**Dhaka University Calendar Management Application**

Submitted To

Dr. Kazi Muheymin-Us-Sakib

Professor

Institute of Information Technology,

University of Dhaka

Submitted by

Pritom Kumar Das BSSE0919

Miran Al Mehrab BSSE0931

**Date: 18/11/2018**

**Letter of Transmittal**

Dr. Kazi Muheymin-Us-Sakib

Professor

Institute of Information Technology,

University of Dhaka

Subject: Submission of term report on “Dhaka University Calendar Management Application”.

Sir,

With due respect, we would like to submit you the report on Software Requirement Specification on the above topic you assigned us. The report reflects our effort to gather requirements and analyzing them. We have included every steps what we have done through the whole time for requirement specification of the mentioned topic.

Therefore, we earnestly hope that you will excuse our error and obliged thereby.

Yours Sincerely,

Pritom Kumar Das

BSSE0919

Miran Al Mehrab

BSSE0931

3nd year, 5th semester, 9th batch

Institute of Information Technology

University of Dhaka

Session: 2016-17

**Acknowledgement**

We are highly indebted for getting such a tremendous opportunity to prepare the report on Dhaka University Calendar Management Application .We would like to thank whole-heartedly our supervisor, Dr. Kazi Muheymin-Us-Sakib , Professor , Institute of Information Technology, University of Dhaka, for giving us guidelines about how we can prepare this report. We are also thankful of those people who gave requirements to do this job.

**Abstract**

The calendar committee of Dhaka University publishes a calendar for the very first year students of the university to be familiar with the University campus and its glorious history every year . We are fortunate to take the task of creating an android application that represents Dhaka University Calendar . So, this is the SRS (Software Requirements Specification) of Dhaka University Calendar Management Application from the gathered requirements. The report represents the objective and also describes the need of this trendy strategy.

5

**Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| [Chapter 1: Introduction: **.........................................................................................................................**](#page12) | | | [**12**](#page12) |
| [**1.1 Purpose** ...................................................................................................................................................](#page12) | | | [12](#page12) |
| [**1.2 Intended Audience** .................................................................................................................................](#page12) | | | [12](#page12) |
| [**1.3**](#page12) | [**Conclusion**...............................................................................................................................................](#page12) | | [12](#page12) |
| [**Chapter 2: Inception ..............................................................................................................................**](#page13) | | | [**13**](#page13) |
| [**2.1**](#page13) | [**Introduction** ............................................................................................................................................](#page13) | | [13](#page13) |
| [**2.1.1**](#page13) | | [**Identifying Stakeholders** .................................................................................................................](#page13) | [13](#page13) |
| [**2.1.2**](#page14) | | [**Recognizing Multiple Viewpoints** ...................................................................................................](#page14) | [14](#page14) |
| [**2.1.3**](#page15) | | [**Working towards collaboration** ......................................................................................................](#page15) | [15](#page15) |
| [**2.1.4**](#page16) | | [**Asking the First Questions** ..............................................................................................................](#page16) | [16](#page16) |
| [**2.2.Conclusion** ..............................................................................................................................................](#page16) | | | [16](#page16) |
| [Chapter 3 : Elicitation **.............................................................................................................................**](#page17) | | | [**17**](#page17) |
| [**3.1**](#page17) | [**Introduction** ............................................................................................................................................](#page17) | | [17](#page17) |
| [**3.2**](#page17) | [**Eliciting Requirements** ...........................................................................................................................](#page17) | | [17](#page17) |
| [**3.3**](#page17) | [**Collaborative Requirements Gathering** .................................................................................................](#page17) | | [17](#page17) |
| [**3.4 Quality Function Deployment** ................................................................................................................](#page17) | | | [17](#page17) |
| [**3.4.1 Normal Requirements** .....................................................................................................................](#page18) | | | [18](#page18) |
| [**3.4.2**](#page18) | | [**Expected Requirements** ..................................................................................................................](#page18) | [18](#page18) |
| [**3.4.3**](#page19) | | [**Exciting requirements** .....................................................................................................................](#page19) | [19](#page19) |
| [Chapter 4: Scenario Based Modeling **.......................................................................................................**](#page23) | | | [**23**](#page23) |
| [**4.1**](#page23) | [**Introduction** ............................................................................................................................................](#page23) | | [23](#page23) |
| [**4.2 Use Case Scenario** ...................................................................................................................................](#page23) | | | [23](#page23) |
| [**4.3**](#page24) | [**Use Case Descriptions** ............................................................................................................................](#page24) | | [24](#page24) |
| [**4.3.1**](#page24) | | [**Authentication** .................................................................................................................................](#page24) | [24](#page24) |
| [**4.3.2 View** .................................................................................................................................................](#page26) | | | [26](#page26) |
| [**4.3.3**](#page28) | | [**Order** ................................................................................................................................................](#page28) | [28](#page28) |
| [**4.3.4 Review** .............................................................................................................................................](#page30) | | | [30](#page30) |
| [**4.3.5 Payment Method** .............................................................................................................................](#page31) | | | [31](#page31) |
| [**4.3.6 Product Management** .....................................................................................................................](#page32) | | | [32](#page32) |
| [**4.3.7**](#page33) | | [**Contact** .............................................................................................................................................](#page33) | [33](#page33) |
| [**4.4 Use Case Diagram** ...................................................................................................................................](#page36) | | | [36](#page36) |
| [**4.5 Activity Diagram and Swimlane Diagram of generated Use Cases:** .....................................................](#page41) | | | [42](#page41) |
| [Chapter 5: Data Modeling **.......................................................................................................................**](#page81) | | | [**81**](#page81) |
|  |  | 6 |  |

**[5.1 Data modeling concept](#page81)** [**81**](#page81)

[**5.2 Data objects**](#page81)[**81**](#page81)

[Chapter 6: Class Based Modeling](#page93) [**94**](#page93)

[**6.1 Class Based Modeling Concept**](#page94)[**94**](#page94)

[**6.2 General Classification**](#page94)[**94**](#page94)

[**6.3 Selection Criteria**](#page98)[**98**](#page98)

[**6.4 Associate Noun and Verb Identification**](#page100)[**100**](#page100)

[**6.5 Attribute Selection**](#page103)[**103**](#page103)

[**6.6 Methods Identification:**](#page106)[**106**](#page106)

[**6.7 Finalizing Classes**](#page110)[**110**](#page110)

[**6.8 Class Cards**](#page110)[**110**](#page110)

[**6.9 UML Diagram**](#page119)[**119**](#page119)

[Chapter 7: Flow-Oriented Model](#page120) [**120**](#page120)

[**7.1 Introduction**](#page120)[**120**](#page120)

[**7.2 Data Flow Diagram (DFD)**](#page120)[**120**](#page120)

[Chapter 8: Behavioral Model](#page127) [**127**](#page127)

[**8.1 State Diagram**](#page127)[**127**](#page127)

[**8.2 Sequence Diagram**](#page136)[**136**](#page136)

[Chapter 9: Conclusion](#page146) [**146**](#page146)

[**References**](#page147)

[**Appendix**](#page147)

**Table of Figures**

|  |  |
| --- | --- |
| [Figure 1: Level 0 for F-commerce based client management system .................................................](#page36) | [36](#page36) |
| [Figure 2: Level 1 for F-commerce based client management system .................................................](#page36) | [36](#page36) |
| [Figure 3: Level 2.1 for F-commerce based client management system ..............................................](#page37) | [37](#page37) |
| [Figure 4: Level 2.2 for F-commerce based client management system .............................................](#page38) | [38](#page38) |
| [Figure 5: Level 2.3 for F-commerce based client management system ..............................................](#page39) | [39](#page39) |
| 7 |  |

|  |  |
| --- | --- |
| [Figure 6: Level 2.4 for F-commerce based client management system ..............................................](#page39) | [39](#page39) |
| [Figure 7 : Level 2.5 for F-commerce based client management system ............................................](#page40) | [40](#page40) |
| [Figure 8: Level 2.6 for F-commerce based client management system ..............................................](#page40) | [40](#page40) |
| [Figure 9: Level 2.7 for F-commerce based client management system ..............................................](#page41) | [41](#page41) |
| [Figure 10: Activity Diagram for Sign-Up ...........................................................................................](#page42) | [42](#page42) |
| [Figure 11: Swimlane Diagram for Sign-Up ........................................................................................](#page43) | [43](#page43) |
| [Figure 12: Activity Diagram for Login ...............................................................................................](#page44) | [44](#page44) |
| [Figure 13: Swimlane Diagram for Login ............................................................................................](#page45) | [45](#page45) |
| [Figure 14: Activity Diagram for Sign-Out ..........................................................................................](#page46) | [46](#page46) |
| [Figure 15: Swimlane Diagram for Sign-out ........................................................................................](#page46) | [46](#page46) |
| [Figure 16: Activity Diagram for Manage Account .............................................................................](#page47) | [47](#page47) |
| [Figure 17: Swimlane Diagram for Manage Account ..........................................................................](#page48) | [48](#page48) |
| [Figure 18: Activity Diagram for Browse Category and Search ..........................................................](#page49) | [49](#page49) |
| [Figure 19: Swimlane Diagram for Browse Category and Search .......................................................](#page50) | [50](#page50) |
| [Figure 20: Activity Diagram for Add to Cart ......................................................................................](#page51) | [51](#page51) |
| [Figure 21: Swimlane Diagram for Add to Cart ...................................................................................](#page52) | [52](#page52) |
| [Figure 22: Activity Diagram for Purchase ..........................................................................................](#page53) | [53](#page53) |
| [Figure 23: Swimlane Diagram for Purchase .......................................................................................](#page54) | [54](#page54) |
| [Figure 24: Activity Diagram for Cancel Order Before delivery .........................................................](#page55) | [55](#page55) |
| [Figure 25: Swimlane Diagram for Cancel Order Before delivery ......................................................](#page56) | [56](#page56) |
| [Figure 26: Activity Diagram for Return Product ................................................................................](#page57) | [57](#page57) |
| [Figure 27: Swimlane Diagram for Return Product .............................................................................](#page58) | [58](#page58) |
| [Figure 28: Activity Diagram for Comment and rate ...........................................................................](#page59) | [59](#page59) |
| [Figure 29: Swimlane Diagram for Comment and rate ........................................................................](#page60) | [60](#page60) |
| [Figure 30: Activity Diagram for Cash on Delivery .............................................................................](#page61) | [61](#page61) |
| [Figure 31: Swimlane Diagram for Cash on Delivery ..........................................................................](#page62) | [62](#page62) |
| [Figure 32: Activity Diagram for Online Payment ...............................................................................](#page63) | [63](#page63) |
| [Figure 33: Swimlane Diagram for Online Payment ............................................................................](#page64) | [64](#page64) |
| [Figure 34: Activity Diagram for Add Product ....................................................................................](#page65) | [65](#page65) |
| [Figure 35: Swimlane Diagram for Add Product .................................................................................](#page66) | [66](#page66) |
| [Figure 36: Activity Diagram for Edit Product .....................................................................................](#page67) | [67](#page67) |
| [Figure 37: Swimlane Diagram for Edit Product ..................................................................................](#page68) | [68](#page68) |
| [Figure 38: Activity Diagram for Remove Product ..............................................................................](#page69) | [69](#page69) |
| [Figure 39: Swimlane Diagram for Remove Product ...........................................................................](#page70) | [70](#page70) |
| [Figure 40: Activity Diagram for Manage Product Advertisement ......................................................](#page71) | [71](#page71) |
| [Figure 41: Swimlane Diagram for Manage Product Advertisement ...................................................](#page72) | [72](#page72) |
| [Figure 42: Activity Diagram for Chat .................................................................................................](#page73) | [73](#page73) |
| [Figure 43: Swimlane Diagram for Chat ..............................................................................................](#page74) | [74](#page74) |
| [Figure 44: Activity Diagram for Email ...............................................................................................](#page75) | [75](#page75) |
| [Figure 45: Swimlane Diagram for Email ............................................................................................](#page76) | [76](#page76) |
| [Figure 46: Activity Diagram for Helpline ...........................................................................................](#page77) | [77](#page77) |
| [Figure 47: Swimlane Diagram for Helpline ........................................................................................](#page78) | [78](#page78) |
| [Figure 48: Activity Diagram for Follow on Social Media ..................................................................](#page79) | [79](#page79) |
| [Figure 49: Swimlane Diagram for Follow on Social Media ...............................................................](#page80) | [80](#page80) |
| [Figure 50: ER diagram of F-commerce based client Management System ........................................](#page86) | [86](#page86) |
| [Figure 51: UML Diagram of F-Commerce Based Client Management System ...............................](#page119) | [119](#page119) |
| 8 |  |

[Figure 52: Level 0 for F-commerce based client management system](#page120) [120](#page120)

[Figure 53: Level 1.1 for F-commerce based client management system](#page121) [121](#page121)

[Figure 54: Level 1.2 for F-commerce based client management system](#page122) [122](#page122)

[Figure 55: Level 2.1 for F-commerce based client management system](#page122) [122](#page122)

[Figure 56: Level 2.2 for F-commerce based client management system](#page123) [123](#page123)

[Figure 57: Level 2.3 for F-commerce based client management system](#page124) [124](#page124)

[Figure 58: Level 2.4 for F-commerce based client management system](#page124) [124](#page124)

[Figure 59: Level 2.5 for F-commerce based client management system](#page125) [125](#page125)

[Figure 60: Level 2.6 for F-commerce based client management system](#page126) [126](#page126)

[Figure 61: Level 2.7 for F-commerce based client management system](#page126) [126](#page126)

[Figure 62: State Diagram (Account Class)](#page127) [127](#page127)

[Figure 63: State Diagram (Admin Class)](#page128) [128](#page128)

[Figure 64: State Diagram (Customer Class)](#page129) [129](#page129)

[Figure 65: State Diagram (Database Class)](#page130) [130](#page130)

[Figure 66: State Diagram (Delivery Service Class)](#page131) [131](#page131)

[Figure 67: State Diagram (Notification Class)](#page131) [131](#page131)

[Figure 68: State Diagram (Order Class)](#page132) [132](#page132)

[Figure 69: State Diagram (Payment Class)](#page133) [133](#page133)

[Figure 70: State Diagram (Search Class)](#page133) [133](#page133)

[Figure 71: State Diagram (Product Class)](#page134) [134](#page134)

[Figure 72: State Diagram (System Class)](#page135) [135](#page135)

[Figure 73: Sequence diagram (Sign Up)](#page136) [136](#page136)

[Figure 74: Sequence diagram (Sign In)](#page137) [137](#page137)

[Figure 75: Sequence diagram (Sign Out)](#page138) [138](#page138)

[Figure 76: Sequence diagram (Browse Category)](#page139) [139](#page139)

[Figure 77: Sequence diagram (Search)](#page139) [139](#page139)

[Figure 78: Sequence diagram (Add to Cart)](#page140) [140](#page140)

[Figure 79: Sequence diagram (Purchase)](#page140) [140](#page140)

[Figure 80: Sequence diagram (Cancel Order)](#page141) [141](#page141)

[Figure 81: Sequence diagram (Add Product)](#page141) [141](#page141)

[Figure 82: Sequence diagram (Edit Product)](#page142) [142](#page142)

[Figure 83: Sequence diagram (Remove Product)](#page142) [142](#page142)

[Figure 84: Sequence diagram (Manage Product Advertisement)](#page143) [143](#page143)

[Figure 85: Sequence diagram (Review and Comment Product)](#page143) [143](#page143)

[Figure 86: Sequence diagram (Chat)](#page144) [144](#page144)

[Figure 87: Sequence diagram (Email)](#page145) [145](#page145)

9

**List of Tables:**

[Table 1: Use Case Scenario](#page23) [23](#page23)

[Table 2: Noun Identification for data modelling](#page81) [81](#page81)

[Table 3: Final Data Objects](#page85) [85](#page85)

[Table 4: Schema for Admin](#page87) [87](#page87)

[Table 5: Schema for Customer](#page87) [87](#page87)

[Table 6: Schema for Product](#page88) [88](#page88)

[Table 7: Schema for Order](#page88) [88](#page88)

[Table 8: Schema for Pathao Service](#page89) [89](#page89)

[Table 9: Schema for Courier Service](#page89) [89](#page89)

[Table 10: Schema for Delivery\_ Service](#page89) [90](#page89)

[Table 11: Schema for Notification](#page90) [90](#page90)

[Table 12: Schema for Order Summary](#page91) [91](#page91)

[Table 13: Schema for Advertisement](#page91) [91](#page91)

[Table 14: Schema for Advertisement\_publish](#page91) [91](#page91)

[Table 15: Schema for Product\_edit](#page92) [92](#page92)

[Table 16: General Classifications of Nouns](#page94) [94](#page94)

[Table 17: Selection Criteria of Potential Classes](#page99) [99](#page99)

[Table 18: Associate Noun and Verbs](#page100) [100](#page100)

[Table 19: Attribute Selection of Classes](#page103) [103](#page103)

[Table 20: Methods of Class](#page106) [106](#page106)

[Table 21: Class Card of Customer](#page110) [110](#page110)

[Table 22: Class Card of Admin](#page112) [112](#page112)

[Table 23: Class Card of Database](#page112) [112](#page112)

[Table 24: Class Card of Product](#page113) [113](#page113)

[Table 25: Class Card of System](#page114) [114](#page114)

[Table 26: Class Card of Account](#page114) [114](#page114)

[Table 27: Class Card of Activity](#page115) [115](#page115)

[Table 28: Class Card of Searching](#page115) [115](#page115)

[Table 29: Class Card of Payment](#page116) [116](#page116)

10

[Table 30: Class Card of Notification](#page116) [116](#page116)

[Table 31: Class Card of SocialMedia](#page117) [117](#page117)

[Table 32: Class Card of DeliveryService](#page117) [117](#page117)

[Table 33: Class Card of Order](#page117) [118](#page117)

11

**Chapter 1: Introduction**

This chapter is a part of “Dhaka University Calendar Management Application” intended to specify the purpose of this document and the intended audience of it.

**1.1 Purpose**

This document is the Software Requirements Specification (SRS) for “Dhaka University Calendar Management Application”. It contains detailed functional, non-functional, and support requirements and establishes a requirements baseline for development of the system. The requirements contained in the SRS are independent, uniquely numbered, and organized by topic. The SRS serves as the official means of communicating user requirements to the developer and provides a common reference point for both the developer team and stakeholder community. The SRS will evolve over time as users and developers work together to validate, clarify and expand its contents.

**1.2 Intended Audience**

This SRS is intended for several audiences, including the customer as well as the project managers, designers, developers, and testers.

* The customer will use this SRS to verify that the developer team has created a product that is acceptable to the customer.
* The project managers of the developer team will use this SRS to plan milestones and a delivery date, and ensure that the developing team is on track during development of the system.
* The designers will use this SRS as a basis for creating the system’s design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer’s needs.
* The developers will use this SRS as a basis for developing the system’s functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created software that will fulfill all of the customer’s documented requirements.
* The testers will use this SRS to derive test plans and test cases for each documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

**1.3 Conclusion**

This analysis of the audience helped us to focus on the users who will be using our analysis. This overall document will help each and every person related to this project to have a better idea about the project.

12

**Chapter 2: Inception**

In this chapter, the Inception part of the SRS will be discussed briefly.

**2.1 Introduction**

Inception is the beginning phase of requirements engineering. It defines how does a software project get started and what is the scope and nature of the problem to be solved. The goal of the inception phase is to identify concurrence needs and conflict requirements among the stakeholders of a software project. To establish the groundwork we have worked with the following factors related to the inception phases:

* Identifying Stakeholders
* Recognizing multiple viewpoints
* Working towards collaboration
* Asking the First Questions

**2.1.1 Identifying Stakeholders**

Stakeholder refers to any person or group who will be affected by the system directly or indirectly. Stakeholders include end-users who interact with the system and everyone else in an organization that may be affected by its installation. At inception, a list of people who will contribute input as requirements are elicited. The initial list will grow as stakeholders are contacted because every stakeholder will be asked: “whom else do you think I should talk to?”

The following stakeholders were identified for the “Dhaka University Calendar Management Application”.

* **Dhaka University Governing Body:** As the highest ranking governing body of Dhaka University they take all the important decisions concerning Dhaka University . Only with their support and approval this application can serve its purpose to its fullest potential.
* **Dhaka University Calendar Committee** : As the committee responsible for delivering and maintaining Dhaka University Calendar each year they are considered the Admin of this application . They operate to gather the necessary information needed for the application .
* **Student:** A student is a first year student of Dhaka University . They are the main users of this application. A student provides his/her information to create his/her profile .
* **Teacher :** As they are the faculty members of various departments/institutions of Dhaka University they can use the app to find relative information . They are also part of the Dhaka University Governing Body and Dhaka University Calendar Committee.
* **Software Developer:** A software developer is concerned with facets of the softwaredevelopment process, including the research, design, programming, maintenance and testing of computer software. He/She will be responsible for the outcomes of the software.

13

**2.1.2 Recognizing Multiple Viewpoints**

Different stakeholders demand different features from the software. To satisfy the stakeholders, most of these features should be included in the software.

**Dhaka University Governing Body’s viewpoint**

* Error free system
* Getting all the useful information about Dhaka University
* Checking the security of the system
* Check the validity of provided information
* Keeping record of users
* Update any information that might be wrong
* Maintain public image of Dhaka University
* Strong authentication system

**Dhaka University Calendar Committee’s viewpoint**

* User friendly
* Error free system
* Keeping a track of new information provided by departments/institutes
* Getting accurate information and check validity of the information
* Easy maintainability
* Add new structures that’s not currently in the provided information
* Respond to any complaints and provide solutions
* Strong authentication system
* Response to security issues
* High scalability
* Keeping a backup of the data

**Student’s viewpoint**

* User friendly
* Quick search result for information
* Error free system
* Easy to use
* Fast response
* Easy to connect with friends
* Check history on conversation

**Teacher’s viewpoint**

* User friendly
* Quick search result for information
* Error free system
* Check validity of the information
* Check history

**Developer’s viewpoint**

* Easy to built
* Error free effective software
* Readable and effective code review
* No conflicting requirement
* Easy to add expand or change software
* Create secure software

**2.1.3 Working towards collaboration**

While working with different stakeholders, some conflicting and common viewpoints can be noticed. For this reason, final requirements can be gotten by collaborating the viewpoints. We followed following steps to merge these requirements:

* Identify the common and conflicting requirements
* Categorize the requirements
* Take priority points for each requirements from stakeholders and on the basis of this voting prioritize the requirements
* Make final decision about the requirements

**Common Requirements:**

* Web based interfaces
* The application can be accessed from any computer that has internet access.
* Allow any user to search for Information
* Attractive and easy to use User Interface
* Secure application
* Easy to maintain
* Maintain a database for all users and information in the system

**Conflicting Requirements:**

* Strong authentication problem when check out
* Whether login required or not to use the system
* Design of the User Interface
* Weather or not add messenger in application
* Separate application for admins

**Final Requirements:**

* Error free easy accessible system
* Android application
* Separate web based interface for admins
* Accessible via the Internet.
* Allow valid users to login and logout.
* Restrict access to functionality of the system based upon user roles
* Allow administrators of the system to change provided information and configure parameters of the system
* Allow any user to search for information in the application without having to log in to the system
* Allow valid users that log in to use exciting features of application.
* Allow Administrators to delete , modify, add or update any information regarding Dhaka University
* This application can be used by any android device with connection to internet.
* A dedicated server have to run all the time to help users get information.
* Maintain proper security for all the data.
* Maintain a database of all users and information.
* Allows user option to create profile or not.
* Allows user option to change language.
* Allows user to contact with admin through email if needed

Restrict access to functionality of the system based upon user roles. For example, only Administrators of the system will be provided functionality to change static information of Dhaka university .

**2.1.4 Asking the First Questions**

We set our first set of context-free questions focuses on the students and other stakeholders, overall project goals and benefits. The questions are mentioned above. These questions helped us to identify all stakeholders, measurable benefit of the successful implementation and possible alternatives to custom software development. Next set of question helped us to gain a better understanding of problem and allows the user to voice his or her perception about the solution. The final set of question focused on the effectiveness of the communication activity itself and the acceptability of the application itself.

**2.2 Conclusion**

Inception phase helped us to establish basic understanding about “Dhaka University Calendar Management Application” and its importance in Dhaka University, identify the people who will be ben**e**fited if Dhaka University Calendar Management System becomes automated, define the nature of the Calendar management software and establish a preliminary communication with our stakeholders. More studies and communication will help both side (developer and client) to understand the future prospect of the project. Our team believes that the full functioning document will help us to define that future prospect.

**Chapter 3: Elicitation**

This chapter specifies the Elicitation phase.

**3.1 Introduction**

Requirements Elicitation is a part of requirements engineering that is the practice of gathering requirements from the students, admin and other stakeholders. Many difficulties were faced, like understanding the problems, making questions for the stakeholders, limited communication with the stakeholders due to a short amount of time and volatility. Though it is not easy to gather requirements within a very short time, these problems have been surpassed in an organized and systematic manner.

**3.2 Eliciting Requirements**

The main task of this phase is to combine the elements of problem solving, elaboration, negotiation and specification. The collaborative working approach of the stakeholders is required to elicit the requirements. The following tasks were done for eliciting requirements:

1. Collaborative Requirements Gathering
2. Quality Function Deployment
3. Usage Scenarios
4. Elicitation work products

**3.3 Collaborative Requirements Gathering**

Many different approaches to collaborative requirements gathering have been proposed. Each makes use of a slightly different scenario. We completed following steps to do it.

* The meetings were conducted with the Dhaka University Calendar Committee members and they were questioned about their requirements and expectations from the Dhaka University Calendar Management Application.
* The Calendar Committee was asked to see there satisfaction with the current system.
* At last we selected our final requirements from the meetings

**3.4 Quality Function Deployment**

Quality Function Deployment (QFD) is a technique that translates the needs of the users into technical requirements for software. It concentrates on maximizing user satisfaction from the Software engineering process. With respect to our project the following requirements are identified by a QFD.

17

**3.4.1 Normal Requirements**

The normal requirements are generally the objectives and goals that are stated for a product or system during meetings with the user. The presence of these requirements fulfills users’ satisfaction. These are the normal requirements for the project.

1. Error free easy accessible system
2. Android application
3. Effective System
4. Separate web based interface for admins
5. Accessible via the Internet.
6. Allow valid users to login and logout.
7. Restrict access to functionality of the system based upon user roles
8. Allow administrators of the system to change provided information and configure parameters of the system
9. Allow any user to search for information in the application without having to log in to the system
10. Allow valid users that log in to use exciting features of application.
11. Allow Administrators to delete , modify, add or update any information regarding Dhaka University
12. This application can be used by any android device with connection to internet.
13. A dedicated server have to run all the time to help users get information.
14. Maintain proper security for all the data.
15. Maintain a database of all users and information.
16. Allows user option to create profile or not.
17. Allows user option to change language.
18. Allows user to contact with admin through email if needed
19. Fast loading application.
20. Provide users with Dhaka University Map.

**3.4.2 Expected Requirements**

These requirements are intrinsic to the product or system and may be so elementary that the customer does not explicitly state them. Their absence will be a cause for significant dissatisfaction. Below the expected requirements for our project are briefly described.

1. Error free software
2. Strong authentication system
3. User friendly
4. Effective system
5. No ambiguous feature
6. Data backup
7. Sending notification to users if necessary.

