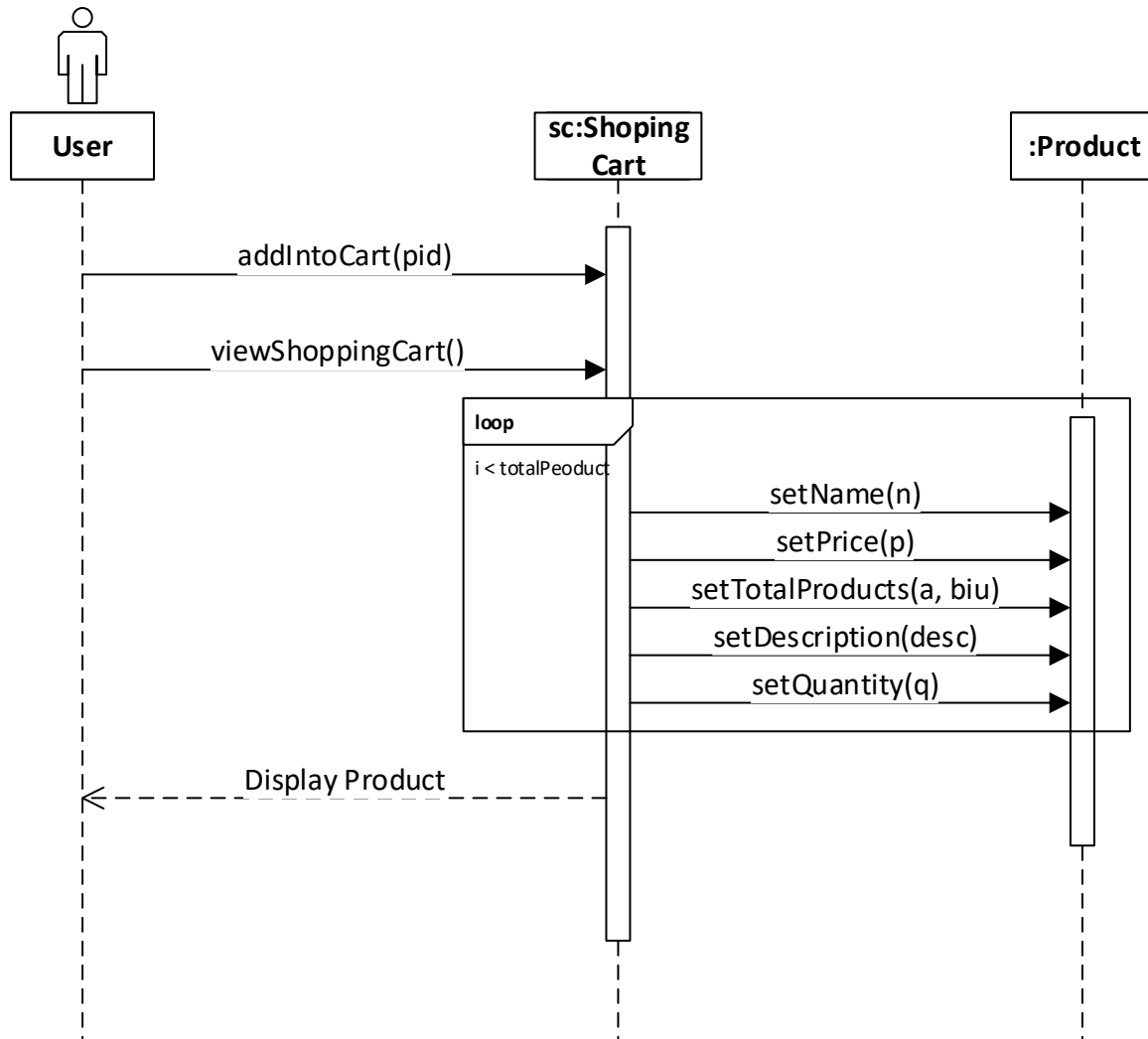


Figure 4.20 Process Shopping Cart

Rating Product:

1) Brand mark the order as delivered.

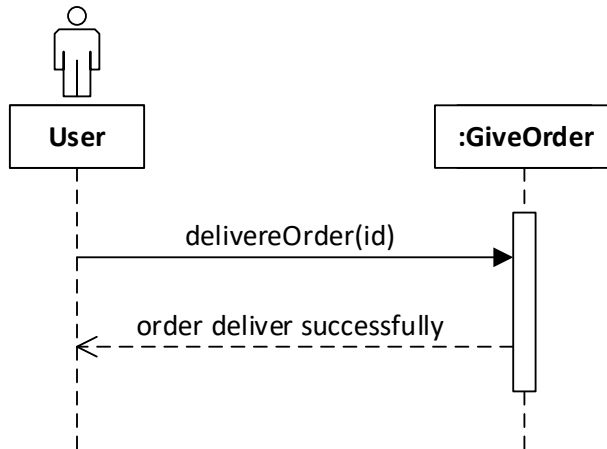


Figure 4.21 Rating Product

2) Then Customer get Notification

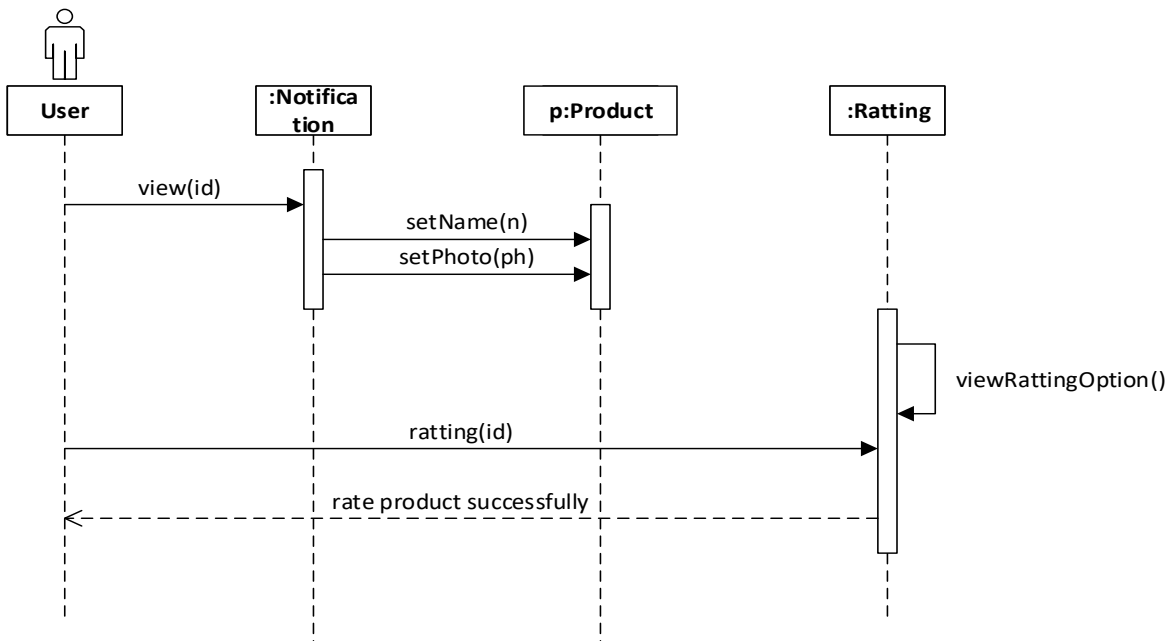


Figure 4.22 Rating Product

Search:

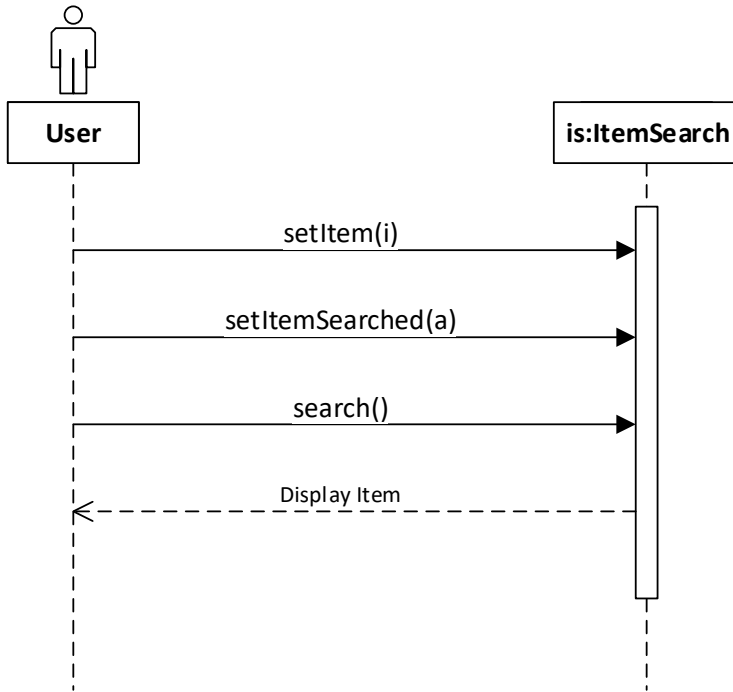


Figure 4.23 Search

Give Order

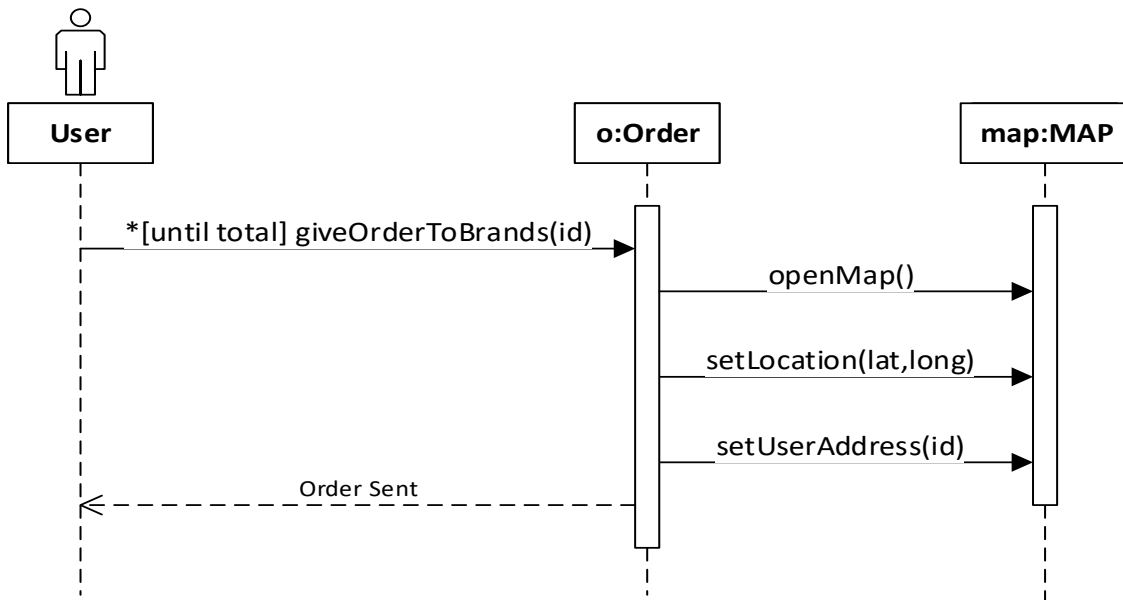


Figure 4.24 Give Order

Update Product:

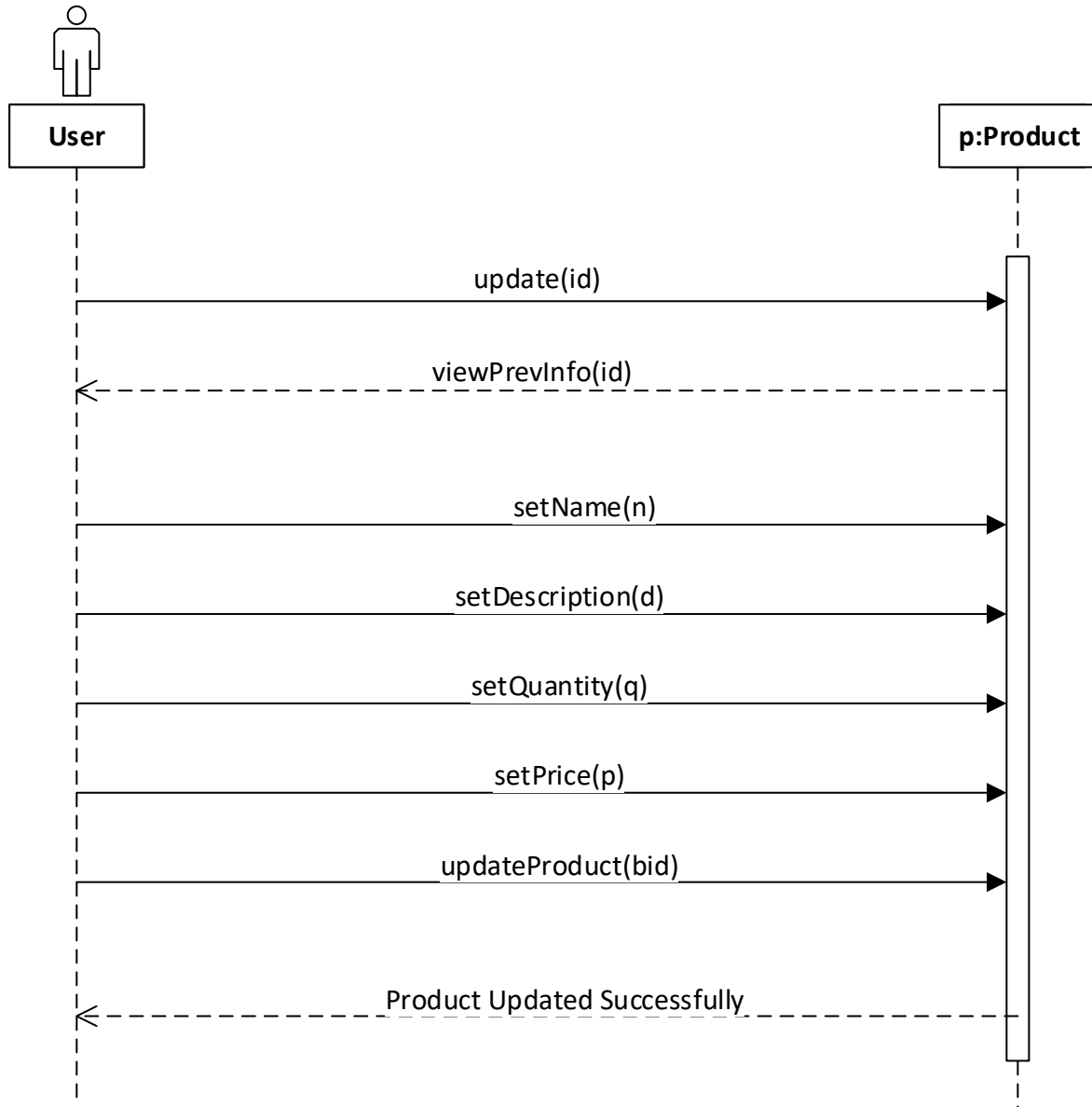


Figure 4.26 Update Profile

Remove Product

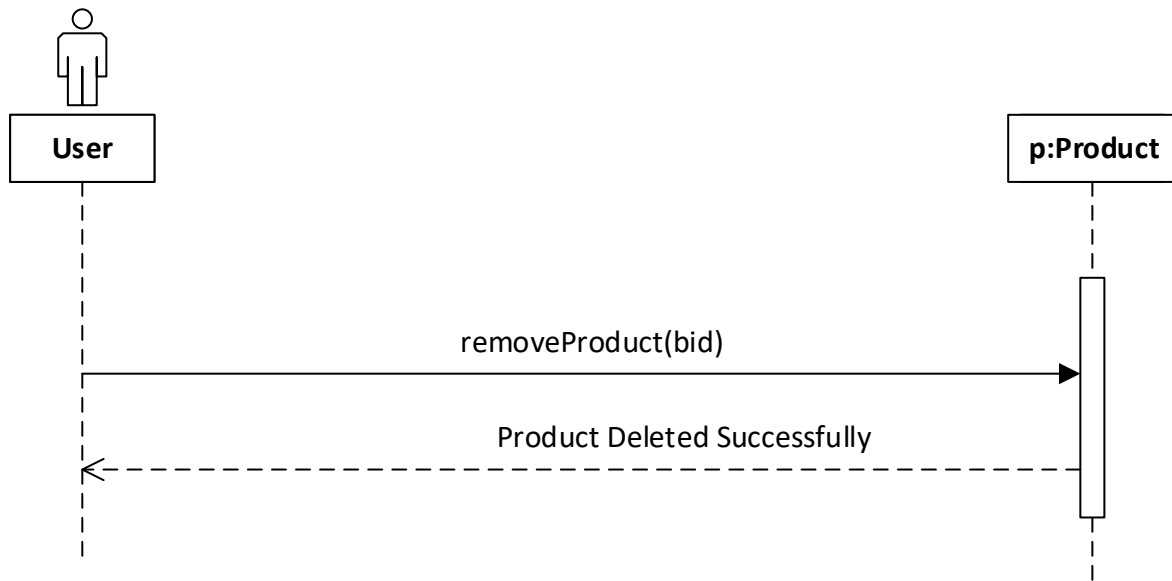


Figure 4.27 Remove Product

Customer Profile

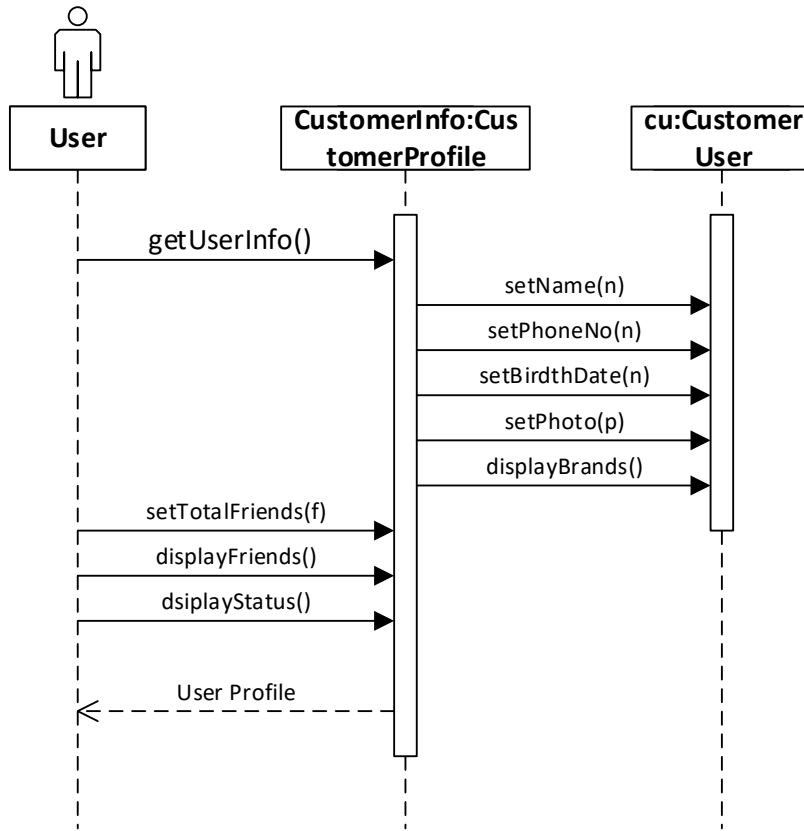


Figure 4.29 Customer Profile

Update Profile

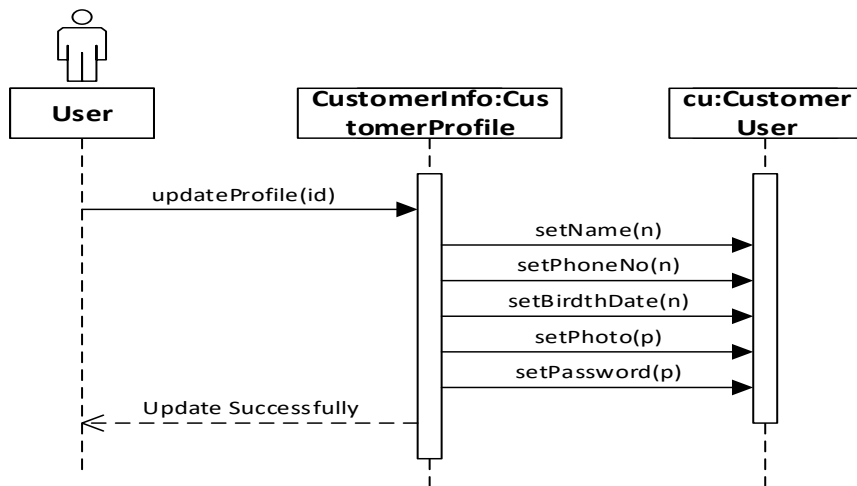


Figure 4.30 Update Profile

4.6. Operation contracts

Operation Contracts

User

1) Login Account

Operation: setEmail(e)

Cross References:

Use case: Login Account

Scenario: Login Account

Pre-Condition: none.

Post-Condition :

- ul.Email was initialized.

Operation: setPassword (p)

Cross References:

Use case: Login Account

Scenario: Login Account

Pre-Condition: none.

Post-Condition :

- ul.Password was initialized.

Operation: login ()

Cross References:

Use case: Login Account

Scenario: Login Account

Pre-Condition: login account is underway.

Post-Condition :

NewsFeed instance nf was created.

cup associated with ProductInfo:Product.

Operation: displayProductWithBrand() ()

Cross References:

Use case: Login Account

Scenario: Login Account

Pre-Condition: Customer User was registered.

Post-Condition:

The Attributes of BrandInfo was initialized.

The Attributes of ProductInfo was initialized.

Product instance p was created.

Operation: setName(a)

Cross References:

Use case: login Account

Scenario: login Account

Pre-Condition: add product is underway.

Post-Condition:

p.Name was initialized.

Operation: setDescription(a)

Cross References:

Use case: login Account

Scenario: login Account

Pre-Condition: add product is underway.

Post-Condition:

p.Description was initialized.

Operation: setPrice(a)

Cross References:

Use case: login Account

Scenario: login Account

Pre-Condition: add product is underway.

Post-Condition:

p.Price was initialized.

Operation: setQuantity(a)

Cross References:

Use case: login Account

Scenario: login Account

Pre-Condition: add product is underway.

Post-Condition:

p.Quantity was initialized.

Operation: addProduct()

Cross References:

Use case: login Account

Scenario: login Account

Pre-Condition: add product is underway.

Post-Condition: p was associated with BrandUser.

Operation: viewProduct()

Cross References:

Use case: login Account

Scenario: login Account

Pre-Condition: add product is underway.

Post-Condition: none.

2) Register Account

Operation: registerCustomerUser()

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: none.

Post-Condition:

- CustomerUser instance cu was created.

Operation: setFirstName(a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.FirstName was initialized.

Operation: setLastName(a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.LastName was initialized.

Operation: setEmail (a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.Email was initialized.

Operation: setPassword (a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.Password was initialized.

Operation: setBirthDate(a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.BirthDate was initialized.

Operation: setPhoneNo (a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.PhoneNo was initialized.

Operation: setPassword (a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.Password was initialized.

Operation: setPhoto (a)

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cu.Photo was initialized.

Operation: endRegistration()

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: Customer User Registration is underway.

Post-Condition:

cr.isRegister was initialized.

Operation: gotoConfirmPage()

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: registration is underway.

Post-Condition: none

Operation: confirm()

Cross References:

Use case: Register Account

Scenario: Register Account

Pre-Condition: registration is underway.

Post-Condition: none

3) View Profile

Operation: getUserInfo()

Cross References:

Use case: View Profile

Scenario: View Profile

Pre-Condition: open profile is underway.

Post-Condition:

cp was associated with cu:CustomerUser.

Attributes of cu were initialized.

Operation: displayFriends()

Cross References:

Use case: View Profile

Scenario: View Profile

Pre-Condition: none.

Post-Condition:

Friend instance f was created.

Attributes of f:Friend were initialized.

4) Like Product

Operation: likeProduct(ue,pid)

Cross References:

Use case: Like Product

Scenario: Like Product

Pre-Condition: Like product is underway.

Post-Condition: none

Operation: displayLikeOnProduct(pid)

Cross References:

Use case: Like Product

Scenario: Like Product

Pre-Condition: Like product is underway.

Post-Condition: none.

5) Process Comment

Operation: commentProduct(ue,pid,comments)

Cross References:

Use case: Process Comment

Scenario: Process Comment

Pre-Condition: comment on product is underway.

Post-Condition: none

Operation: displayCommentsOnProduct(pid)

Cross References:

Use case: Process Comment

Scenario: Process Comment

Pre-Condition: Like product is underway.

Post-Condition: none.

6) Share Product

Operation: shareProduct(ue,pid)

Cross References:

Use case: Share Product

Scenario: Share Product

Pre-Condition: share product is underway.

Post-Condition: none

7) Process Shopping Cart

Operation: addInToCart(pid)

Cross References:

Use case: Process Shopping Cart

Scenario: Process Shopping Cart

Pre-Condition: none.

Post-Condition: none.

Operation: viewShoppingCart()

Cross References:

Use case: Process Shopping Cart

Scenario: Process Shopping Cart

Pre-Condition: none.

Post-Condition:

P instance was created of Product class.

Operation: setName(n)

Cross References:

Use case: Process Shopping Cart

Scenario: Process Shopping Cart

Pre-Condition: none.

Post-Condition:

p.Name was initialized.

Operation: setPrice(p)

Cross References:

Use case: Process Shopping Cart

Scenario: Process Shopping Cart

Pre-Condition: none.

Post-Condition:

p.Price was initialized.

Operation: setQuantity(q)

Cross References:

Use case: Process Shopping Cart

Scenario: Process Shopping Cart

Pre-Condition: none.

Post-Condition:

p.Quantity was initialized.

Operation: setDescription(d)

Cross References:

Use case: Process Shopping Cart

Scenario: Process Shopping Cart

Pre-Condition: none.

Post-Condition:

p.Description was initialized.

8) Give Order

Operation: giveOrder()

Cross References:

Use case: Give Order

Scenario: Give Order

Pre-Condition: none.

Post-Condition:

GiveOrder instance go was created.

go was associated with ShoppingCartInfo:ShoppingCart.

go was associated with AddressInfo:Address.

Operation: giveOrderToBrands(cid)

Cross References:

Use case: Give Order

Scenario: Give Order

Pre-Condition: none.

Post-Condition:

map of 'MAP was created.

Operation: setLatitude(lat)

Cross References:

Use case: Give Order

Scenario: Give Order

Pre-Condition: none.

Post-Condition:

map.Latitude was intialized.

Operation: setLongitude(long)

Cross References:

Use case: Give Order

Scenario: Give Order

Pre-Condition: none.

Post-Condition:

map.Longitude was intialized.

9) Search Item

Operation: setItem(a)

Cross References:

Use case: Search Item

Scenario: Search Item

Pre-Condition: search is underway.

Post-Condition:

Is.Item was initialized.

Operation: setItemSearched(a)

Cross References:

Use case: Search Item

Scenario: Search Item

Pre-Condition: search is underway.

Post-Condition:

Is.ItemSearch was initialized.

Operation: search()

Cross References:

Use case: Search Item

Scenario: Search Item

Pre-Condition: search is underway.

Post-Condition:

Is.isSearch was changed to true.

Brand

10) Register Page

Operation: registerBrandUser()

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: none.

Post-Condition:

- BrandUser instance bu was created.

Operation: setBrandName(a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Customer User Registration is underway.

Post-Condition:

bu.FirstName was initialized.

Operation: setEmail (a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Brand User Registration is underway.

Post-Condition:

bu.Email was initialized.

Operation: setPassword (a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Brand User Registration is underway.

Post-Condition:

bu.Password was initialized.

Operation: setPhoneNo (a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Brand User Registration is underway.

Post-Condition: bu.PhoneNo was initialized.

Operation: setPassword (a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Brand User Registration is underway.

Post-Condition:

bu.Password was initialized.

Operation: setPhoto (a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Customer User Registration is underway.

Post-Condition:

bu.Photo was initialized.

Operation: setCategory(a)

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Customer User Registration is underway.

Post-Condition:

bu.Category was initialized.

Operation: endRegistration()

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Brand User Registration is underway.

Post-Condition:

br.isRegister was initialized.

Operation: gotoWaitingPage()

Cross References:

Use case: Register Page

Scenario: Register Page

Pre-Condition: Brand User User was registered.

Post-Condition:

BrandUser's association was broken.

11) Add Product

Operation: addProduct()

Cross References:

Use case: Add Product

Scenario: Add Product

Pre-Condition: none.

Post-Condition:

Product instance p was created.

Operation: setName(a)

Cross References:

Use case: Add Product

Scenario: Add Product

Pre-Condition: add product is underway.

Post-Condition:

p.Name was initialized.

Operation: setDescription(a)

Cross References:

Use case: Add Product

Scenario: Add Product

Pre-Condition: add product is underway.

Post-Condition:

p.Description was initialized.

Operation: setPrice(a)

Cross References:

Use case: Add Product

Scenario: Add Product

Pre-Condition: add product is underway.

Post-Condition:

p.Price was initialized.

Operation: setQuantity(a)

Cross References:

Use case: Add Product

Scenario: Add Product

Pre-Condition: add product is underway.

Post-Condition:

p.Quantity was initialized.

Operation: addProduct()

Cross References:

Use case: Add Product

Scenario: Add Product

Pre-Condition: add product is underway.

Post-Condition: p was associated with BrandUser.

12) Update Product

Operation: update (id)

Cross References:

Use case: Update Product

Scenario: Update Product

Pre-Condition: none.

Post-Condition:

Product instance p was created.

Operation: setName(a)

Cross References:

Use case: Update Product

Scenario: Update Product

Pre-Condition: add product is underway.

Post-Condition:

p.Name was initialized.

Operation: setDescription(a)

Cross References:

Use case: Update Product

Scenario: Update Product

Pre-Condition: add product is underway.

Post-Condition:

p.Description was initialized.

Operation: setPrice(a)

Cross References:

Use case: Update Product

Scenario: Update Product

Pre-Condition: add product is underway.

Post-Condition:

p.Price was initialized.

Operation: setQuantity(a)

Cross References:

Use case: Update Product

Scenario: Update Product

Pre-Condition: add product is underway.

Post-Condition:

p.Quantity was initialized.

Operation: updateProduct(pid)

Cross References:

Use case: Update Product

Scenario: Update Product

Pre-Condition: add product is underway.

Post-Condition:

p was associated with BrandUser.

p.Id was initialized.

13) Remove Product

Operation: removeProduct(pid)

Cross References:

Use case: Remove Product

Scenario: Remove Product

Pre-Condition: remove product is underway.

Post-Condition:

p was associated with BrandUser.

p.Id was initialized.

14) View Brand Profile

Operation: viewBrandInfo()

Cross References:

Use case: View Brand Profile

Scenario: View Brand Profile

Pre-Condition: open profile is underway.

Post-Condition:

BrandUser instance bu was created.

bup was associated with bu:BrandUser.

bu.Name was initialized.

Bu.category was initialized.

Bu.TotalProducts was initialized.
Bu.Followers was initialized.
p instance of Product was initialized.
p.name was initialized.
p.Quantity was initialized.
p.Description was initialized.
p.Name was initialized.

Admin

15) View Profile

Operation: openBrandProfile()

Cross References:

Use case: View Profile

Scenario: View Profile

Pre-Condition: none.

Post-Condition:

BrandUserProfile instance bup was created.

Operation: viewBrandProfileAndPage()

Cross References:

Use case: View Profile

Scenario: View Profile

Pre-Condition: open profile is underway.

Post-Condition:

BrandUser instance bu was created.

bup was associated with bu:BrandUser.

Attributes of bu were initialized.

Operation: showProductInfo()

Cross References:

Use case: view Profile

Scenario: view Profile

Pre-Condition: View Profile is underway.

Post-Condition:

Product instance p was created.

p.TotalProducts was initialized.

Operation: showFollower()

Cross References:

Use case: view Profile

Scenario: view Profile

Pre-Condition: View Profile is underway.

Post-Condition:

bu.Followes was intitialized.

Operation: openUserProfile()

Cross References:

Use case: view Profile

Scenario: view Profile

Pre-Condition: none.

Post-Condition: CustomerUserProfile instance cup was created.

Operation: viewCustomerAccount(cid)

Cross References:

Use case: View Profile

Scenario: view Profile

Pre-Condition: open profile is underway.

Post-Condition:

CustomerUser instance cu was created.

cup was associated with cu:BrandUser.

Attributes of cu ware initialized.

16) Approve Brand

Operation: openRequests()

Cross References:

Use case: Approve Brand

Scenario: Approve Brand

Pre-Condition: none.

Post-Condition: none.

Operation: viewAllNews()

Cross References:

Use case: Approve Brand

Scenario: Approve Brand

Pre-Condition: Approve request is underway.

Post-Condition: none.

Operation: viewRequest(id)

Cross References:

Use case: Approve Brand

Scenario: Approve Brand

Pre-Condition: Approve request is underway.

Post-Condition:

brandUser instance bu was created.

Bu.Email was initialized.

Bu.Name was initialized.

Bu.Category was initialized.

Bu.Photo was initialized.

Bu.Proofs was initialized.

Operation: openProfoos()

Cross References:

Use case: Approve Brand

Scenario: Approve Brand

Pre-Condition: Approve request is underway.

Post-Condition: none.

Operation: approveRequest(bid)

Cross References:

Use case: Approve Brand

Scenario: Approve Brand

Pre-Condition: approve brand is underway.

Post-Condition: adm.isApprove was changed to true.

17) Delete Brand

Operation: deleteBrand()

Cross References:

Use case: Delete Brand

Scenario: Delete Brand

Pre-Condition: delete brand is underway.

Post-Condition:

Admin instance adm was created.

Operation: deleteBrandAccount(id)

Cross References:

Use case: Delete Brand

Scenario: Delete Brand

Pre-Condition: delete brand is underway.

Post-Condition:

BrandUser instance bu was created.

bu.Email was initialized.

18) Remove User

Operation: deleteCustomer()

Cross References:

Use case: Remove User

Scenario: Remove User

Pre-Condition: delete customer is underway.

Post-Condition:

Admin instance adm was created.

Operation: deleteustomerUserAccount(id)

Cross References:

Use case: Remove User

Scenario: Remove User

Pre-Condition: delete brand is underway.

Post-Condition:

CustomerUser instance cu was created.

cu.Email was initialized.