18

**3.4.3 Exciting requirements**

These requirements are for features that go beyond the customer's expectations and prove to be very satisfying when present

1. The user interface should provide appropriate error messages for invalid input or show message if search results are not found.
2. The user interface should follow standard web practices such that the web interface is consistent with typical internet applications.
3. Offer log in with mobile phone
4. Users will also be given a list of exciting places within Dhaka University .
5. There will be a personalized calendar for each student which will contain academic schedule.
6. Messenger for first year students.

**Usage scenario**

This is an android application that is entitled to the name Dhaka University calendar management application. People can avail this application to be familiar with the Dhaka University. People can create account to access some more exciting feature along with the basic features.

**Account Creation**

**Sign Up:**

The user can download the application from google play store or Dhaka University main website page (admission page). Any person can sign in with google account. To complete their profile the users will have to provide their user name and type. Depending on their type they will be required to enter their , department/institute name and academic year.

**Sign In:**

Aftersuccessful sign up the user can sign in to his\her account with Firebase Auth Sign In. User can update their profile information. User has an option to log out.

**Role Based Access**

There are three types of user such as teacher , student and other. All users can access basic functionalities of the application such as Map , transport , information about University and general calendar . They can also update or delete their account according to their wish.

Exciting features access ---

**Student :** Students can use group chat . Student can access specified calendar based on their given academic year.

**Teachers :** Teachers can choose from a dropdown menu which year calendar they want to see .

**Function Based Management**

**Information management:**

It includes DU brief history, notable achievement, information about governing body, department / institute description, halls, Club or society details, pictures of famous places in DU campus and other descriptive information. Only the admin can modify this information. All user can view and search for this information .

Dhaka University governing body : This includes the picture , name and rank of the Dhaka University governing body .

Administrative committee : This includes the name and rank of the administrative committee members.

Editorial committee : This includes the name and rank of the editorial committee members.

List of Faculty Deans : This includes a list of faculty names and its Deans .

Department /Institute/hall/offices : This includes department /institute id , name , establishment year , description of the department/institutes/hall/offices and locations.

**Transport Management:**

Users can see the schedule of buses on different route and their stoppage on the way.

**Map management:**

Users can see their current position. They can also search and get route for their destination. They can also see exciting places near their position within the university area.

**Communication Management:**

A group chat option is available for the the students based on their academic session and department. Students can also leave this chat group if they wish.

**Calendar Management:**

There will be to kinds of calendar . A general calendar that entails all holiday and events marked by Dhaka University . And there will be a specific calendar for each department / institute that will mark down all the academic activities as well as university holidays.

**Administrative Responsibilities**

Admin will have a different web interface for their convenience. They can upload, update or delete information to the server and simultaneously database will be updated accordingly. Admins will also have to maintain the database and server that’s providing service for the application . They will also have to maintain data transparency and security.

**Design Components**

**Homepage:**

Users will find a search option to search information in the general homepage . There will be icons for all the functions such as map, calendar , transport , messenger etc. on top right side there will be accounts and settings option . In the settings they can change the language to Bengali or English.

**Other Pages:**

Users can go to other pages from home page . The layout will be consistent on the other pages . The design will follow a tree based structure to link the activities.

**Search**

Users can search for any information . The search will be generated as database query to find information from the database. If not found the search will return an empty page with no results found. Information search will return result on searching all the Dhaka University information available on database . All users can use this search .

**Chapter 4: Scenario Based Modeling**

**4.1 Introduction**

In this model the system is described from the user’s point of view. As this is the first model, it serves as input for creation of other modeling elements.

**4.2 Use Case Scenario**

Table 1: Use Case Scenario

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Level 0 |  | Level 1 |  | Level 2 |  |
|  |  |  |  |  |  |
|  |  | Authentication |  | Sign up |  |
|  |  |  |  |  |  |
|  |  |  |  | Sign in |  |
|  |  |  |  |  |  |
|  |  |  |  | Sign out |  |
|  |  |  |  |  |  |
|  |  |  |  | Manage account |  |
|  |  |  |  |  |  |
|  |  | View |  | Browse Category |  |
| F-commerce | |  |  |  |  |
| based | client |  |  |  |  |
|  |  | Search |  |
| management | |  |  |  |
|  |  |  |  |
| system |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  | Discounted Product view |  |
|  |  |  |  |  |  |
|  |  | Order |  | Add to cart |  |
|  |  |  |  |  |  |
|  |  |  |  | Purchase |  |
|  |  |  |  |  |  |
|  |  |  |  | Cancel order before delivery |  |
|  |  |  |  |  |  |
|  |  |  | 23 | |  |

|  |  |  |
| --- | --- | --- |
|  |  | Return product |
|  |  |  |
|  | Review | Comment |
|  |  |  |
|  |  | Rate product |
|  |  |  |
|  | Payment | Cash on delivery |
|  |  |  |
|  |  | Advance online payment |
|  |  |  |
|  | Product management | Add product |
|  |  |  |
|  |  | Edit product |
|  |  |  |
|  |  | Remove product |
|  |  |  |
|  |  | Manage product advertisement |
|  |  |  |
|  | Contact | Chat |
|  |  |  |
|  |  | Email |
|  |  |  |
|  |  | Helpline |
|  |  |  |
|  |  | Follow on social media |
|  |  |  |

**4.3 Use Case Descriptions**

**4.3.1 Authentication**

4.3.1.1 Sign up

**Use Case:** Sign up

**Primary Actor:** Admin, Customer

**Goal in context:** to create an account

24

**Preconditions:**

1. System has been designed for Sign up
2. System has interface for Sign up **Triggers:** Admin and customer has to sign up

**Scenario:**

1. Visit page and sign up
2. Provided the required information
3. Check availability of user name, email and password
4. Verify information
5. Send email for confirmation

**Exception:**

* Same user name
* Same password
* Same email
* Verification failed

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.1.2** **Login**

**Use Case:** Login

**Primary Actor:** Admin, Customer

**Goal in context:** To enter the system

**Preconditions:**

1. System has been designed for login
2. System has interface for login
3. Visit page and login
4. Provide the required information
5. Proceed for shopping

**Exception:** Unrecognized Username Wrong Password

User is blocked

**Priority:** Essential, must be implemented.

**When Available**: First increment

**4.3.1.3** **Sign out**

**Use Case:** Sign out

**Primary Actor:** Admin, Customer

**Goal in context:** sign out from site

**Preconditions:**

25

1. System has been designed for sign out
2. System has interface for sign out
3. User must be logged in
4. Visit page
5. Click Sign out button

**Exception:**

User not logged in

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.1.4** **Manage Account**

**Use Case:** Manage Account

**Primary Actor**: Admin, Customer

**Goal in context:** to manage an account (change user information)

**Preconditions:**

1. System has been designed to change username and password
2. System has interface to Change username and password **Triggers:** customer has to change username and password

**Scenario:**

1. Visit page and login
2. Click change username or password button
3. Provide required information for login
4. Provide information that need to be change
5. Send message for successful change

**Exception:**

·Information is same as the previous one

·Username ambiguous

· Password not in range **Priority:** Depends on user **When Available:** First increment

**4.3.2 View**

**4.3.2.1** **Browse Category**

**Use Case:** Browse

**Primary Actor:** Admin, Customer

**Goal in context:** to browse category and view

26

**Preconditions:**

1. System has been designed for browse category
2. System has interface for browse category
3. Has list of product

**Triggers:** Customer wants to browse product category wise to buy product

1. Visit page
2. Users have something to view to shop
3. Users have to browse category

**Exception:**

·Product not available

·Invalid Input

**Priority:** Essential, depends on user.

**When Available:** First increment

**4.3.2.2** **Search**

**Use Case:** Search

**Primary Actor:** Admin, Customer

**Goal in context:** To search specific product and view product

**Preconditions:**

1. System has been designed for search
2. System has interface for search
3. Has list of product with keywords

**Triggers:** Admin and customer has to search and view product for shopping

1. Visit page
2. Users have something to search
3. Users have to enter keyword or price range to search and view products

**Exception:**

·Product not available

·Invalid Input

**Priority:** Essential, depends on user

When Available: First increment

**4.3.2.3** **Discount Product**

**Use Case:** Discount product

**Primary Actor:** Admin, Customer

**Goal in context:** To view any offer or discount available

**Preconditions:**

1. System has been designed for discount product view
2. System has interface for discount product view
3. Has list of product with discount price
4. Visit page
5. Users have something to view

27

1. Users have discounted product with discount item

**Exception:**

·Offer has expired

· Discount not available for particular product **Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.3 Order**

**4.3.3.1** **Add to cart**

**Use Case:** Add to cart

**Primary Actor:** Customer

**Goal in context:** To add product to cart for shopping

**Preconditions:**

1. System has been designed for Add to cart
2. System has interface for Add to cart
3. Has list of product

**Triggers:** Customer has to add product to card

**Scenario:**

1. Visit page
2. Users have something to order
3. Users have something to select
4. Users have to add product to cart

**Exception:** System not ready

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.3.2** **Purchase**

**Use Case:** Purchase

**Primary Actor:** Customer

**Goal in context:** To purchase a product

**Preconditions:**

1. System has been designed for purchase
2. System has interface for purchase
3. System has list of product from which user can purchase **Triggers:** customer has to purchase the product

**Scenario:**

1. Visit page
2. Users have something to view from the list of product
3. Users select product(s) and add them to cart

28

1. Users have something to purchase

**Exception:**

·Products out of stock

·Product not available

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.3.3** **Cancel order before delivery**

**Use Case:** Cancel order before delivery

**Primary Actor:** Customer

**Goal in context:** To cancel an order before delivery

**Preconditions:**

1. System has been designed for cancel order before delivery
2. System has interface for cancel order before delivery
3. System has list of product
4. System has deadline time for cancel

**Triggers:** Customer has to cancel order before delivery

1. Visit page
2. Users must know deadline time for canceling the order
3. Users should cancel the order before delivery
4. Admin has to confirm the cancellation

**Exception:**

* Cancel request not approved by admin
* Deadline for request to cancel has crossed **Priority:** Essential, depends on user

**When Available:** First increment

**4.3.3.3** **Return Product**

**Use Case:** Return Product

**Primary Actor:** Customer

**Goal in context:** To return product

**Preconditions:**

1. System has been designed for return product
2. System has deadline time for return
3. System has list of reason about returning product **Triggers:** Customer has to return product

**Scenario:**

1. Users must know deadline time for returning product
2. Users must select the reason for returning product
3. Users must know that they will be penalize some amount

**Exception:**

1. The deadline of returning the product is over **Priority:** Essential, depends on user

29

**When Available:** First increment

**4.3.4 Review**

**4.3.4.1** **Comment**

**Use Case:** comment

**Primary Actor:** Customer

**Goal in context:** To comment on product

**Preconditions:**

1. System has been designed for comment
2. System has interface for comment
3. System has list of product

**Triggers:** customer wants to comment on product

1. Visit page
2. Users have something to view
3. Users have to add comment

**Exception:**

·Invalid Input

· Product is not available for commenting **Priority:** Essential, depends on user.

**When Available:** First increment

**4.3.4.2** **Rate product**

**Use Case:** Rate product

**Primary Actor:** Customer

**Goal in context:** to rate product

**Preconditions:**

1. System has been designed for rating the product
2. System has interface for rating the product
3. System has list of product

**Triggers:** customer wants to comment on item

1. Visit page
2. Users have something to view
3. Users have to rate product

**Exception:**

·Decimal point not allowed

· Product not available for rating **Priority:** Essential, must be implemented. **When Available:** First increment

30

**4.3.5 Payment Method**

**4.3.5.1** **Cash on Delivery**

**Use Case:** Cash on delivery

**Primary Actor:** Customer

**Goal in context**: To pay cash on delivery

**Preconditions:**

1. System has been designed for Cash on delivery
2. System has interface for cash on delivery **Triggers:** customer wants to pay cash on delivery

**Scenario:**

1. Customer has an order list for which he/she wants to pay
2. Customer should select payment method.
3. Admin should check place
4. Inside Dhaka pathao delivery
5. Outside Dhaka Courier delivery (S.A Paribahan)

**Exception:**

·Delivery organization currently not available for delivery

·Location ambiguous

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.5.2** **Advance Online Payment**

**Use Case:** Advance Online Payment

**Primary Actor:** Customer

**Goal in context:** To pay cash online

**Preconditions:**

1. System has been designed for Advance online payment
2. System has interface for Advance online payment
3. System has list of online payment method

Triggers: customer wants to pay cash online

**Scenario:**

1. Customer has an order list for which he/she wants to pay
2. Customer should select Payment method
3. Customer should give required information
4. Admin should verify the given information
5. Send message for complete payment

**Exception:**

* system not working properly
* pin number may not matching **Priority:** Essential, must be implemented. **When Available:** First increment

31

**4.3.6 Product Management**

**4.3.6.1** **Add Product**

**Use Case:** Add Product

**Primary Actor:** Admin

**Goal in context:** To add Product

**Preconditions:**

1. System has been designed for adding Product
2. System has interface for adding Product
3. System has a database of products
4. Is a Admin
5. Admin has something to add
6. Admin should set category with number of product.
7. Admin adds product with corresponding product id, quantity and price

**Exception:**

·Product ID wrong

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.6.2** **Remove Product**

**Use Case**: Remove Product

**Primary Actor:** Admin

**Goal in context:** To remove Product

**Preconditions:**

1. System has been designed for removing Product
2. System has interface for removing Product
3. System has a database of products
4. Is a Admin
5. Admin has something to remove
6. Admin can remove the product

**Exception:**

·Product does not exist

·Doesn’t get product by searching

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.6.3** **Edit Product**

32

**Use Case:** Edit Product

**Primary Actor:** Admin

**Goal in context:** To edit Product

**Preconditions:**

1. System has been designed for editing Product
2. System has interface for editing Product
3. System has a database of products
4. Is a Admin
5. Admin has something to edit
6. Admin has search product and edit required information
7. Admin confirms the changes

**Exception:**

* product may not exist
* New information is wrong **Priority:** Essential, must be implemented. **When Available:** First increment

**4.3.6.4** **Manage Product Advertisement**

**Use Case:** Manage Product Advertisement

**Primary Actor:** Admin

**Goal in context:** To manage product advertisement

**Preconditions:**

1. System has been designed for managing product advertisement
2. System has interface for managing product advertisement
3. System has a database of products

**Triggers:** Admin has to manage product advertisement

1. Is a Admin
2. Admin has something to manage
3. Admin has advertisement options
4. Send notification message via SMS or Email for particular products

**Exception:**

· System not ready for sending Email or SMS **Priority:** Essential, must be implemented. **When Available:** First increment

**4.3.7 Contact**

**4.3.7.1** **Chat**

**Use Case:** Chat, Email

**Primary Actor:** Admin and Customer

33

**Goal in context:** To chat with admin

**Preconditions:**

1. System has been designed for chat
2. System has interface for chat

**Triggers:** Admin and customer has to chat and Email

1. Admin availability is required
2. User sends messages
3. Admin retrieves messages
4. If admin not available message request is sent

**Exception:**

·Invalid input

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.7.2** **Help Line**

**Use Case:** Help Line

**Primary Actor:** Customer

**Goal in context:** To help customer

**Preconditions:**

1. System has been designed for help line
2. System has interface for help line

**Triggers:** customer has to request for help through help line

1. Is a customer
2. Customer should request for help
3. If on phone, number is loaded
4. If on Computer, number is displayed

**Exception:**

·contact number not available

**Priority:** Essential, must be implemented.

**When Available:** First increment

**4.3.7.3** **Follow on Social Media**

**Use Case:** Follow on Social Media

**Primary Actor:** customer

**Goal in context:** To follow on social media

**Preconditions:**

1. System has been designed for follow on social media
2. System has interface for follow on social media
3. Has list of Social media

**Triggers:** Customer has to follow on social media

1. Is a customer
2. Visit page

34

1. Customer has to follow on social media
2. Customer has to loin on social media

**Exception:**

·some social media may not be available

· Login to social media failed **Priority:** Essential, must be implemented. **When Available:** First increment

**4.3.7.5** **Chat**

**Use Case:** Email

**Primary Actor:** Admin and Customer

**Goal in context:** To Email to admin

**Preconditions:**

1. System has been designed for Email
2. System has interface for Email **Triggers:** Customer has to Email to Admin

**Scenario:**

1. Admin has an email address
2. Customer sends email to admin

**Exception:**

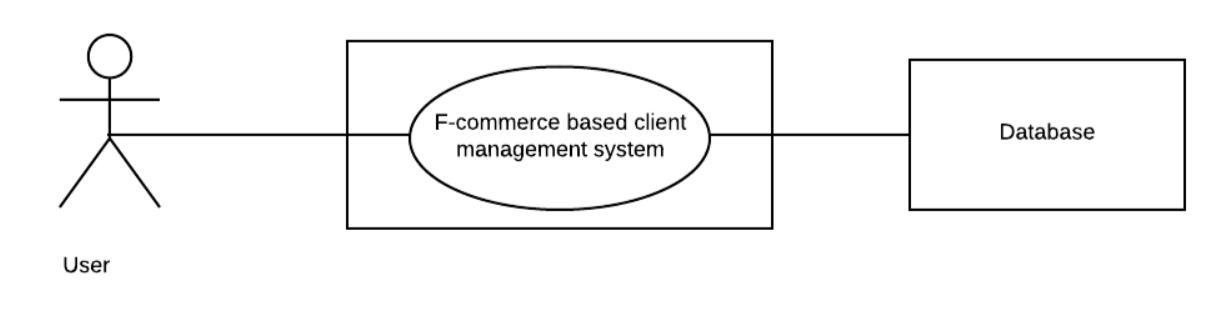
·Invalid input

**Priority:** Essential, must be implemented.

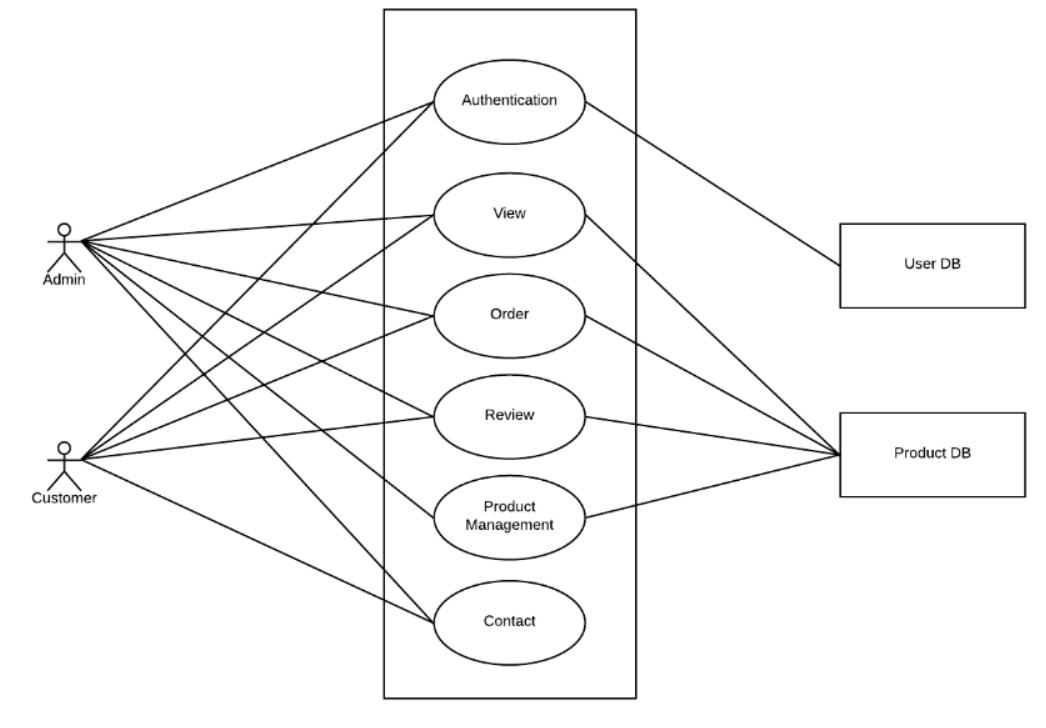
**When Available:** First increment

35

**4.4 Use Case Diagram**

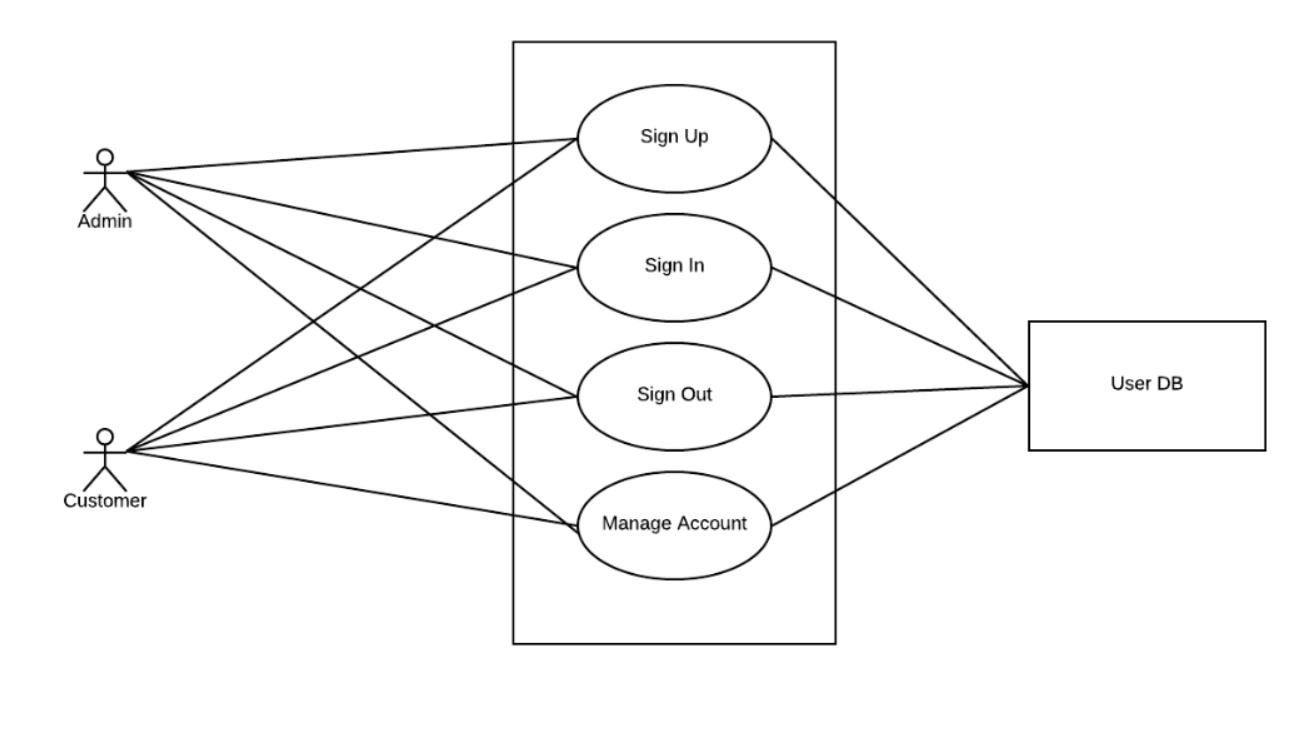


**Figure 1: Level 0 for F-commerce based client management system**



**Figure 2: Level 1 for F-commerce based client management system**

36



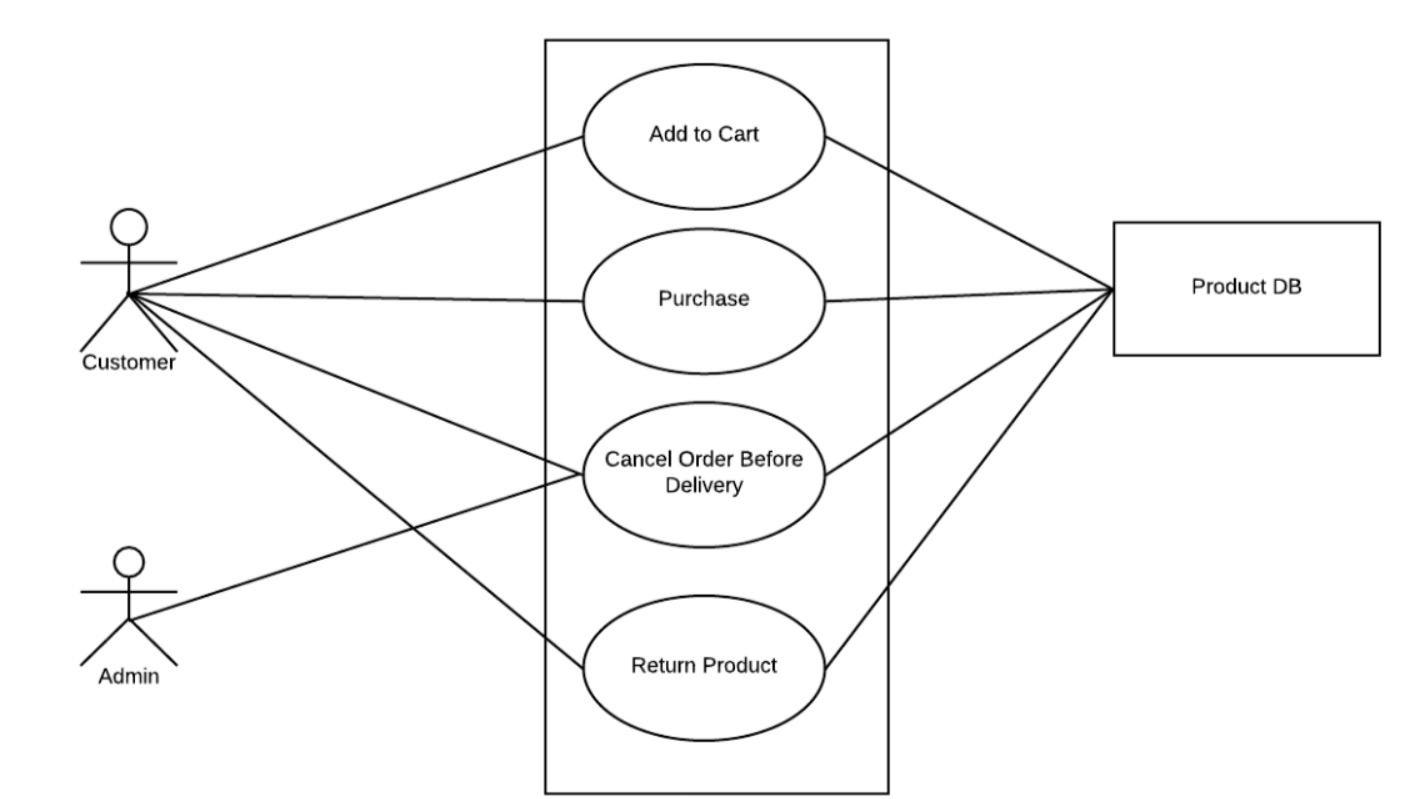
**Figure 3: Level 2.1 for F-commerce based client management system**

37

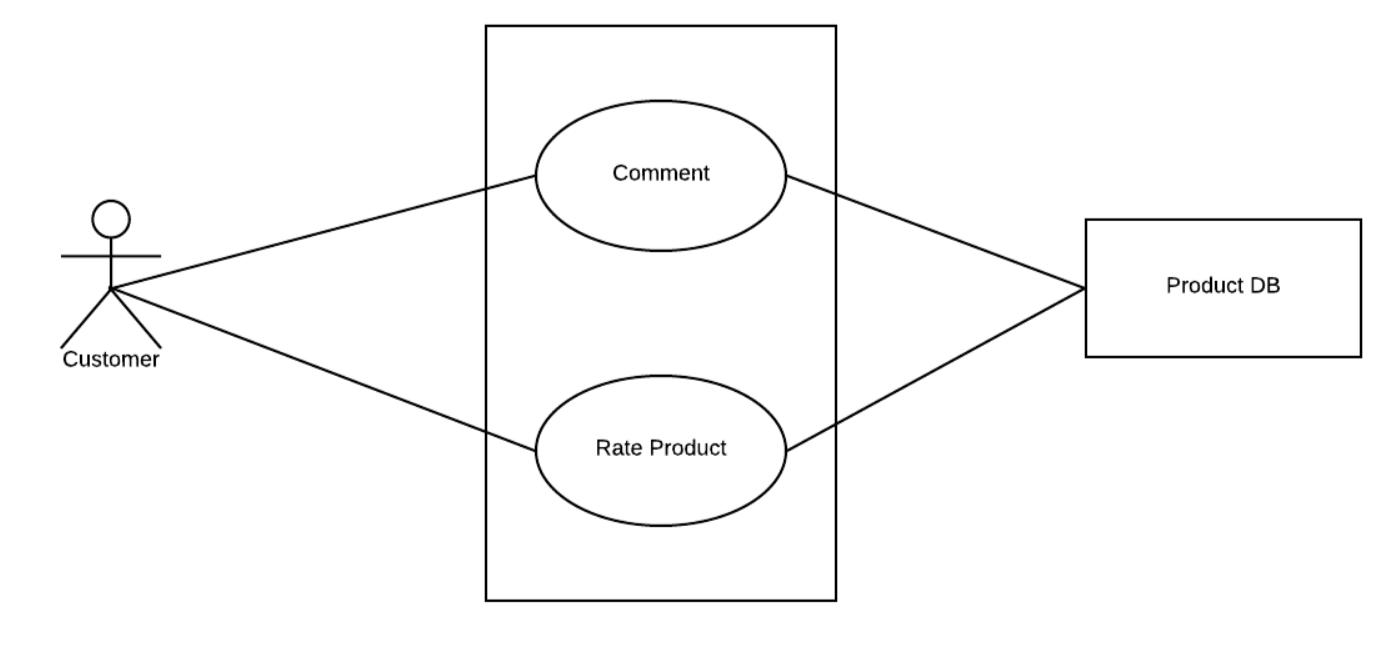


**Figure 4: Level 2.2 for F-commerce based client management system**

38

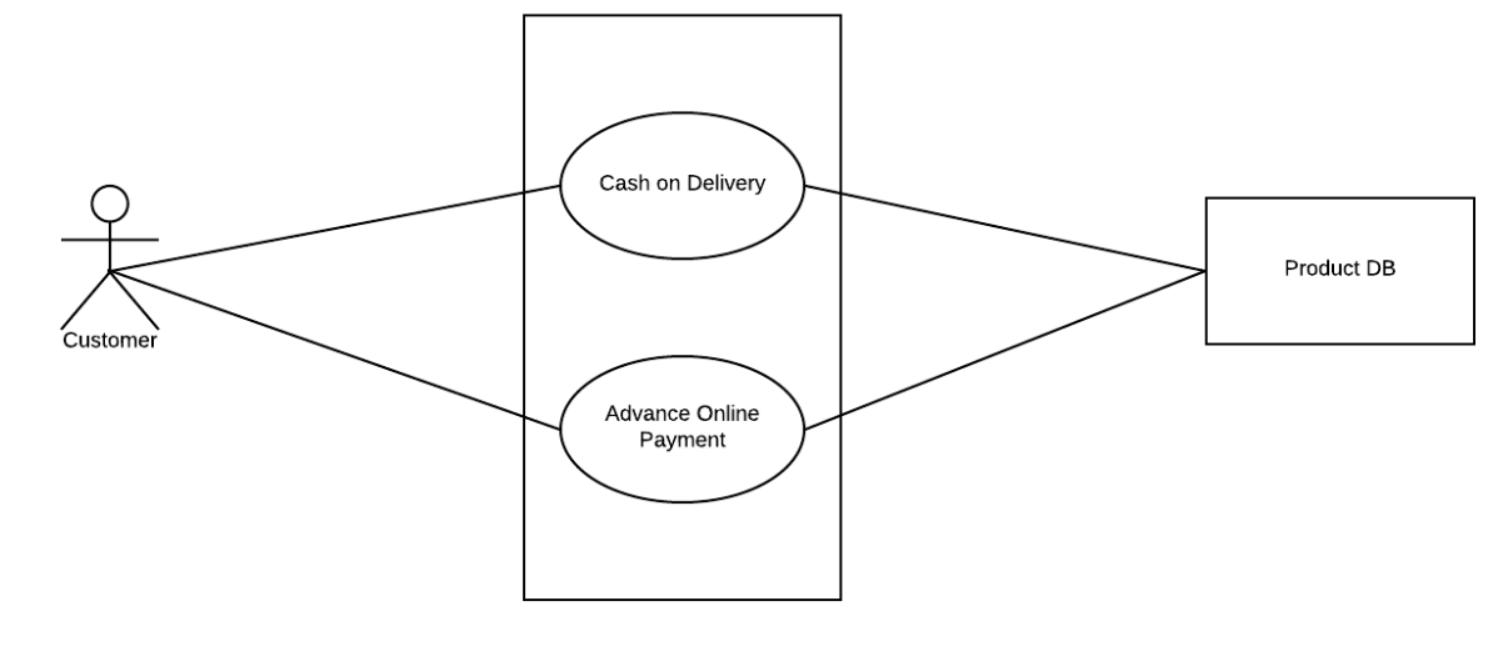


**Figure 5: Level 2.3 for F-commerce based client management system**

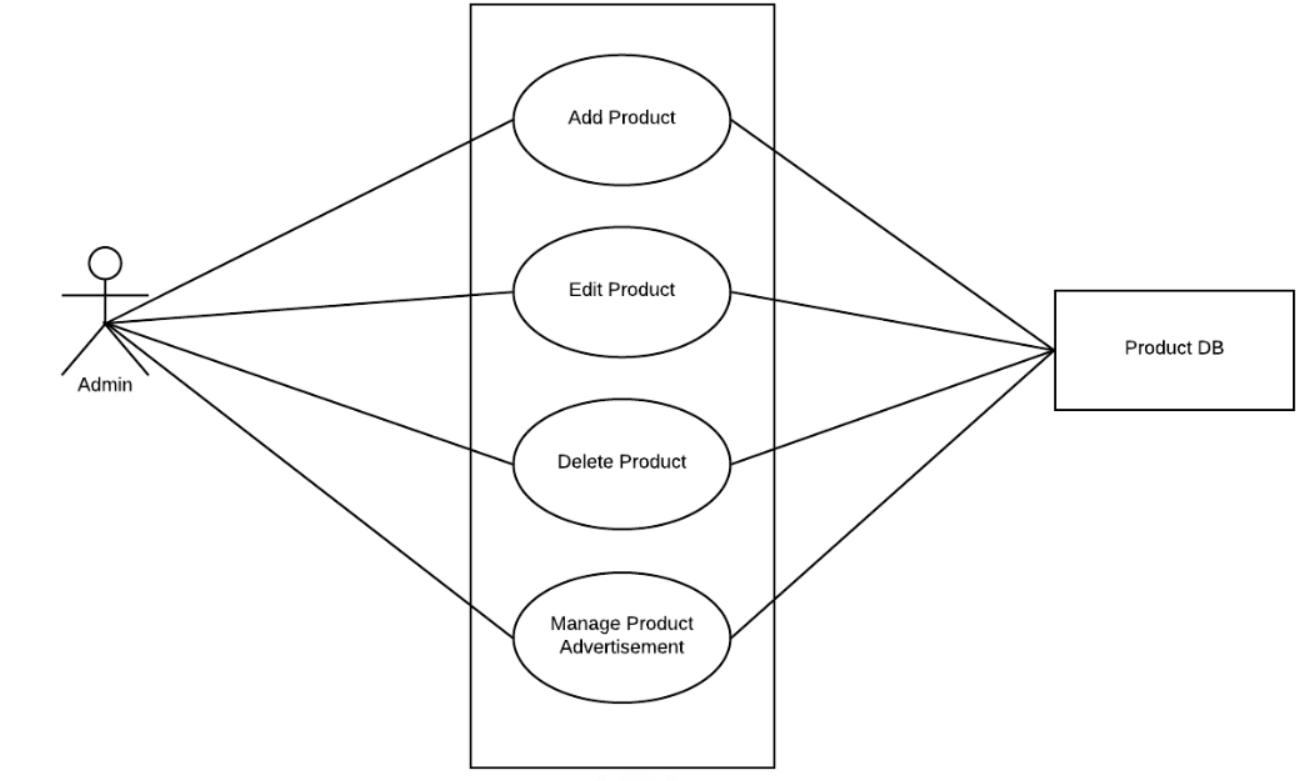


**Figure 6: Level 2.4 for F-commerce based client management system**

39

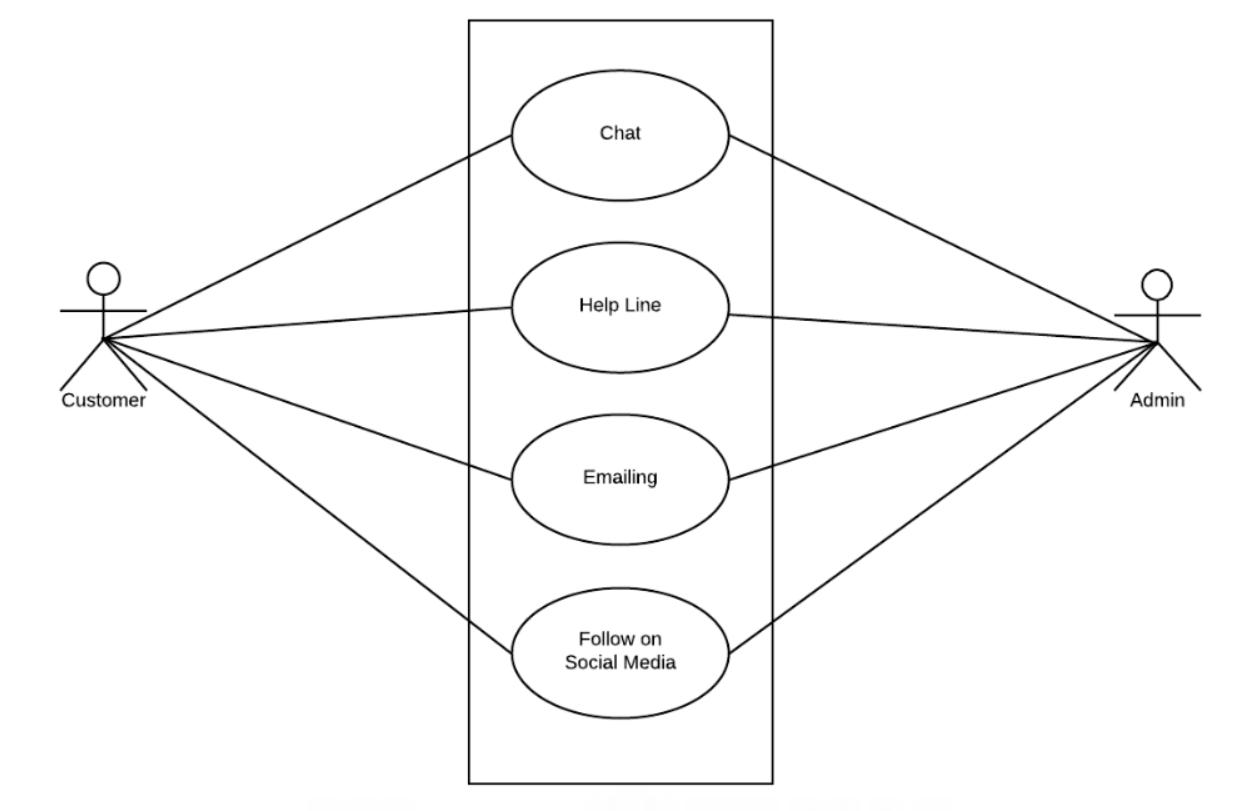


**Figure 7 : Level 2.5 for F-commerce based client management system**



**Figure 8: Level 2.6 for F-commerce based client management system**

40



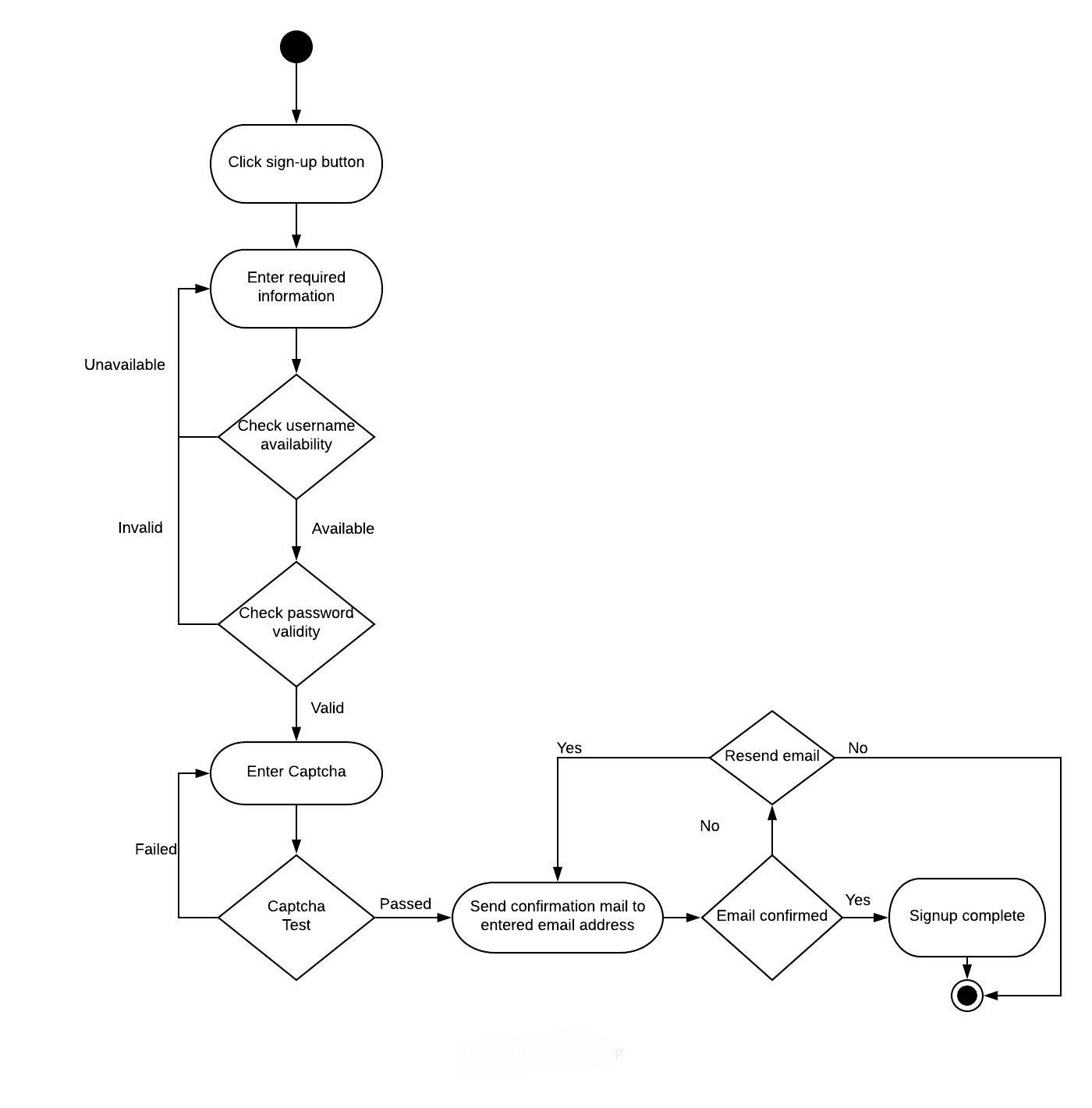
**Figure 9: Level 2.7 for F-commerce based client management system**