4.7. Activity Diagram

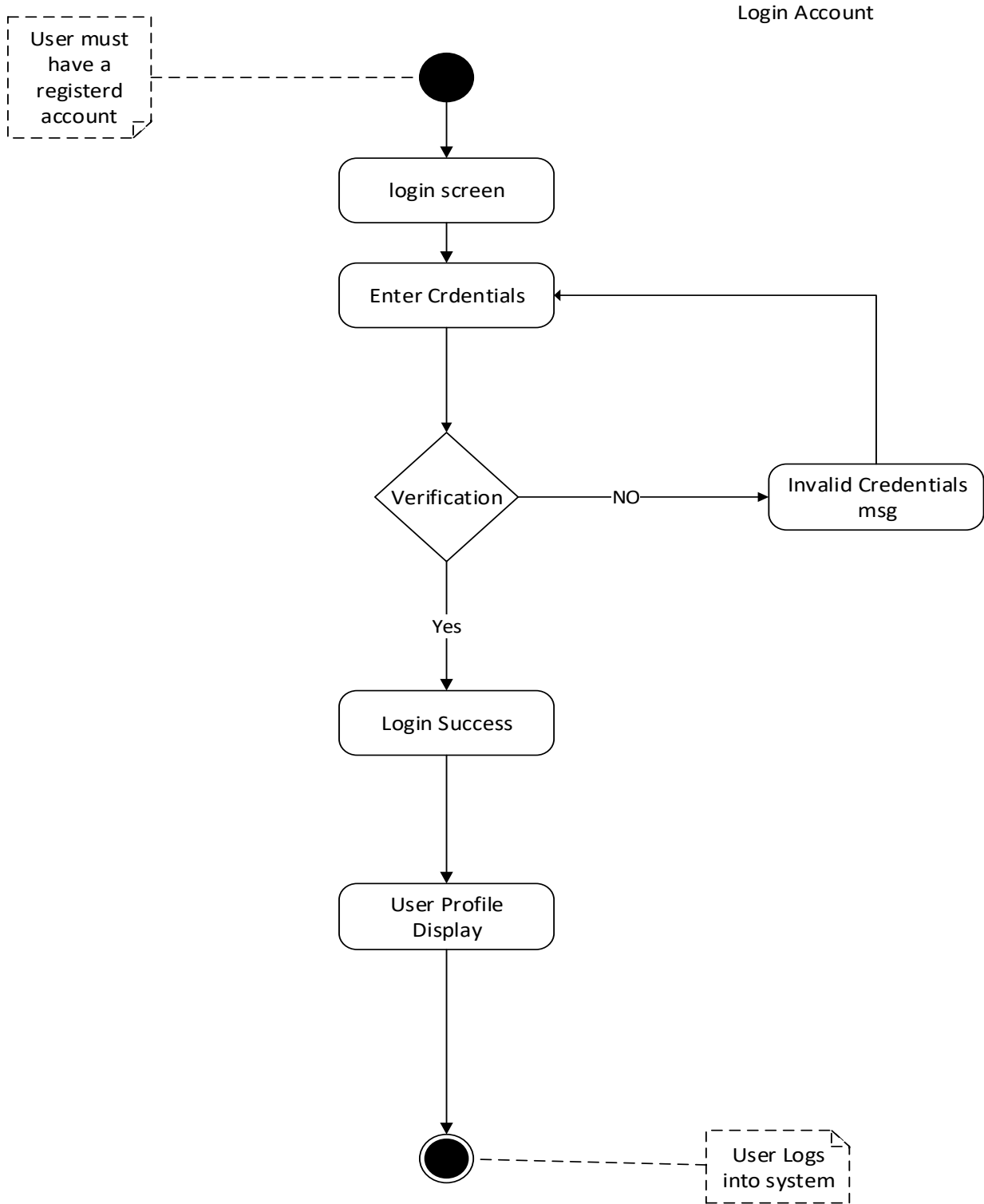


Figure 4.31 Login Account

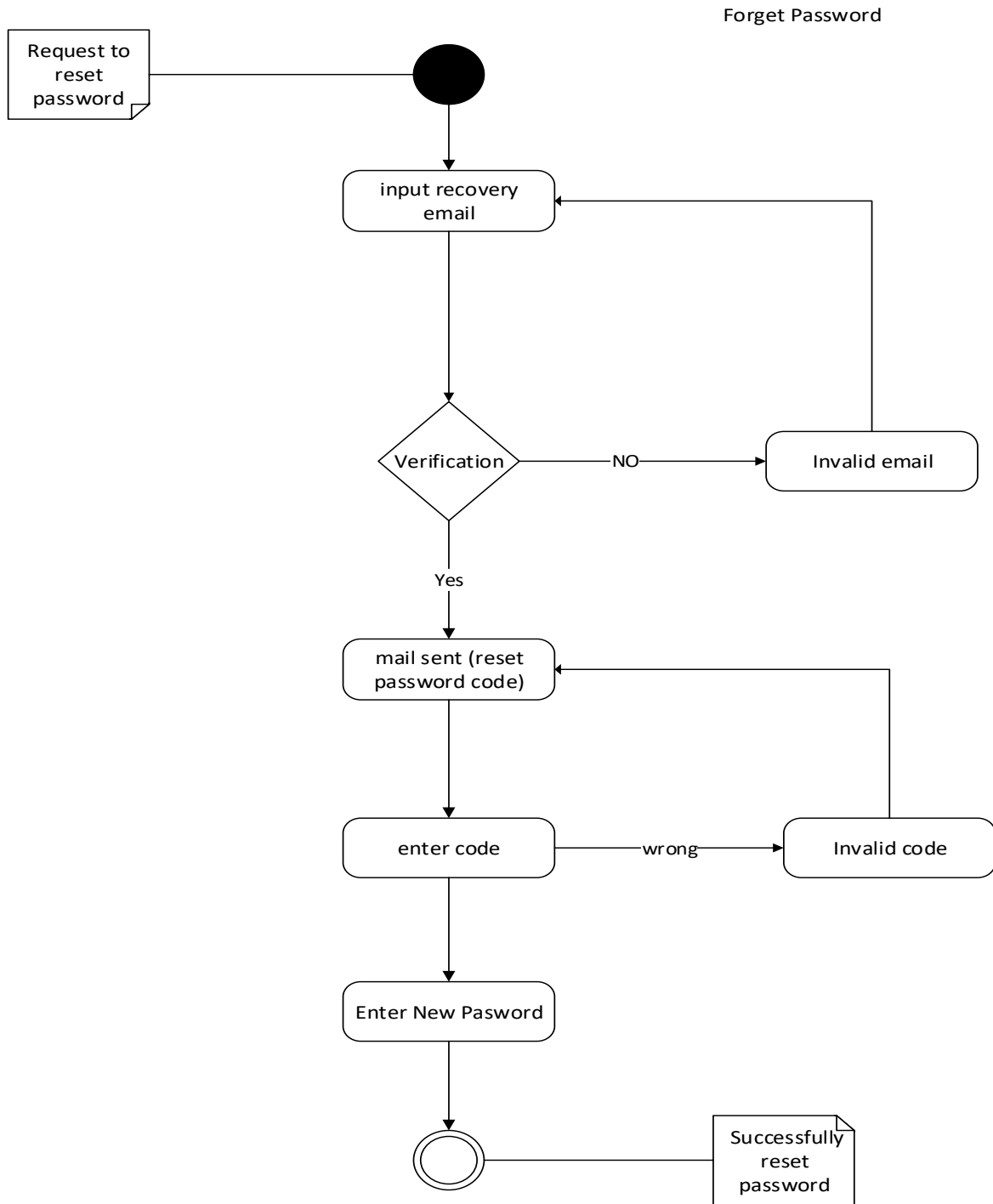


Figure 4.32 Forget Password

Update Profile

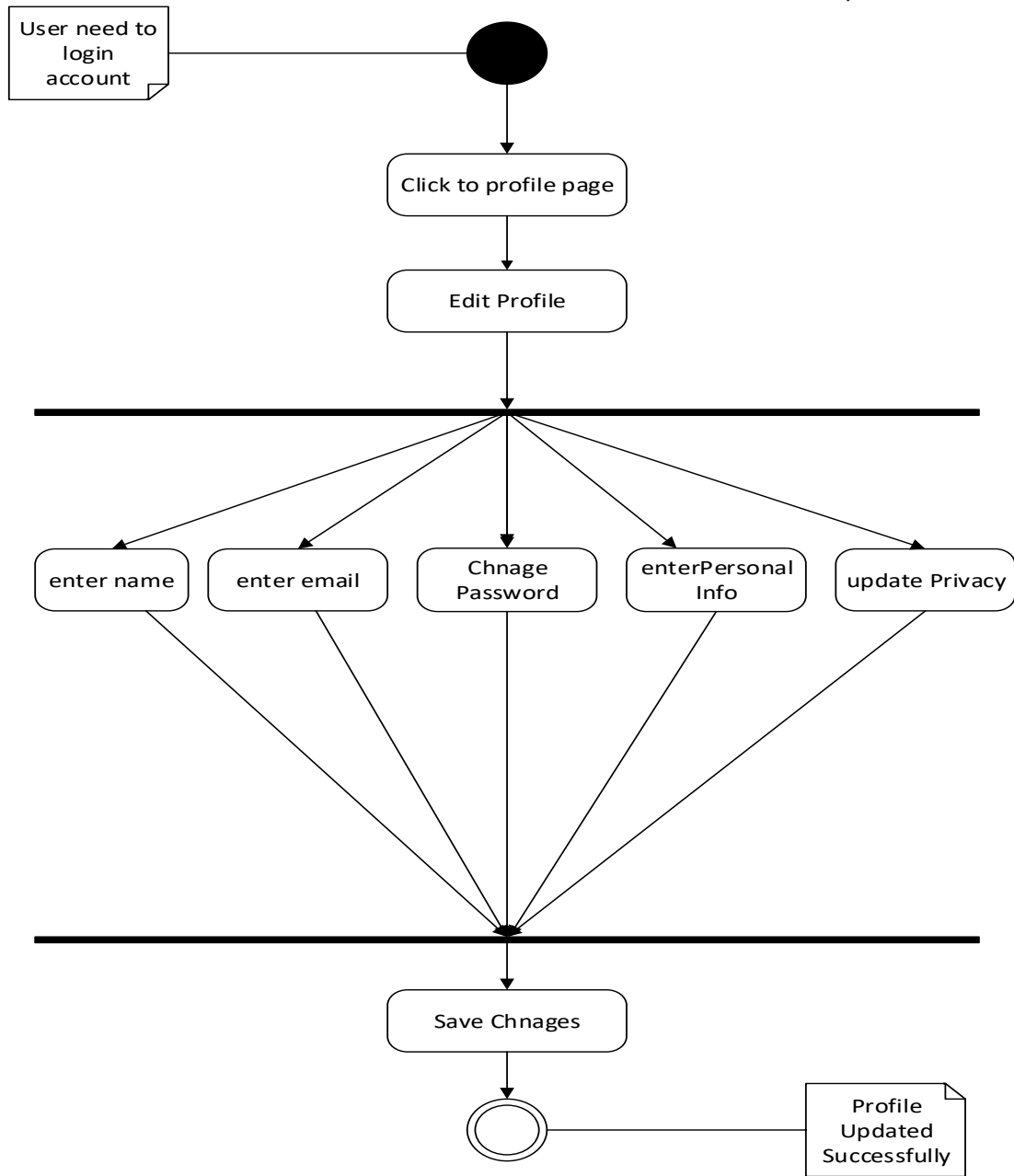


Figure 4.33 Update Profile

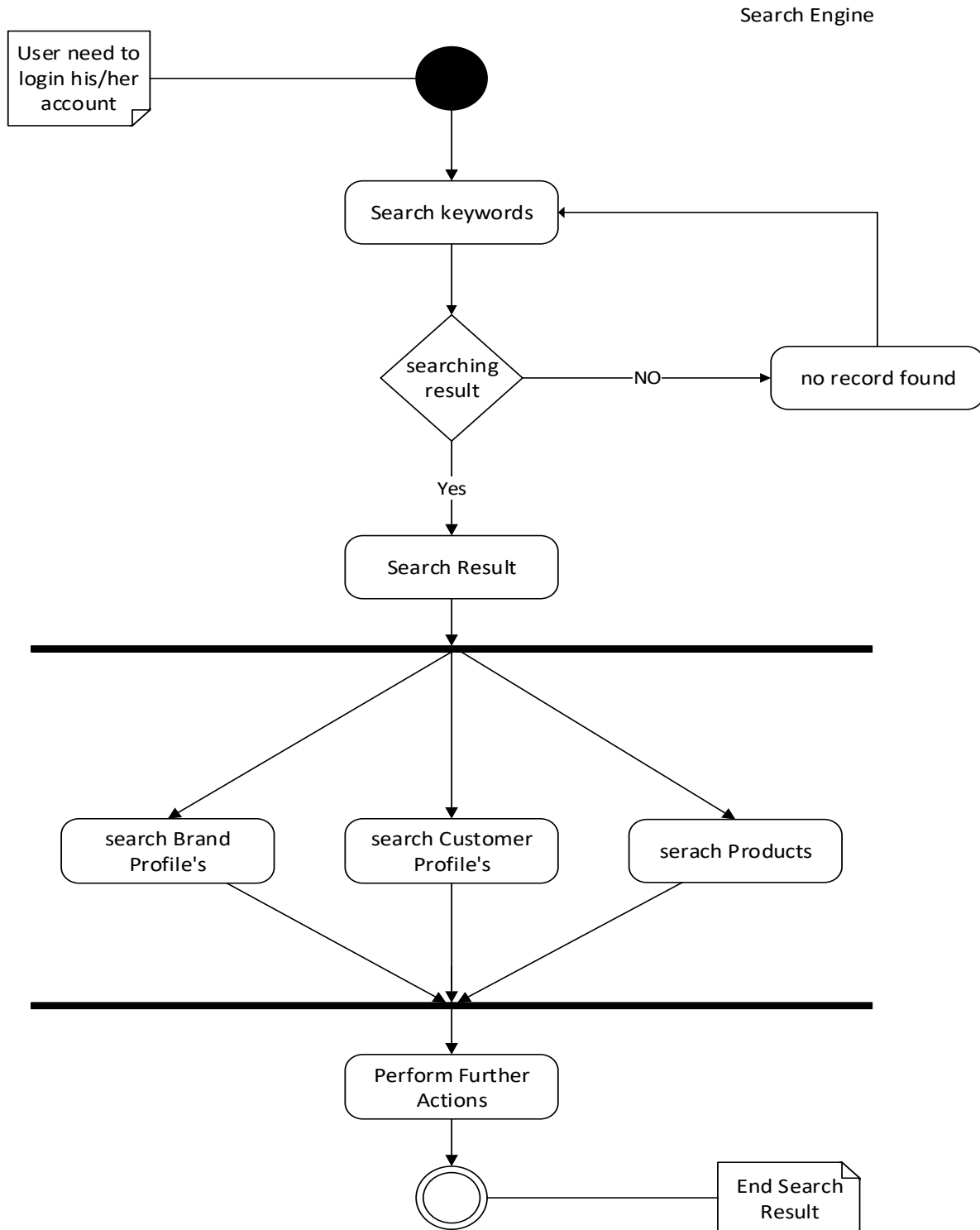


Figure 4.34 Search Engine

Timeline

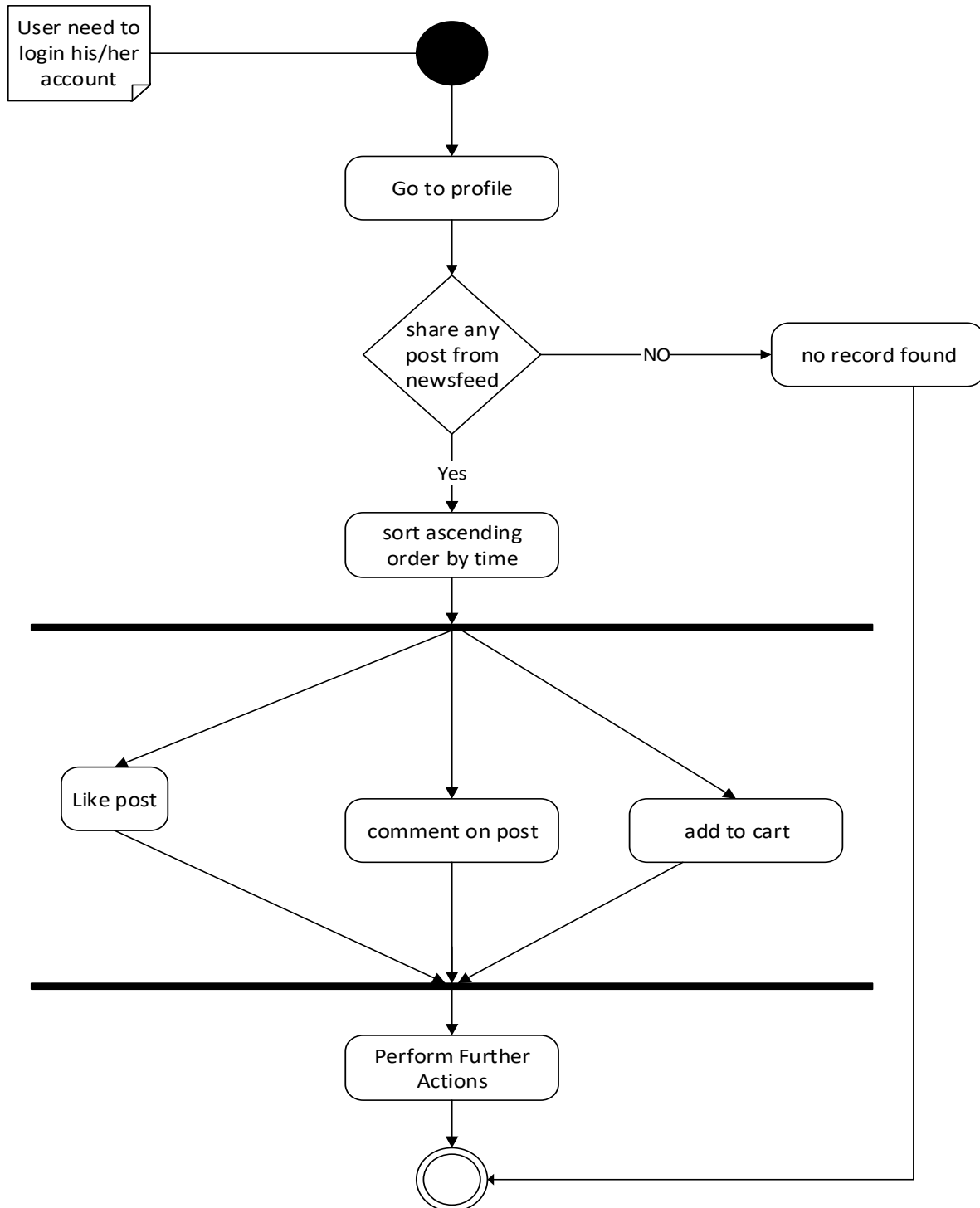


Figure 4.35 timeline

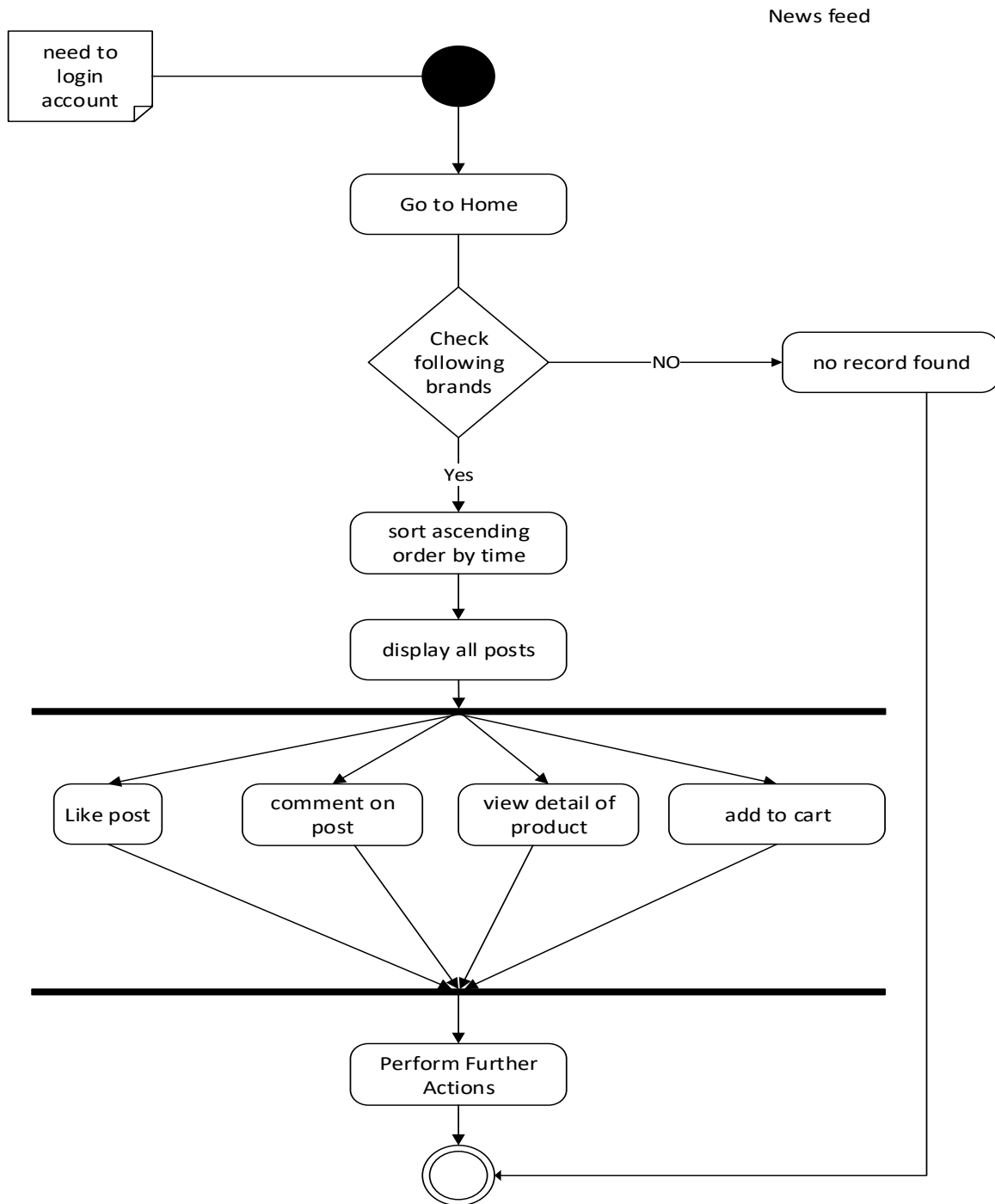


Figure 4.36 NewsFeed

Follow

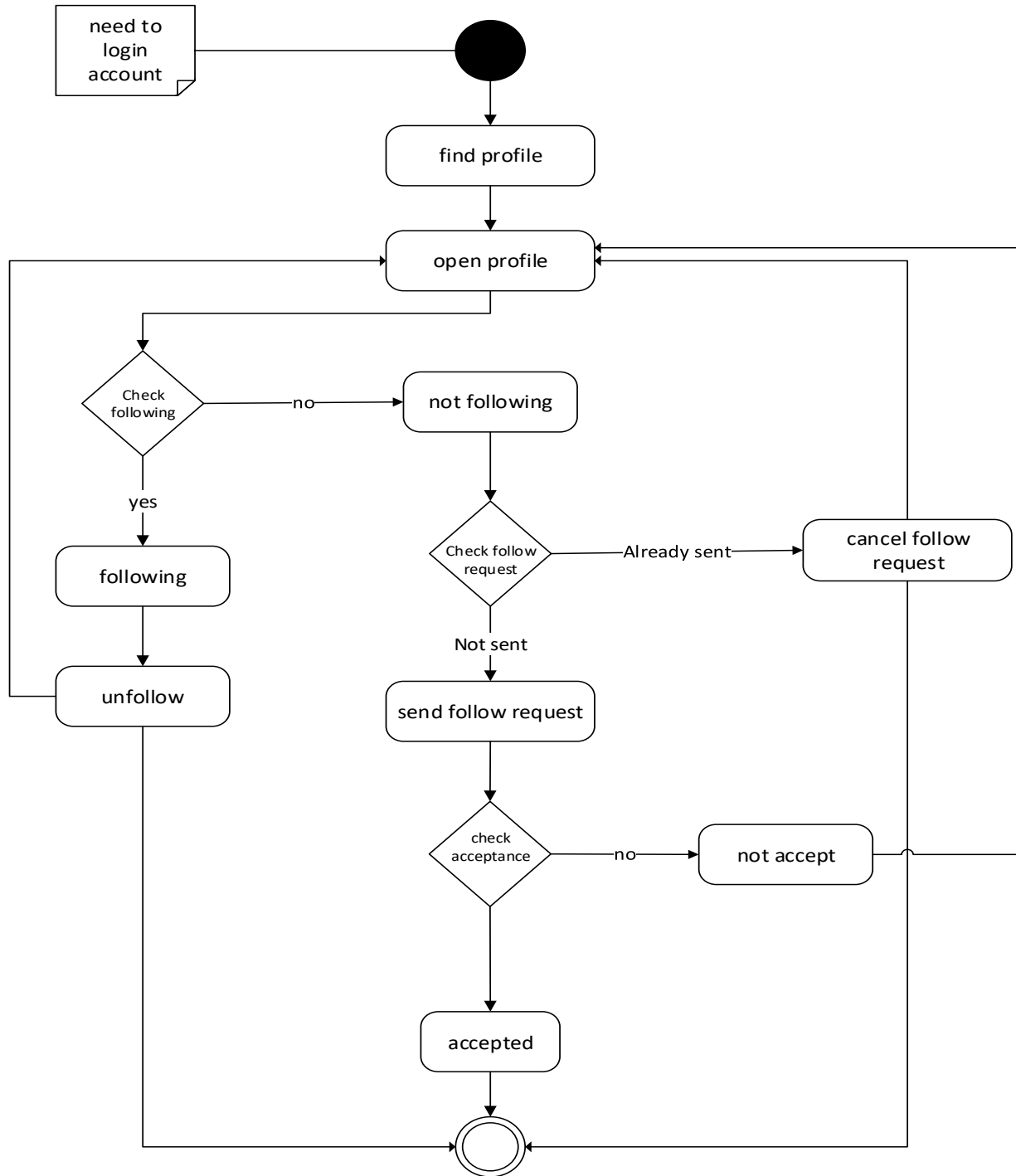


Figure 4.37 Follow

Notifications

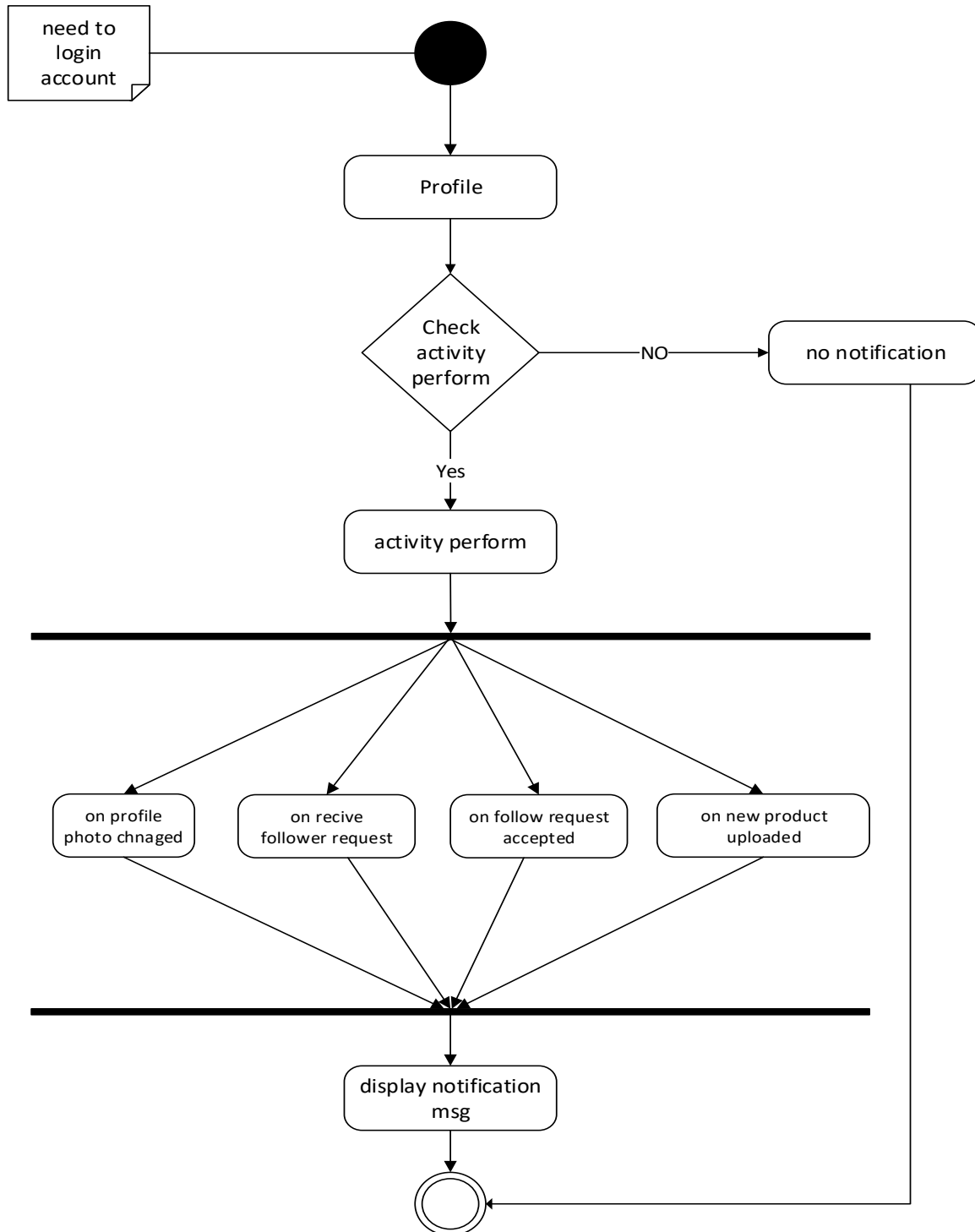


Figure 4.38 Notification