41

**4.5 Activity Diagram and Swimlane Diagram of generated Use Cases:**

**Use case 1:** Sign Up

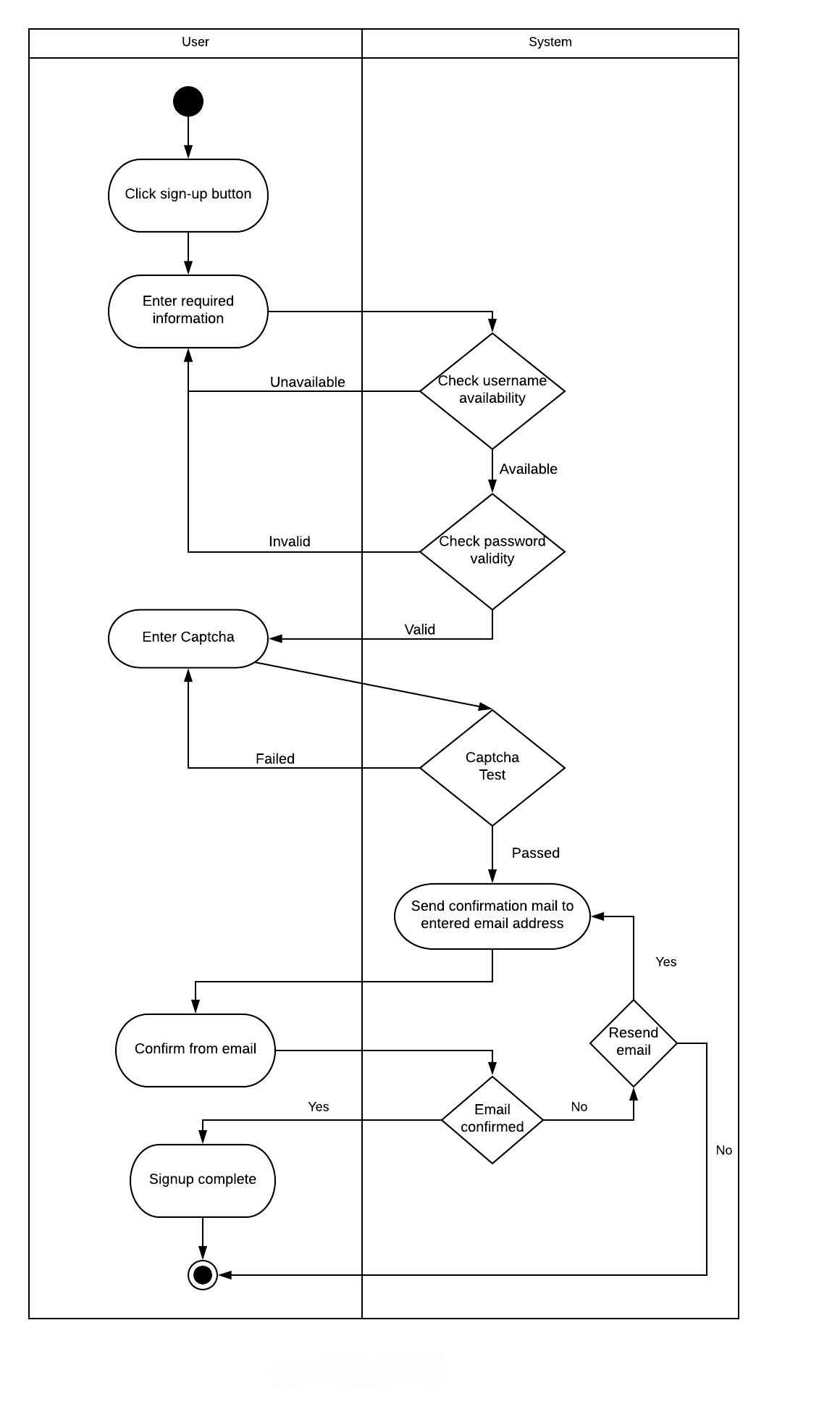
**Activity Diagram:**



**Figure 10: Activity Diagram for Sign-Up**

42

**Swimlane Diagram:**

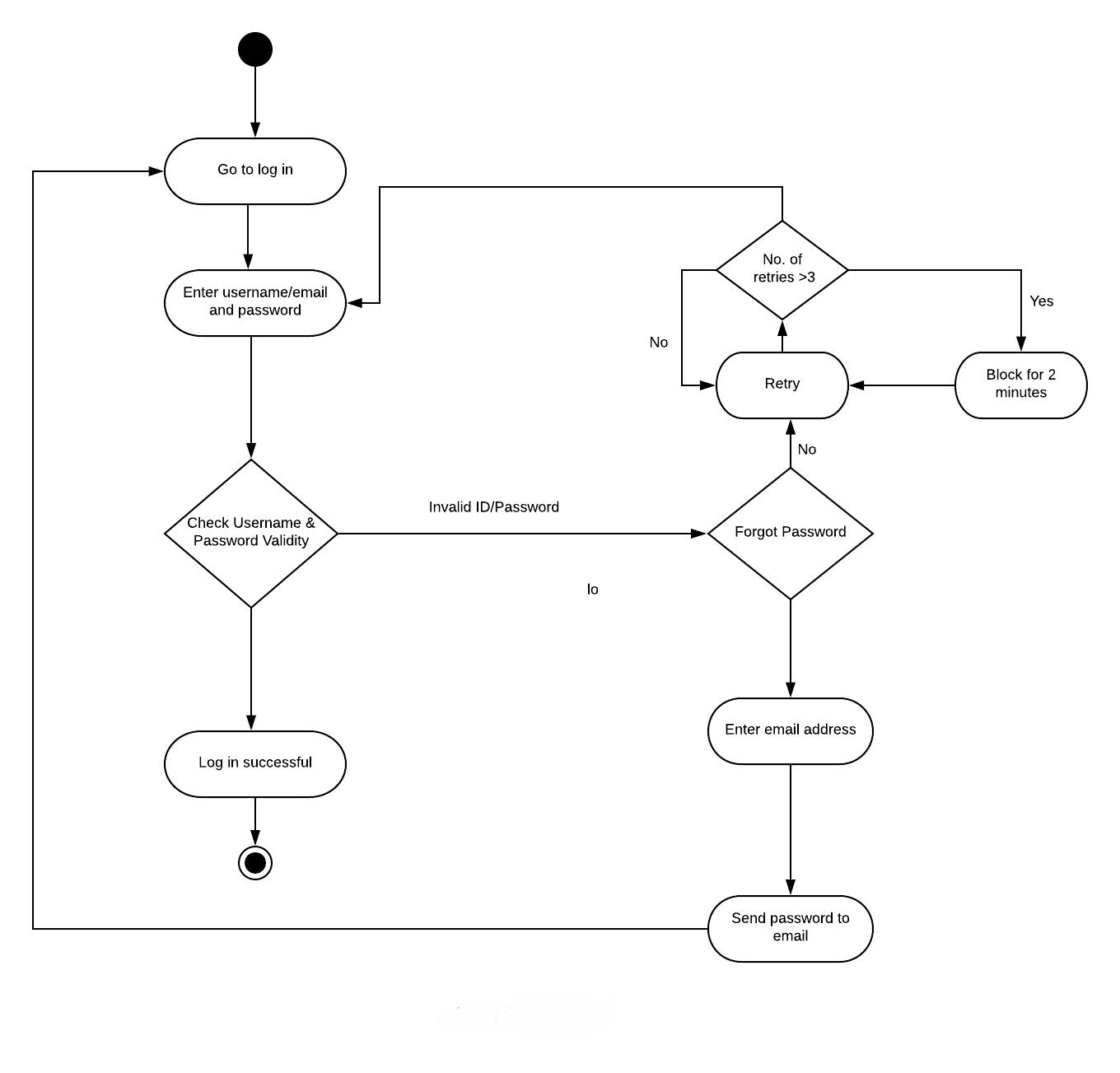


**Figure 11: Swimlane Diagram for Sign-Up**

43

**Use case 2:** Sign In

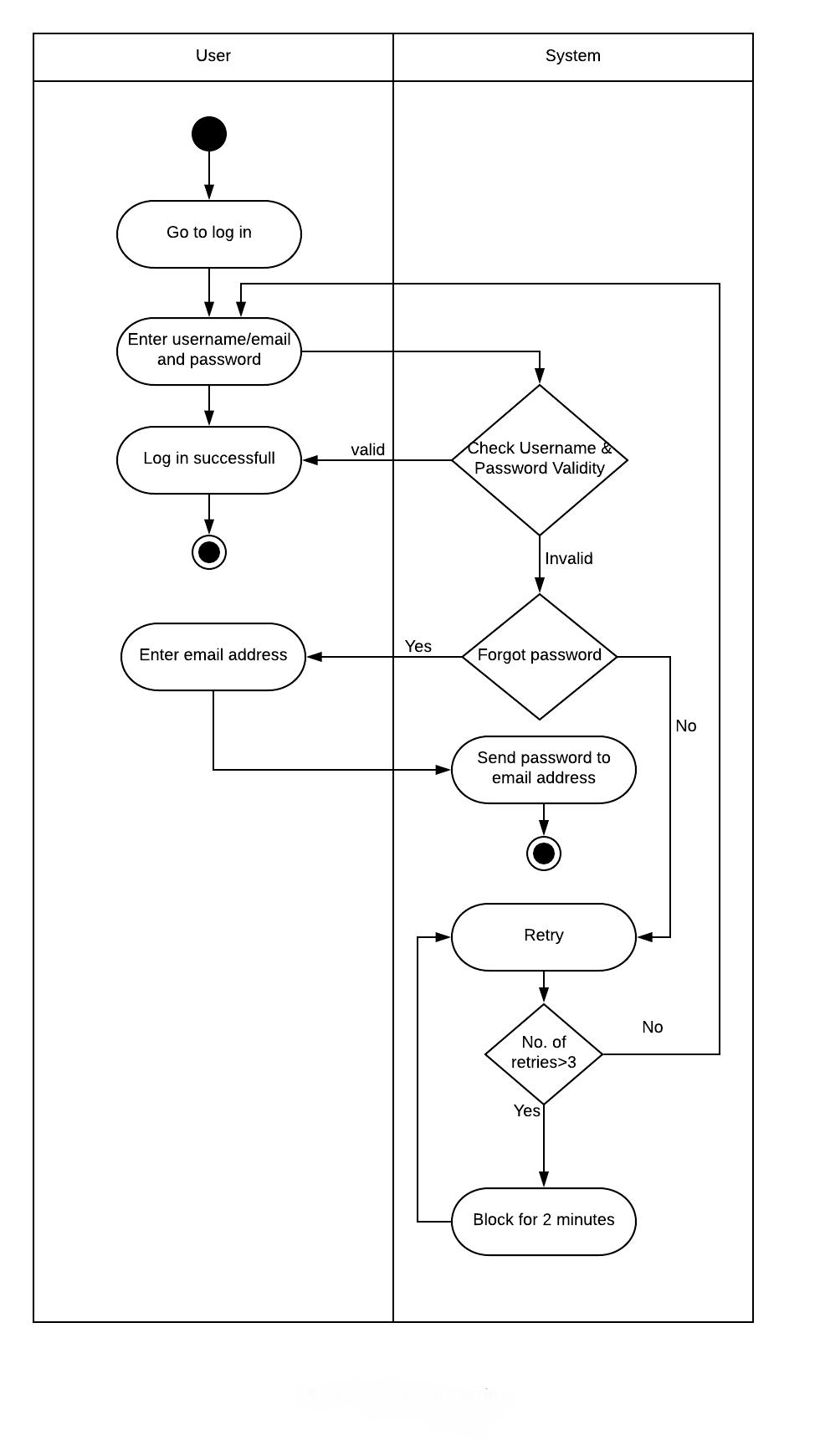
**Activity Diagram:**



**Figure 12: Activity Diagram for Login**

44

**Swimlane Diagram:**

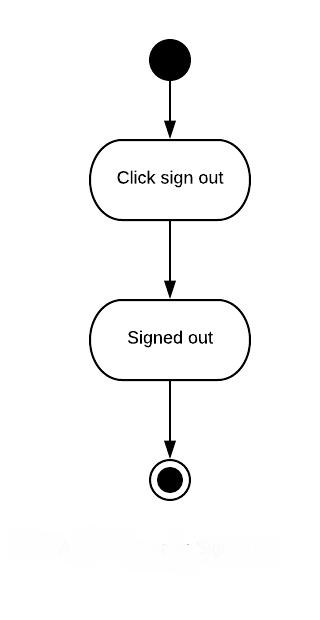


**Figure 13: Swimlane Diagram for Login**

45

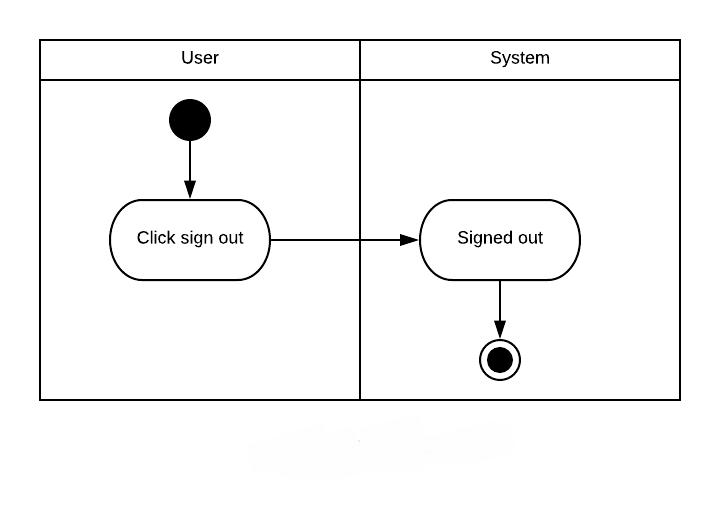
**Use case 3:** Sign Out

**Activity Diagram:**



**Figure 14: Activity Diagram for Sign-Out**

**Swimlane Diagram:**



**Figure 15: Swimlane Diagram for Sign-out**

46

**Use case 4:** Manage Account

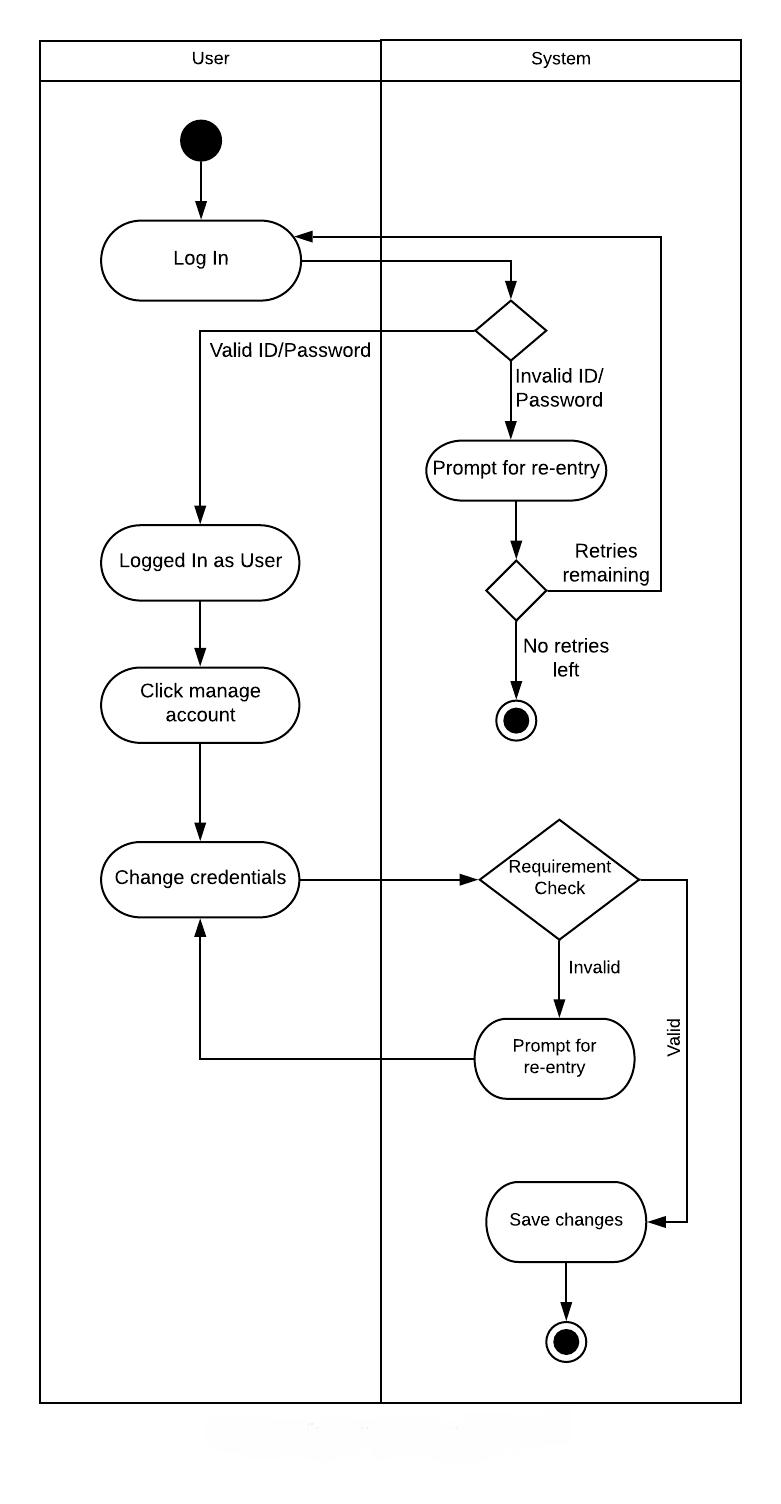
**Activity Diagram:**



**Figure 16: Activity Diagram for Manage Account**

47

**Swimlane diagram:**

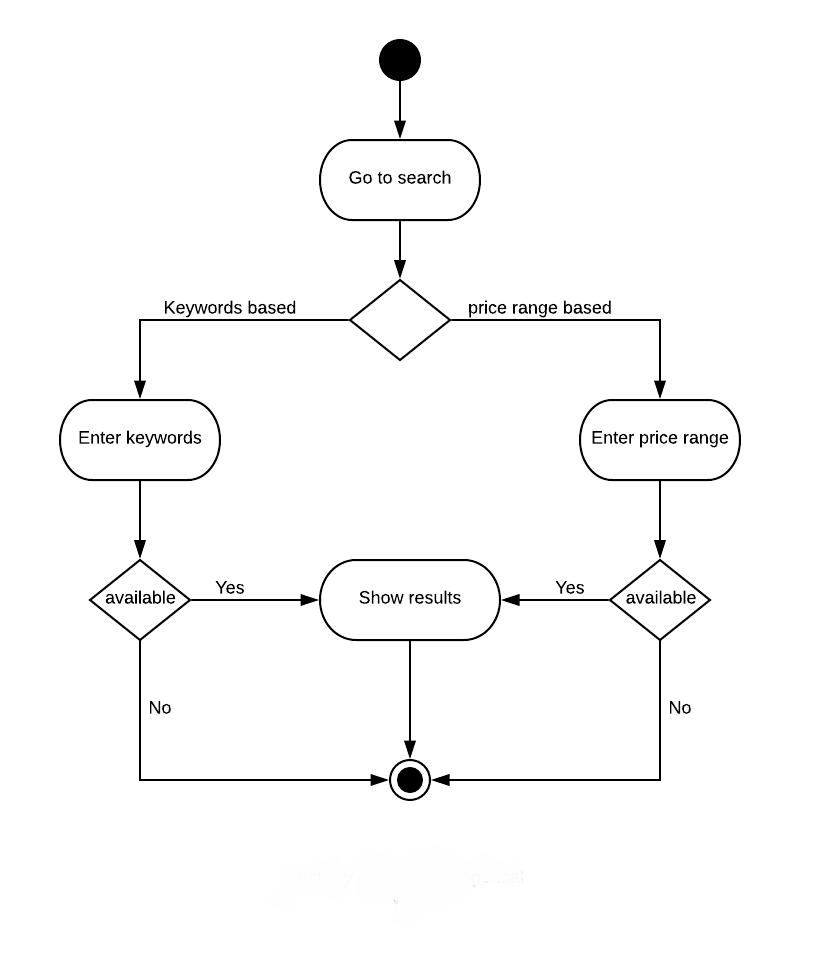


**Figure 17: Swimlane Diagram for Manage Account**

48

**Use case 5:** Browse Category and Search

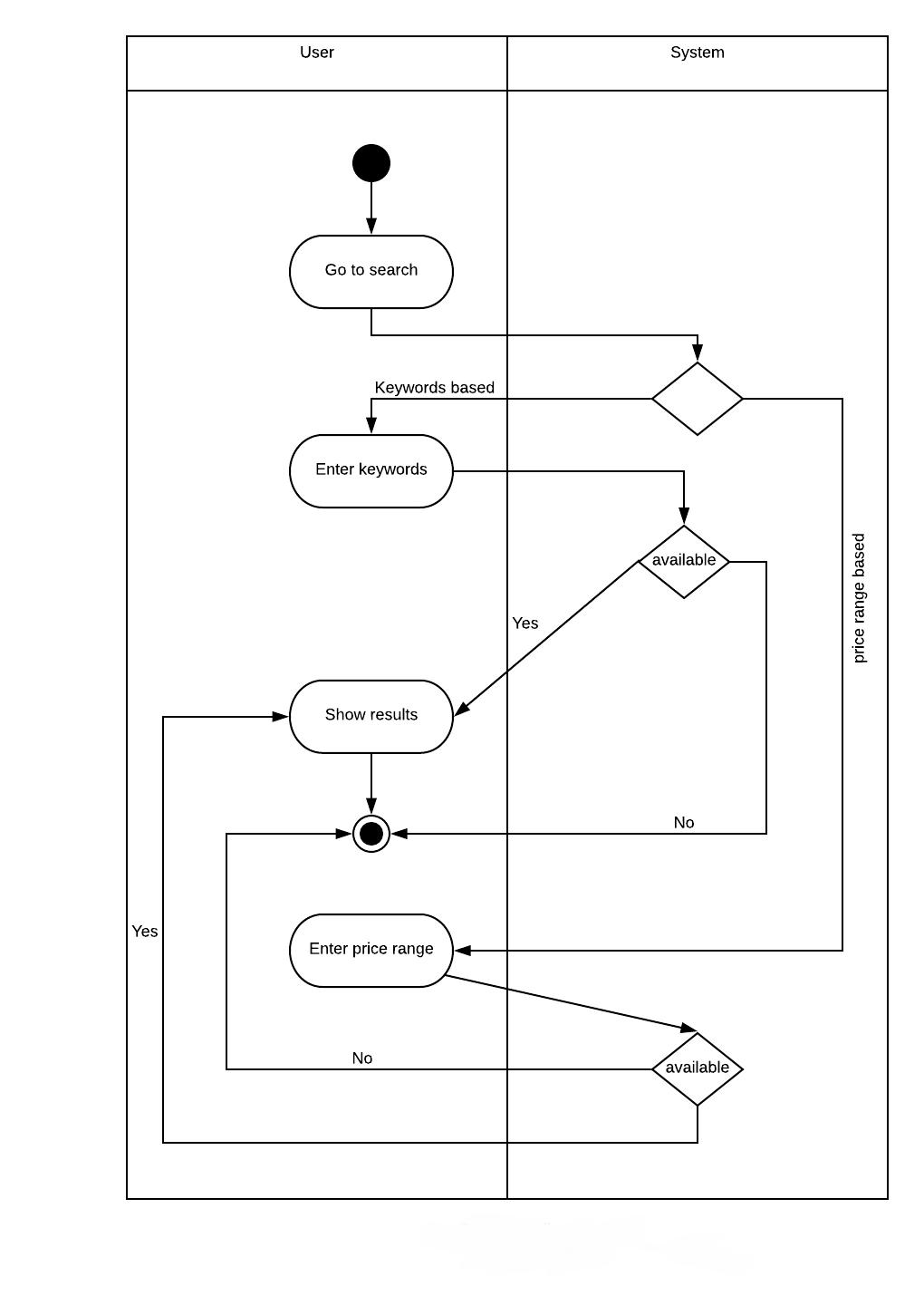
**Activity Diagram:**



**Figure 18: Activity Diagram for Browse Category and Search**

49

**Swimlane Diagram:**

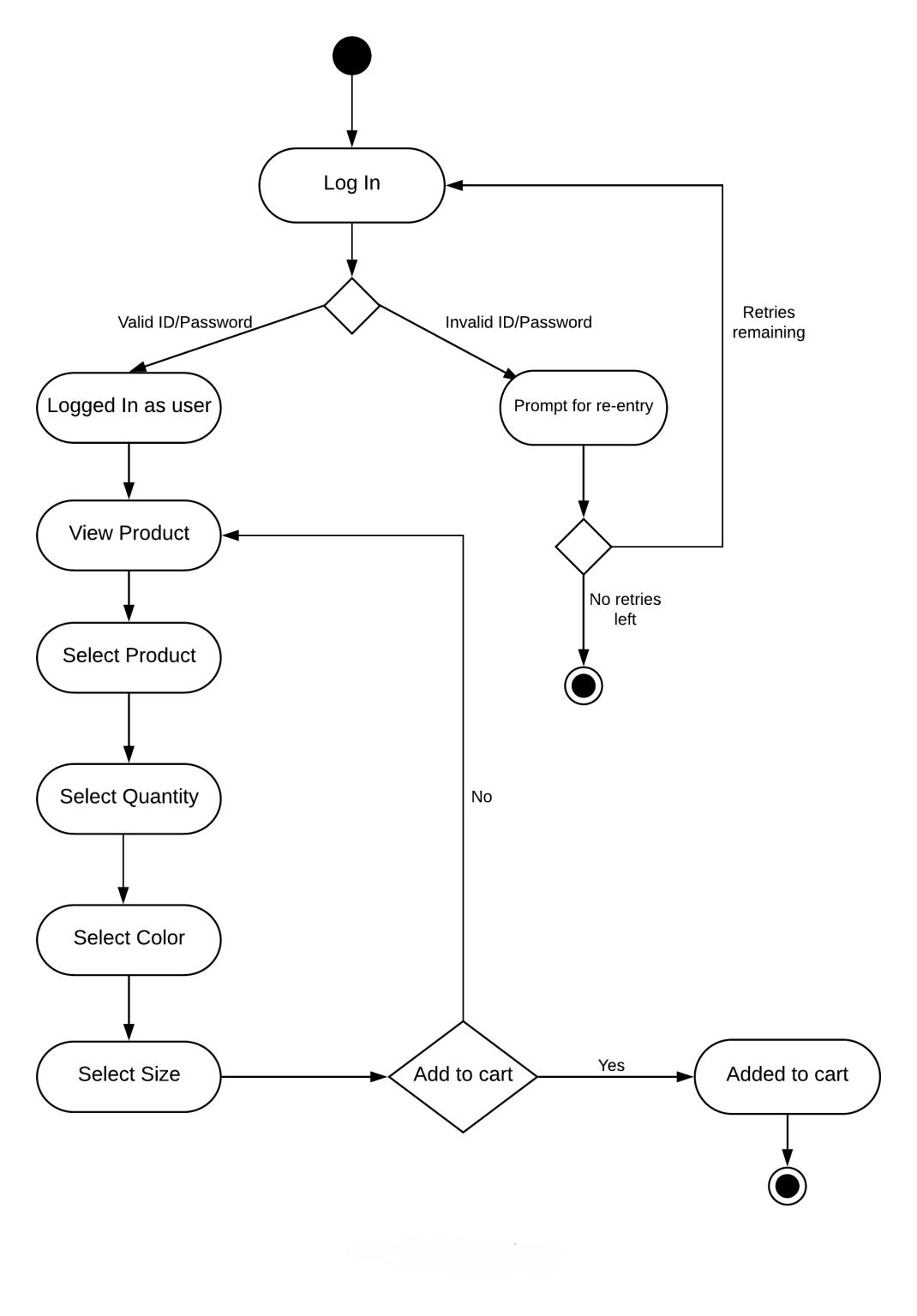


**Figure 19: Swimlane Diagram for Browse Category and Search**

50

**Use case 6: Add to cart**

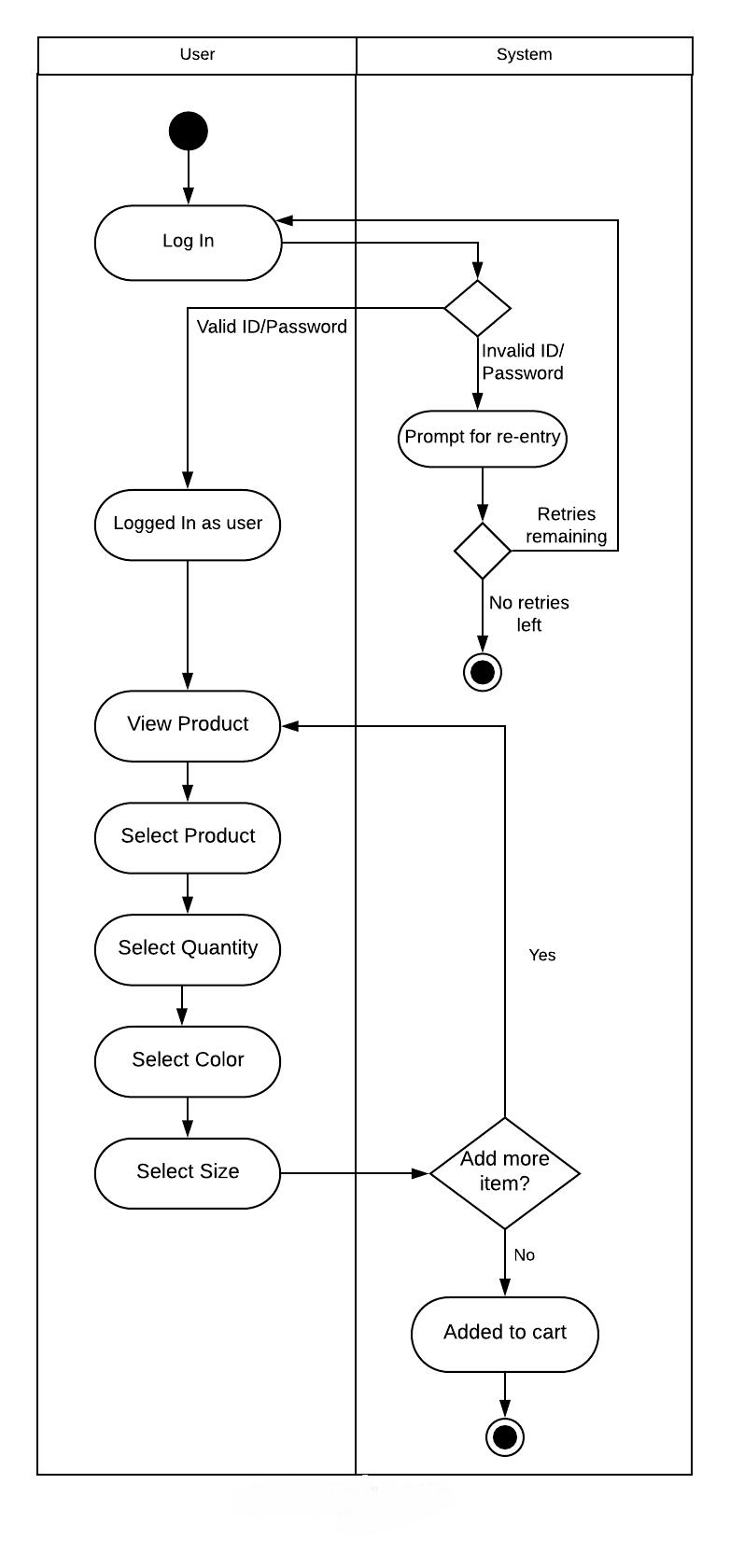
**Activity Diagram:**



**Figure 20: Activity Diagram for Add to Cart**

51

**Swimlane Diagram:**

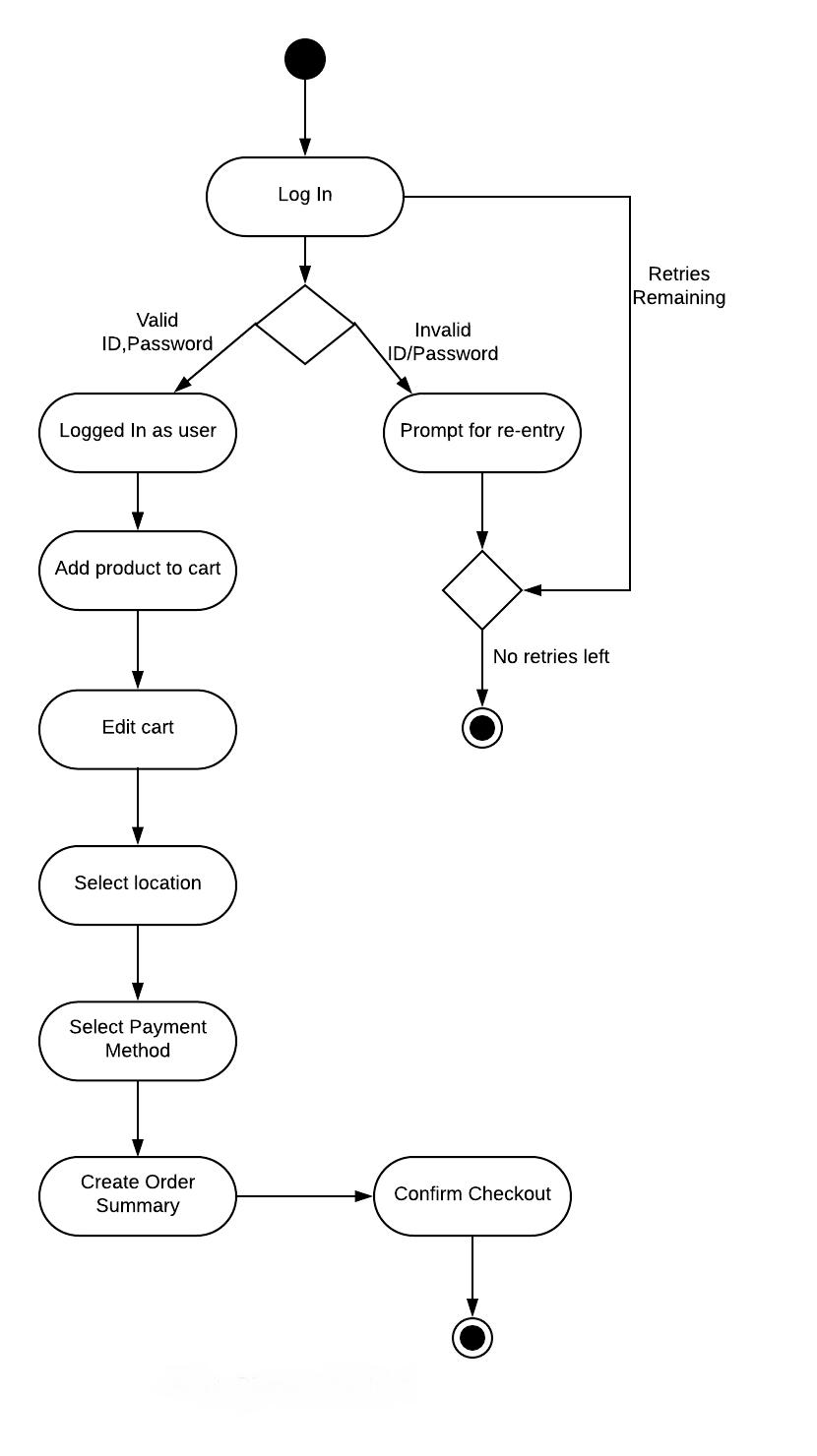


**Figure 21: Swimlane Diagram for Add to Cart**

52

**Use case 7:** Purchase

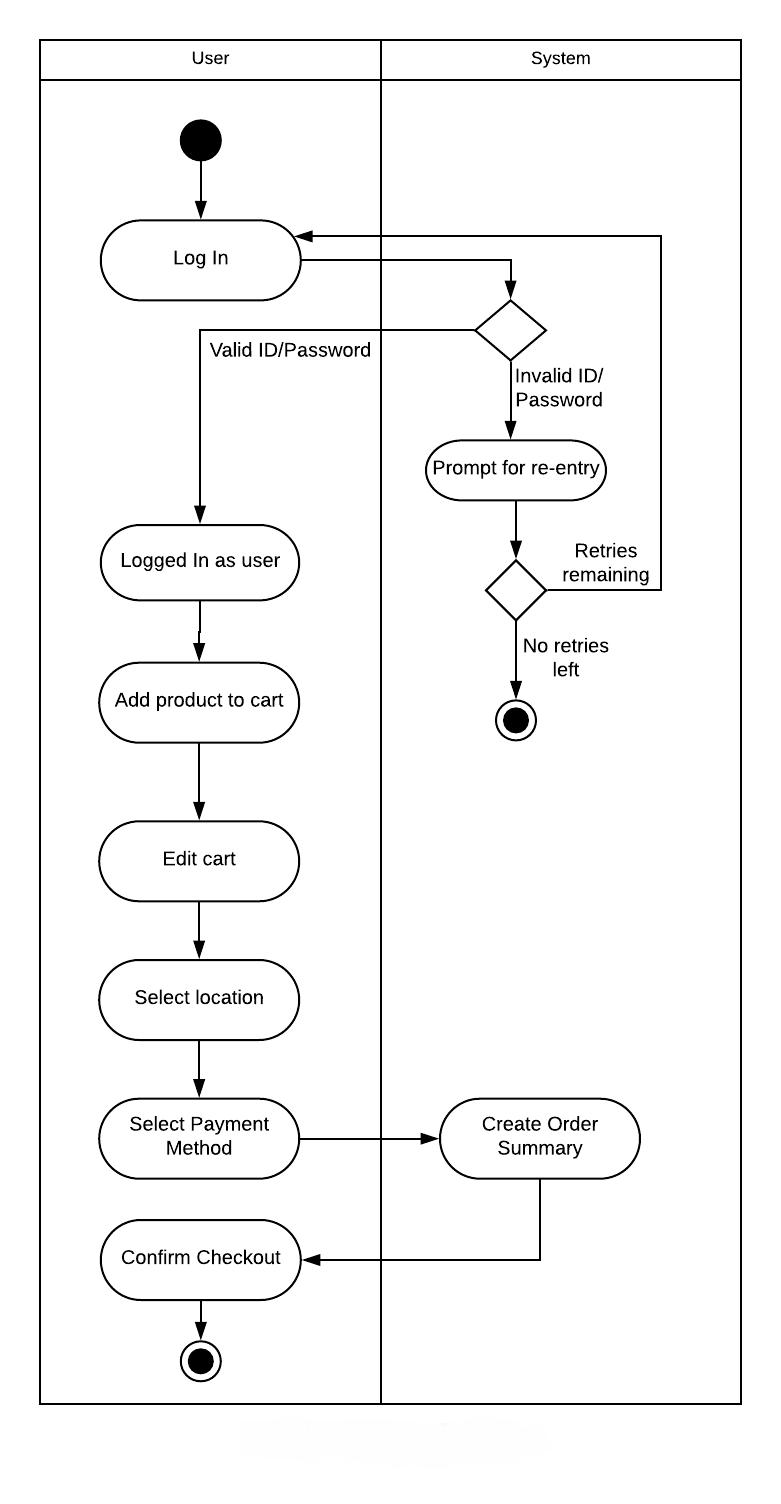
**Activity Diagram:**



**Figure 22: Activity Diagram for Purchase**

53

**Swimlane Diagram:**

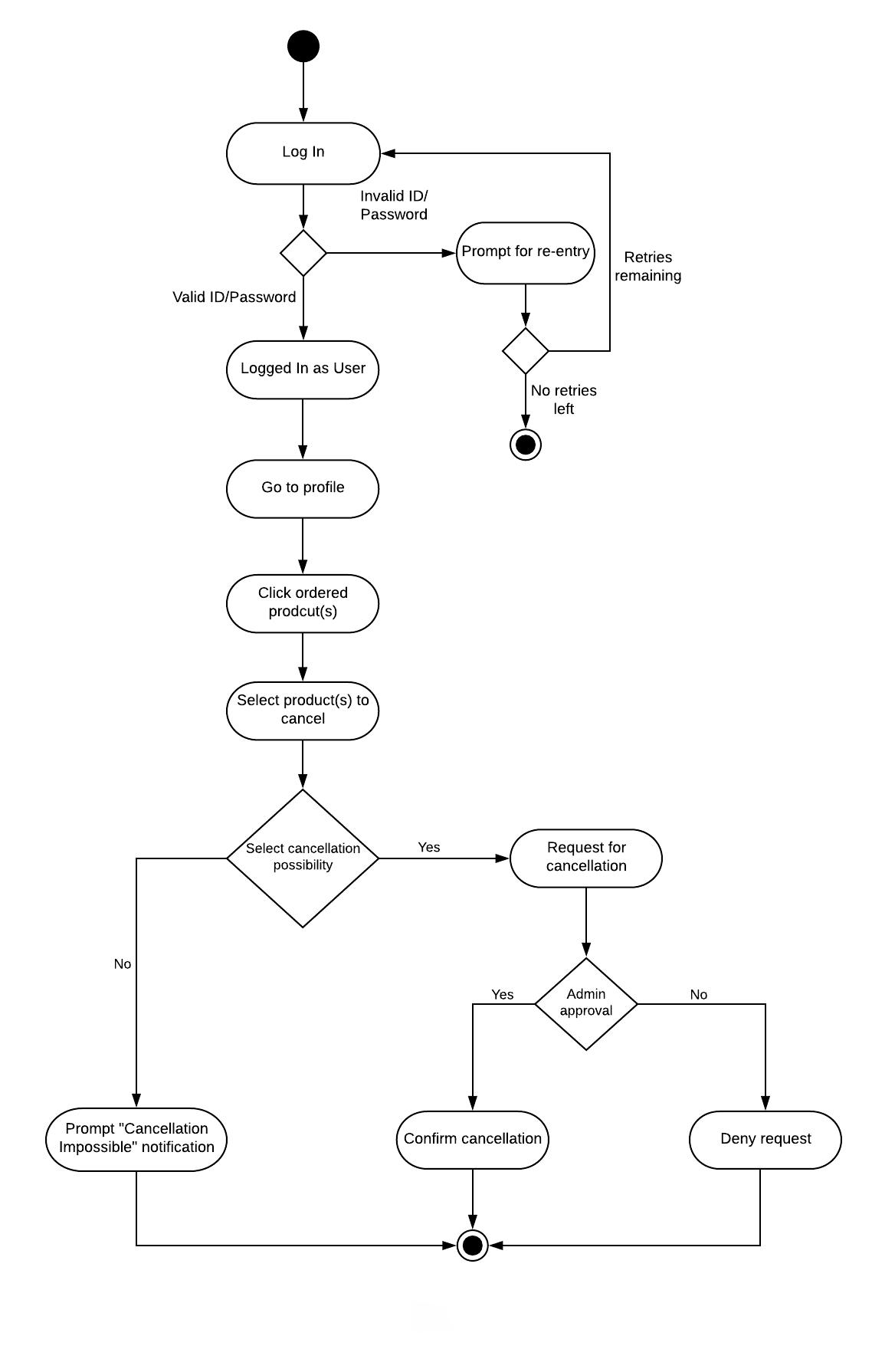


**Figure 23: Swimlane Diagram for Purchase**

54

**Use case 8:** Cancel order before delivery

**Activity Diagram:**



**Figure 24: Activity Diagram for Cancel Order Before delivery**

55

**Swimlane Diagram:**

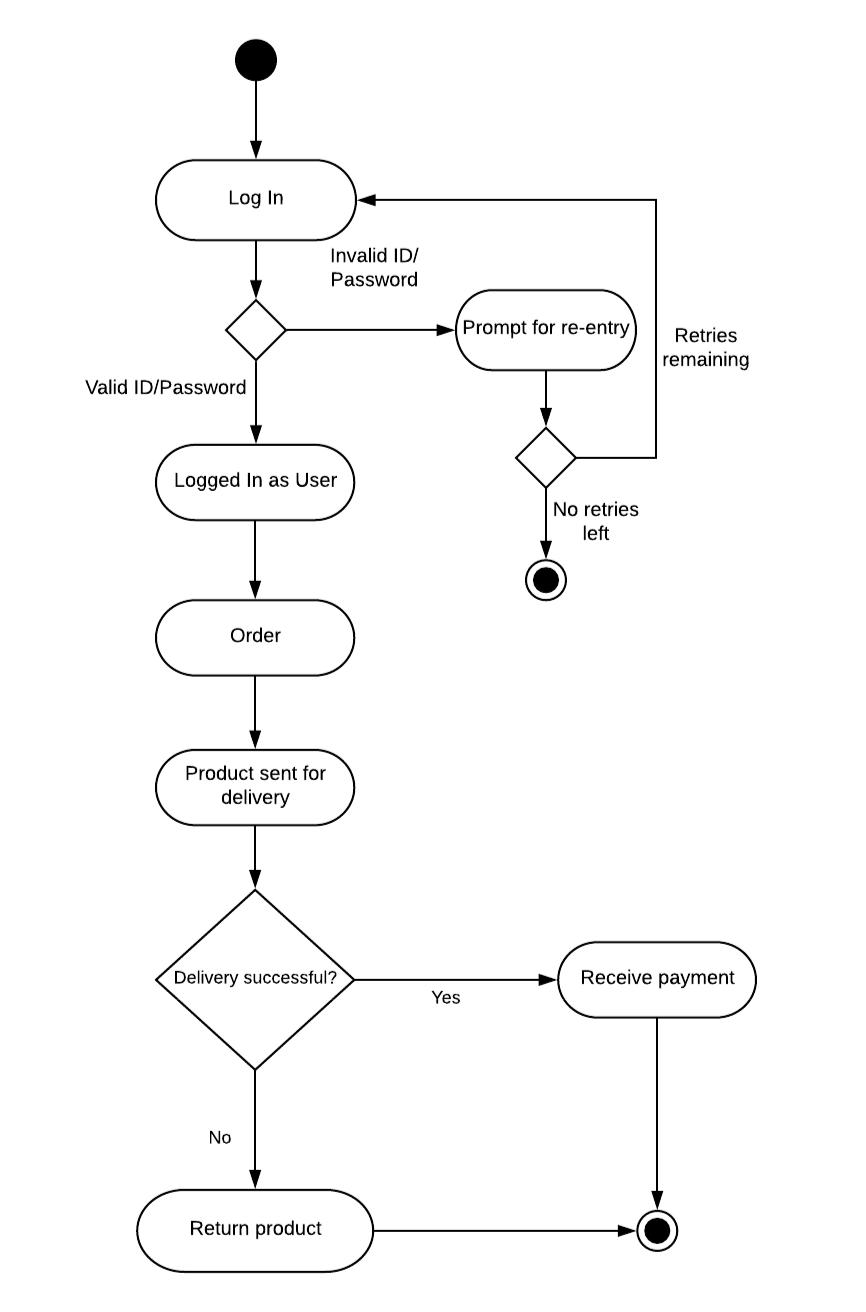


**Figure 25: Swimlane Diagram for Cancel Order Before delivery**

56

**Use case 9:** Return product

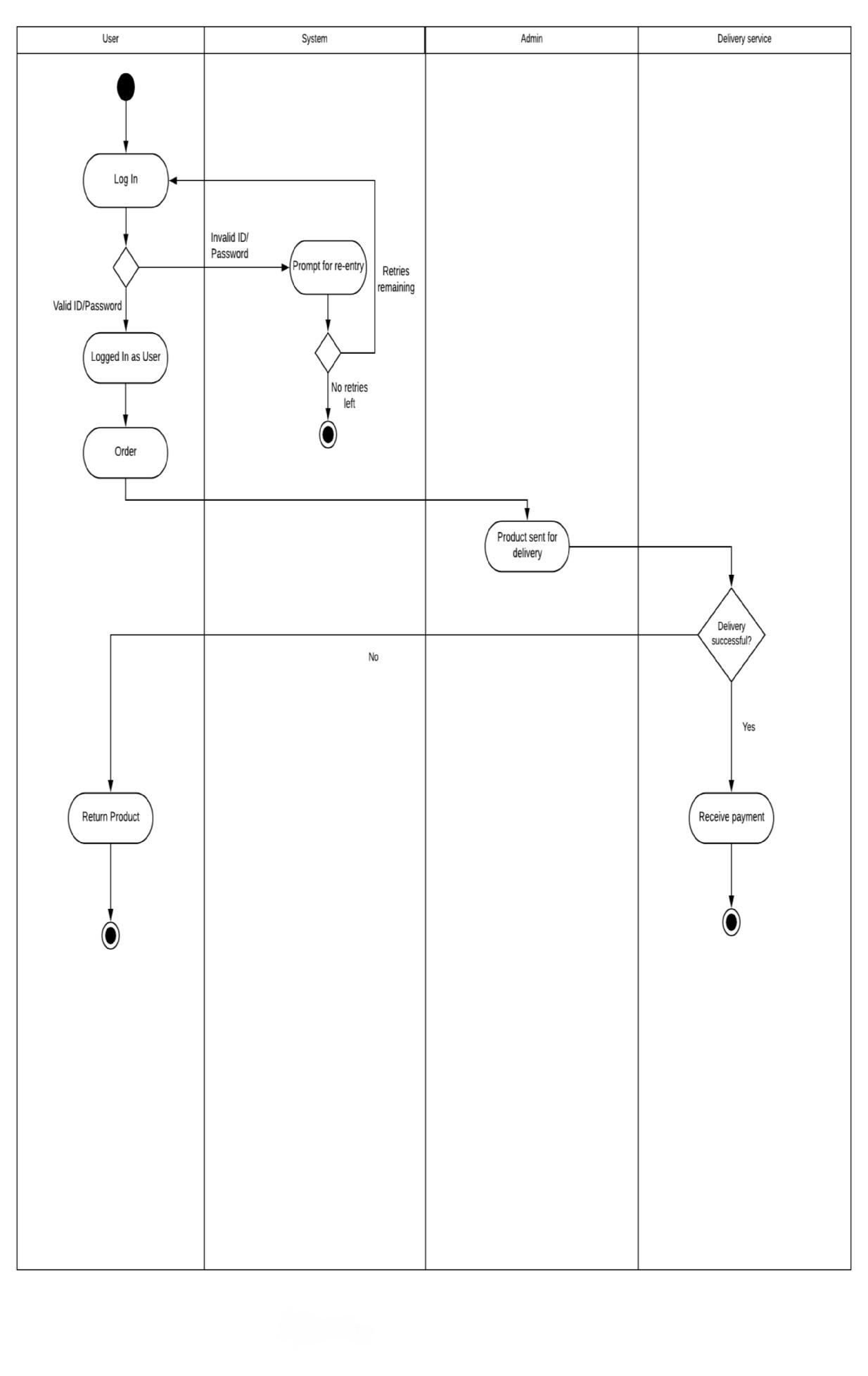
**Activity Diagram:**



**Figure 26: Activity Diagram for Return Product**

57

**Swimlane Diagram:**

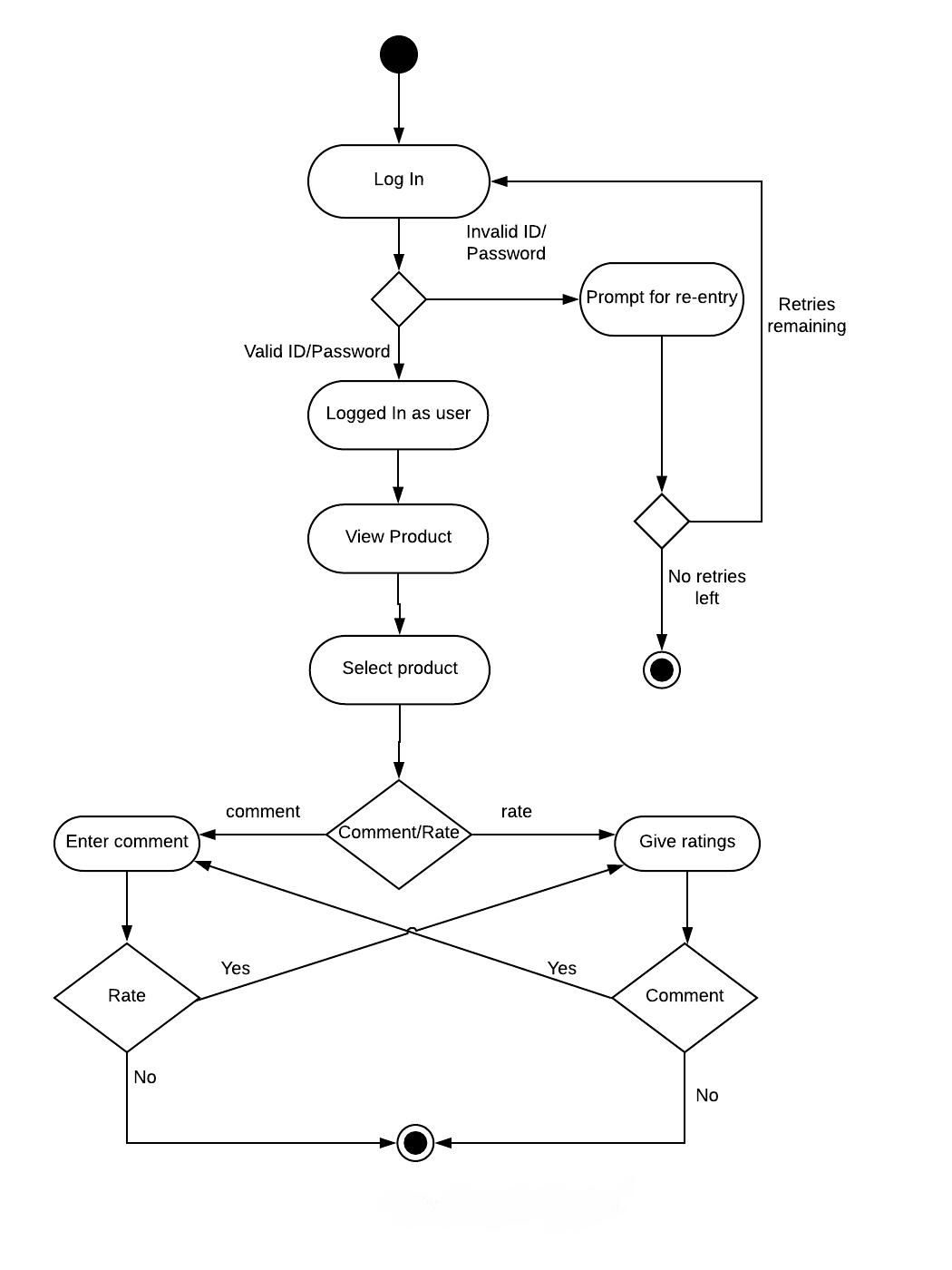


**Figure 27: Swimlane Diagram for Return Product**

58

**Use case 10:** comment and rate

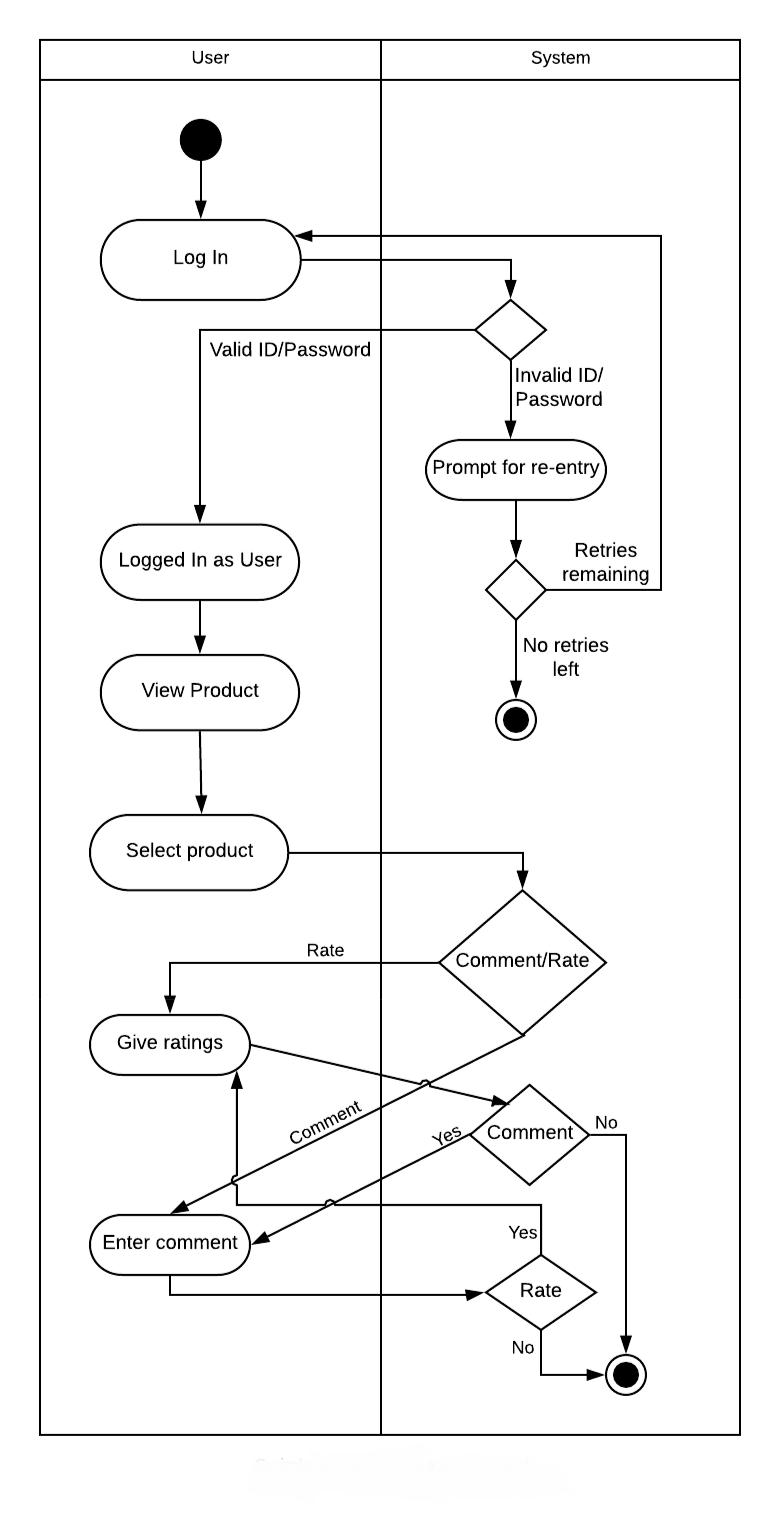