Verified Address Using Map

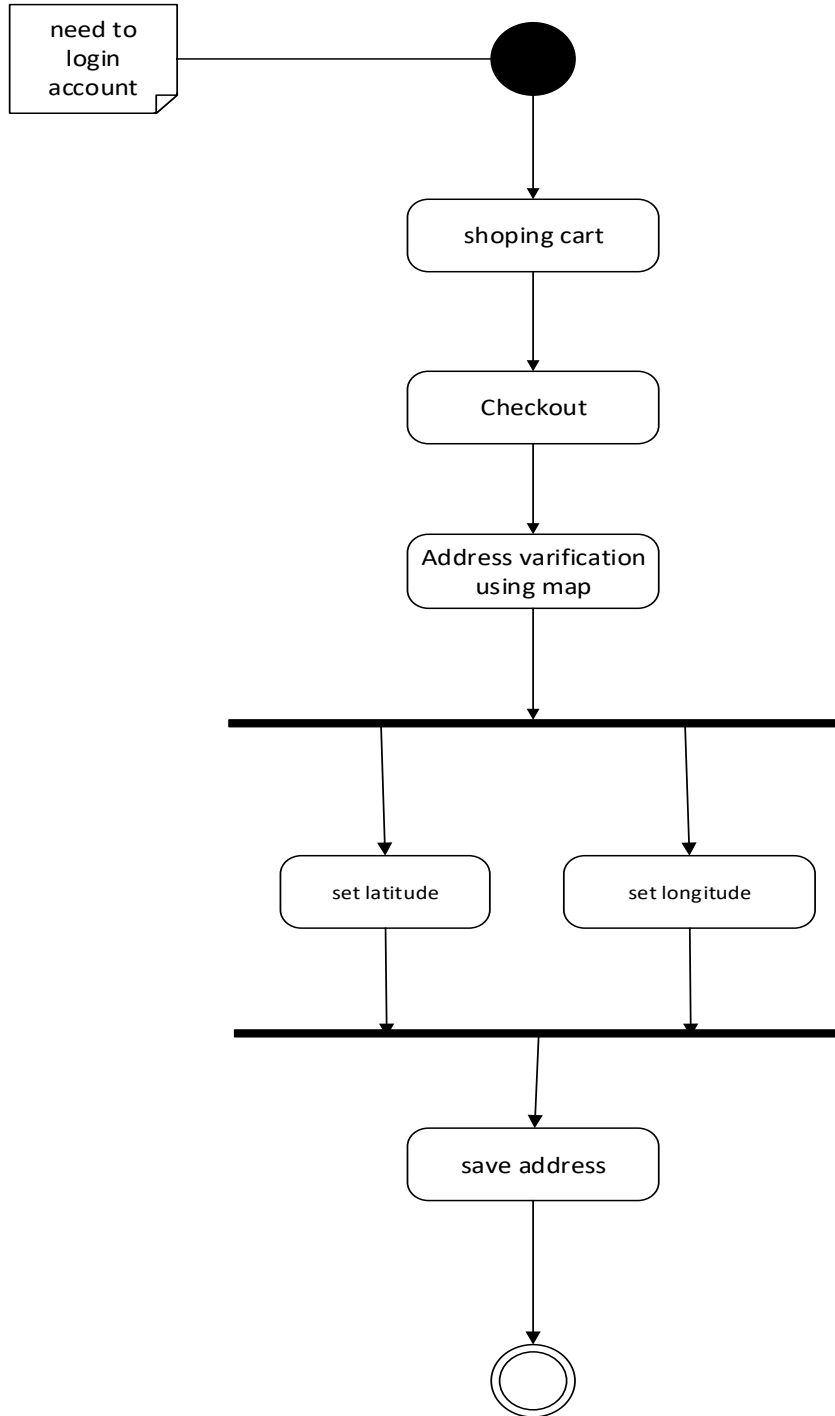


Figure 4.39 Verified Address Using Map

Create Customer Account

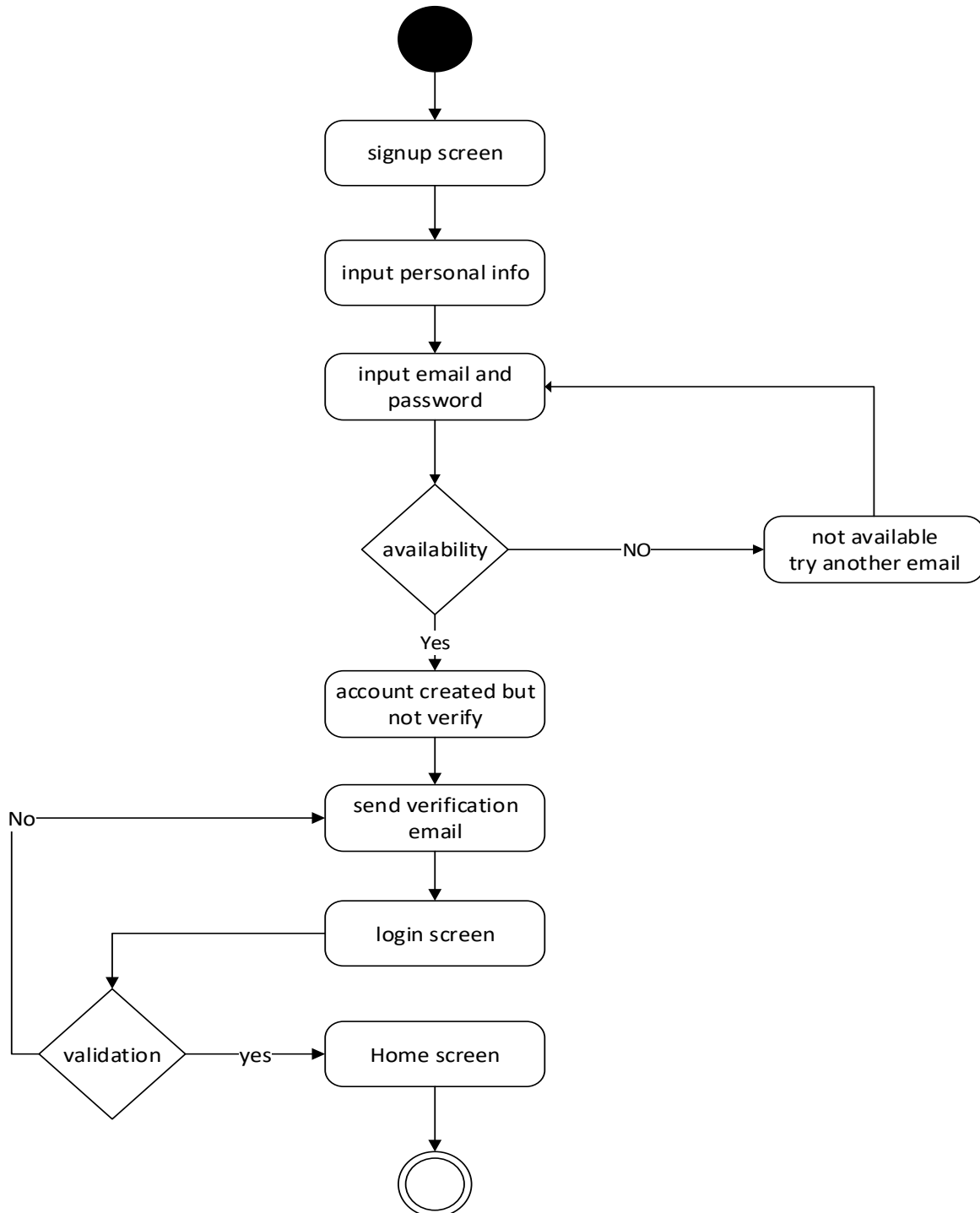


Figure 4.41 Create Customer Account

Shopping Cart

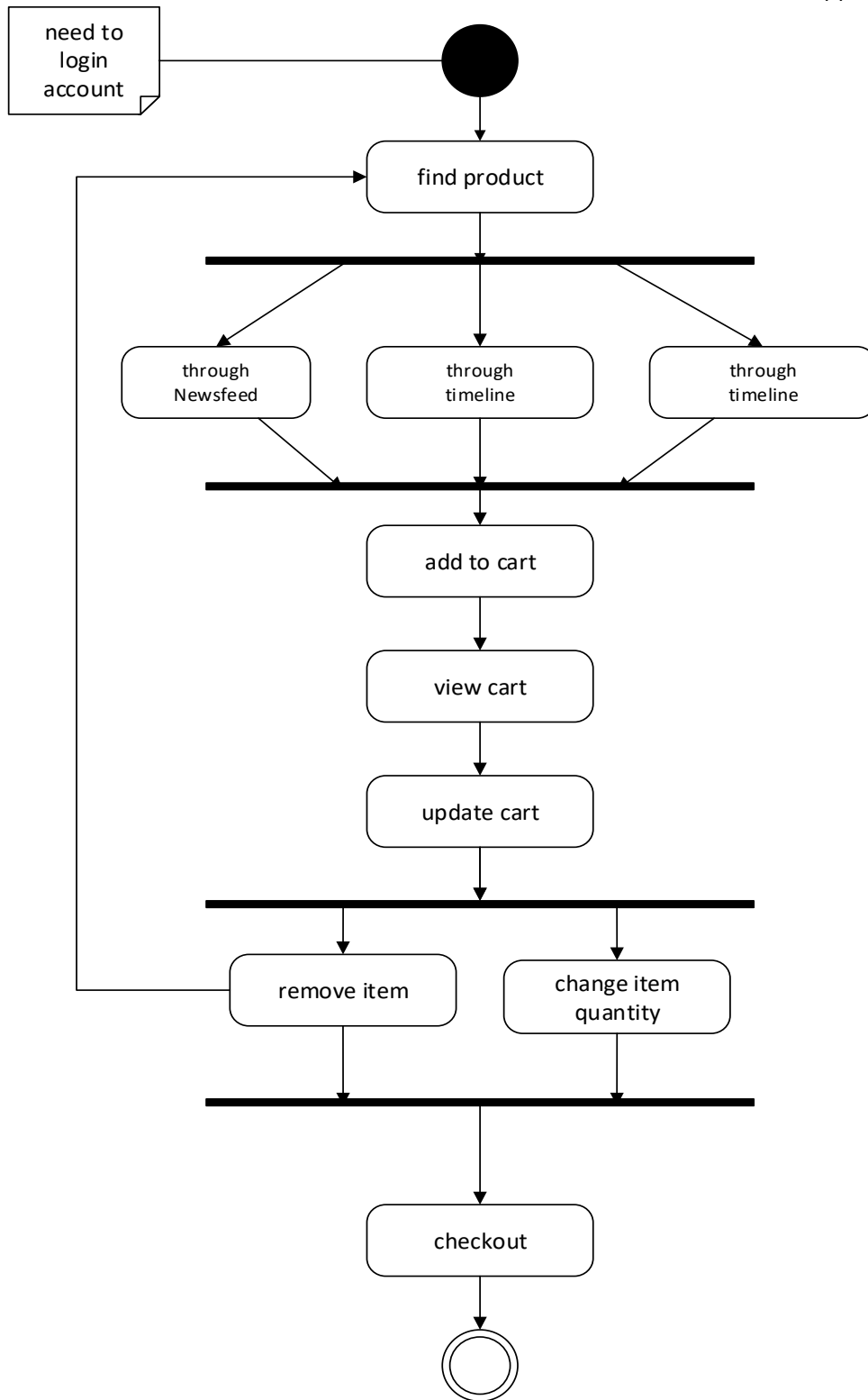


Figure 4.42 Shopping Cart

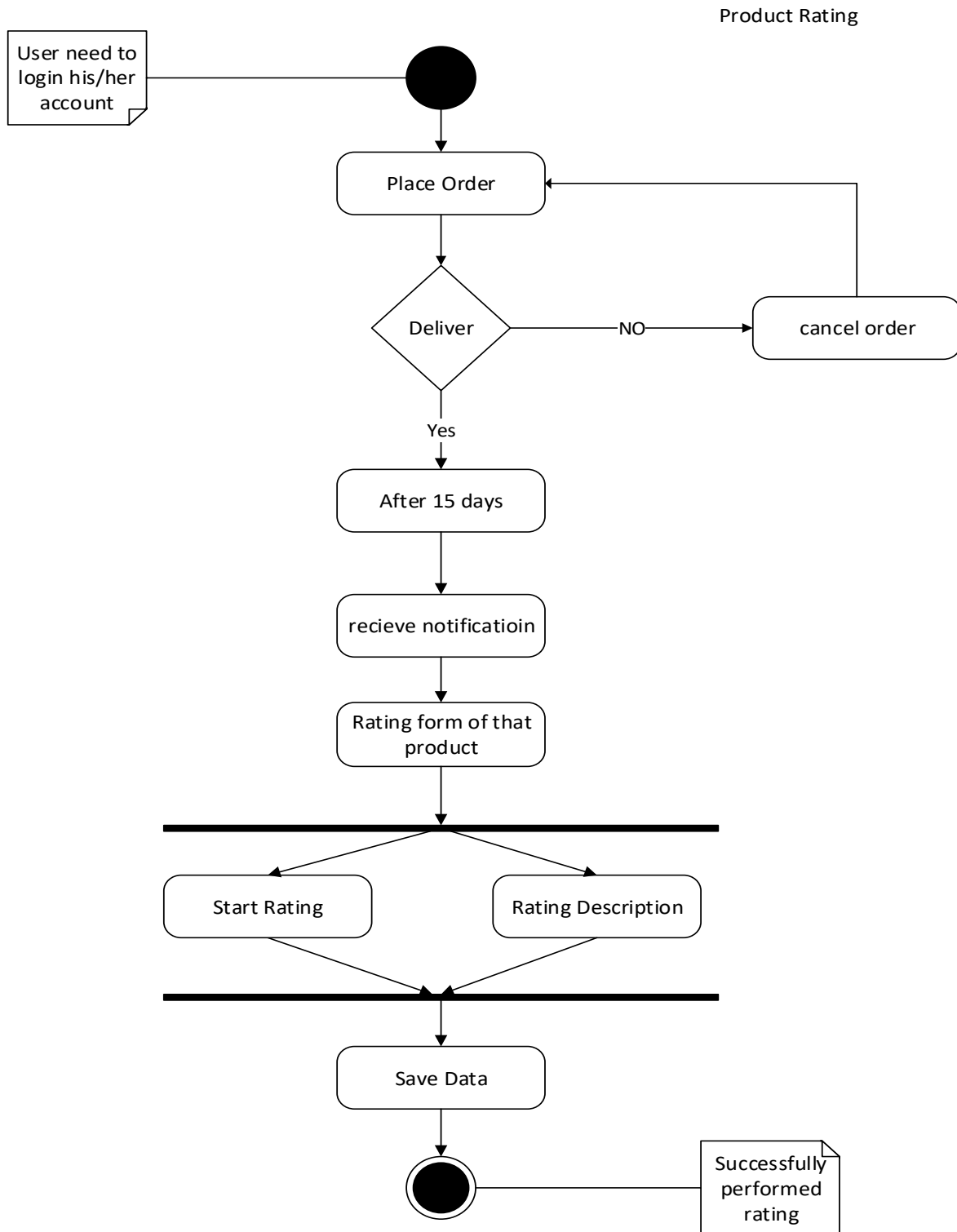


Figure 4.44 Product Rating

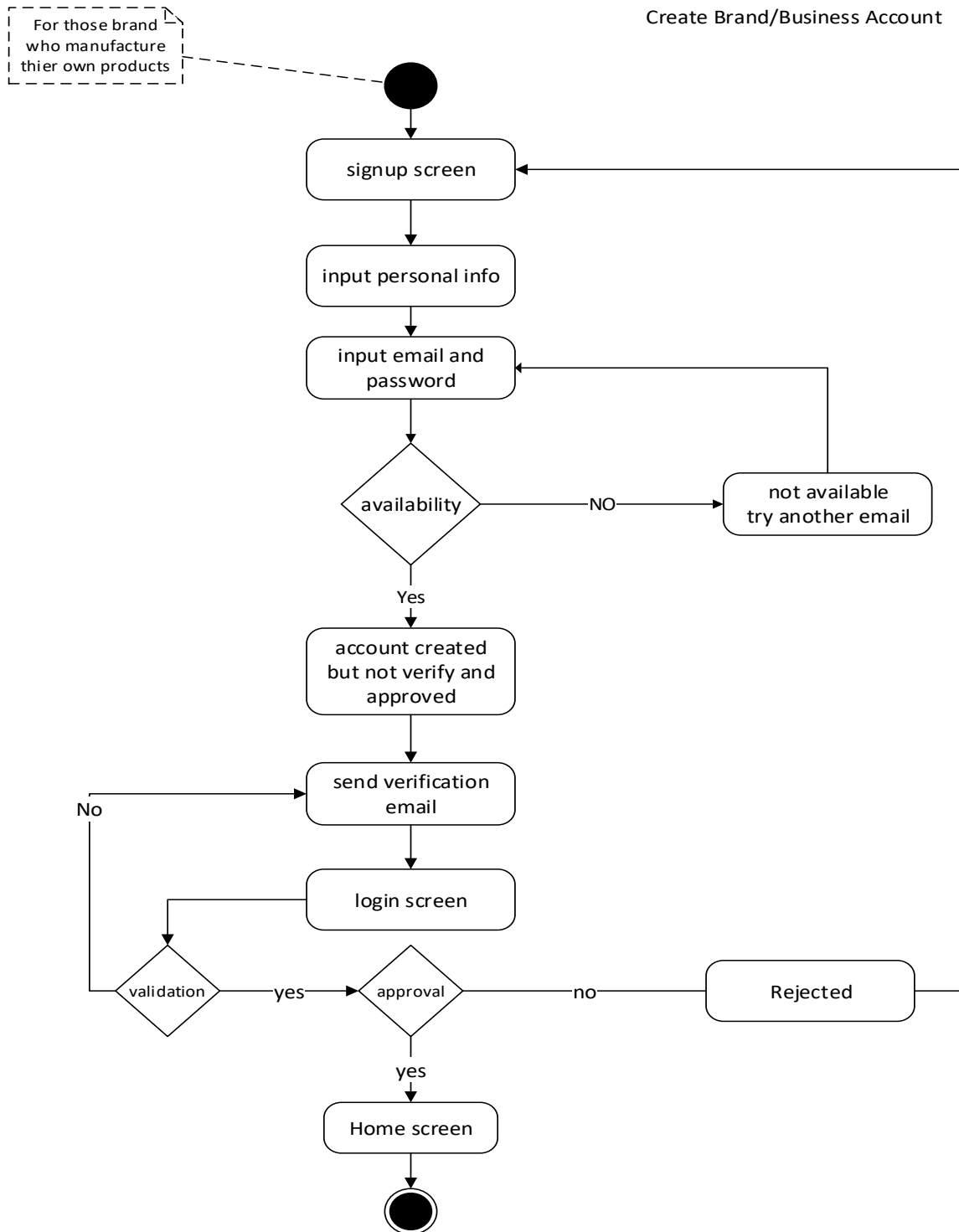


Figure 4.46 Create Brand Account

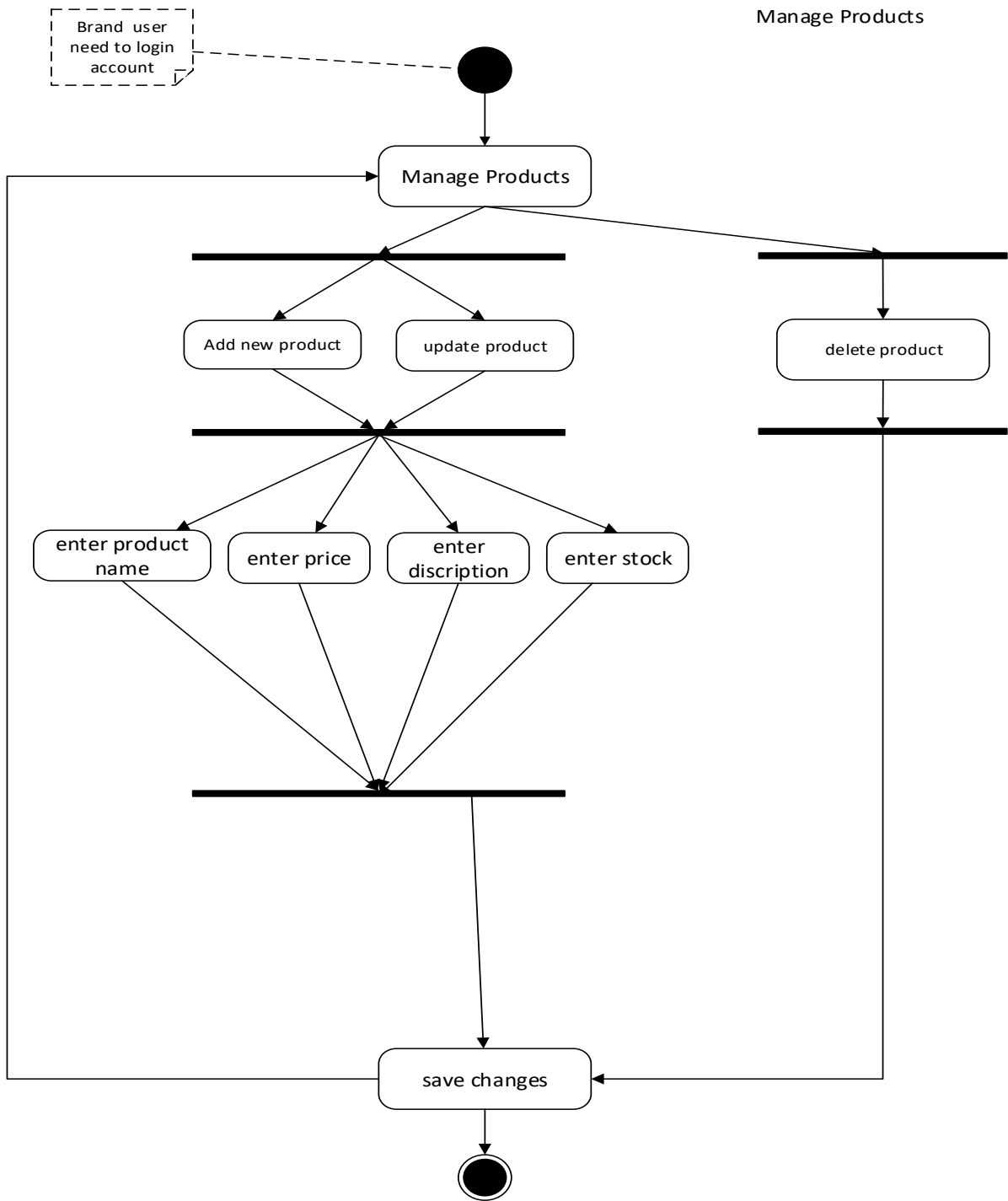


Figure 4.47 Manage Products

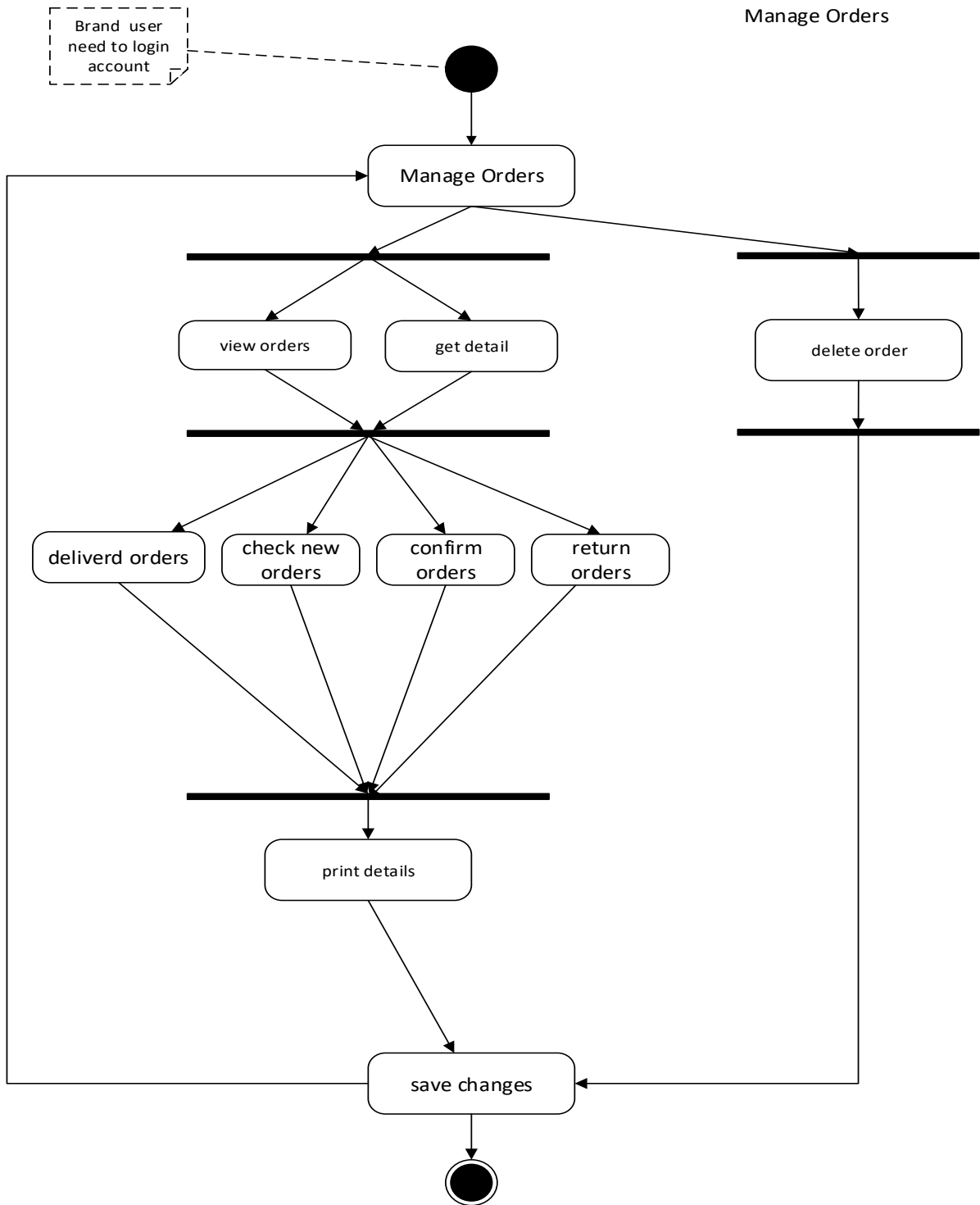
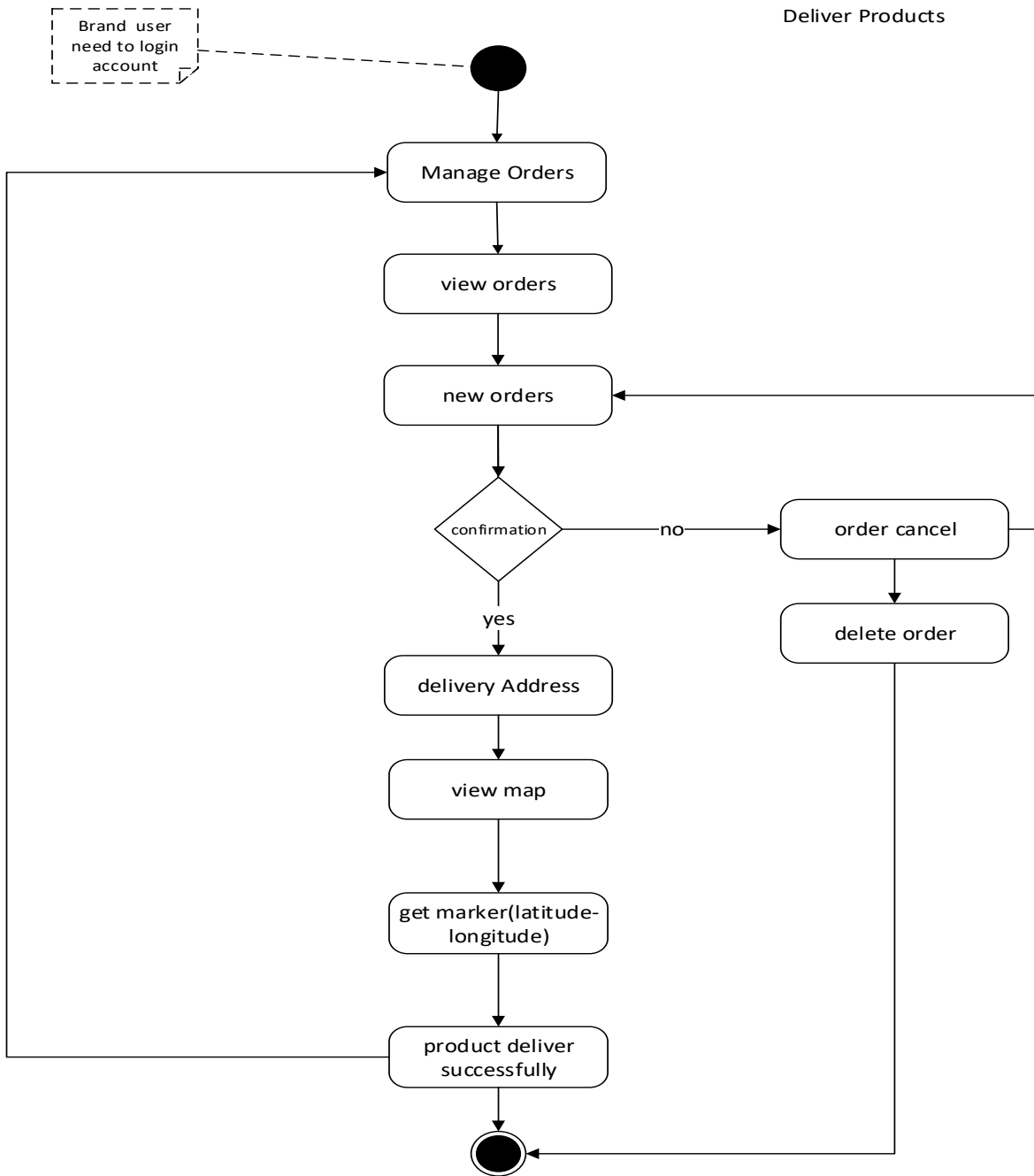