**Activity Diagram:**



**Figure 28: Activity Diagram for Comment and rate**

59

**Swimlane Diagram:**

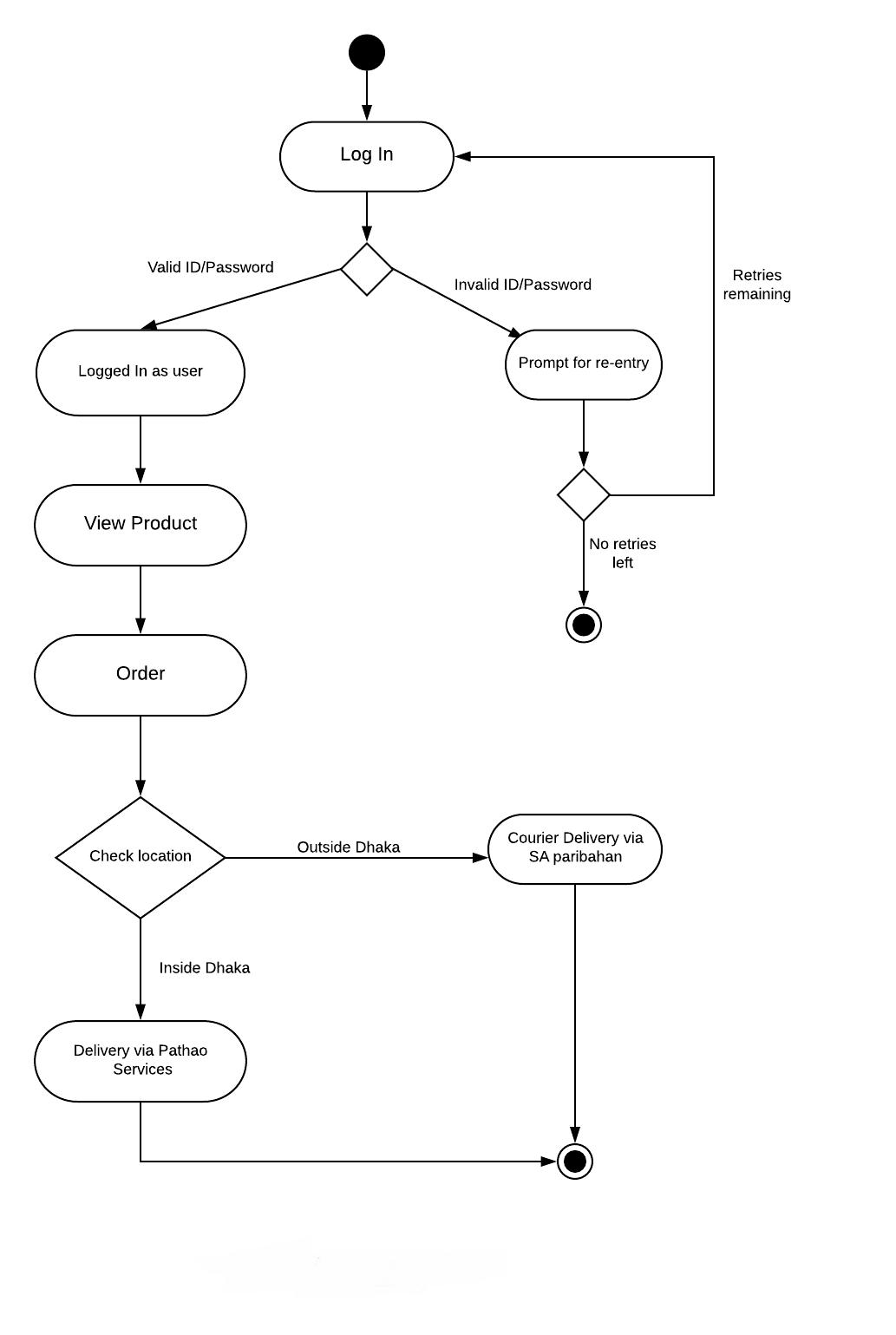


**Figure 29: Swimlane Diagram for Comment and rate**

60

**Use case 11:** Cash on delivery

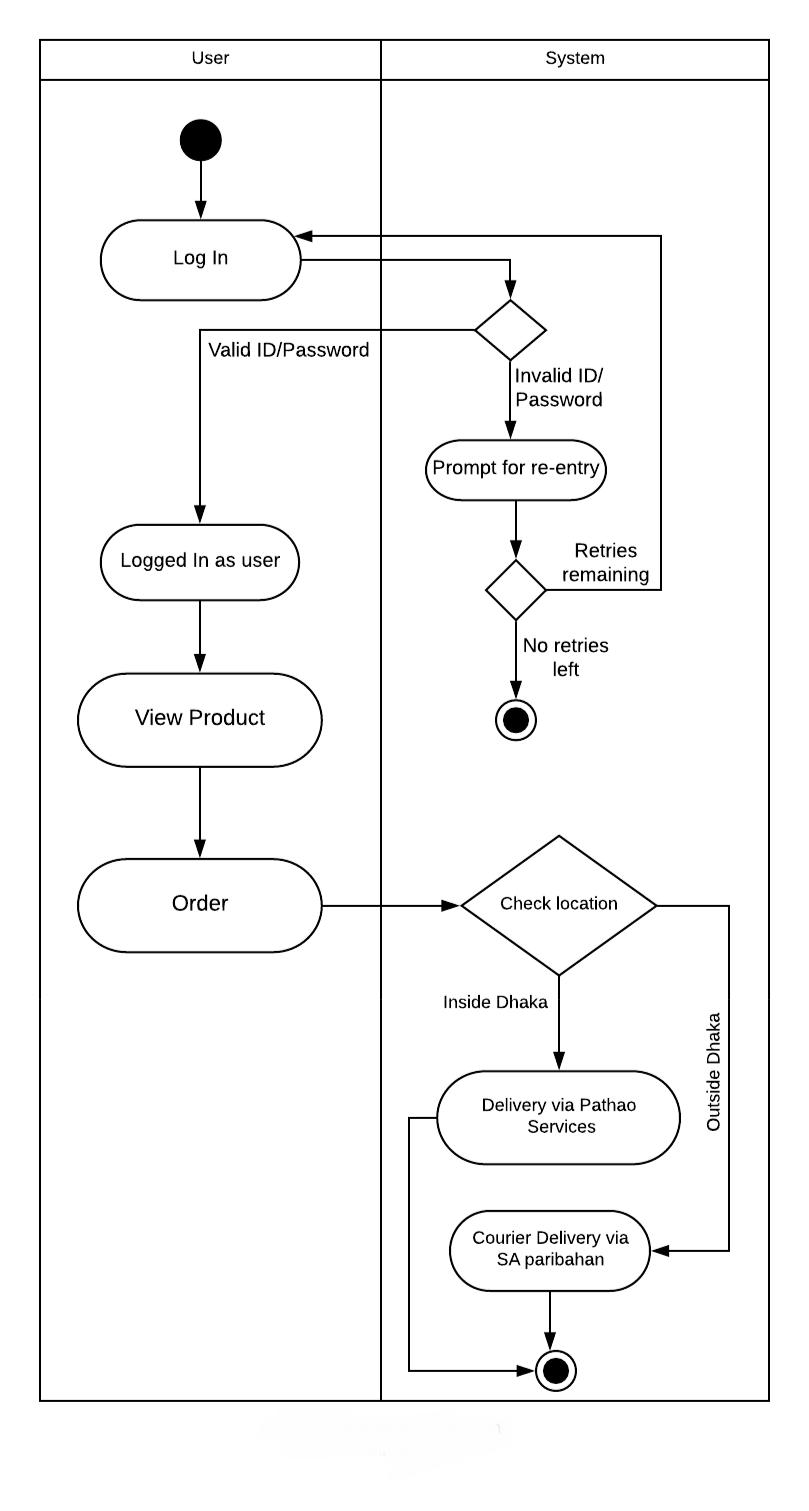
**Activity Diagram:**



**Figure 30: Activity Diagram for Cash on Delivery**

61

**Swimlane Diagram:**

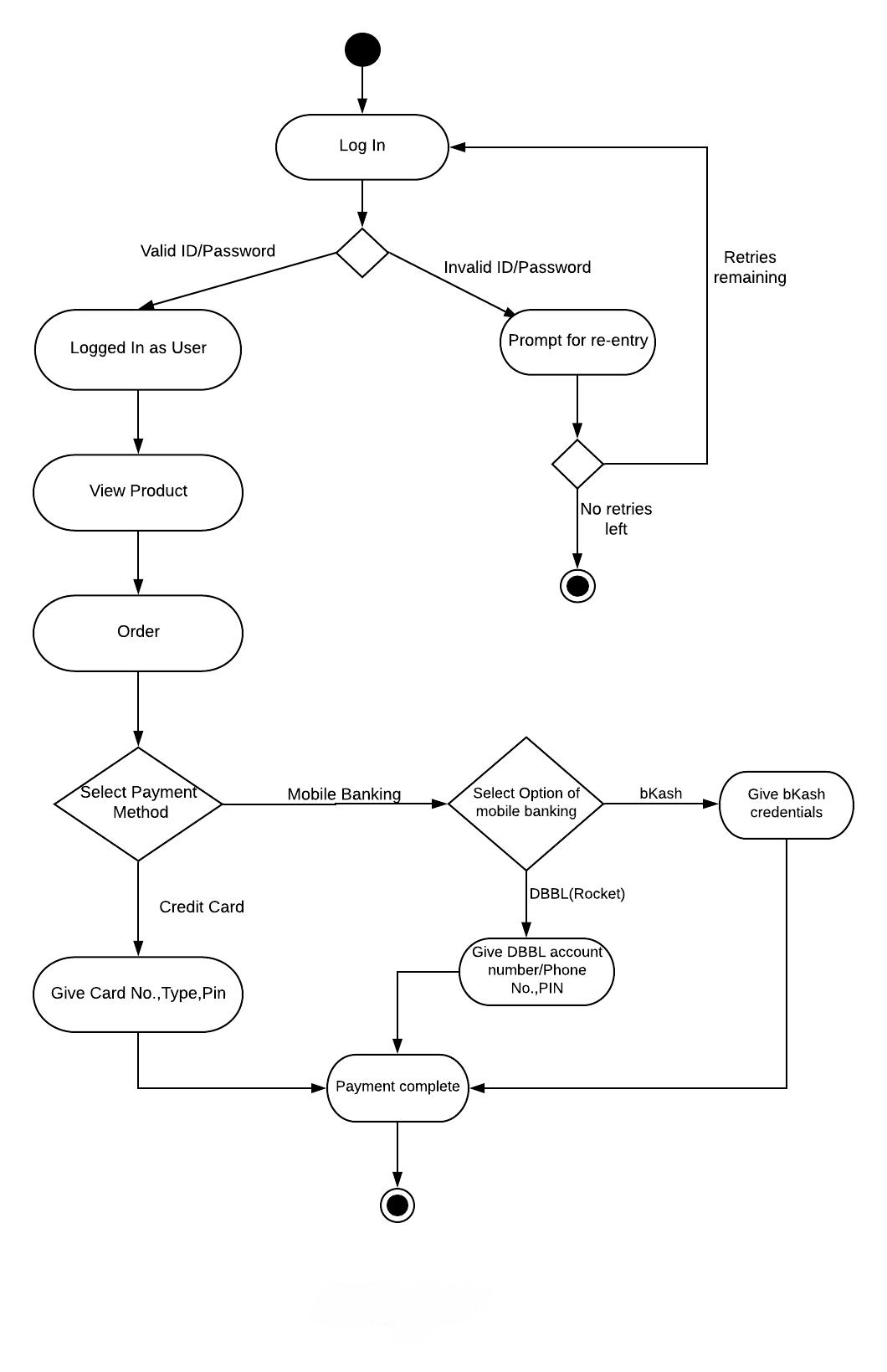


**Figure 31: Swimlane Diagram for Cash on Delivery**

62

**Use case 12:** Online payment

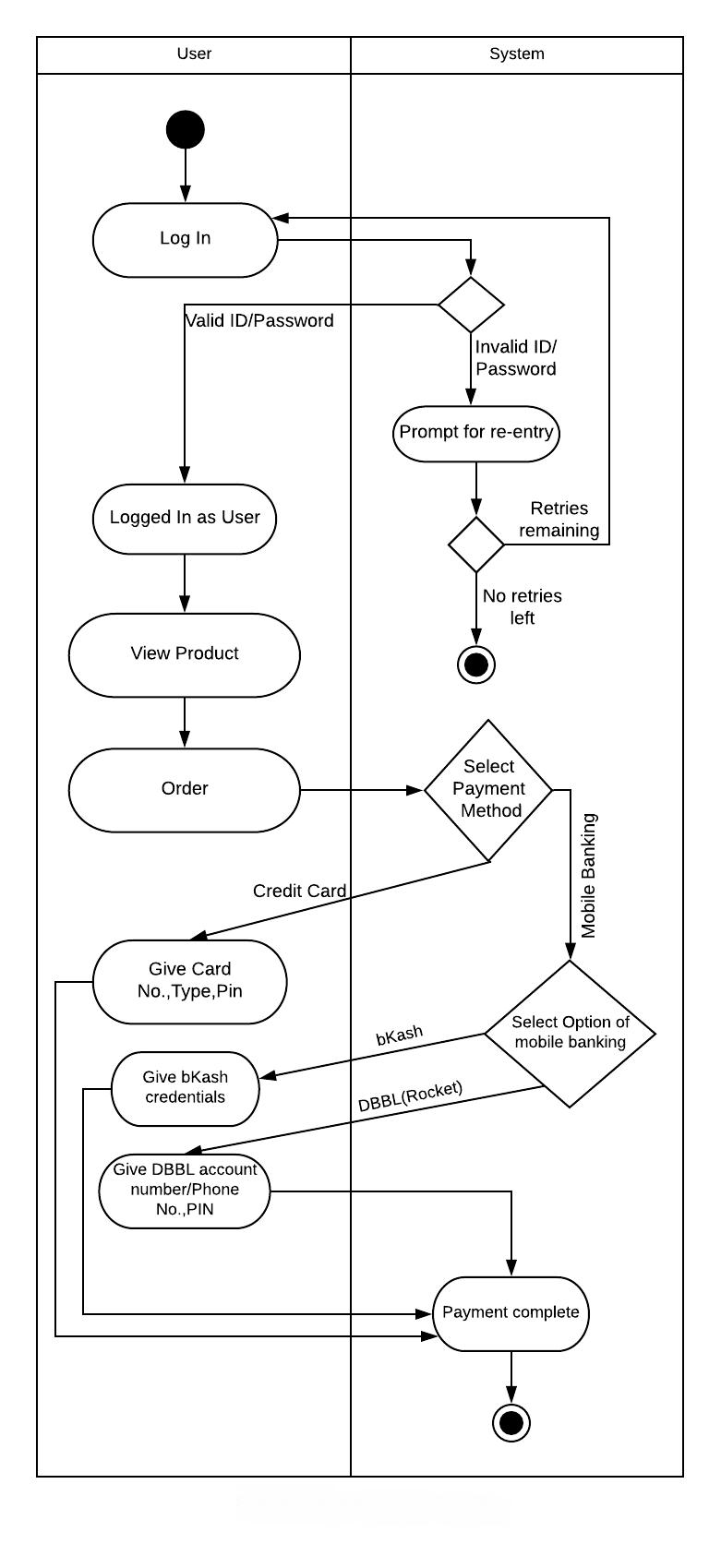
**Activity Diagram:**



**Figure 32: Activity Diagram for Online Payment**

63

**Swimlane Diagram:**

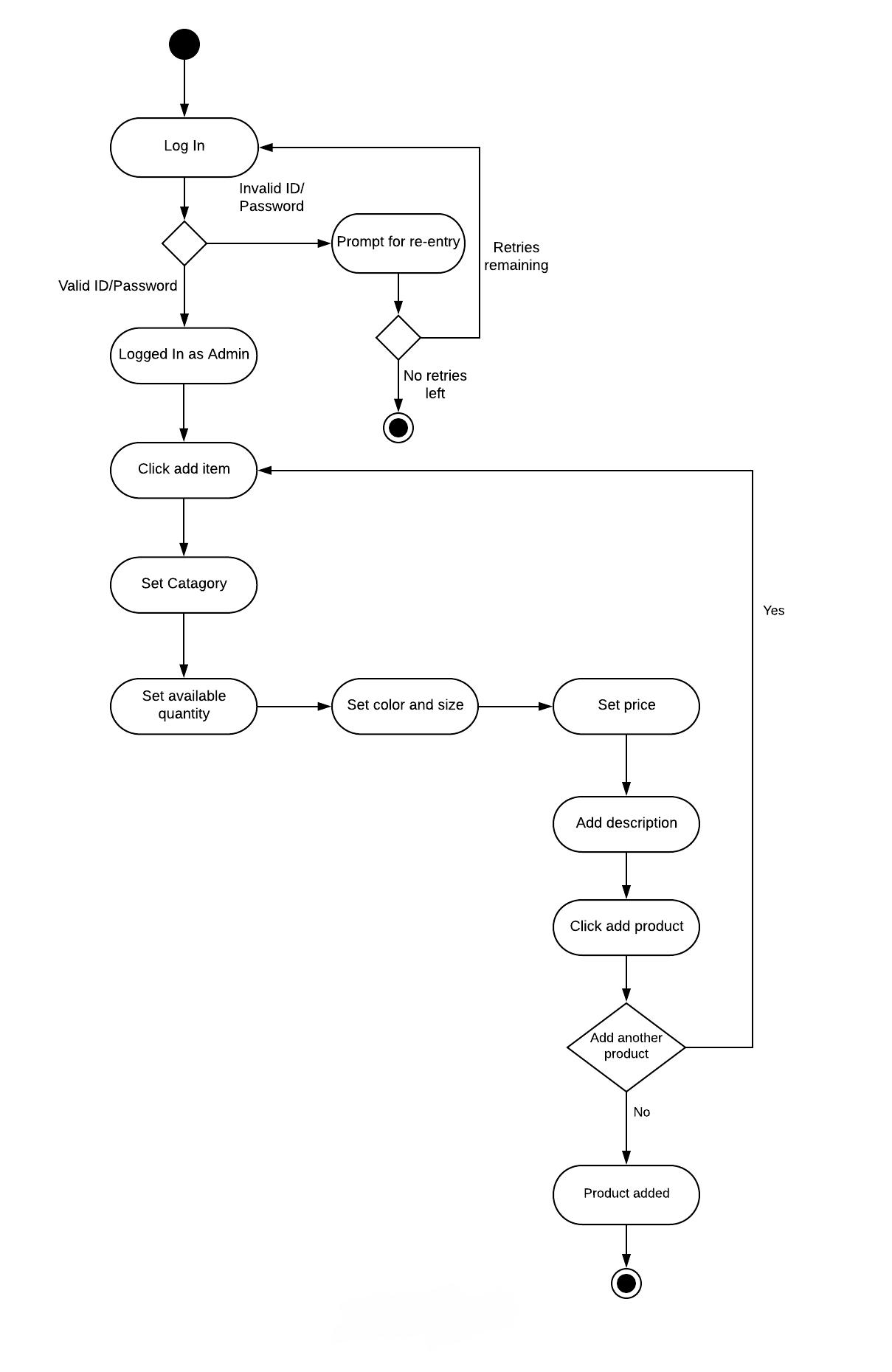


**Figure 33: Swimlane Diagram for Online Payment**

64

**Use case 13: Add Product**

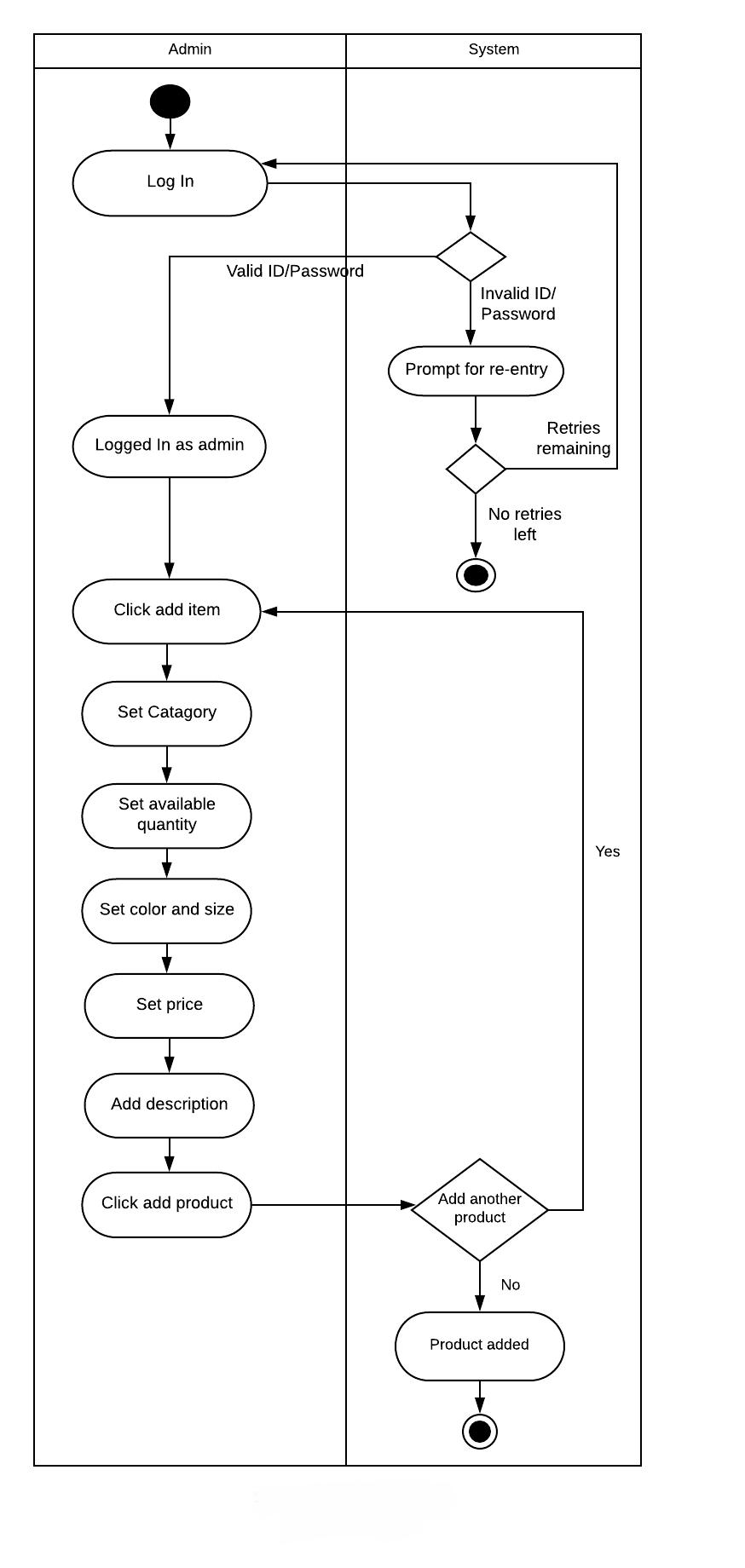
**Activity Diagram:**



**Figure 34: Activity Diagram for Add Product**

65

**Swimlane Diagram:**

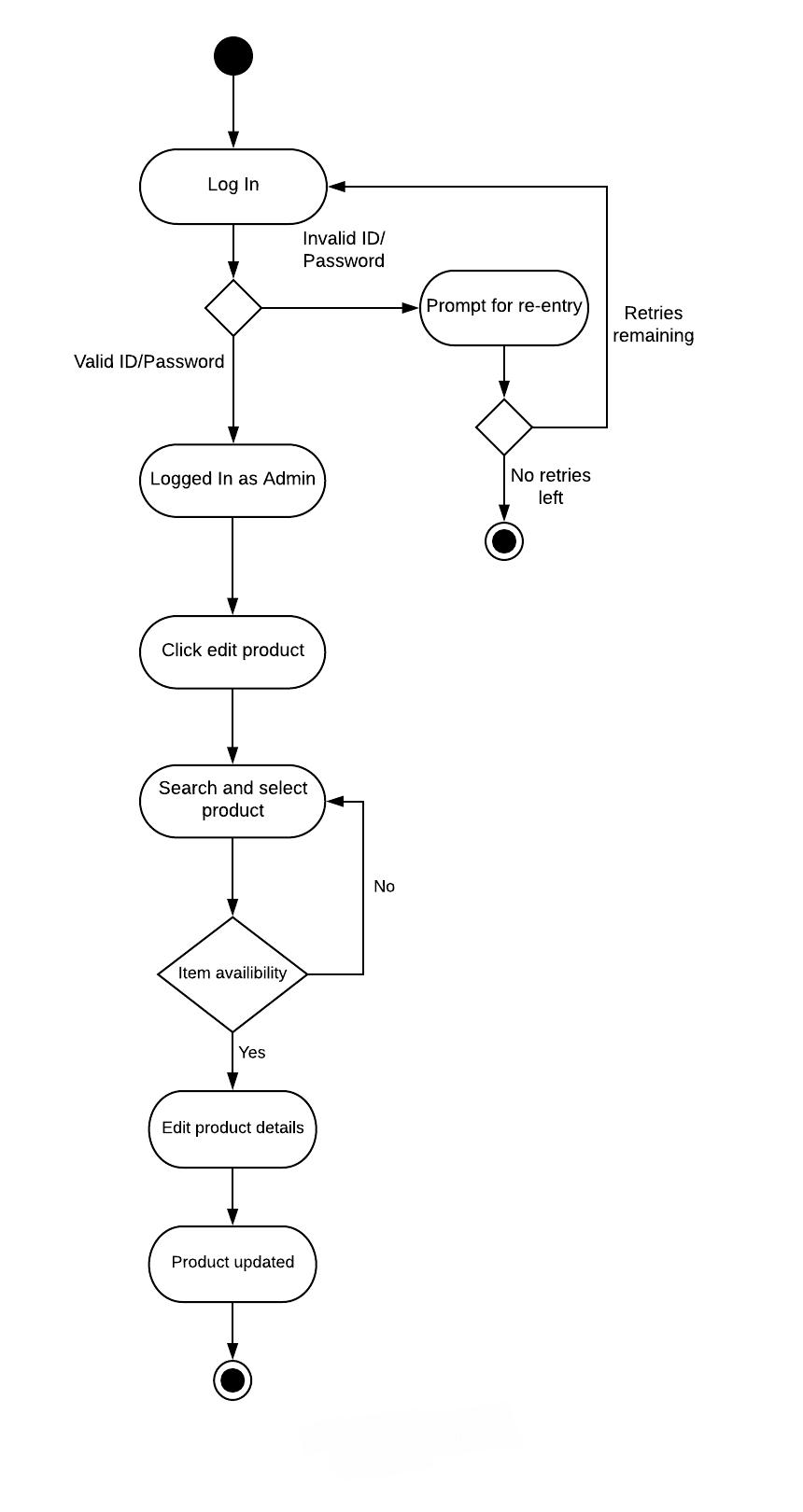


**Figure 35: Swimlane Diagram for Add Product**

66

**Use case 14: Edit Product**

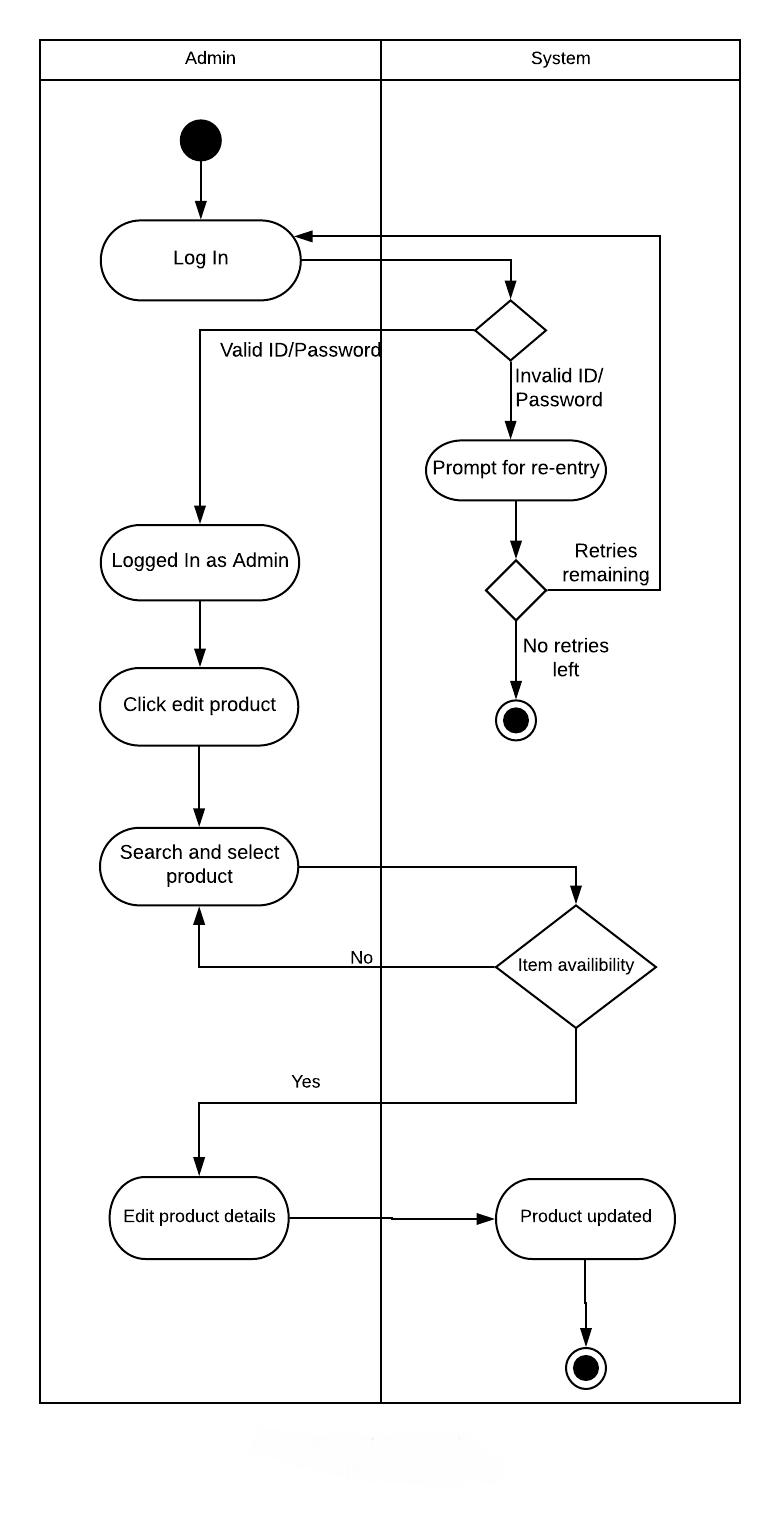
**Activity Diagram:**



**Figure 36: Activity Diagram for Edit Product**

67

**Swimlane Diagram:**



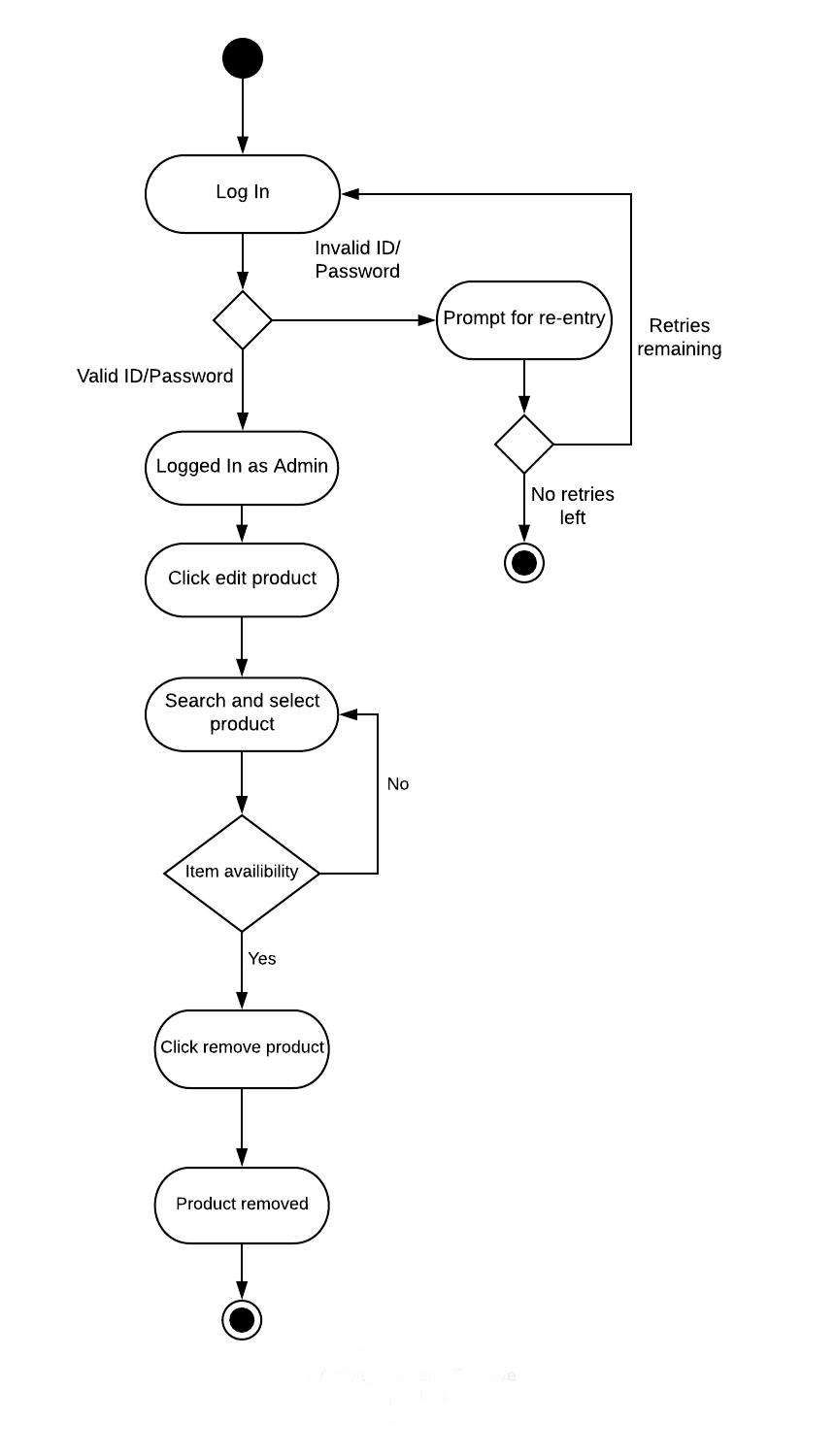
**Figure 37: Swimlane Diagram for Edit Product**

68

**Use case 15:**

**Remove Product**

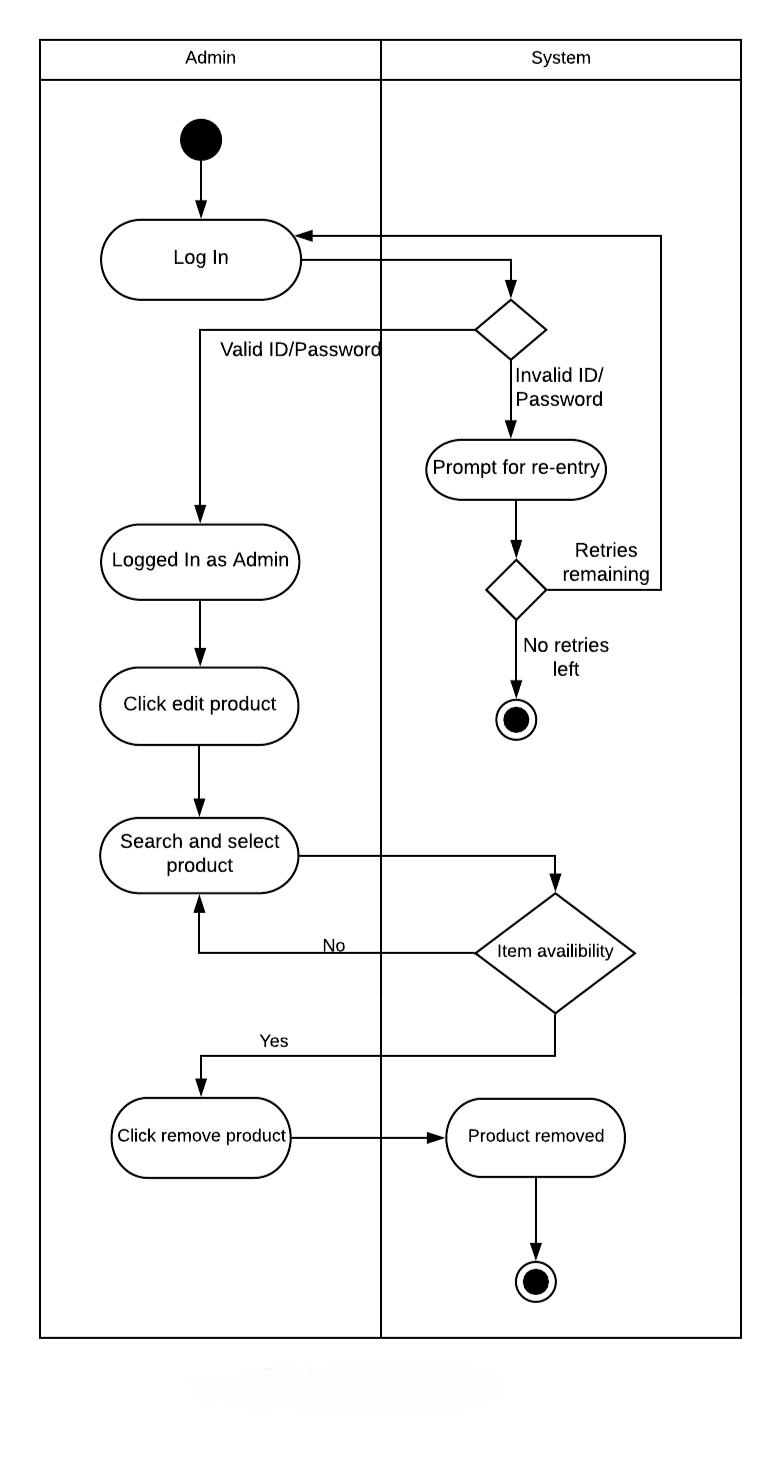
**Activity Diagram:**



**Figure 38: Activity Diagram for Remove Product**

69

**Swimlane Diagram:**

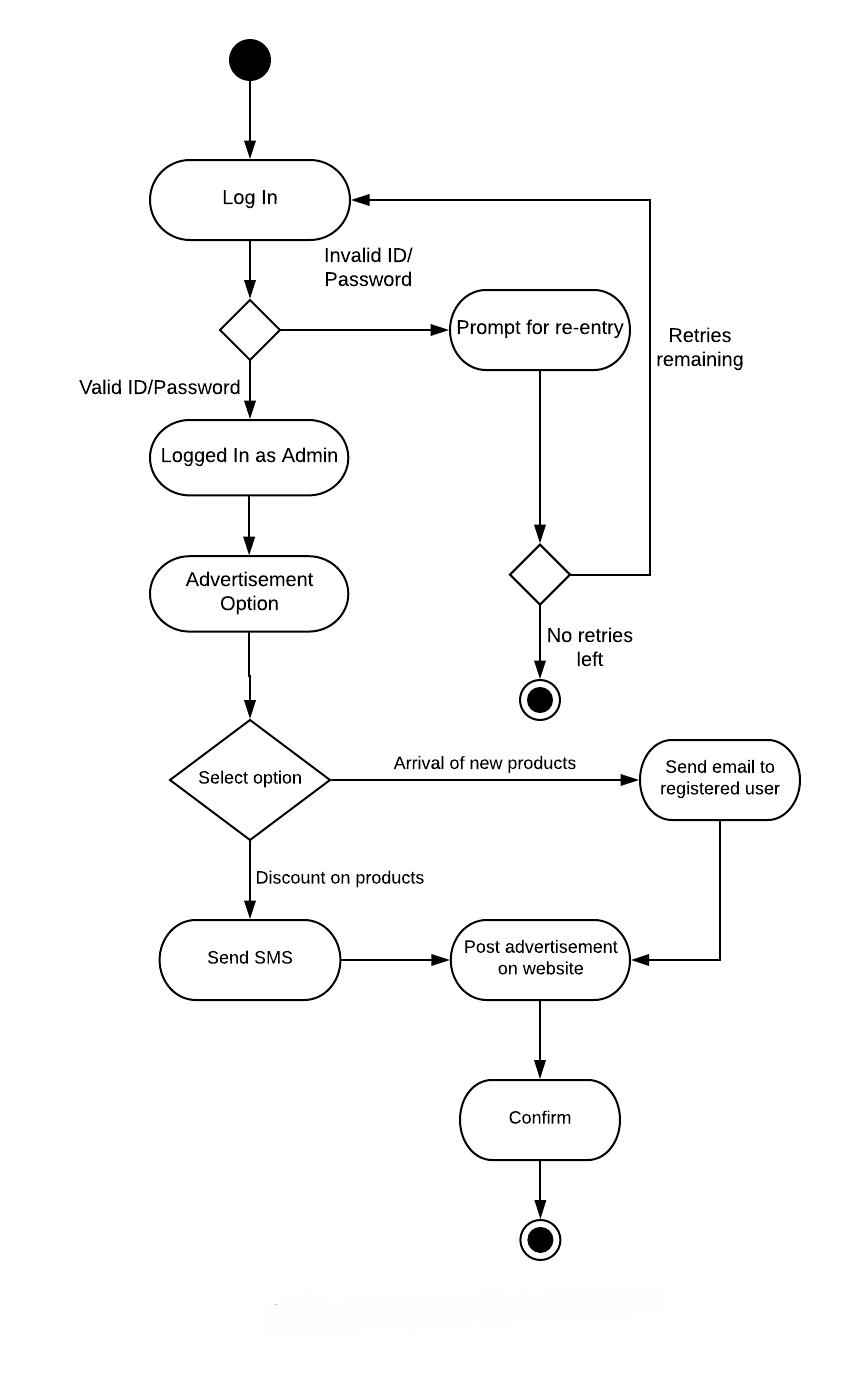


**Figure 39: Swimlane Diagram for Remove Product**

70

**Use case 16: Manage product advertisement**

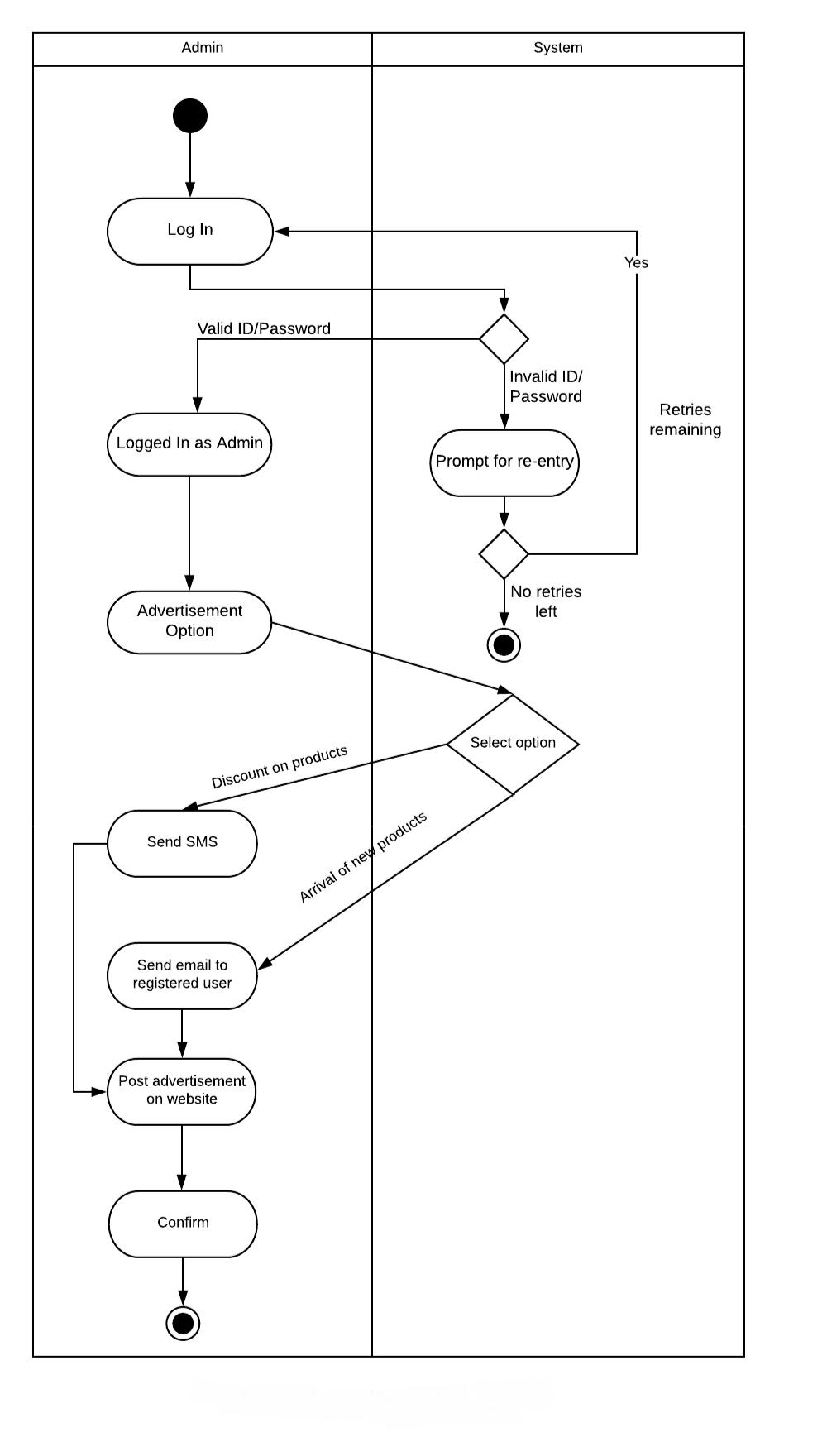
**Activity Diagram:**



**Figure 40: Activity Diagram for Manage Product Advertisement**

71

**Swimlane Diagram:**

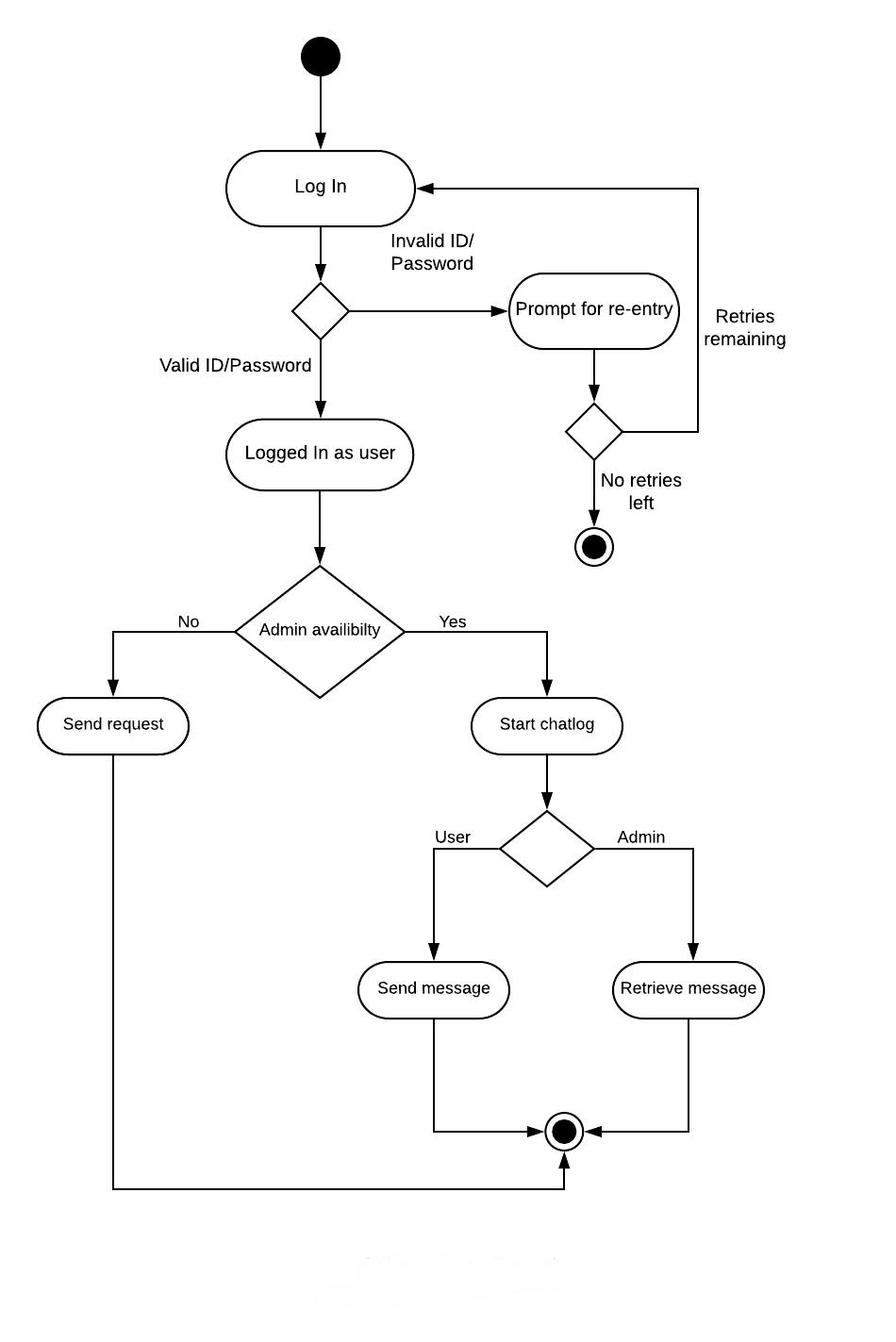


**Figure 41: Swimlane Diagram for Manage Product Advertisement**

72

**Use case 17: Chat**

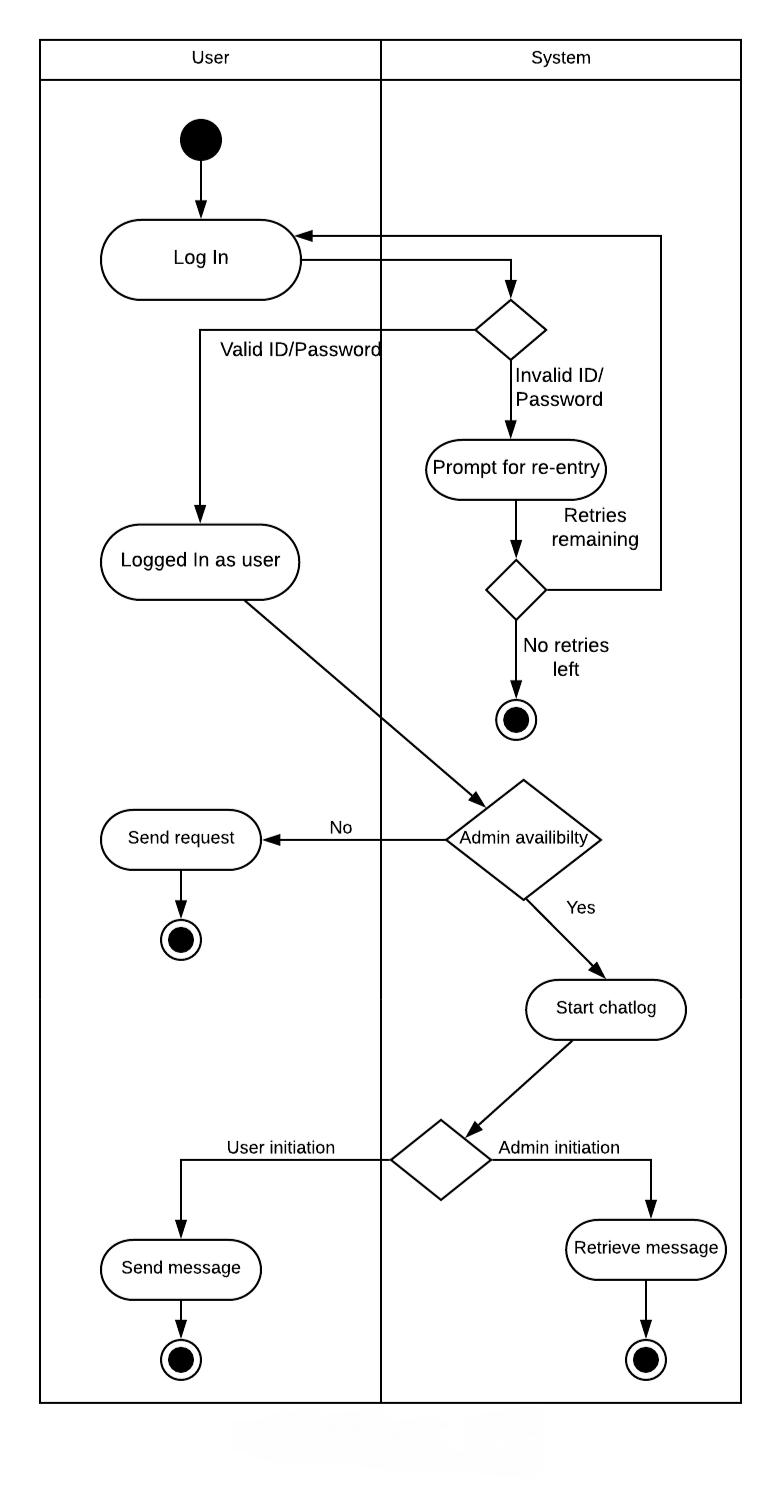
**Activity Diagram:**



**Figure 42: Activity Diagram for Chat**

73

**Swimlane Diagram:**

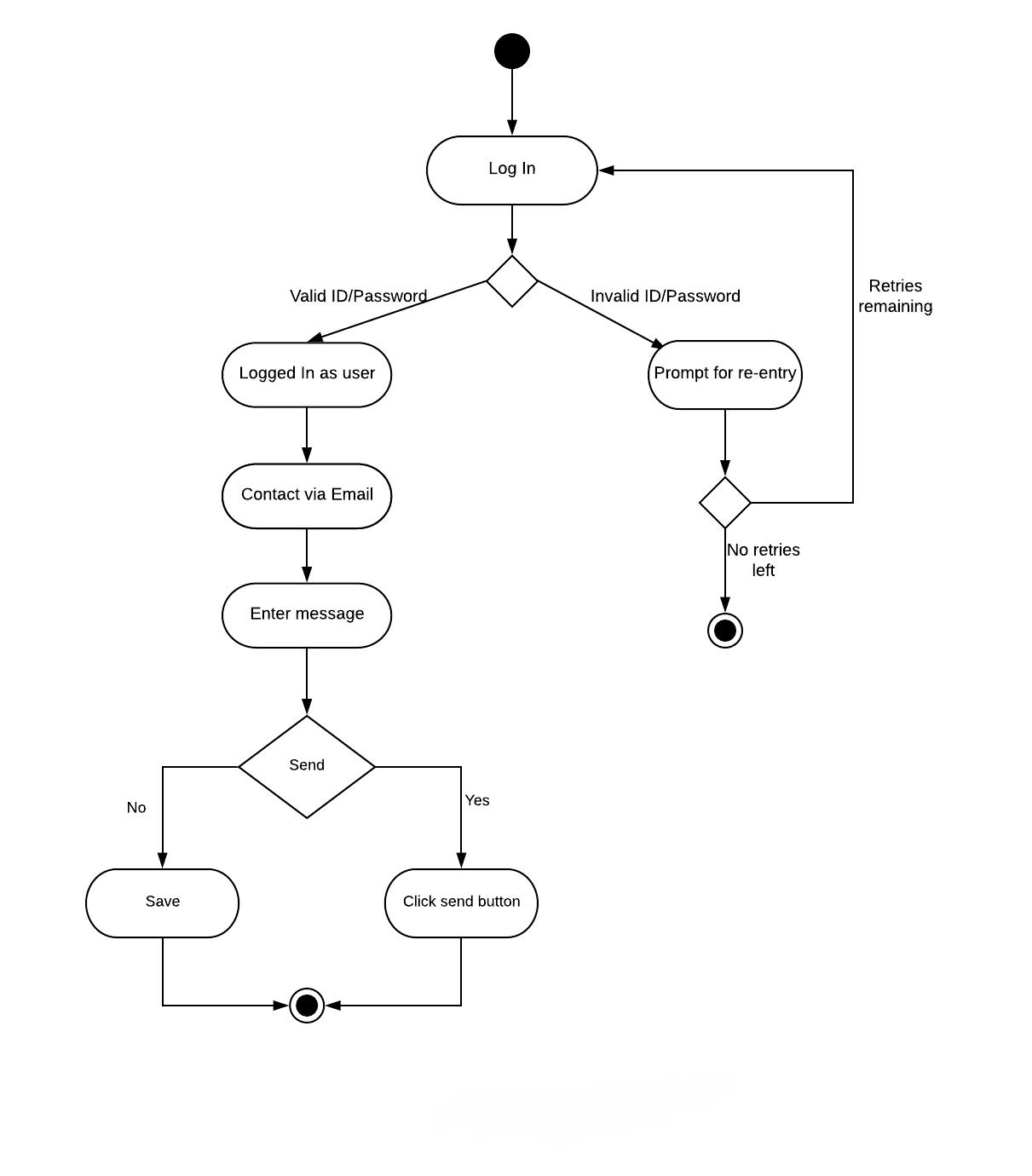


**Figure 43: Swimlane Diagram for Chat**

74

**Use case 18: Email**

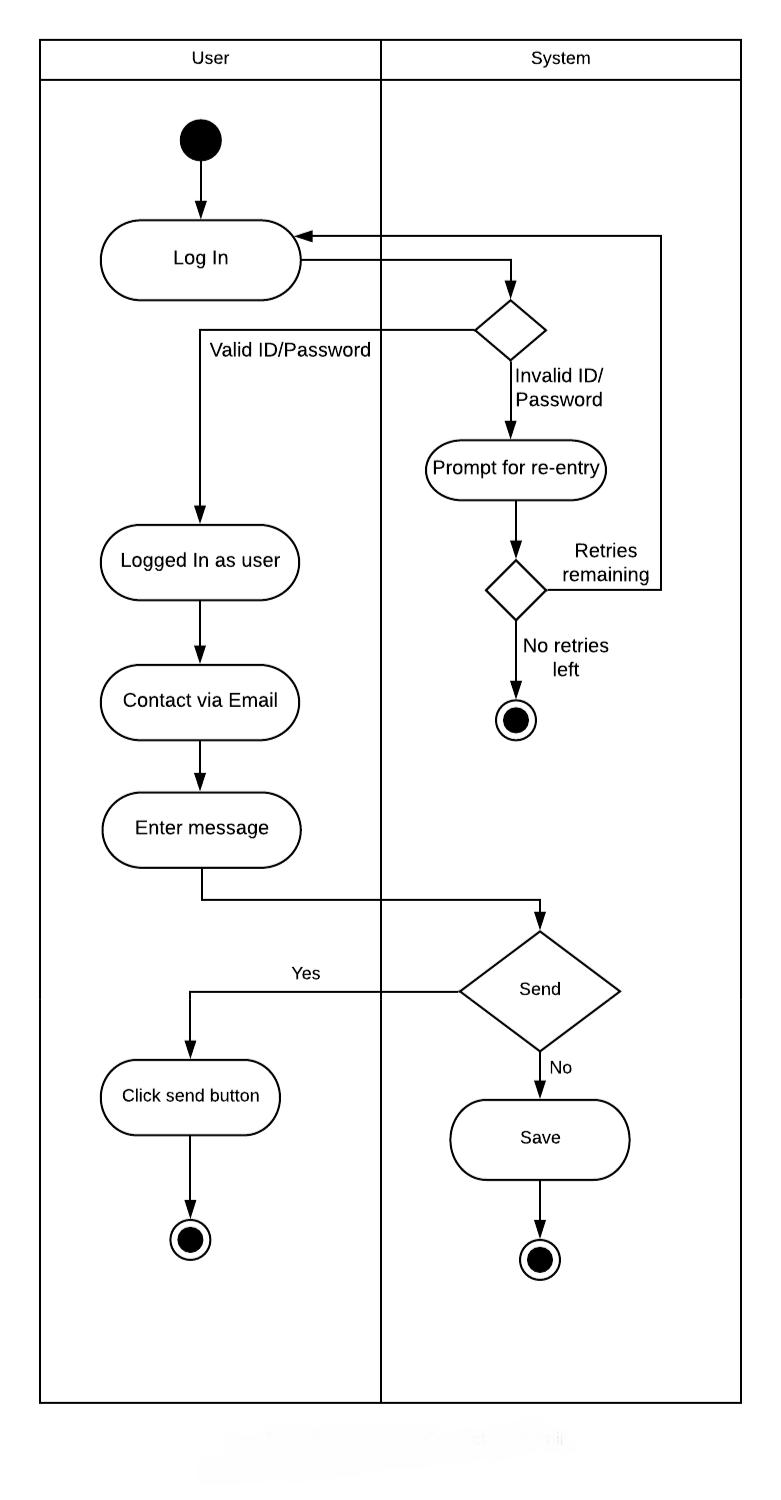
**Activity Diagram:**



**Figure 44: Activity Diagram for Email**

75

**Swimlane Diagram:**

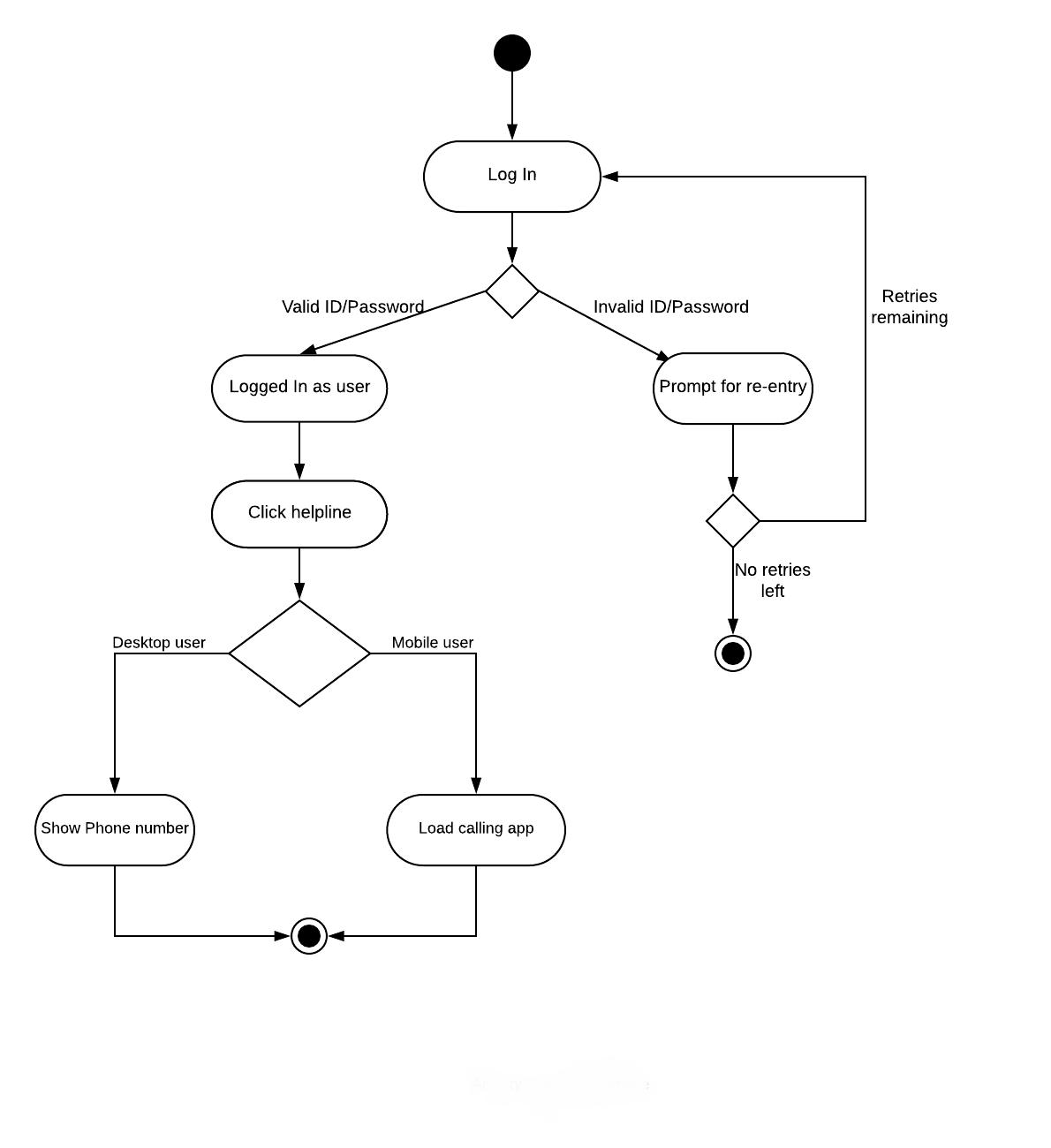


**Figure 45: Swimlane Diagram for Email**

76

**Use case 19: Helpline**

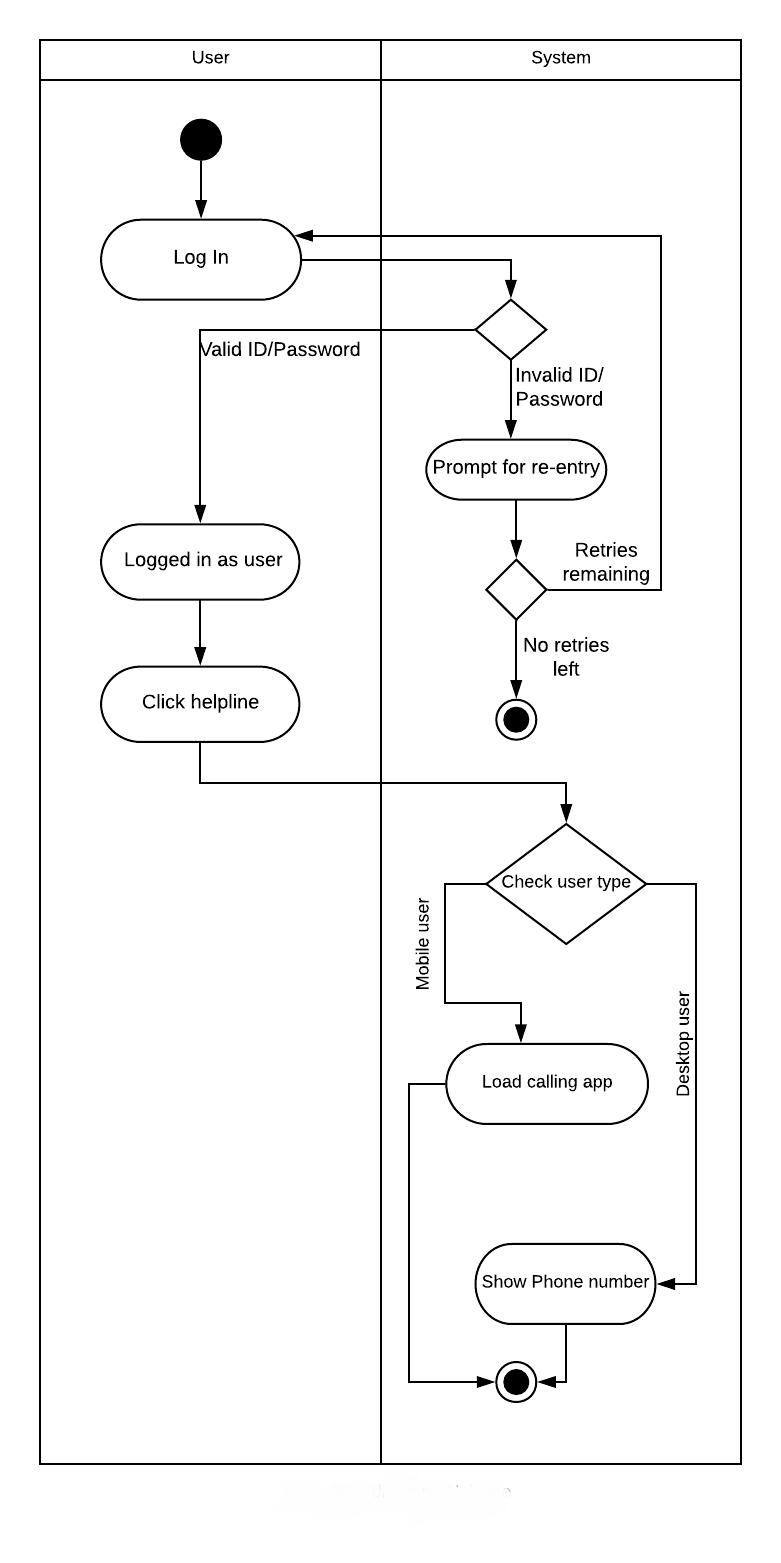
**Activity Diagram:**



**Figure 46: Activity Diagram for Helpline**

77

**Swimlane Diagram:**

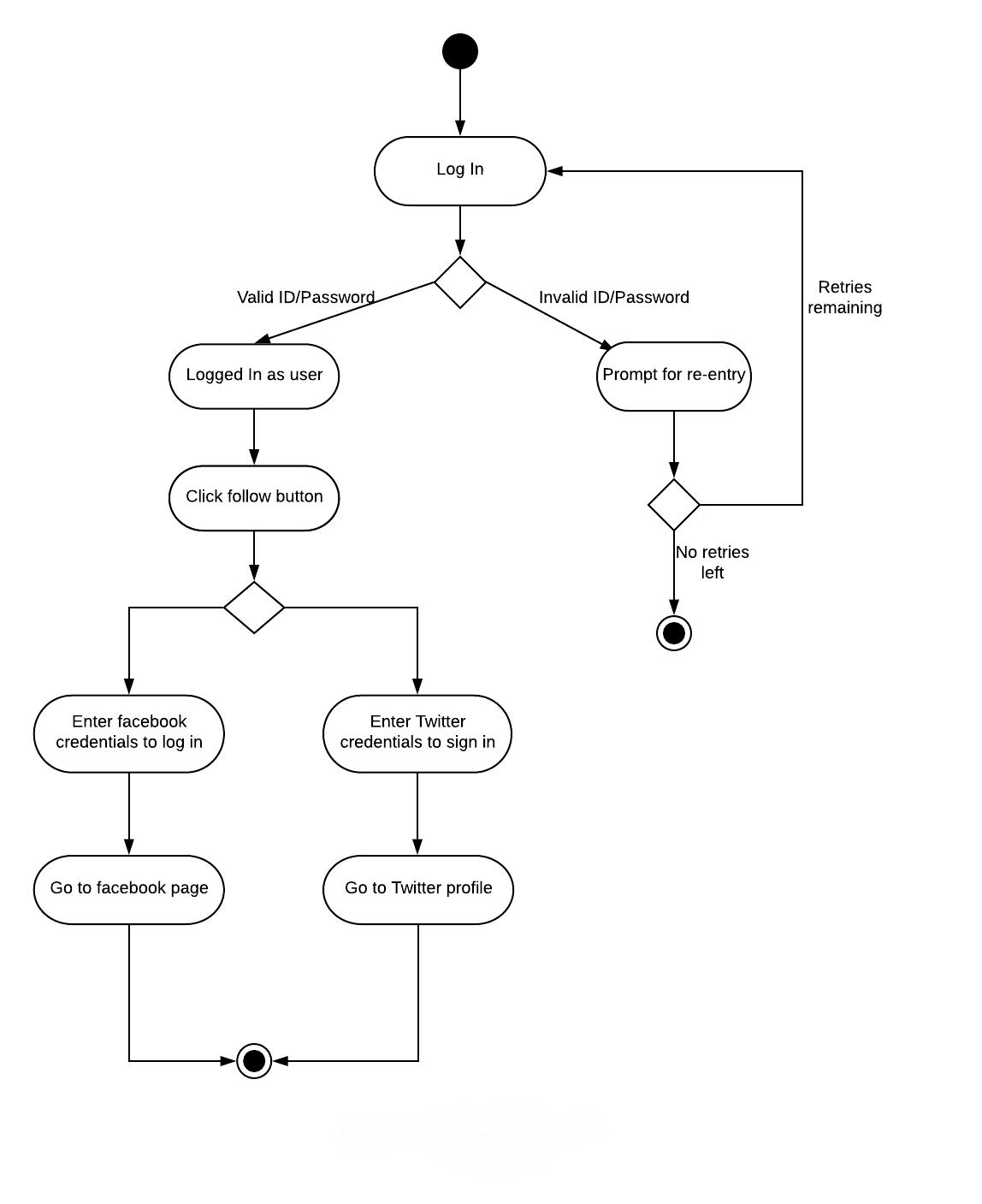


**Figure 47: Swimlane Diagram for Helpline**

78

**Use case 20: Follow on social media**

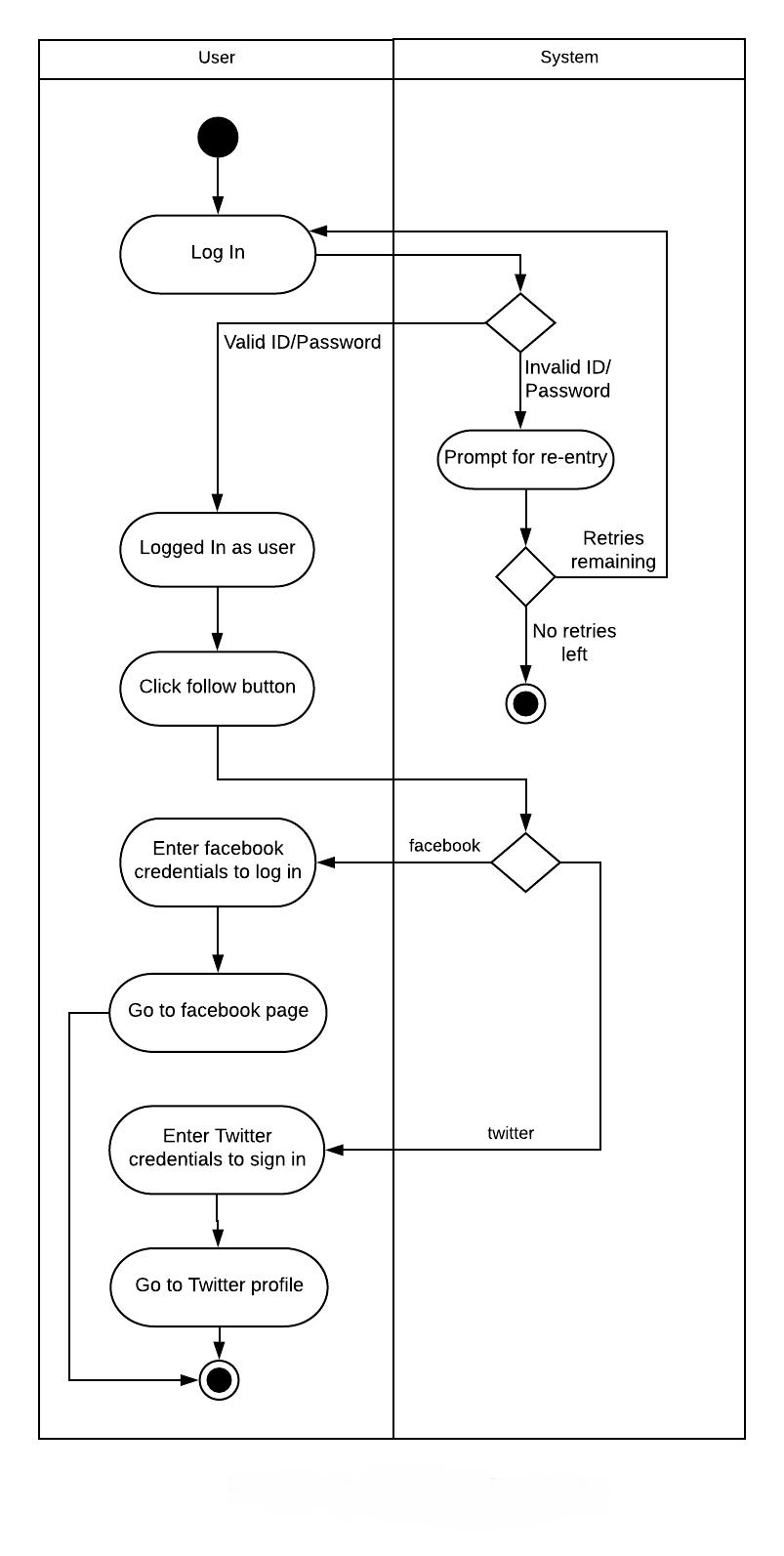
**Activity Diagram:**



**Figure 48: Activity Diagram for Follow on Social Media**

79

**Swimlane Diagram:**



**Figure 49: Swimlane Diagram for Follow on Social Media**

80

**Chapter 5: Data Model**

**5.1 Data modeling concept**

If software requirements include the necessity to create, extend or interact with a database or complex data structures need to be constructed and manipulated, then the software team chooses to create data models as part of overall requirements modeling. The entity-relationship diagram (ERD) defines all data objects that are processed within the system, the relationships between the data objects and the information about how the data objects are entered, stored, transformed and produced within the system.

**5.2 Data objects**

A data object is a representation of composite information that must be understood by the software. Here, composite information means an information that has a number of different properties or attributes. A data object can be an external entity, a thing, an occurrence, a role, an organizational unit, a place or a structure.

**Noun Identification**

* All the nouns in the scenario were identified.

Table 2: Noun Identification for data modelling

|  |  |  |  |
| --- | --- | --- | --- |
| **SL NO** | **NOUN** | **P/S** | **Attributes** |
|  |  |  |  |
| 1 | Account | S | 6 |
| 2 | User | S | 6,11 |
| 3 | Application | P | 71,72,74,75,76 |
| 4 | Google play store | S | - |
| 5 | Dhaka University | P | 13, 37-43 |
| 6 | Google account | P | 17,18 |
| 7 | Profile | S | 8,9,10,11,105 |
| 8 | User Name | S | - |
| 9 | Academic Year | S | - |
| 10 | Email | S | 6 |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 11 | Type | S |  | - |
| 12 | delete | S |  | 14 |
| 13 | Department | S |  | 87,96,100,101,102,104,105 |
| 14 | specified | S |  | - |
| 15 | Sign up | S |  | - |
| 16 | Sign in | S |  | - |
| 17 | Authentication | S |  | 14 , 18 |
| 18 | Firebase Auth 2 | P |  | - |
| 19 | Information | P |  | - |
| 20 | Log out | P |  | - |
| 21 | dropdown | P |  | - |
| 22 | Menu | S |  | - |
| 23 | Admin | P |  | 8,10,35,115 |
| 24 | features | P |  | - |
| 25 | year | P |  | 26 |
| 26 | Keyword | S |  | - |
| 27 | exciting | P |  | - |
| 28 | calendar | S |  | 60,61 |
| 29 | group chat | P |  | 30,54,64 |
| 30 | messenger | P |  | 83 |
| 31 | activities | P |  | - |
| 32 | update | P |  | - |
| 33 | Confirmation | S |  | - |
| 34 | separate | S |  | - |
| 35 | website | P |  | - |
| 36 | wish | S |  | - |
| 37 | DU brief history | P |  | - |
| 38 | achievement | S |  | - |
| 39 | Governing Body | S |  | 88,89,90 |
| 40 | Halls | S |  | 87,100,101,102 |
| 41 | Club | S |  | 87,100,101,102 |
| 42 | society | S |  | - |
| 43 | places | P |  | - |