Figure 4.48 Manage Orders



Dashboard Login

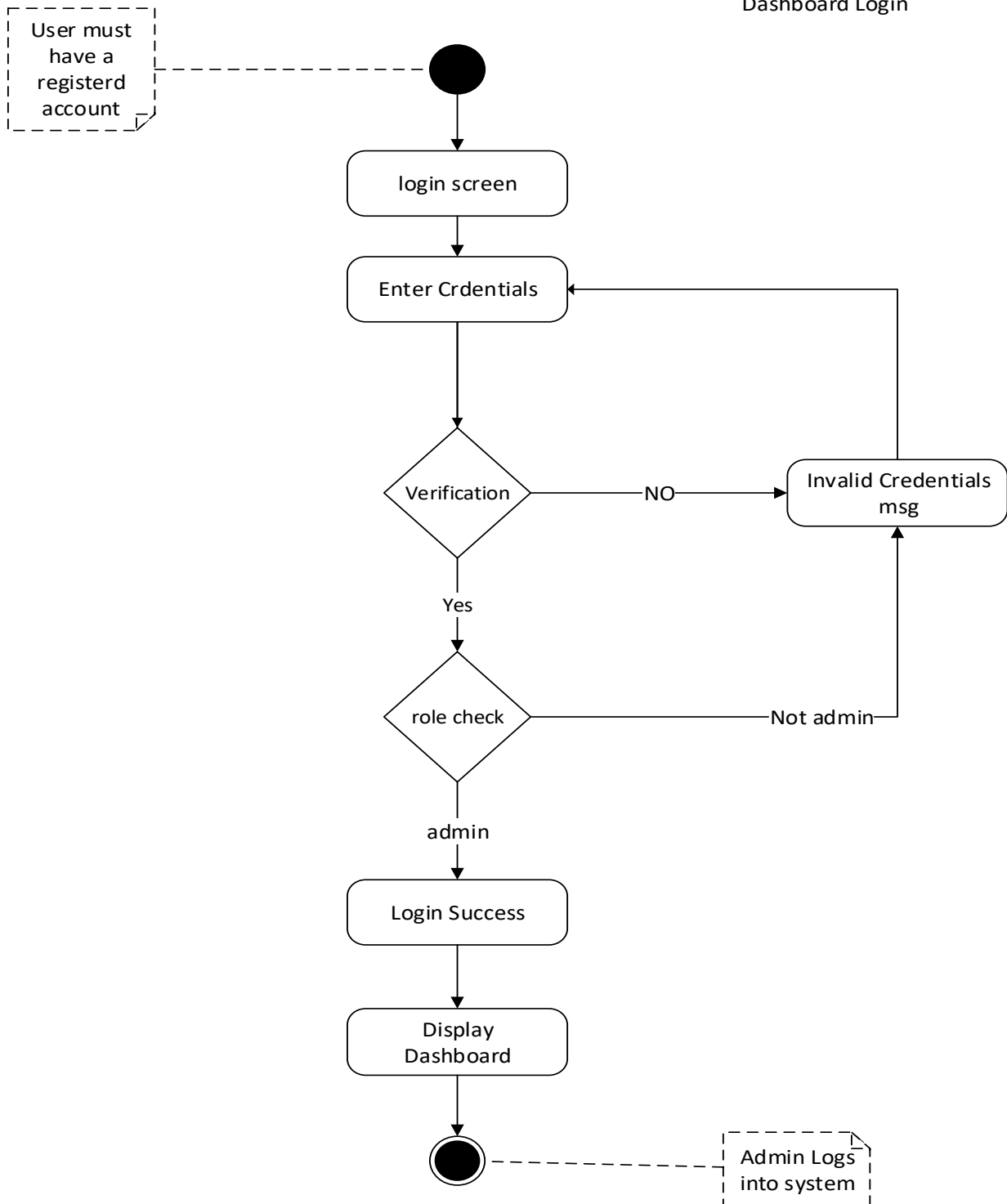


Figure 4.49 Dashboard Login

Approve Brand

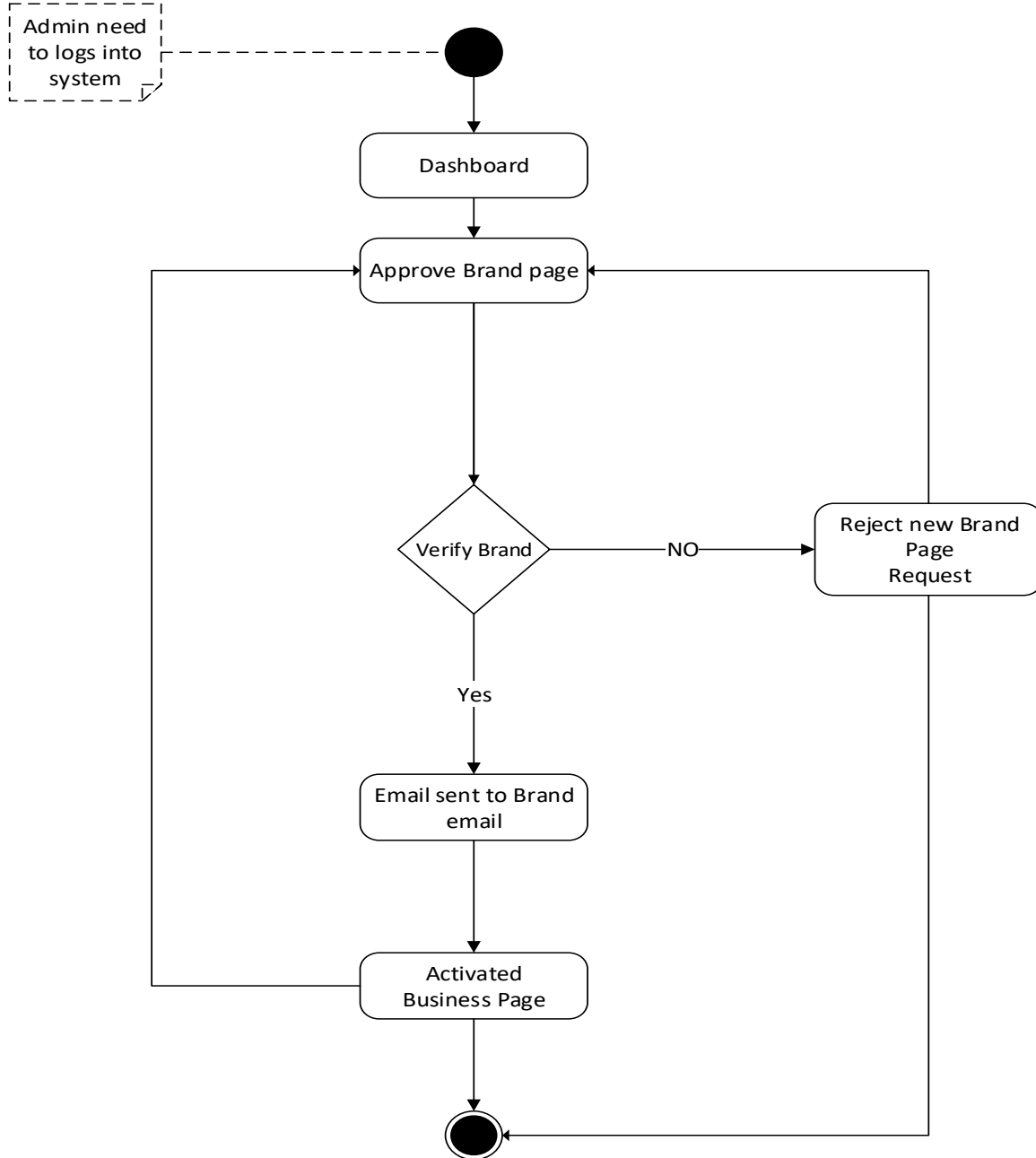


Figure 4.49 (a) Dashboard Login

Figure

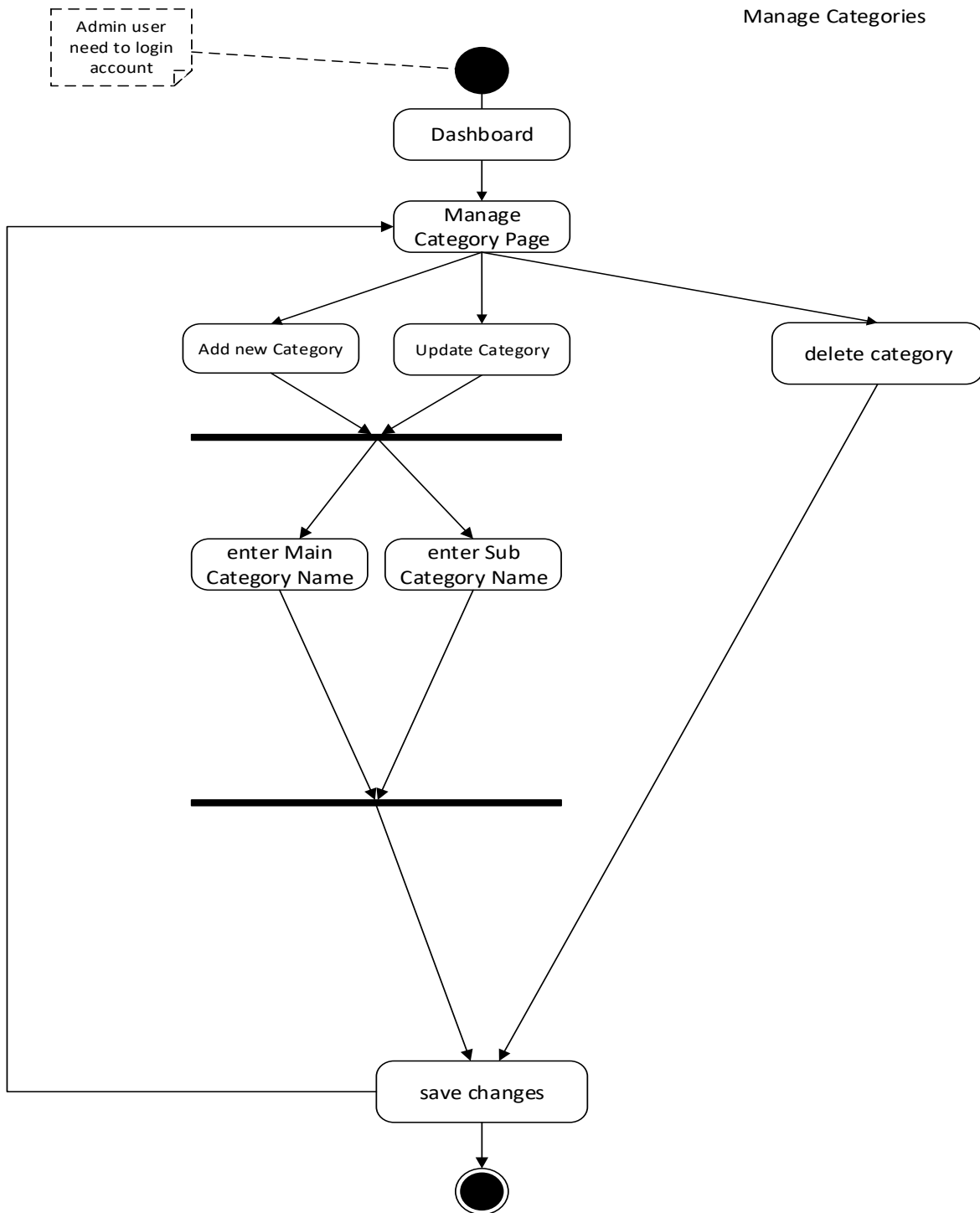


Figure 4.50 Manage Categories

Manage Accounts

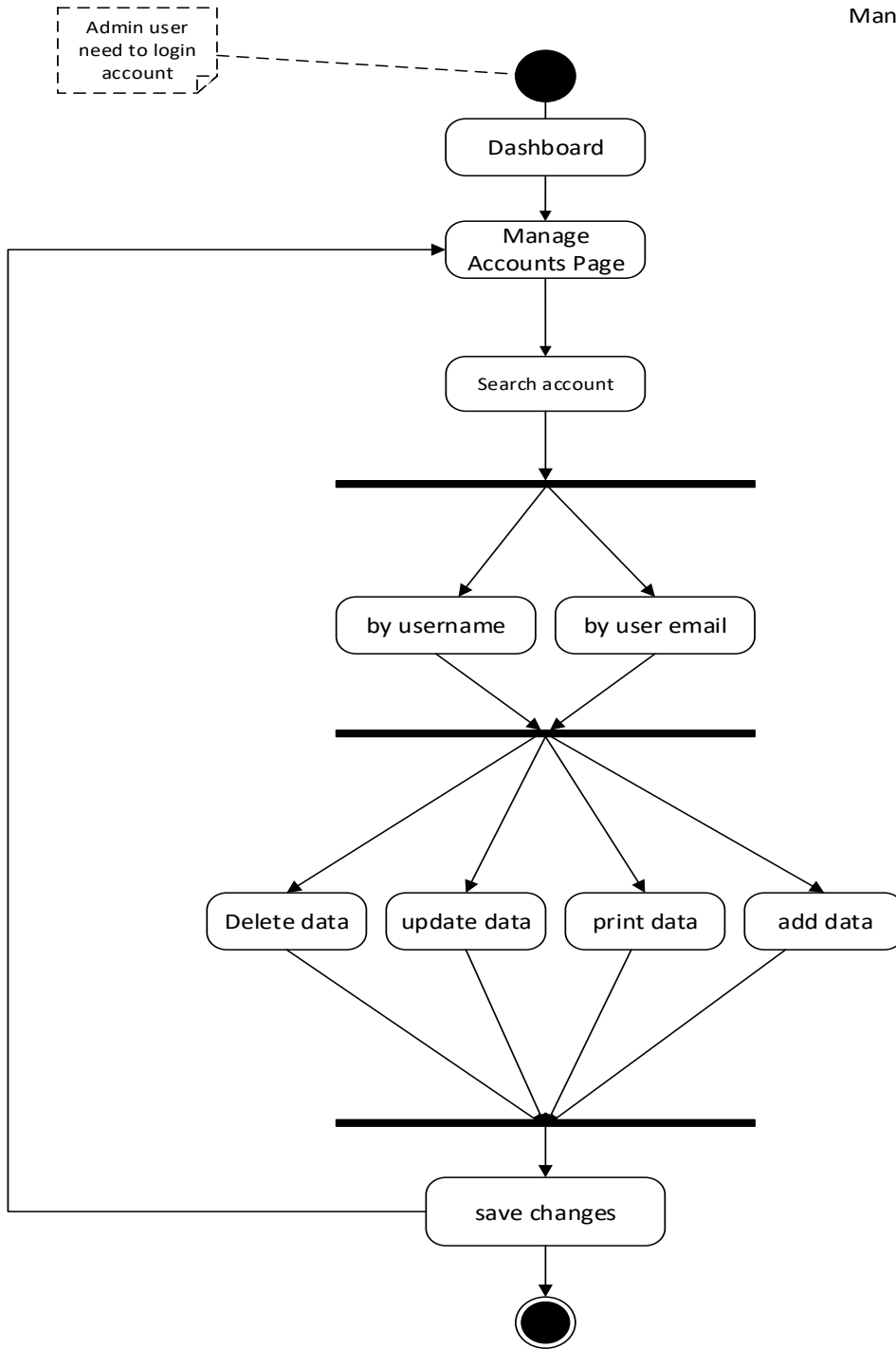


Figure 4.51 Manage Accounts

Delete Business Page

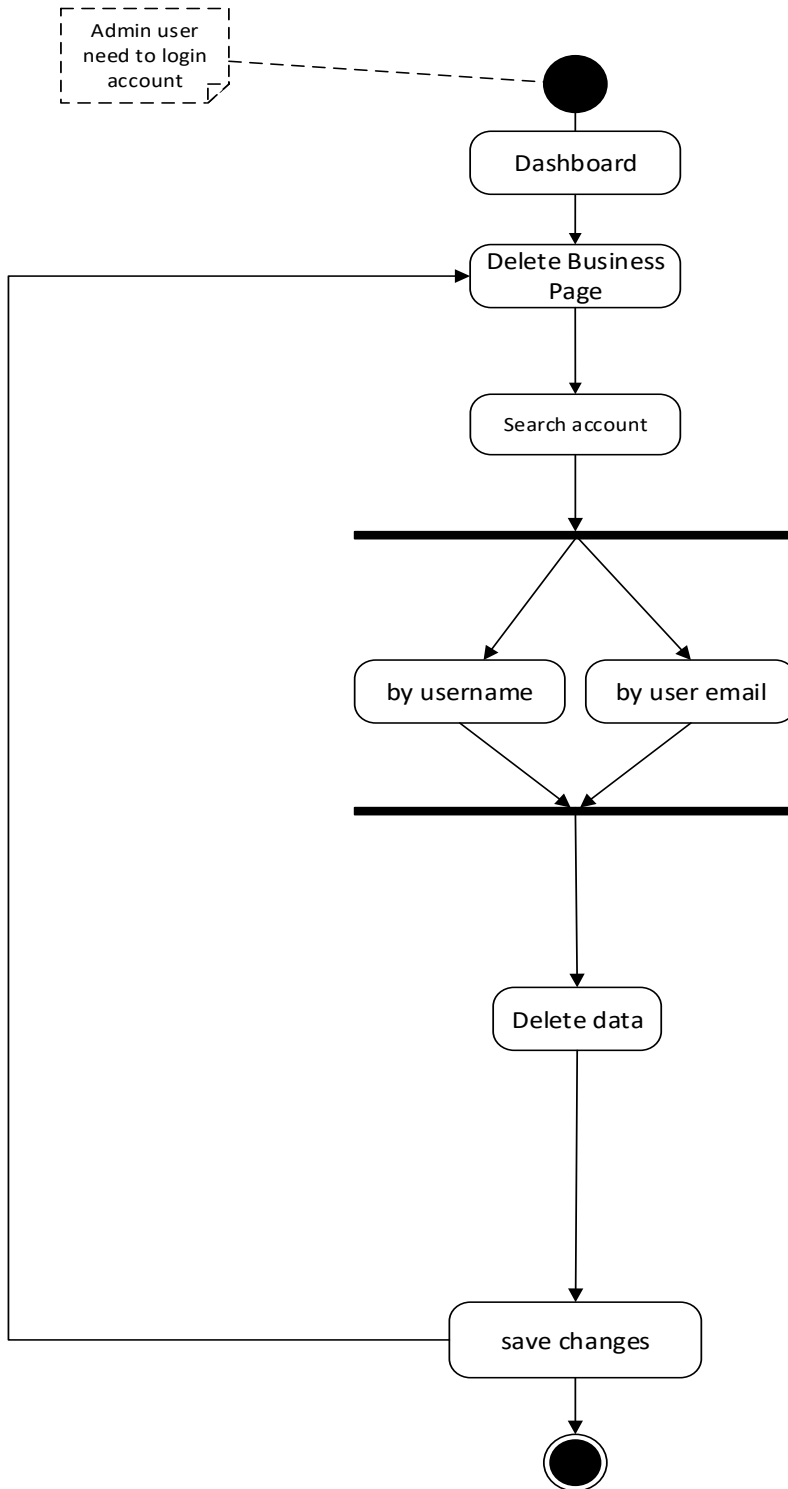


Figure 4.52 Delete Business Page

Manage Reports

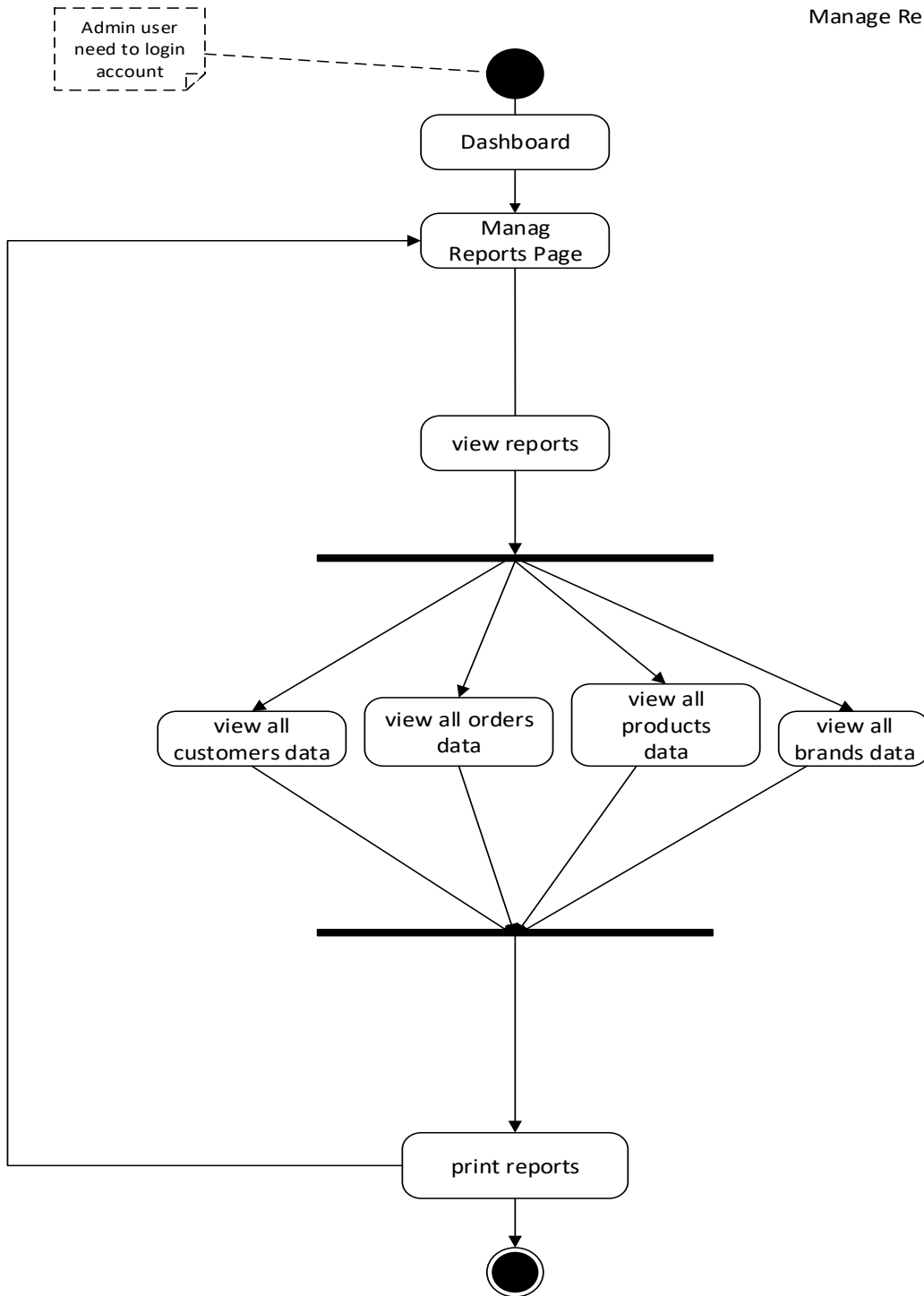


Figure 4.53 Manage Reports

Reset Password

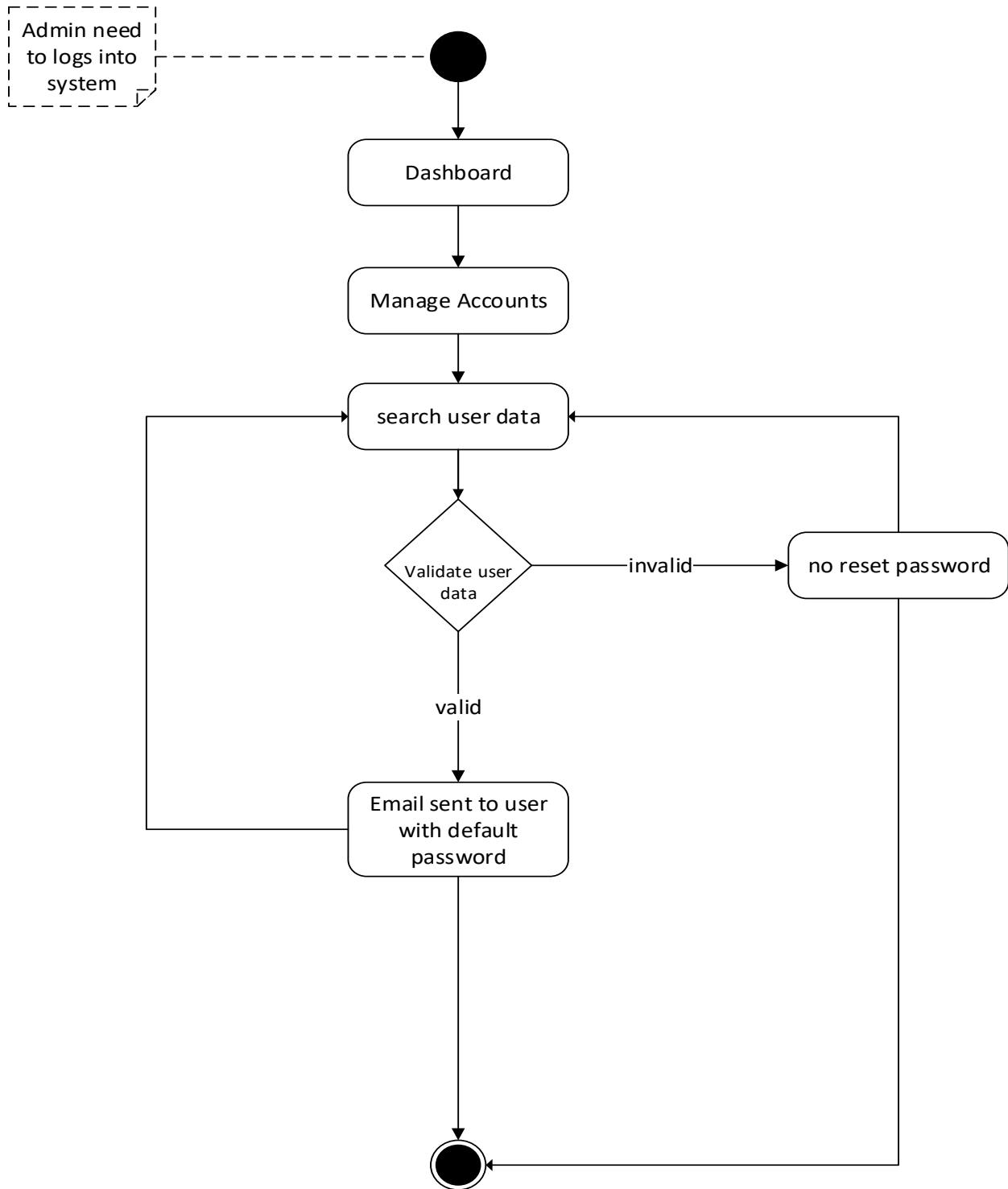


Figure 4.53 (b) Reset Password

Figure 4.54 State Transition Diagram

4.9. Component Diagram

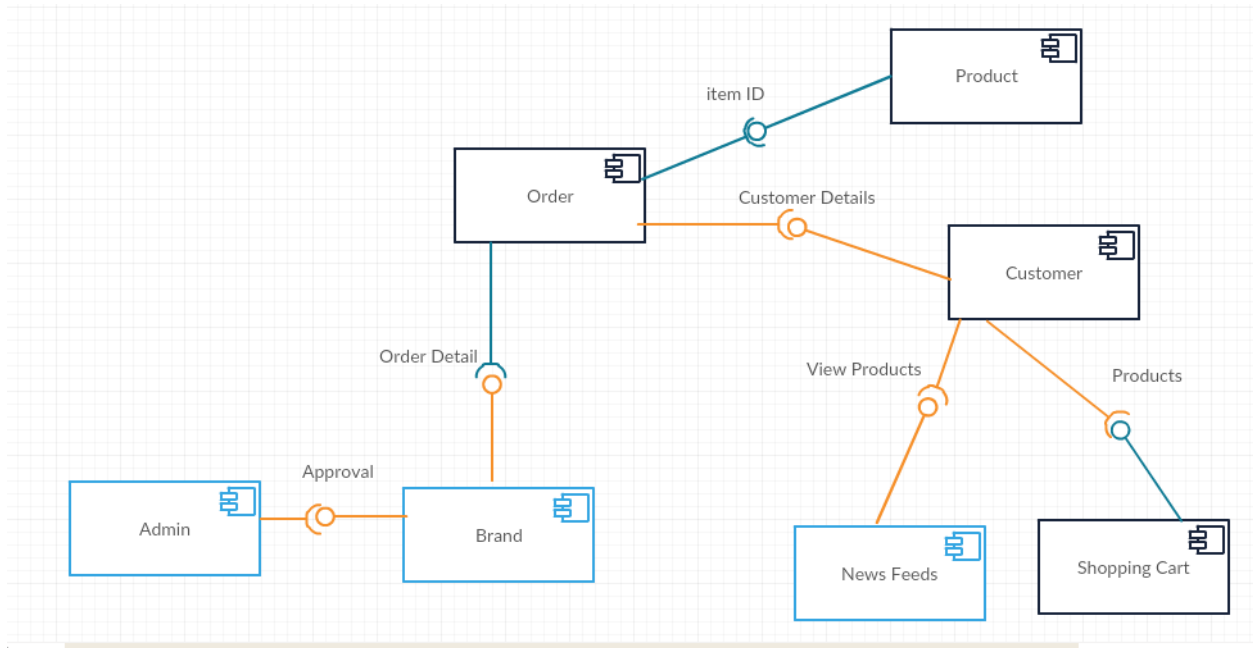


Figure 4.55 Component Diagram

4.10. Deployment Diagram

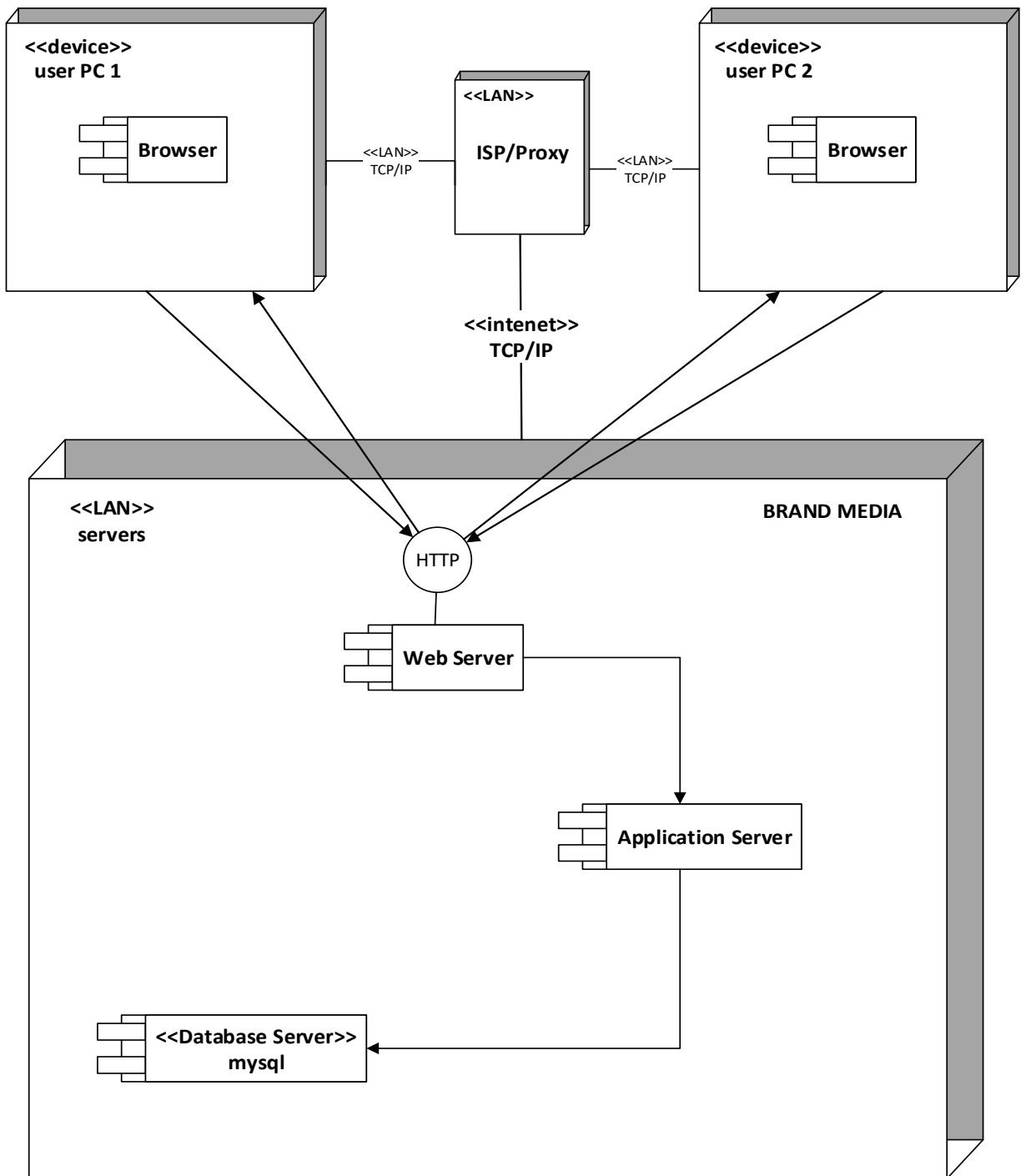


Figure 4.55 Deployment Diagram

Chapter 5

Implementation

Chapter 5: Implementation

It describes the implementation behaviors of system. It describes the important pseudo code of the systems. It describes the components of system, libraries we used in coding, and what web services we used to develop the system. It describes the tools and techniques that we need to develop a system. What coding standard are used. What programming language's versions used, also mention in this chapter.

5.1. Important Flow Control/Pseudo codes

Register Brand

Start

```
READ BrandName, Email, Password, RePassword, Category, photo, Proofs
```

```
IF Password = RePassword THEN
```

```
    VIEW Confirmation Page
```

```
    IF isConfirm = TRUE
```

```
        SAVE BrandName, Email, Password, Category, photo
```

```
        DISPLAY waiting page
```

```
    ENDIF
```

```
ENDIF
```

END

Login Account

Start

```
READ userEmail AND UserPassword
```

```
IF userEmail AND UserPassword Exist THEN
```

```
GOTO Home Page
ENDIF
END
```

Register Customer

```
Start
  READ FirstName, LastName, Email, Password, RePassword, Category, photo, Address
  IF Password = RePassword THEN
    VIEW Confirmation Page
    IF isConfirm = TRUE
      SAVE FirstName, LastName, Email, Password, Category, photo, Address
      DISPLAY home activity
    ENDIF
  ENDIF
END
```

Add Product into Cart

```
START
  READ ProductID, CustomerEmail
  IF Sale = TRUE
    SAVE ProductID, CustomerEmail
  ENDIF
END
```

View Cart

```
START
  TOTALPRICE = 0
```

```
WHILE FETCH DATA
  READ ProductName, ProductBrandName, ProductQuantity, ProductPrice
  DISPLAY ProductName, ProductBrandName, ProductQuantity, ProductPrice
  TOTALPRICE = ADD TOTALPRICE AND ProductPrice
ENDWHILE
DISPLAY TOTALPRICE
END
```

Approve Request

```
START
  WHILE FETCH
    READ BrandName, Email, Password, RePassword, Category, photo
    IF isApprove = FALSE
      DISPLAY BrandName, Email, Password, RePassword, Category, photo, Proofs
    ENDIF
  ENDWHILE
  VIEW Request
  VIEW Proofs
  Approve BrandName
END
```

View Notification

```
START
  WHILE FETCH
    READ Notifications FROM Database
    DISPLAY Notification
  ENDWHILE
END
```

View Specific Notification

START

GOTO Specific Notification Page

WHILE FETCH

 READ Data

 DISPLAY Data

ENDWHILE

END

View News Feeds

START

WHILE FETCH

 READ ProductPhoto, ProductName, ProductQuantity, ProductDescription, ProductPrice, PBN

 READ likes, comments, shares

 DISPLAY ProductPhoto, ProductName, ProductQuantity, ProductDescription, ProductPrice,

 DISPLAY likes, comments, shares

ENDWHILE

END

View Customer Profile

START

WHILE FETCH

 READ CustomerName, CustomerEmail, CustomerFriends, Timeline, AND ALL his/her data

 DISPLAY CustomerName, CustomerEmail, CustomerFriends, Timeline, AND ALL his/her data

ENDWHILE

END

ADD Product

START

 READ ProductPhoto, ProductName, ProductQuantity, ProductDescription, ProductPrice

 SAVE ProductPhoto, ProductName, ProductQuantity, ProductDescription, ProductPrice

 UPDATE BRAND's PAGE

END

View Order List

START

 WHILE FETCH

 READ OrderInformation

 DISPLAY OrderInformation in LIST

 ENDWHILE

END

View Specific Order

START

GOTO Specific Order Page

 WHILE FETCH

 READ Order

 DISPLAY Order

 ENDWHILE

END

5.2. Components, Libraries, Web Services and stubs

We use the map service from the google in both website and internet. Google provide the all map related services. We make the APIs of map to get into our system.

In android studio we use the following dependencies to get all the features from website. These dependencies are

```
compile 'com.squareup.retrofit2:retrofit:2.2.0'
```

```
compile 'com.squareup.retrofit2:converter-gson:2.2.0'
```

These dependencies are helpful to do transactions from database (MYSQL). We make the APIs to do transactions from database and we use the built-in libraries to make a programs in both website and android application. Also we use the “android.widget.*” libraries in android studio. In PHP, we use the mysql library and his functions to do transactions with database.

5.3. Deployment Environment

Login:



Figure 5.1 Login

Category:

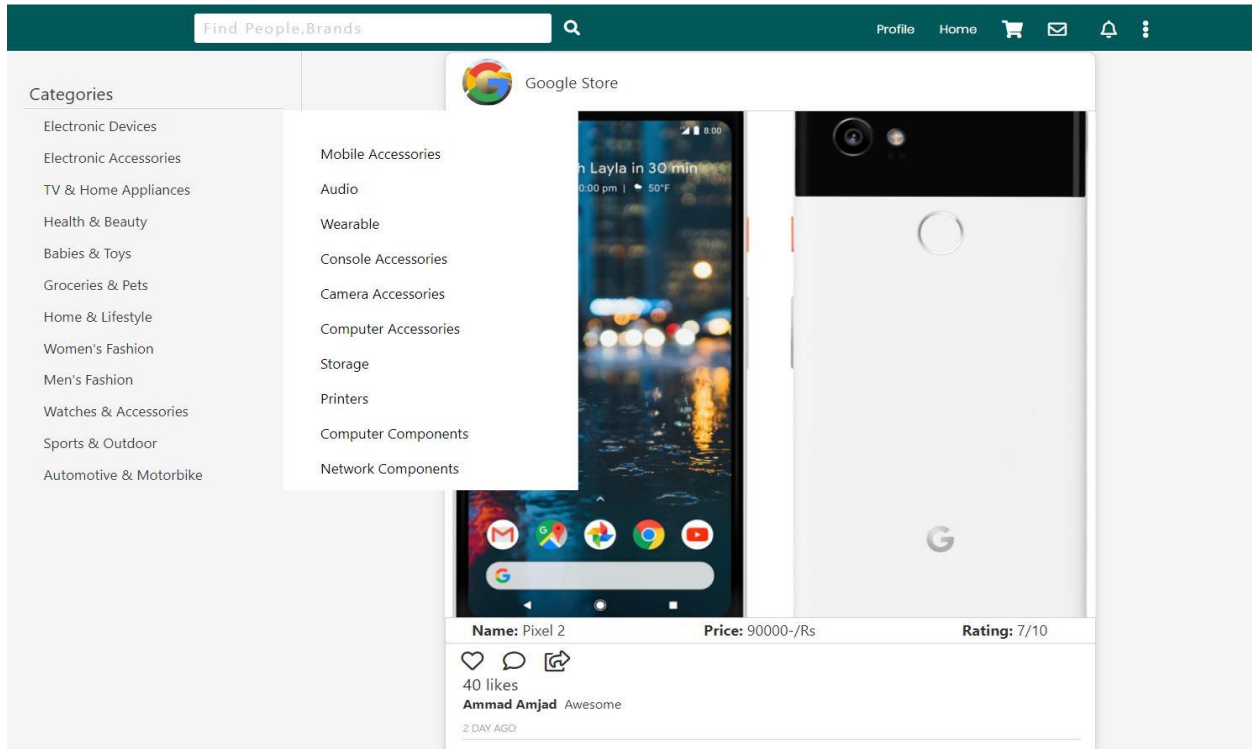


Figure 5.2 Category Diagram

Admin Dashboard:

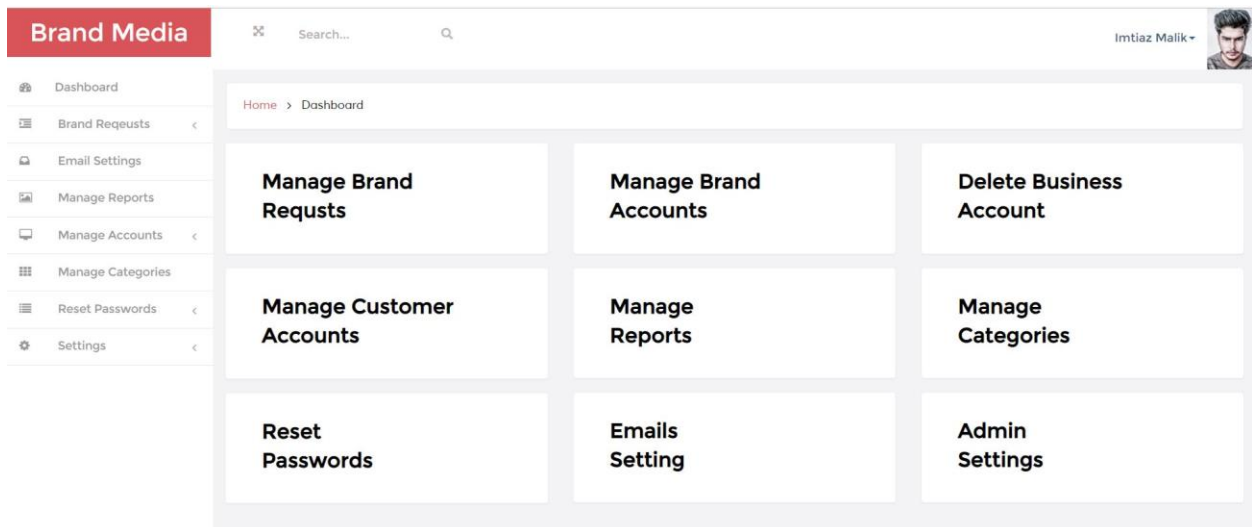


Figure 5.3 Admin Dashboard

Add New Profile Photo:

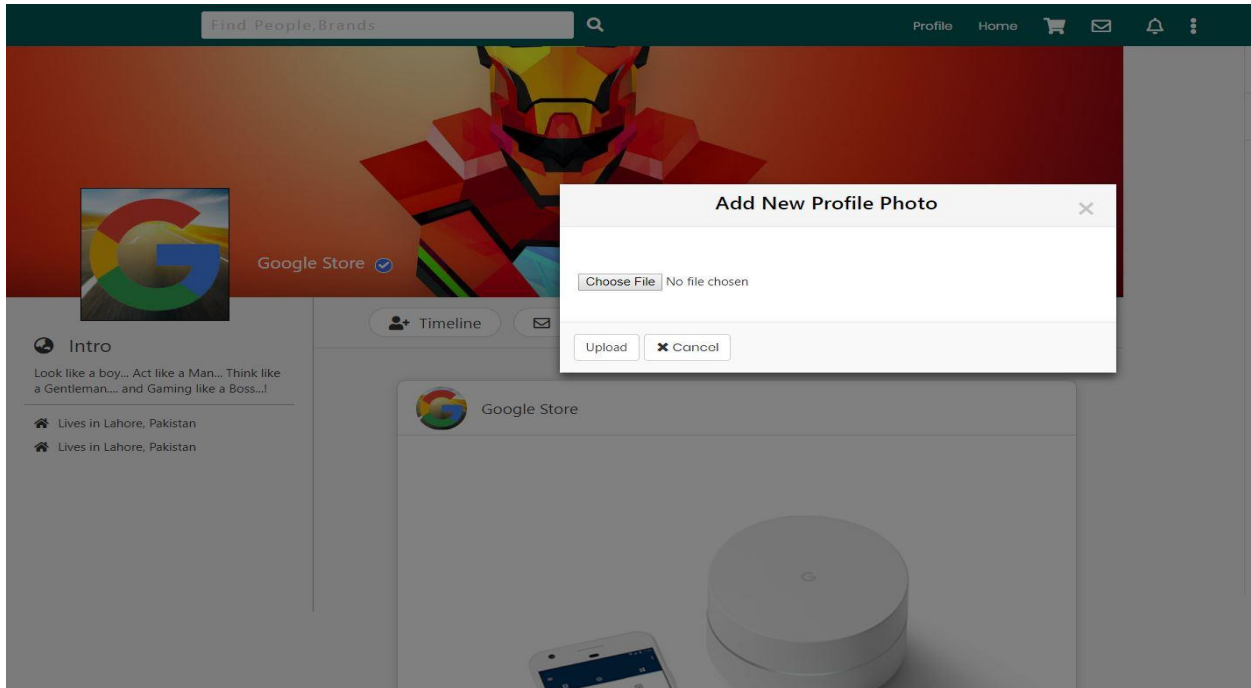


Figure 5.4 Add New Product

Add New Product:

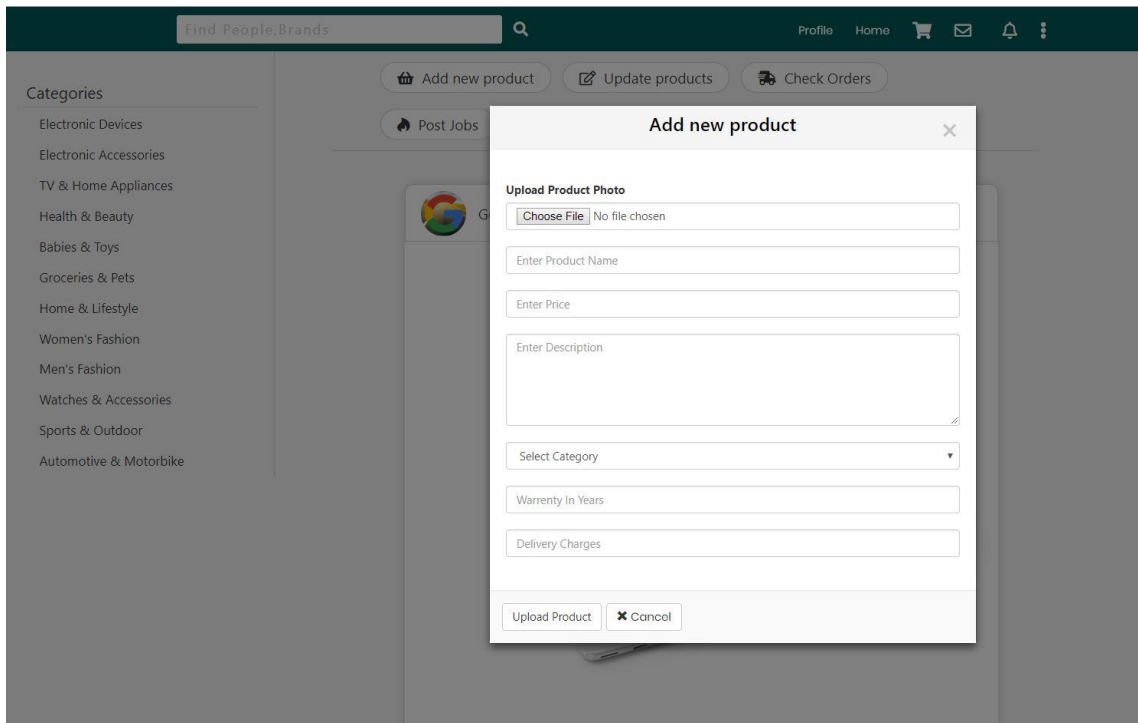


Figure 5.5 New Product Added

Newsfeeds:

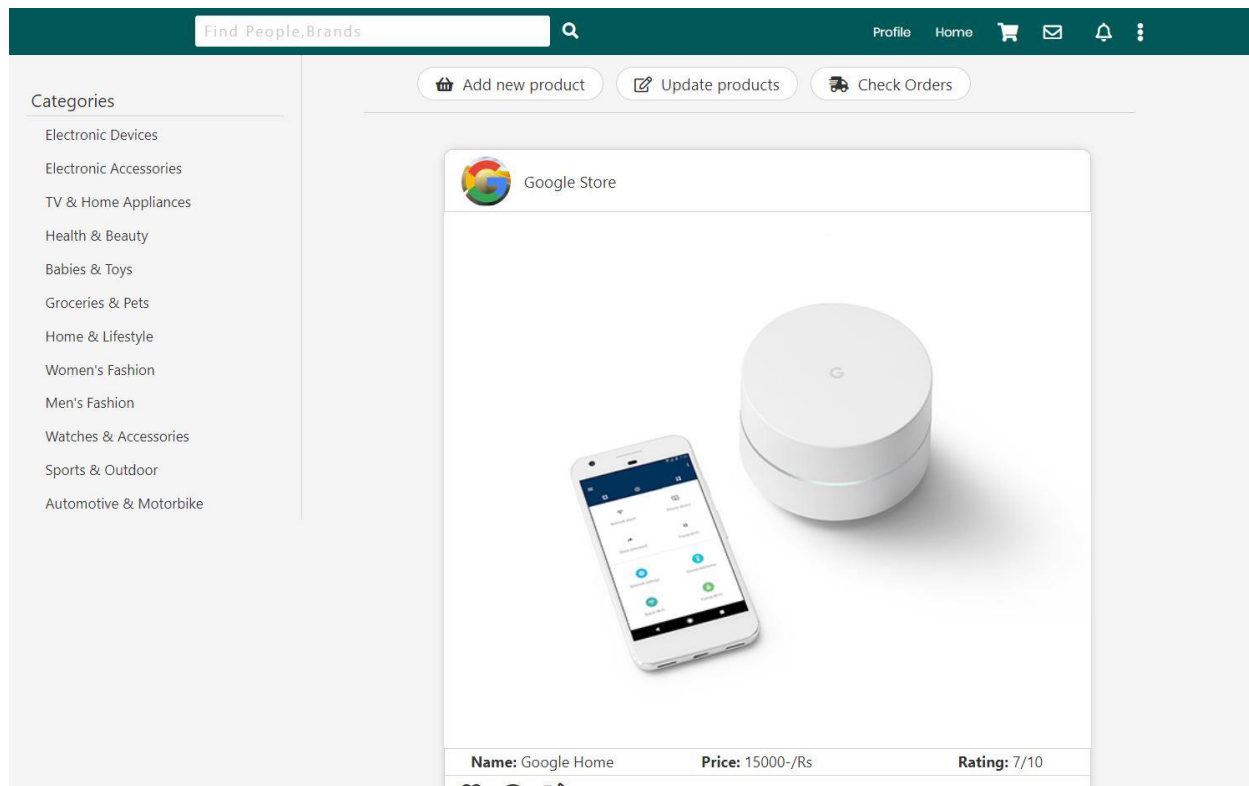


Figure 5.6 NewsFeeds

Profile:

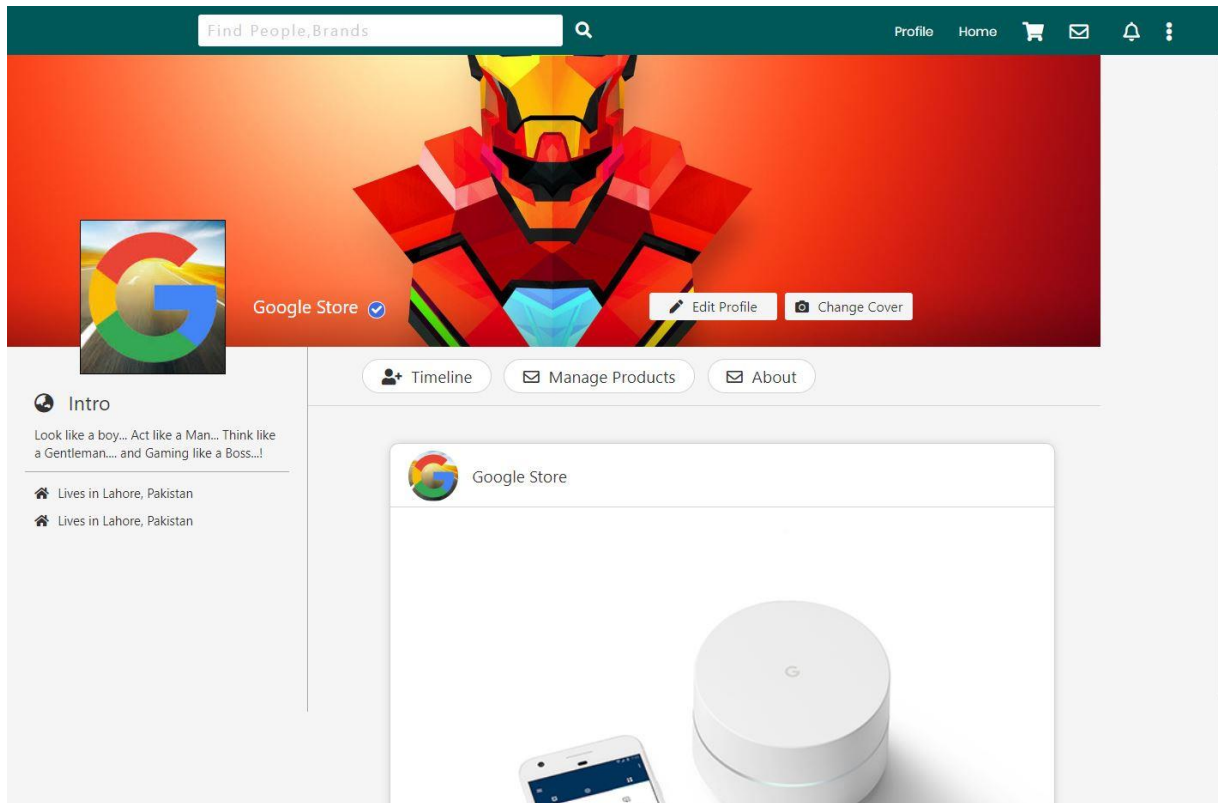


Figure 5.7 Profile

Search Result:

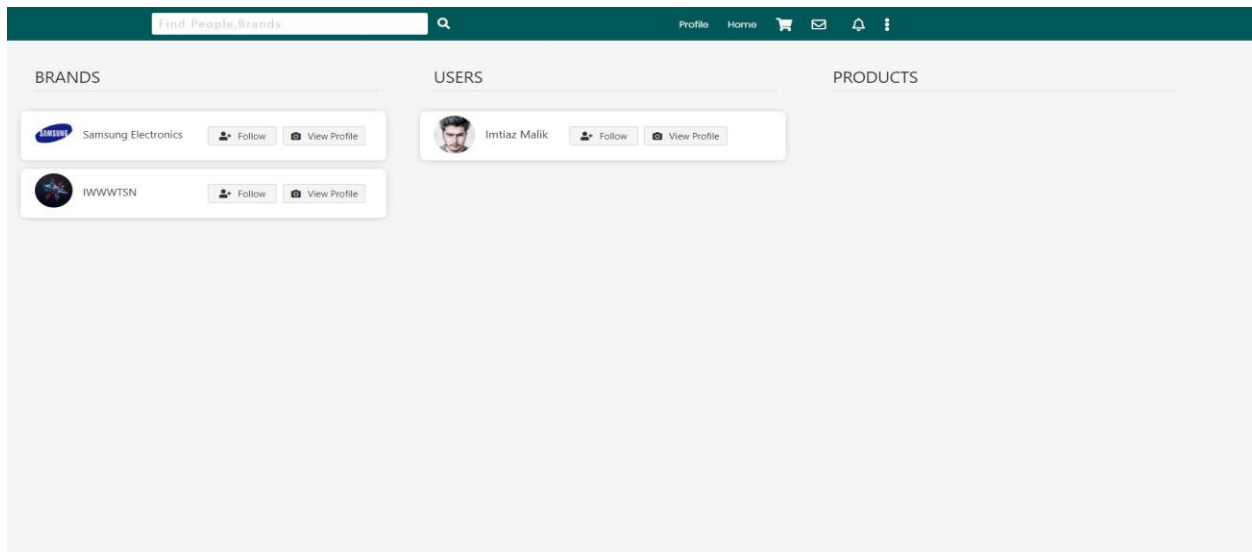


Figure 5.8 Search Result

Registration:



Sign Up

First Name
Name_

Last Name
Name_

Email
Email address_

Phone
Username_

Password

Choose Account Type
 Customer Account
 Business Account

I agree to the [Terms of User](#)

[Sign Up](#) [Sign in](#)

Figure 5.9 Registration

5.4. Tools and Techniques

Tools

Web Development

Front End

HTML5, CSS3, Ajax, JQuery

Backend

PHP, Codigniter

Centralized Database for web and Android

MySQL

Android Development

Java Native Android App using Android Studio

Developing Environments

Sublime, Notepad ++, Android Studio, Google Chrome

Techniques

We used the object programming techniques to develop a system. In android application we make standard algorithms to do transactions with MYSQL central database. These algorithms includes the APIs that are useful to do transactions with MYSQL database.

5.5. Best Practices / Coding Standards

We use the PHP 7.x in website and we used JAVA 8.x in developing android application. We use the object oriented programing languages for both website and android application. We follow the good standard for coding. Class name is a noun, attributes of classes are nouns, and method name start with verb. Like in class diagram one class name is "CustomerUser" class the attribute is "Name" and their function name is "setName(n)". Class name's first letter start with capital letter, the attribute's first letter start with capital letter, and method name's first letter start with small letter. We use the relationship between classes. These relationships are "association, inheritance, composition, and aggregation." These coding standard are helpful for us to generate a code for system components.

5.6. Version Control

For Version control and management of our source code for the project (Brand Media) we are using GitHub platform.

Repository name: brandmedia

Team members: (Imtiaz Malik, Faizan Anwar, Ammad Amjad)

Repository Link: <https://github.com/iwwwtsn/brandmedia>

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

First to understand more clearly you need to break a software product into several parts. First to test software solution this involve the comparing the solution with the objectives and design specifications of the original software idea. There are essentially two ways to test this solution framework both black box and white box testing. Black box is also known as functional testing the inputs and outputs are known but the processes are not. White box testing is also known as structural testing which is a testing with the internal working of item. There are four different types of testing these are unit testing, use case testing, data flow testing, integration testing. There are many types of online and offline software to test and evaluation of software project.

6.1. Use Case Testing

Test Case	Step Details	Test Data	Expected Result	Actual Result	Pass/Fail
Login	Navigate to brandmedia.tk		Site should be open	As Expected	Pass
	Enter Userid	Email = iwwwtsn@gmail.com	Email can be entered	As Expected	pass
	Enter Password	Pass = 077	Password can be entered	As Expected	pass
	Enter login		login should be successful	As Expected	pass
Account Register	Navigate to		Site/app should be	As Expected	pass

	brandmedia.tk		opened		
	Display registration message		Message should be displayed	As Expected	pass
	Enter First Name	Name = Faizan	First name should be entered	As Expected	pass
	Enter Last Name	Last Name= Anwar	Last name should be entered	As Expected	pass
	Enter email	Email= fa35@gmail.com	Email should be entered	As Expected	pass
	Enter password	Pass = 123	password should be entered	As Expected	pass
	Enter again password	Pass = 123	Re pass should be match	As Expected	pass
	Enter Phone	03204835425	Phone should be entered	As Expected	pass
	Enter Register		Registration should be successful	As Expected	pass
Admin login	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Enter Account Name	Account= Imtiaz Malik	Account should be displayed in search list	As Expected	pass
	Search Account		User profile should be	As Expected	pass

			opened		
Manage Cart	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Click on "Add into cart"	Id=1	Product with price and quantity should be displayed on Cart	As Expected	pass
	Remove product From cart	Id=1	Product should be removed from cart	As Expected	pass
Order	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Click on "Add into cart"	Id=1	Product with price and quantity should be displayed on Cart	As Expected	pass
	Order the product to brand		Map should be displayed with current location of that user	As Expected	pass
	User set	Latitude = 71 &	Marker	As	pass

	the marker	Longitude = 73	should be displayed on given location	Expected	
	Place order		Order will be sent and reached to brand account.	As Expected	Pass
View Notification	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Enter Account Name	Account= Imtiaz Malik	Account should be displayed in search list	As Expected	pass
	Search Account		User profile should be opened	As Expected	pass
	Follow User		Following should be started	As Expected	pass
	Check Follow Request		Followers request displayed	As Expected	pass
Block Account	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Enter	Account= Imtiaz	Account	As	pass

	Account Name	Malik	should be displayed in search list	Expected	
	Search Account		User profile should be opened	As Expected	pass
	Block User		Account should be blocked for blocked user	As Expected	pass
Follow Account	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email = fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Enter Account Name	Account= Imtiaz Malik	Account should be displayed in search list	As Expected	pass
	Search Account		User profile should be opened	As Expected	pass
	Follow User		Following should be started	As Expected	pass
	Check Follow Request		Followers request displayed	As Expected	pass
	Decline request		Follower item should be removed from list	As Expected	pass
	Follow	Navigate to brandmedia.tk		Site/App should be opened	As Expected

	ia.tk				
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Enter Account Name	Account= Imtiaz Malik	Account should be displayed in search list	As Expected	pass
	Search Account		User profile should be opened	As Expected	pass
	Follow User		Following should be started	As Expected	pass
	Check Follow Request		Followers request displayed	As Expected	pass
	Accept Request		Follower should be displayed on list	As Expected	pass
Unfollow User	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Open profile	Account=Faizan Anwar	Account should be opened	As Expected	pass
	Open Followers list		List should be displayed	As Expected	pass
	Unfollow User		User must be removed from follower	As Expected	pass

			list		
Register Brand	Navigate to brandmedia.tk		Site/app should be opened	As Expected	pass
	Display registration message		Message should be displayed	As Expected	pass
	Enter First Name	Name = Deni	First name should be entered	As Expected	pass
	Enter Last Name	Last Name= Zone	Last name should be entered	As Expected	pass
	Enter email	Email= dz@gamil.com	Email should be entered	As Expected	pass
	Enter password	Pass = 123	password should be entered	As Expected	pass
	Enter again password	Pass = 123	Re pass should be match	As Expected	pass
	Enter Phone	03204835425	Phone should be entered	As Expected	pass
Manage Orders	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Open profile	Account=Deni Zone	Account should be opened	As Expected	pass
	Click on manage order		New Orders must be displayed	As Expected	pass

			into list		
	User will deliver order	Id=1	Order marked to delivered	As Expected	pass
Add Product	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email =fa35@gmail.com Pass=123	Login should be successful	As Expected	pass
	Open profile	Account=Deni Zone	Account should be opened	As Expected	pass
	Click on Manage Products	Bid=1	Products should be displayed	As Expected	pass
	Click Add Product		Product Form should be displayed	As Expected	pass
	Enter Product Image	Image=galaxyjf.jpg	Image should be entered	As Expected	pass
	Enter name	Name=JF	Name should be entered	As Expected	pass
	Enter quantity	Quantity=4	Quantity should be entered	As Expected	pass
	Enter description	Description=this is the good mobile.	should be entered	As Expected	pass
	Enter price	30000	Price should be entered	As Expected	pass
	Enter category	Mobiles	Category should be	As Expected	pass

			entered		
	Click on Add		Product should be stored and displayed into list	As Expected	pass
Update Product	Click on Manage Products	Bid=1	Products should be displayed	As Expected	pass
	Click update icon	Pid=2	Product Form should be displayed with previous details	As Expected	pass
	Enter Product Image	Image=galaxy jf.jpg	Image should be entered	As Expected	pass
	Enter name	Name=JF	Name should be entered	As Expected	pass
	Enter quantity	Quantity=4	Quantity should be entered	As Expected	pass
	Enter description	Description=this is the good mobile.	should be entered	As Expected	pass
	Enter price	30000	Price should be entered	As Expected	pass
	Enter category	Mobiles	Category should be entered	As Expected	pass
	Click on Update		Product should be updated and displayed into list with updated information	As Expected	pass

Delete Product	Click on Manage Products	Bid=1	Products should be displayed	As Expected	pass
	Click Delete Icon	Pid=2	Product should be deleted	As Expected	pass
Approve Brand	Navigate to brandmedia.tk		Site/App should be opened	As Expected	pass
	Login account	Email= admin@gmail.com Pass=admin123	Login should be successful	As Expected	pass
	Enter Account Name	Account= Imtiaz Malik	Account should be displayed in search list	As Expected	pass
	Click on Request List		List must be opened	As Expected	pass
	Approve request	Bid=1	Brand page should be approved	As Expected	Pass
	Open Page	Bid=1	Page opened and brand should have access to his functions	As Expected	Pass
Manage Reports	View Report	Uid=2	User message will displayed	As Expected	pass
Delete Brand	Delete Brand	Bid=1	Brand page should be deleted	As Expected	pass
Delete Customer	Delete customer	Cid=1	Customer page should be deleted	As Expected	pass
Process	Enter	comment="this is	Comment	As	pass

Comment	Comment	nice product with good features”	must be entered	Expected	
	Click on Comment icon	Uid=4	Comment should be displayed into comment list	As Expected	pass
Like Product	Click on Like Button/icon	Pid=1 & uid=2	“liked” should be showed	As Expected	pass
Share Product	Click on share Button/icon	Pid=1 & uid=2	Shared product should be stored on timeline	As Expected	pass

Table 6.1 Use Case Testing

6.2. Equivalence partitioning

For customers Sing-up:

Attributes	Valid Partition	Invalid Partition
Name	All characters between 1 to 50	Below 1 and above 50 will not accepted
Password	Any type numbers and alphabet will be accepted between 1 to 50	Below 1 and above 50 will not valid in the system or any other slash, dot, comma will not accepted
E-mail	Must be write in @gmail.com	Alphabet or numbers invalid
Description	Characters between 1 to 250	Below 1 or above 250 invalid
Phone number	11 numbers	Below 1 and above 11 invalid

Login System:

Attributes	Valid partition	Invalid partition
E-mail	Must write "@gmail.com"	Incorrect e-mail is invalid
Password	Password between 1 to 50	Incorrect password invalid

Manage orders:

Attributes	Valid partition	Invalid partition
Orders	Between 1 to 10 in one day	Below 1 or above 10 invalid
Price	Between 100 to 100000	Below 100 and above 100000 is invalid for the system
Delivery	Between 1 to 6 days	Below 1 or above 6 days is invalid for the system

For brand sign-up:

Attributes	Valid partition	Invalid partition
Name	Characters between 1 to 50	below 1 or above 50 invalid
E-mail	Must write in "@gmail.com"	Alphabet and numbers invalid
Password	text between 1 to 50	below 1 or above 50 invalid
Phone number	Must write 11 numbers	below 11 or above 11 invalid
Description	Characters between 1 to 250	below 1 or above 250 invalid

Order manage by brands:

Attributes	Valid partition	Invalid partition
Orders	Between 1 to 1000	below 1 or above 1000
Price	between 100 to 1000000	below 100 above 1000000

Follow brands and friends by customer:

Attributes	Valid partition	Invalid partition
Follow brands and friends	Between 1 to 500	below 1 or above 500

Product upload by brand:

Attributes	Valid partition	Invalid partition
Product name	between 1 to 50	Below 1 or above 50
Product category	Between 1 to 50	Below 1 or above 50
Product image	Between 100 to 2500 kbs	Below 100 or above 2500 kbs
Description	Between 1 to 250	Below 1 or above 250
Stock	Between 1 to 1000	Below 1 or above 1000
product condition	New product	Old and refurbished
Product rating	Between 1 to 10	Below 1 or above 10

Table 6.2 Equivalence Partitioning

6.3. Boundary value analysis

For customers Sing-up:

Attributes	Invalid partition	Valid partition	Invalid partition
Name	Below 1 char invalid	Between 1 to 50 char	Above 50 char
Password	Below 8 char invalid	Between 8 to 50 char	Above 50 char
E-mail	Below @gmail.com	Must in @gmail.com	Above @gmail.com
Description	Below 1 char	Between 1 to 250	Above 250 char
Phone number	Below 11 char	Must write 11 char	Above 11 char

Login System:

Attributes	Invalid partition	Valid partition	Invalid partition
E-mail	Below @gmail.com	Must in @gmail.com	Above @gmail.com
Password	Below 8 char	Between 8 to 50 char	Above 50 char

Manage orders:

Attributes	Invalid partition	Valid partition	Invalid partition
Orders	Below 1 day is invalid	Between 1 to 10 in one day	Above 10 days is invalid
Price	Below 100 is invalid	Between 100 to 100000	Above 100000 is invalid
Delivery	Below 1 day is invalid	Between 1 to 6 days	Above 6 days

For brand sign-up:

Attributes	Invalid partition	Valid partition	Invalid partition
Name	Below 1 char invalid	Between 1 to 50 char	Above 50 char
Password	Below 8 char invalid	Between 8 to 50 char	Above 50 char
E-mail	Below @gmail.com	Must in @gmail.com	Above @gmail.com

Description	Below 1 char	Between 1 to 250	Above 250 char
Phone number	Below 11 char	Must write 11 char	Above 11 char

Order manage by brands:

Attributes	Invalid partition	Valid partition	Invalid partition
Orders	Below 1 order is invalid because 0 has no order	Between 1 to 1000	Above 1000 orders is invalid because it's out of stock
Price	Below 100 is invalid	100 to 1000000	Above 1000000

Follow brands and friends by customer:

Attributes	Invalid partition	Valid partition	Invalid partition
Follow brands and friends	Below 1 is invalid	Between 1 to 500	Above 500 is invalid

Product upload by brand:

Attributes	Invalid partition	Valid partition	Invalid partition
Product name	Below 1 is invalid	between 1 to 50	Above 50 is invalid
Product category	Below 1 is invalid	Between 1 to 50	Above 50 is invalid
Product image	Below 100 is invalid	Between 100 to 2500 kbs	Above 2500 kbs is invalid it's out of range
Description	Below 1 is invalid	Between 1 to 250	Above 250 is invalid
Stock	Below 1 is invalid	Between 1 to 1000	Above 1000 is invalid
product condition	Old product	New product	Refurnished product
Product rating	Below 1 is invalid	Between 1 to 10	Above 10 is invalid

Table 6.3 Boundary Value Analysis

6.4. Data flow testing

Modules				
Step no:	Step description	Expected result	Actual result	Status
01	Signup {function=user_signup()} Through function	Fetch user data from database	Show user result	Pass
02	Signup for brand {function=brand_signup()}	Fetch brand data from brand media database	Show brand result	Pass
03	Login for brand and user {function=login()}	Fetch e-mail and password	Show home page	pass
04	Upload product {function=upload_product()}	Insert data into server and fetch info	Show uploaded product	Pass
05	Follow brand and user Function=follow_all()	Fetch user id and brand id	Show all products on newsfeed	pass
06	Add to cart Function=add_cart()	Fetch brand products data	Show only these product that carted	Pass
07	Track orders Function=orders()	Fetch orders data	Show orders to user track order page	pass
08	Notification Function=notification()	Fetch numbers of	Show all products on	pass

		products data	notification	
09	Logout Function=logout()	Fetch logout session data	Show sign-in page	pass

Table 6.4 Data Flow Testing

6.5. Unit testing

In the unit testing test each module of the software project individually. Follow a white box testing that test the logic of the program it is done by developer.

Sign-up account testing:

User and brand have the signup page for enter the required information. Basically the user and brand give the input to the system for entering the required information. After this system give user and brand the output of login system for user and brand both.

Sign Up

First Name
First Name...

Last Name
Last Name...

Email
Email address...

Phone
Phone...


Password

Chose Account Type

- Customer Account
- Business Account

I agree to the Terms of User

Sign Up Sign in

Powered by  anqwahhost

Login account testing:

When user and brand signup their account then after this the system automatically open the login system for user and brand then user and brand enter the valid information to login their account such as e-mail and password.

Sign In

Email

Email address...

Password

Remember me

Sign In

Sign up

Figure 6.1 Login Account Testing

Navbar testing:

After login with valid information system open the home screen for the user and brand. So first thing is navbar to show some option for user such as timeline, newsfeed, cart, track orders, notification and settings for logout.

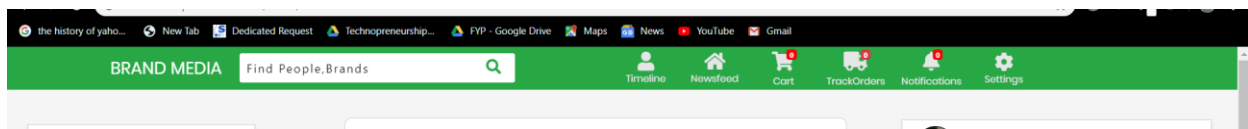


Figure 6.2 Navbar Testing

Categories:

In this function system will show the categories of shopping that what user want to buy the products and categories have many option such as Electronics devices, accessories, TV & Home appliances and more options. All these options system will show to the user home screen.

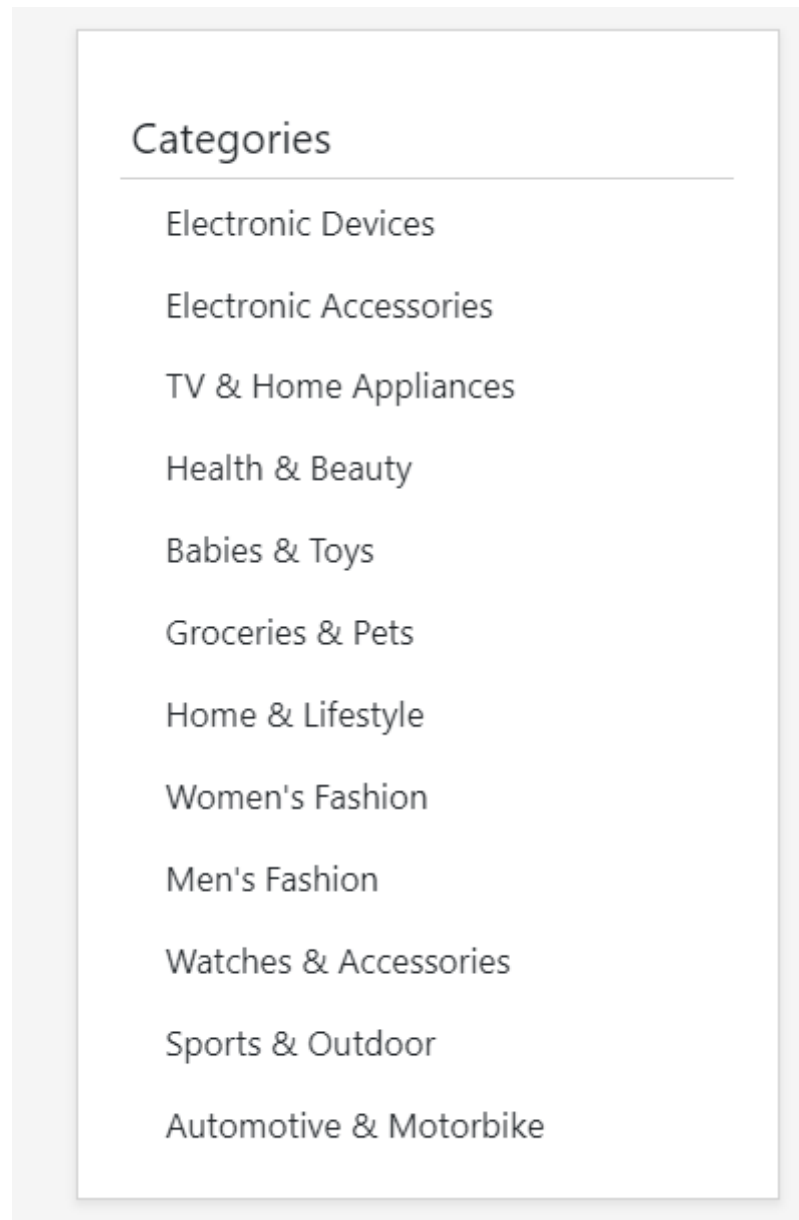


Figure 6.3 Categories Testing

Follow brands and friends:

User can follow brands and friends. When user follow the brand and friends then system will show all products of brand to the user's home screen.

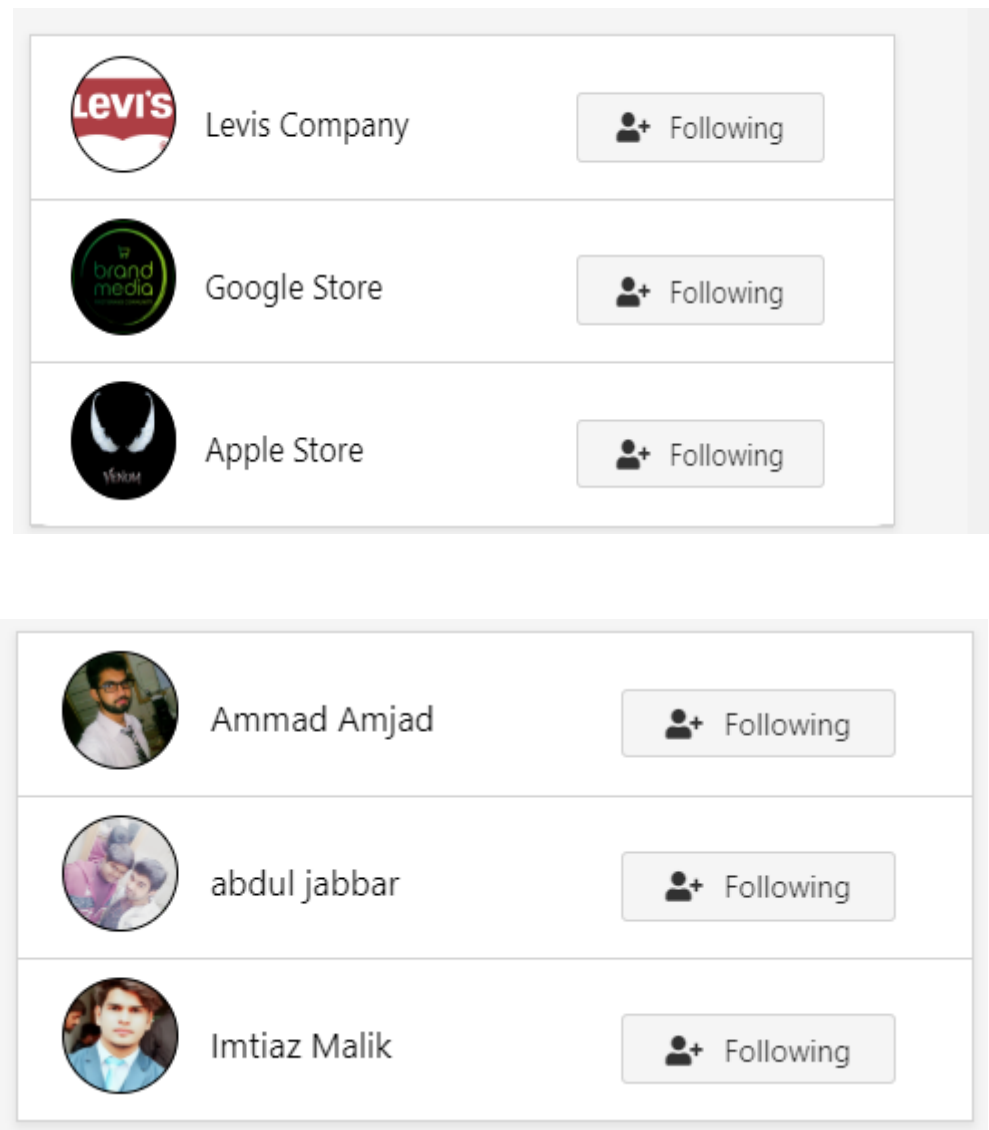


Figure 6.4 Follow Testing

Upload product:

When brand upload the product then the system will show the uploaded product to the newsfeed page of user and brand. This will happen every uploaded product again and again. No matter how many products uploaded by the brands.



Figure 6.5 Upload Product Testing

Add to cart testing:

When user click on the cart button then system will add product in the cart option page and save it for the user. After the add to cart option user can order any product and buy it. And system will calculate the amount.



DISCRPTION	BRAND	UNIT PRICE	QUANTITY	SELECT ITEMS
 <p>Levis T-shirt 2019 10 aood</p>	Levis Company	700/-PKR	1 ▾	<input type="checkbox"/>
 <p>Samsung Camera</p>	Google Store	18000/-PKR	1 ▾	<input type="checkbox"/>

Figure 6.6 Cart Testing

Order testing:

When user add the product in the cart system then system will give option to the user to buy these type of product or not. Order system is totally work on calculation amount on the run time and system give the total amount of product for the user

Location [Change](#)

📍 Currently orders can be deliver only in Pakistan

Order Summery

Subtotal (0 items) Rs.

Shipping Fee Rs.

ENTER VOUCHER CODE [Apply](#)

Grand Total Rs. 0

[Checkout](#)

6.6. Integration testing

Integration Testing is defined as a type of testing where software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. Integration Testing focuses on checking data communication amongst these modules.

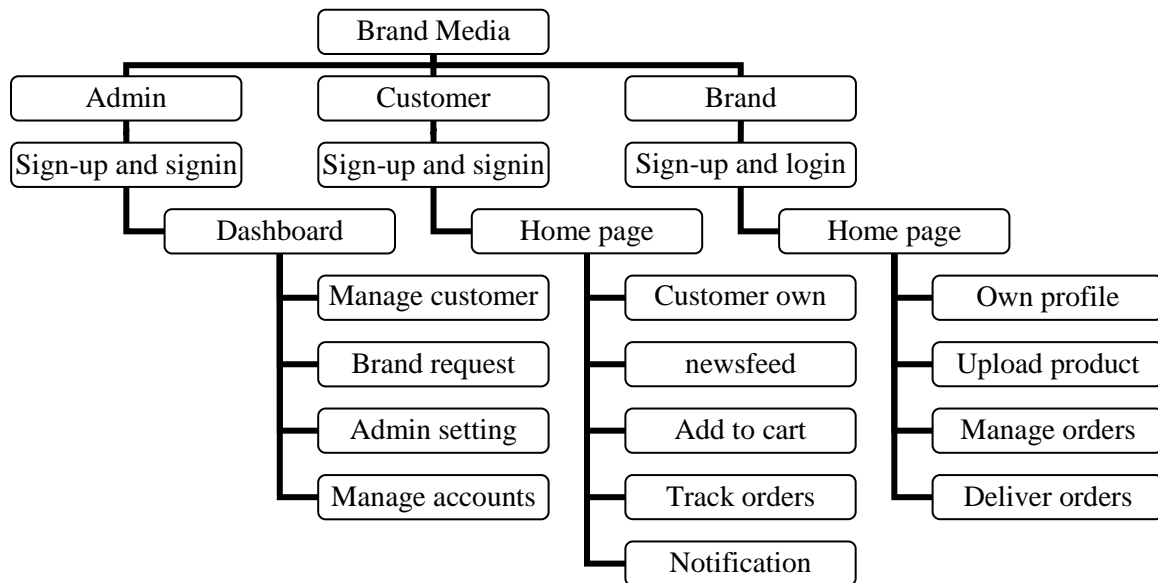


Figure 6.7 Integration Testing

Sign-up page:

In the sign-up page user can enter the required information for the sign-in their account because its compulsory to enter information

Sign Up

First Name
First Name_

Last Name
Last Name_

Email
Email address_

Phone
Phone_

Password

Chose Account Type
 Customer Account
 Business Account

I agree to the Terms of User

Sign Up Sign in

Powered by 000webhost

Sign-in page:

user and brand both have the valid e-mail and password to sign-in the dashboard

Sign In

Email
Email address_

Password

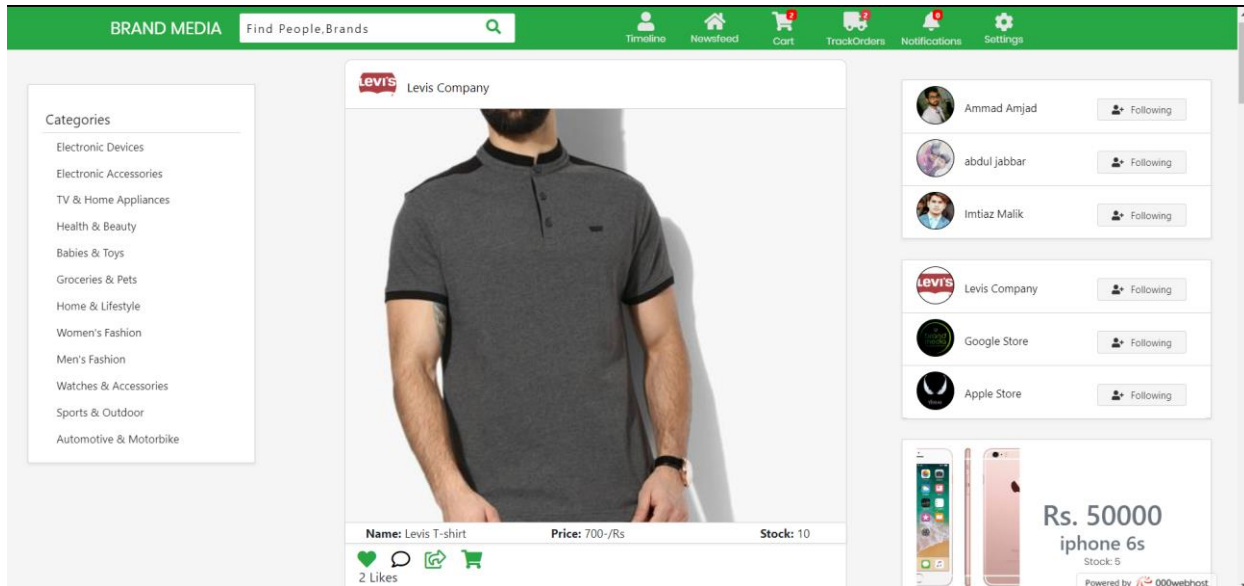
Remember me

Sign In Sign up

Powered by 000webhost

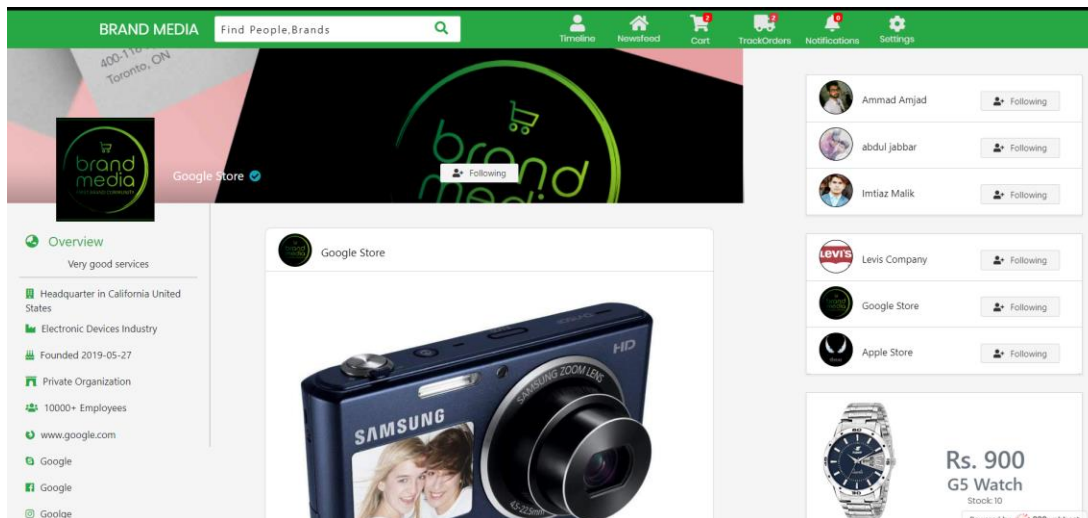
Customer Dashboard:

System will show the customer dashboard after the sign-in the account now user can use these functionalities and enjoy the dashboard. In this dashboard user can see the newsfeed and give comments, like, share and add to cart to any product.



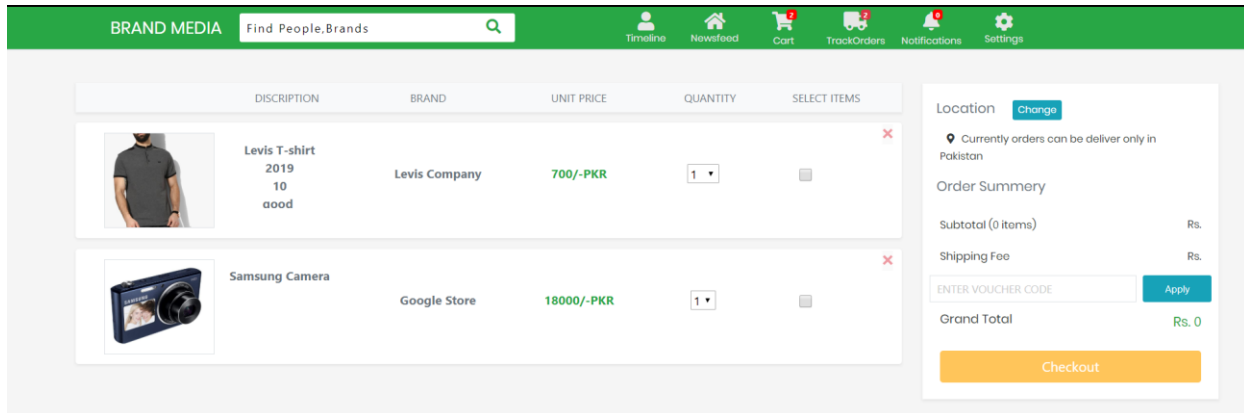
Brand profile page:

System will show the brand profile after the sign in the account in this home page user can upload product see others brand products and manage the cart board manage track orders and many more features



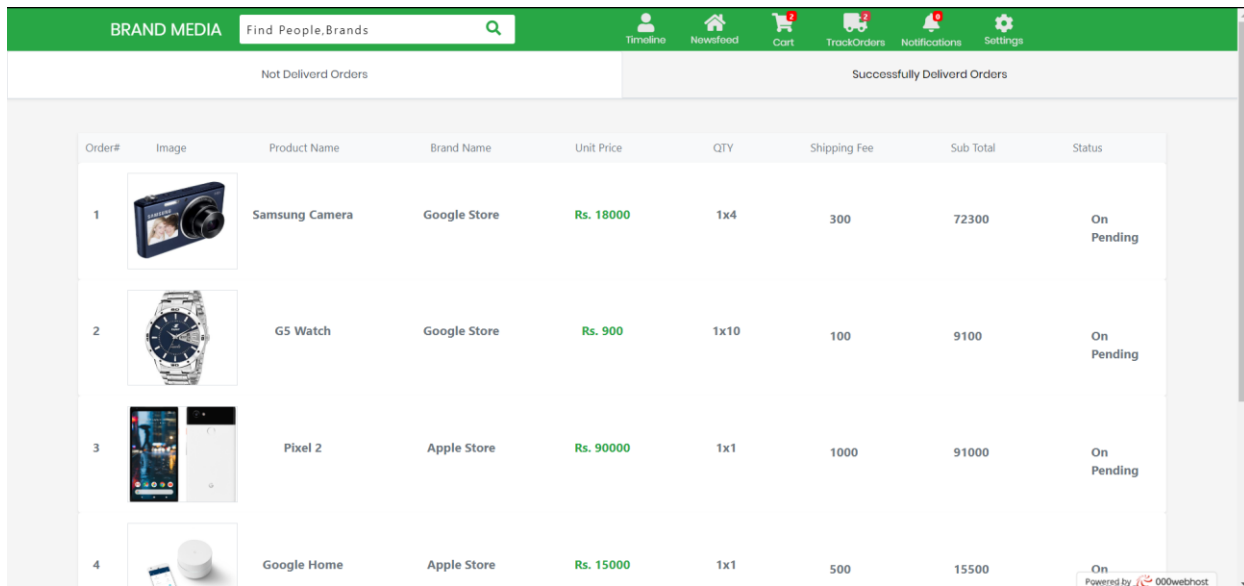
Add to cart page:

In this feature when user will add any product to the cart then system will show all the carted product on the cart board page and user can manage these products, delete it, edit it and order it.



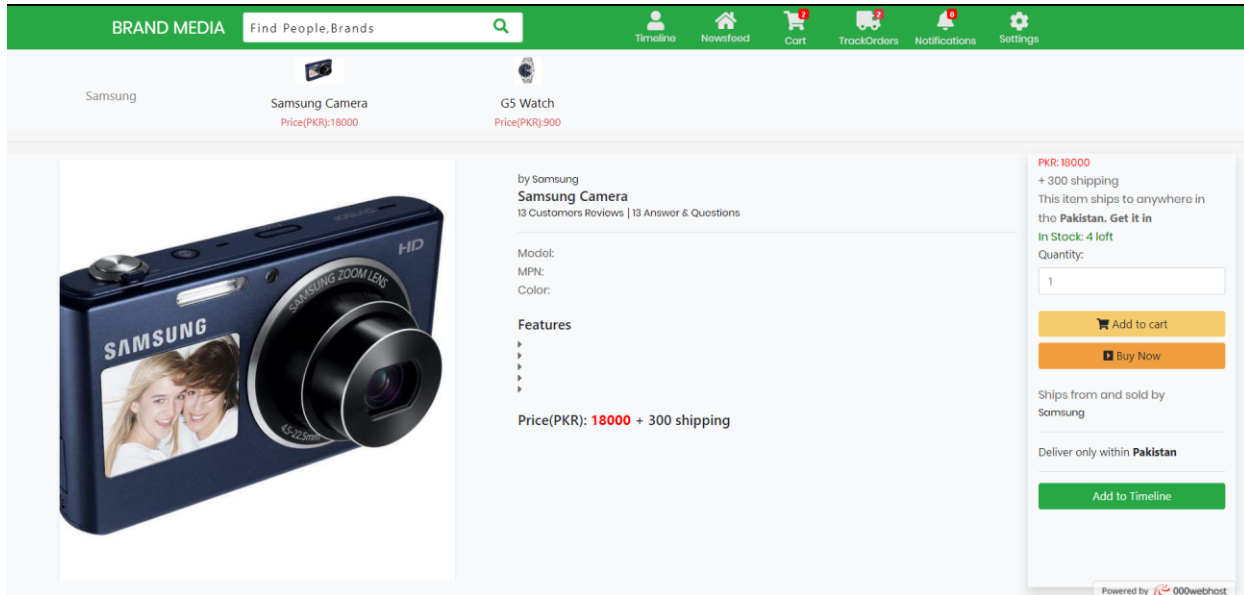
Track orders:

When customer can order any product then user also track the order done by user and user will see the distance between user and ordered product.



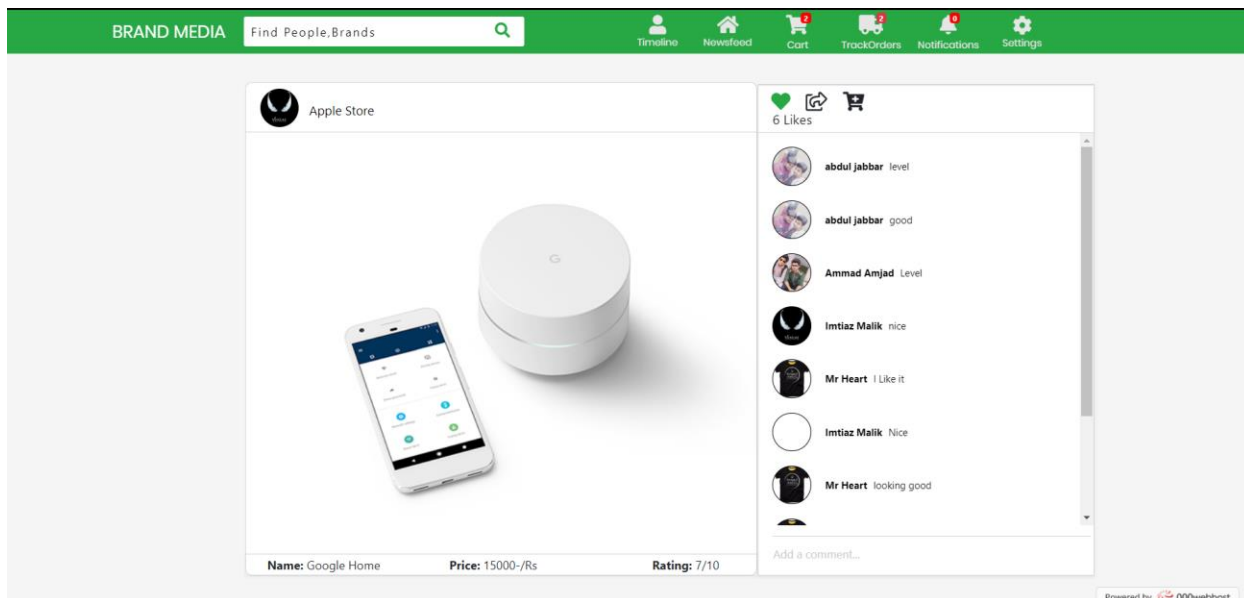
Product specifications:

When user click on the product then system will show all the specification of the products to the user for the better satisfaction.



Product details:

In this feature user can like product, comment, share and also discuss the detail about product and give rating to the product for better performance.



Add a new product:’

this is the main feather for brand. Basically this feather can use to upload the product and all the information required to upload any product. When the brand upload any product then system will add all the specification and show the product for the public view

Add A New Product

Vital Info	Offer	Images	Description
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Product Name*</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>Max. 250 characters</i></p> <input style="width: 90%; border: none; border-bottom: 1px solid #ccc;" type="text"/> <p style="text-align: center; font-size: 0.7em; margin: 0;"><i>Example: Nikon D5300 with 300mm Lens</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Manufacturer Part Number*</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>For Most products this will be identical to the model number; however, some manufacturer distinguish part number from model number.</i></p> <input style="width: 90%; border: none; border-bottom: 1px solid #ccc;" type="text"/> <p style="text-align: center; font-size: 0.7em; margin: 0;"><i>Example: EE</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Model Number*</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>Product code assigned by the manufacturer; can be numbers, letters, or both</i></p> <input style="width: 90%; border: none; border-bottom: 1px solid #ccc;" type="text"/> <p style="text-align: center; font-size: 0.7em; margin: 0;"><i>Example: D5300</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Category</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>Select Category</i></p> <div style="border: 1px solid #ccc; padding: 2px; text-align: center; font-size: 0.7em;">Select Category ▼</div> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Material Type</p> <input style="width: 90%; border: none; border-bottom: 1px solid #ccc;" type="text"/> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Color*</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>The Color of the Product</i></p> <input style="width: 90%; border: none; border-bottom: 1px solid #ccc;" type="text"/> <p style="text-align: center; font-size: 0.7em; margin: 0;"><i>Example: Red, Maro Black, Royal Blue</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p style="text-align: center; margin: 0;">Stock Quantity*</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>Quantity of Items</i></p> <input style="width: 90%; border: none; border-bottom: 1px solid #ccc;" type="text"/> <p style="text-align: center; font-size: 0.7em; margin: 0;"><i>Example: 10, 100</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <p style="text-align: center; margin: 0;">Gender*</p> <p style="text-align: center; font-size: 0.8em; margin: 0;"><i>Select from the list of valid values</i></p> <div style="border: 1px solid #ccc; padding: 2px; text-align: center; font-size: 0.7em;">Male ▼</div> <p style="text-align: center; font-size: 0.7em; margin: 0;"><i>Example: Men</i></p> </div>			
<div style="display: flex; justify-content: center; gap: 10px;"> Cancel Save and finish </div>			

6.7. Performance testing

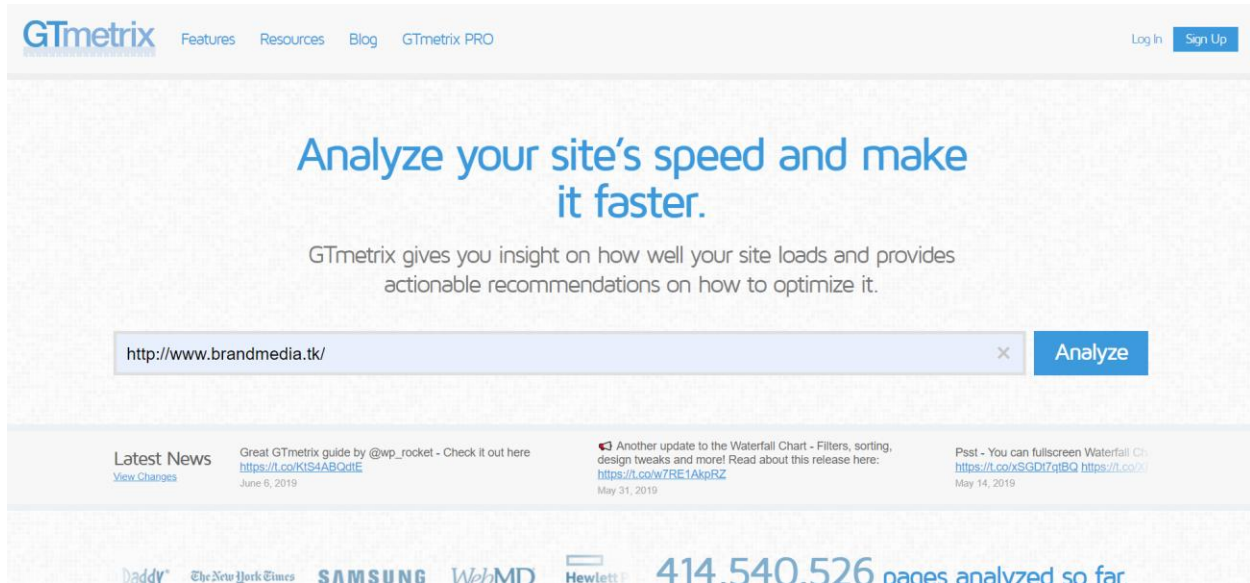
Performance Testing is defined as a type of software testing to ensure software applications will perform well under their expected workload.

There are many tools for testing the performance of web application such as jmeter, USS, new load and many more. To check performance testing by three types load testing, speed testing and scalability testing. There are many tools to test the performance of website testing. We use GTMetricx.


GTMetricx tool:

In this tool just enter the Url of any website and this will help you to test the performance of every page of website also tells the speed performance and page size performance. That helps

the performance testing of website. Basically this tool is very helpful for test the performance of data, the backend languages like PHP, Ajax, Javascript also test the frontend of website and test it correctly and generate the report of testing performance.



Performance score:



Latest Performance Report for:

http://www.brandmedia.tk/

Report generated: Tue, Jun 11, 2019 12:01 PM -0700

Test Server Region: 🇨🇦 Vancouver, Canada

Using: 🌐 Chrome (Desktop) 62.0.3202.94, PageSpeed 1.15-gt1.2, YSlow 3.1.8

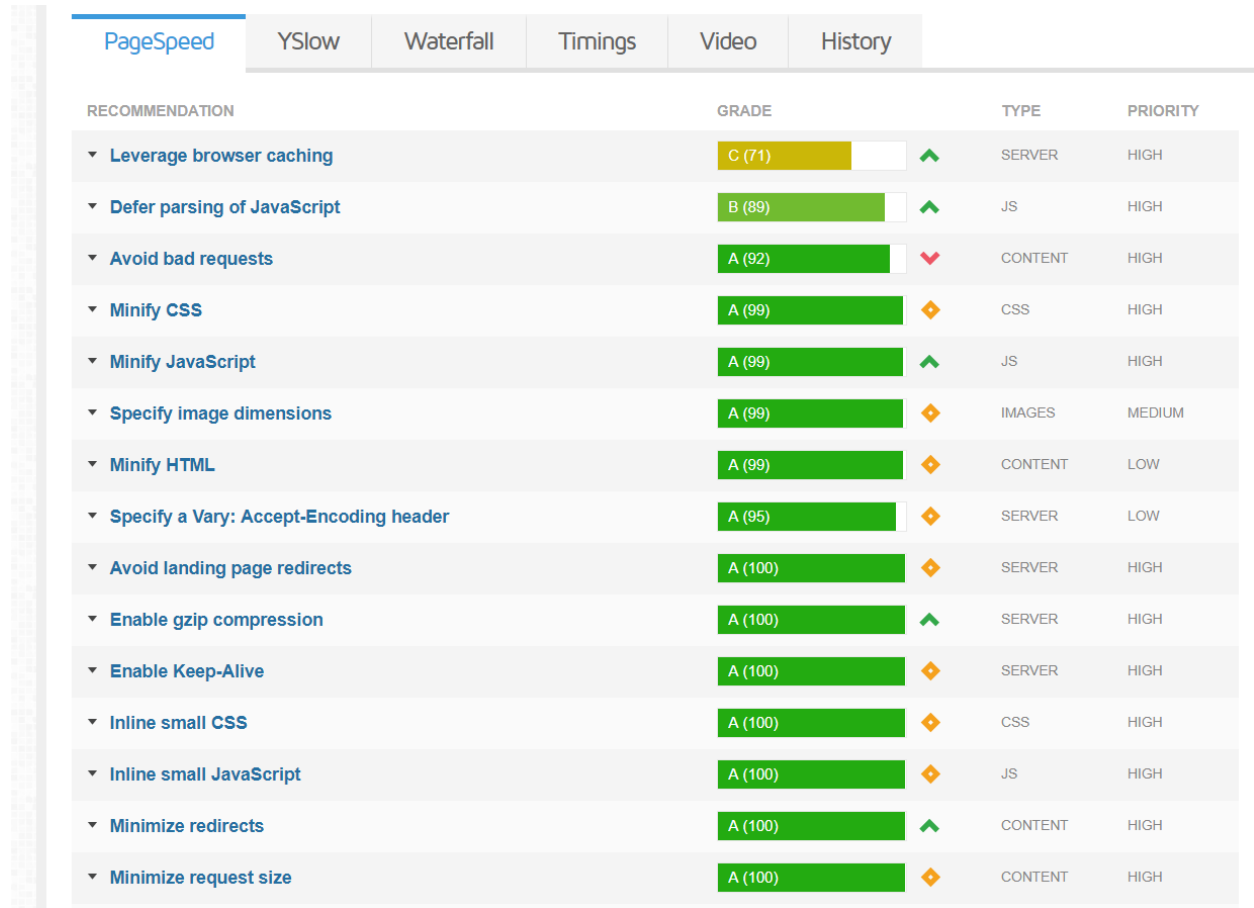
Performance Scores

PageSpeed Score A (94%) ^	YSlow Score B (88%) ^
---	---

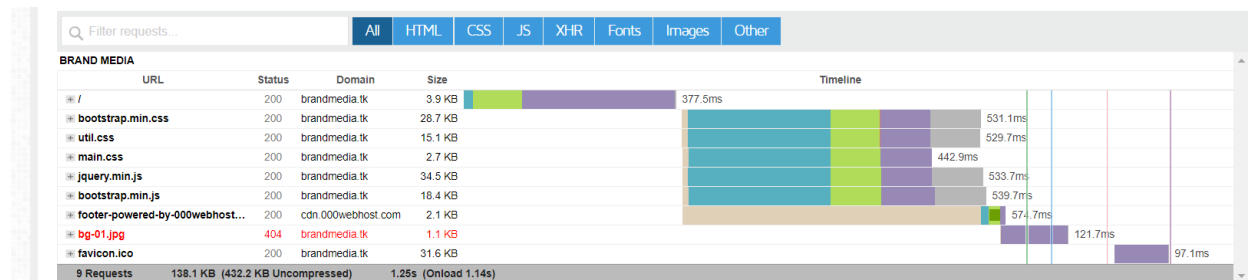
Page Details

Fully Loaded Time 2.1s ^	Total Page Size 138KB ^	Requests 9 ^
--	---	--

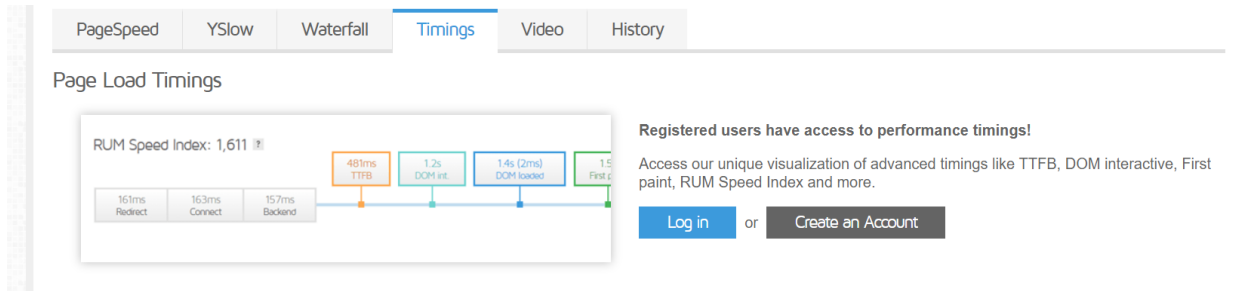
Page load speed:



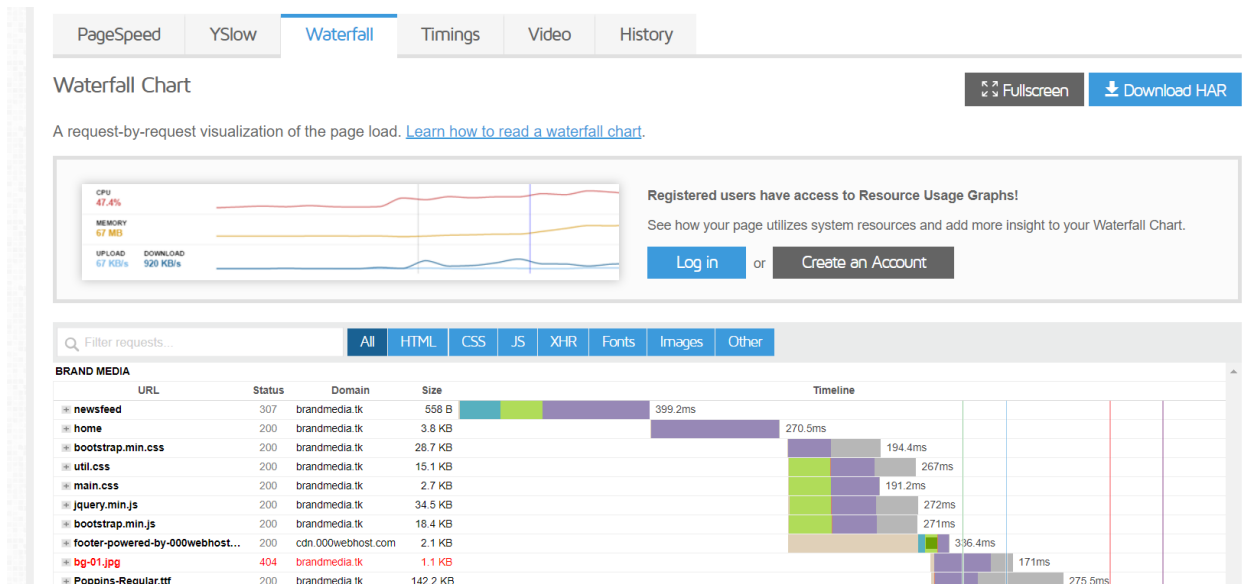
Frontend testing:



Timings of website pages:



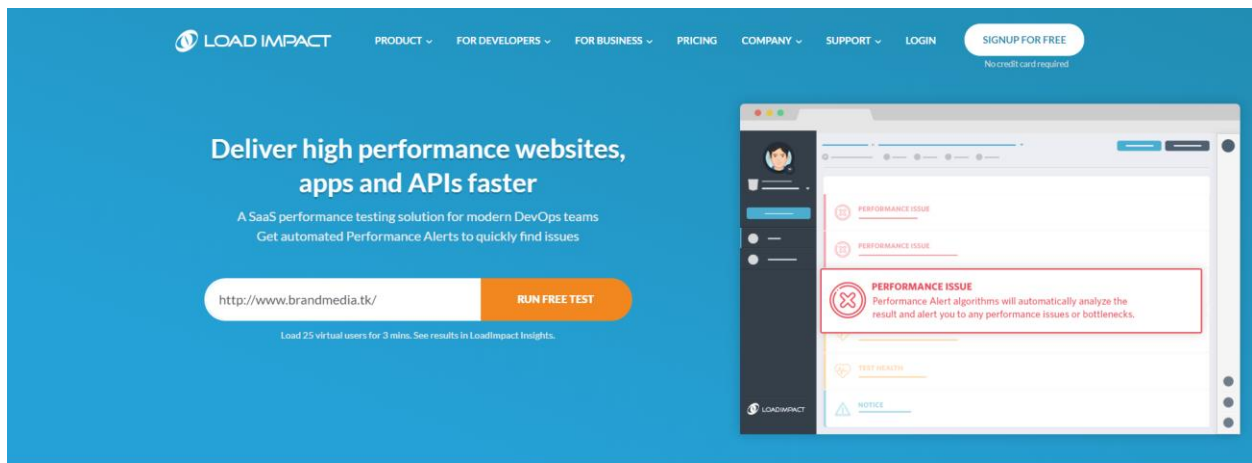
Testing by waterfall:



6.8. Stress Testing

In stress testing website take a load of user data brand data and many more such that orders, carted products. In mysql database just 25gb data can enter and after this database cannot handle this load and our website only handle 1 gb data because of free hosting and limited bandwidth that is 20 Gb. So when 10000 users and 10000 brands will enter our platform then its working great but when its exceed the data then system will slow.

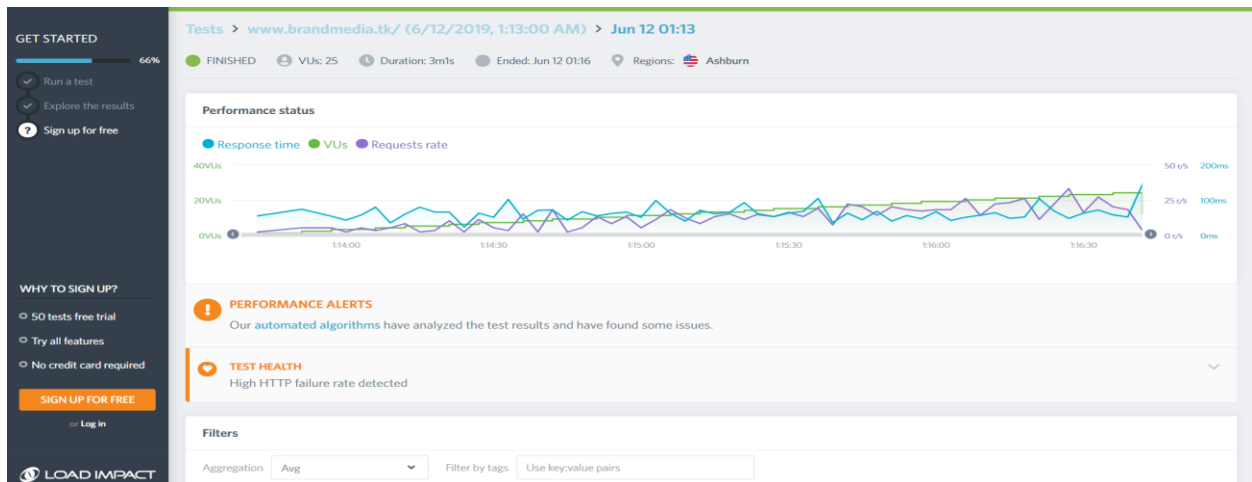
Now for checking stress testing we have online platform that is loadimpact.com that show the performance of the website load and stress performance



Next Generation Performance Testing

Reduce development costs, improve customer satisfaction and grow your revenue faster. A new approach that supports performance testing throughout the development lifecycle.

Stress result:



Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

We have Facebook for social media/community (A community where people interact with other peoples) and Amazon for e-commerce that is the largest Internet retailer in the world as measured by revenue and market capitalization and also LinkedIn that is a business and employment-oriented service. All of these platform have different functionalities, services and features. By inspiration all of these we are going to introduce a new online community/platform that is Brand Media. In Brand Media we give services (entertainment, ecommerce and professional networking). Basically we are going to combine (Amazon +Facebook + LinkedIn) in this project. The intention will to bridge the gap between Consumer and Brand and also to help Pakistani Entrepreneurs (start-ups) to promote their products and brands.

Many platforms like daraz.pk, olx.com, goto.pk. They provide shopping services in their website but not any Social services like Facebook. Actually people want to like, comment, share their favorite brand products and also to see the choice of other peoples like friends, family members and colleagues. In Brand Media community we have not allow to any whole sellers to sale products. Only Manufacturer /owner of the product can sale the products to consumer so people can facilitate with original quality and actual price of product. It will become World first brand community, where a consumer can fully engage with their favorite brands.

A community/platform of official brands where all brands can directly communicate with their consumers. A platform for consumers where they can meet with their needs and demands and get high quality products at one click. A community where quality is more important than money. To facilitate 193.2 million Pakistan's people with high quality product at low price. This platform will become very helpful for Pakistani Entrepreneurs and especially helpful for Superior Technopreneurship program. All students who will make innovative products in Technopreneurship event can be easily promote their product at global level and easily get customer attraction. This platform will be going very helpful to make Pakistan economically superior.

7.2. Achievements and Improvements

During this process of website development we learnt a lot of skills like :-

- Html3 and css5 for frontend
- Java script for more value able for frontend
- Also learn Ajax for run time data entering into the data server
- We learn how to create a social website
- We learn how to create the E-commerce website
- Also learn the creativity to combine the functionality of social web feather and E-commerce web feathers use in the same brand media platform
- We learn also how to create the dynamic website and load data in run time
- It's a great opportunity for us to learn a lot from this brand media platform

7.3. Critical Review

Brand media is the first platform to provide the functionality of social website and e-commerce website on the same platform. Also the first platform who gives the user free hand to achieve the goals and enjoy the shopping on the same platform so that we provide the best platform to user and brand. Brands have also the biggest opportunity to sell their products directly to the user not sell their product by any other market seller. This platform is very helpful for customer and brands to communicate with each other.

7.4. Lessons Learnt

We absorb very much from this project. This project sharpens our skills in the web development such that web designing tools and many management concept as well as how to deal with the problem and how to stick for finding the solution of any problem until you found. As well as technical skills this project also enhance our personal development skills such as team working, dedication. We learnt various types of techniques in web development for the creation of website. Our passion of making a unique website that contains the enjoyment element in an innovative environment and we succeeded it. We continue our work after this to

make it through worldwide. This project is a stepping stone for us in the beginning we don't even know that if we accomplished it or not but due to our hardworking and research we do it and it's an achievement for us that motivates us to do better and more.

7.5. Future Enhancements/Recommendations

- Use Artificial intelligence in our website and android application
- To search brand and user because of more data use of statistics and analysis of algorithm
- We will improve the performance of our platform
- We will improve the load time and speed of website and android application
- We will add more features in the future for better user friendly platform

Appendices

Appendix A: User Manual

In the appendix section we describe the different phases of user interface and also describe how user can use our project. Secondly we describe our promotional plan and promotional materials like broacher, banner, standee and other marketing materials. We also describe other interfaces of our game project.

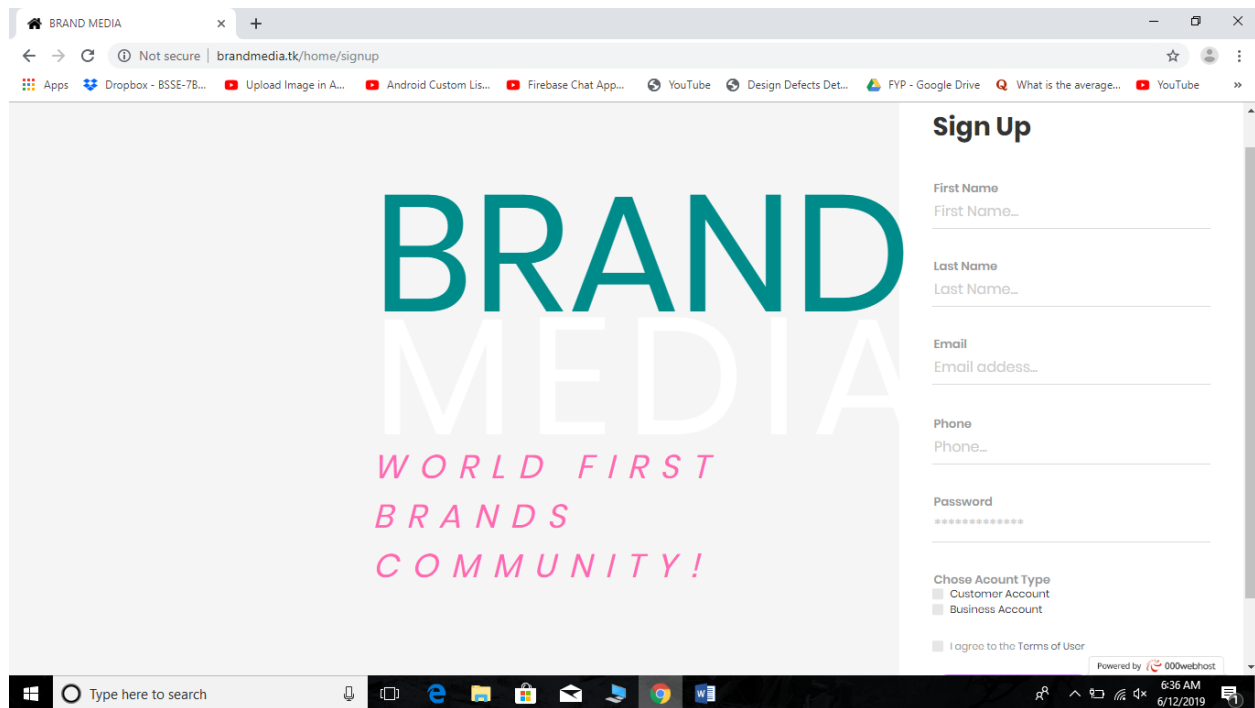
Appendix A: Login Authorization

This is the first activity of brand media. In which user first login the account for do his/her functions provided by the Brand Media. If user has not account then he/she will create their accounts.



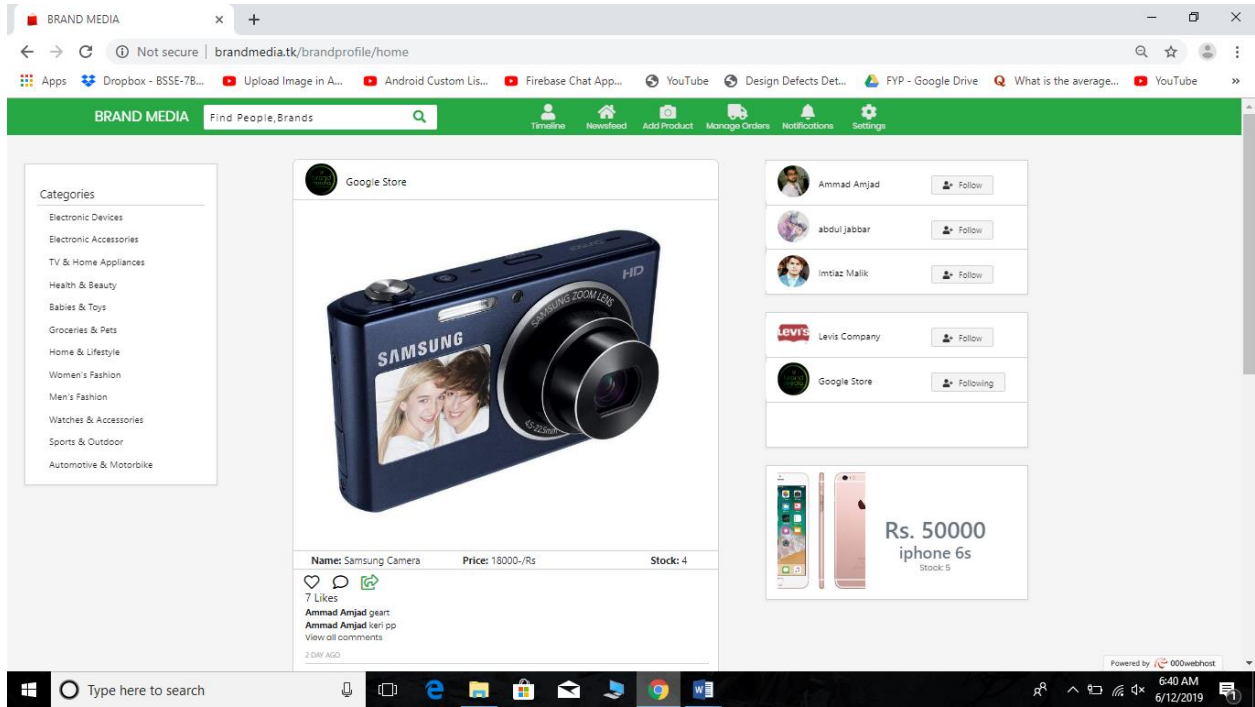
A.1. Create Account

If brand want to create the account then he create the account. When he created the account then he will wait for approval from brand. If Customer want to create the account for buying the products of brand then he/she will create account. Customer account will be registered without approval.



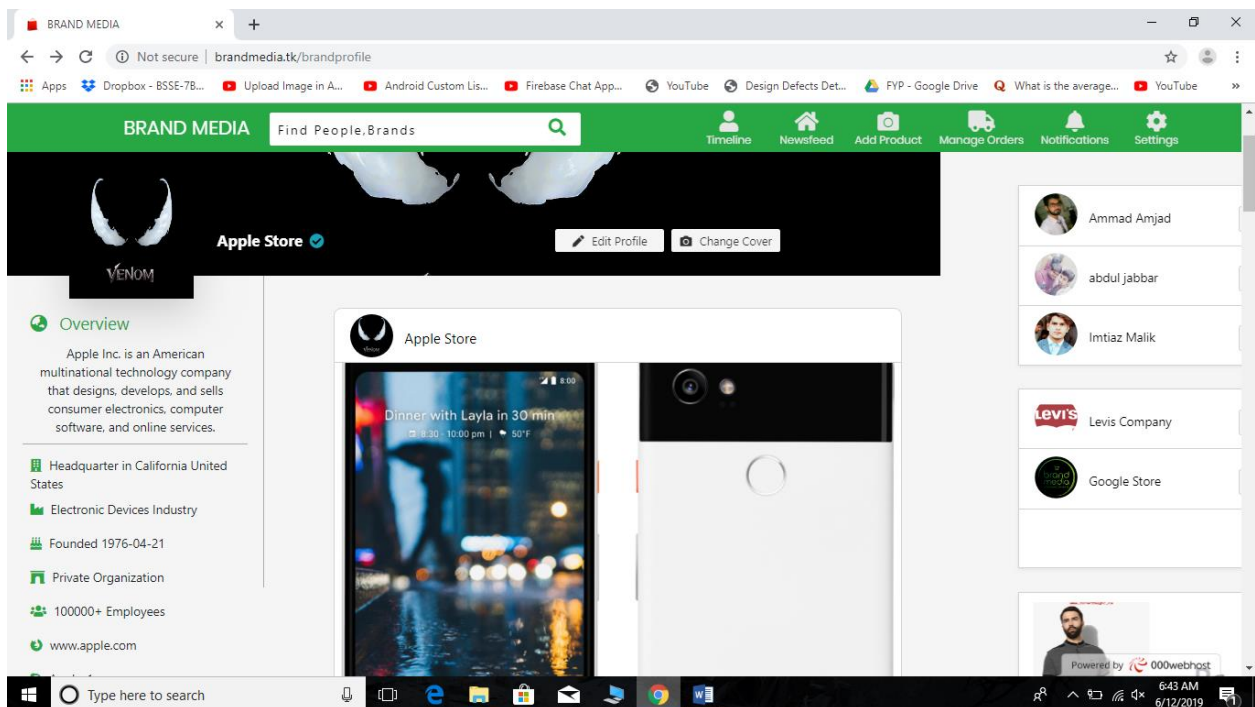
A.1.1. Newsfeed

When user create his/her account then he will go to the home page. This page provide the news feed for users. In this newsfeeds user will be updated for new products of his/her followers. Also user will like, share, comment and put products into the cart.



A.1.1.1. Profile

In profile user can view the time line. Also user can view his/her bio data and can change the bio data. User can change the profile and cover photos.

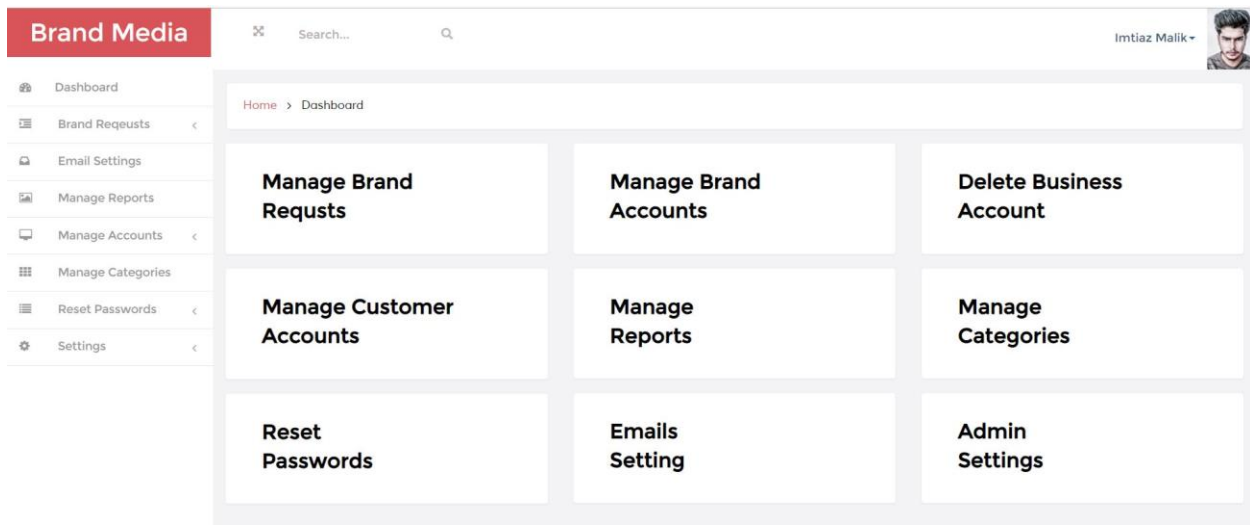


Appendix B: Administrator Manual

Admin manage all the components of brand media. Admin will approve the request of brand pages. Admin can change or delete the accounts of users. Admin can add the categories of products.

B.1. Login-Dashboard

Admin login his account. Admin will see the dashboard. In dashboard admin will manage the accounts of users



B.1.1. Second level heading

N/A

B.1.1.1. Third level heading

N/A

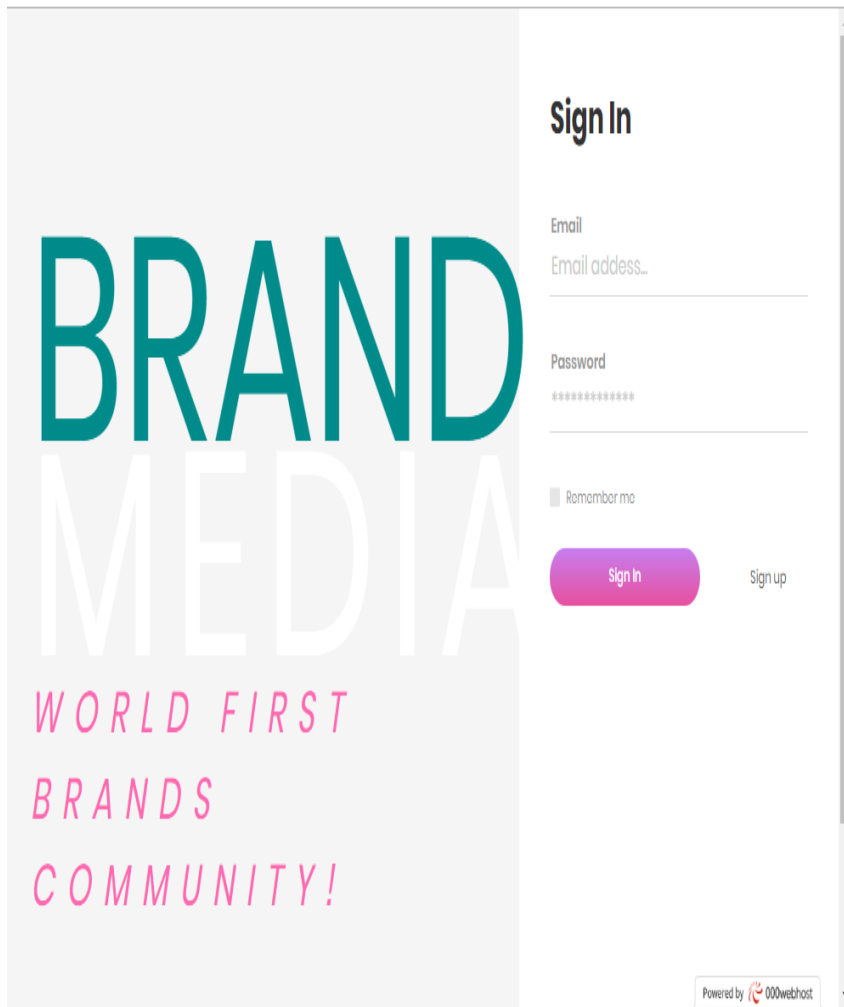
Appendix C: Information / Promotional Material

This chapter describe the promotional material for out project which name is Brand Media.

C.1. Broacher



Flyer





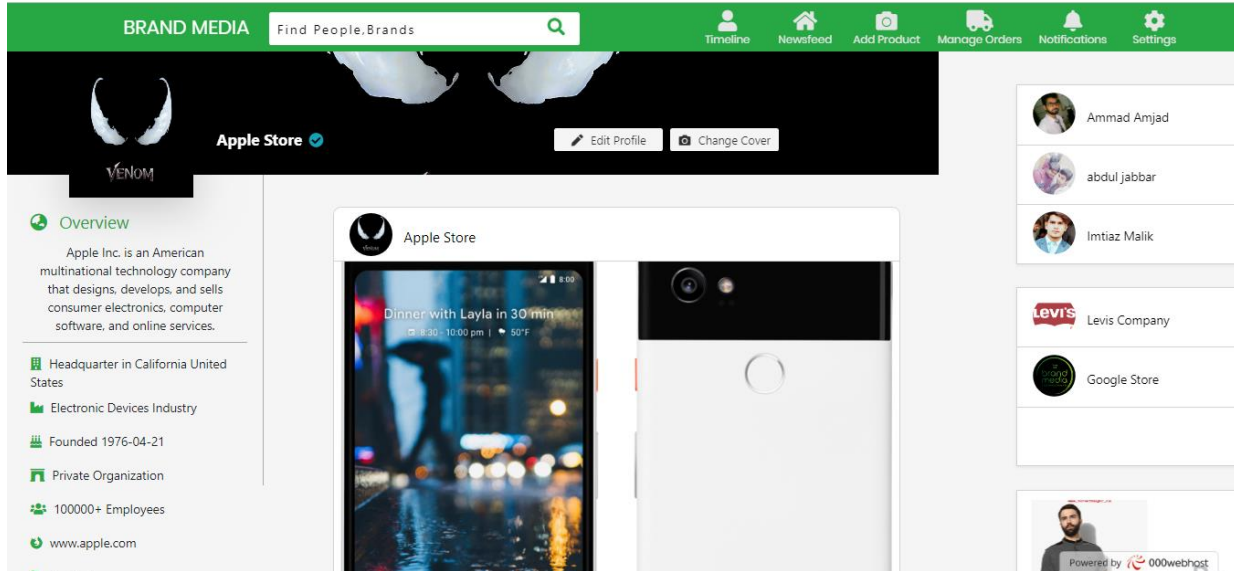
C.2. Standee

The screenshot displays a mobile application interface with a green navigation bar at the top. The navigation bar includes a search field labeled "Find People, Brands" and several icons for "Timeline", "Newsfeed", "Add Product", "Manage Orders", "Notifications", and "Settings".

The main content area is divided into two columns. The left column features a product listing for "Google Home". The product image shows a white cylindrical smart speaker and a smartphone displaying the Google Home app interface. Below the image, the product details are listed: "Name: Google Home", "Price: 15000,-Rs", and "Stock:". Below the product details, there are icons for a heart, a comment bubble, and a share icon, followed by "6 Likes" and two user comments: "abdul jabbar level" and "abdul jabbar good". A link "View all comments" and the text "2 DAY AGO" are also present.

The right column displays a list of users and brands to follow. Each entry includes a profile picture, a name, and a "Follow" button. The entries are: "Ammad Amjad", "abdul jabbar", "Imtiaz Maik", "Levis Company", and "Google Store". Below this list is a promotional banner for a "Levis T-shirt" priced at "Rs. 700" with a "Stock: 10". The banner features a photo of a man wearing a dark grey t-shirt.

C.3. Banner



C.4. First Level heading

N/A

C.1.1. Second level heading

N/A

C.1.1.1. Third level heading

N/A

Reference and Bibliography

Reference and Bibliography

- [1] https://www.codeigniter.com/user_guide/overview/at_a_glance.html, "User Guide About Codigniter"/Version (3.0)
- [2] <https://developer.android.com/studio/>, "Adnroid Studio"/ Version (3.4)
- [3] <https://www.simplifiedcoding.net/retrofit-android-tutorial/>, "Restful API(s) using Retrofit 2"/ Version
- [4] <https://materialdesignicons.com/>, "Metrial Design of icons"