| 44 | view | S |  | - |
| 45 | search | S |  | 26,80 |
| 46 | Transport | S |  | 47,48,51,52 |
| 47 | bus | P |  | 113,48,51,52,114 |
| 48 | stoppage | P |  |  |
| 49 | Map | P |  | 45,51,52,103 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 50 | current position | S |  | - |
| 51 | destination | P |  | - |
| 52 | route | P |  | - |
| 53 | offices | P |  | - |
| 54 | Communication | P |  | 29,30,64 |
| 55 | first | P |  | - |
| 56 | person | P |  | - |
| 57 | Database | S |  | - |
| 58 | private chat | S |  | - |
| 59 | option | S |  | - |
| 60 | holiday | P |  | - |
| 61 | events | P |  | - |
| 62 | Notification | S |  | - |
| 63 | Contact1 | S |  |  |
| 64 | Chat | S |  | - |
| 65 | academic | S |  | - |
| 66 | web interface | S |  | - |
| 67 | server | S |  | - |
| 68 | data transparency | S |  | - |
| 69 | service | S |  | - |
| 70 | security | S |  | - |
| 71 | Homepage | S |  | - |
| 72 | icons | S |  | - |
| 73 | top | S |  | - |
| 74 | settings | S |  | - |
| 75 | language | S |  | - |
| 76 | layout | S |  | - |
| 77 | tree based | S |  | - |
| 78 | design | S |  | 77,76 |
| 79 | structure | S |  | - |
| 80 | database query | S |  | - |
| 81 | two | S |  | - |
| 82 | restriction | P |  | - |
| 83 | Message | P |  | - |
| 84 | link | P |  | - |
| 85 | right | P |  | - |
| 86 | Status | S |  | - |
| 87 | Student | S |  | 8,9,10,11,105 |
| 88 | picture | P |  | - |
| 89 | name | P |  | - |
| 90 | rank | P |  | - |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 91 | Administrative committee | S |  | 11,89,90 |
| 92 | Editorial committee | S |  | 11,89,90 |
| 93 | Faculty Deans | S |  | 11,94,95 |
| 94 | faculty name | P |  | - |
| 95 | Dean | P |  | - |
| 96 | Department ID | P |  | 29,30,64 |
| 97 | Institute ID | P |  | - |
| 98 | Hall ID | P |  | - |
| 99 | Office ID | S |  | - |
| 100 | establishment year | S |  | - |
| 101 | Description | S |  | - |
| 102 | Location | P |  | - |
| 103 | Area | P |  | - |
| 104 | Teacher | S |  | 8,10,11,105 |
| 105 | Department Name | P |  |  |
| 106 | Institute | S |  | 87,97,100,101,102,104,107 |
| 107 | Institute Name | P |  |  |
| 108 | Specific Calendar | S |  | 60,61,109,110 |
| 109 | Date | P |  | - |
| 110 | Academic Schedule | P |  | - |
| 111 | Other user | S |  | 8,10,11 |
| 112 | Infrastructure | S |  | 100,101,102 |
| 113 | Bus ID | P |  | - |
| 114 | Bus name | P |  | - |
| 115 | password | P |  | - |
|  |  |  |  |  |

**Potential Data Objects:**

* Account : 6
* User : 6,7,11
* Application : 71,72,74,75,76
* Dhaka University : 13, 37-43
* Google account : 17,18
* Profile : 8,9,10,11,105
* Email : 6
* Department : 87,96,100,101,102,104,105
* Authentication : 18
* Admin : 8,10,32,35,115
* General Calendar : 60,61,109
* group chat : 30,54,64
* Governing Body : 88,89,90
* Hall : 87,97,100,101,102
* Club : 87,98,100,101,102
* Search : 26,80
* Transport : 47,48,51,52
* Map : 51,52,103
* Office : 99,100,101,102
* Communication : 29,30,64
* Design : 77,76
* Student : 8,9,10,11,12,13
* Administrative committee : 11,89,90
* Editorial committee : 11,89,90
* Faculty Dean : 11,94,95
* Teacher : 8,9,10,11,105
* Department : 87,96,100,101,102,104,107
* Institute : 87,97,100,101,102,104,107
* Specific Calendar : 60,61,109,110
* Other User : 8,10,11
* Infrastructure : 100,101,102
* Bus : 48,51,52,113

**Analysis for finalizing Data objects**

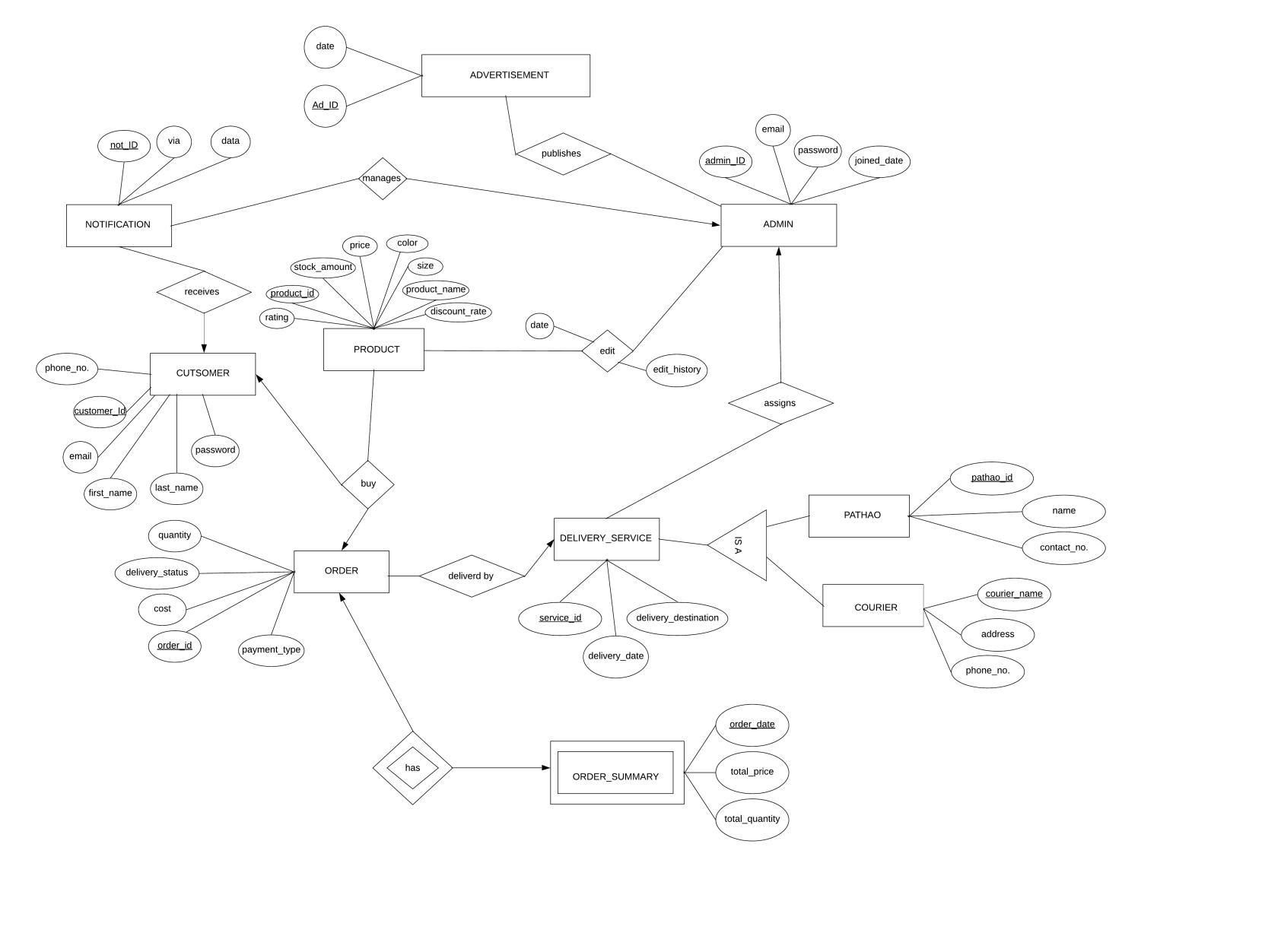
* + Account , user , application , profile , email are all part of other user object . So , they are not considered.
  + Authentication , Search , communication are attributes of other data object. So they are not considerable .
  + Department , Office , Hall , Club are all part of Infrastructure . So infrastructure is the parent class.
  + Governing body , Administrative committee , Editorial committee , Faculty Deans are all part of Governing committee list .
  + All other data objects can be used as data objects as they have enough importance in the system.

**Final Data objects**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Table 3: Final Data Objects** | |  |
|  |  |  |  |  |
| 1 | User : User ID, Email , Type. | | |  |
|  |  |  |  |  |
| 2 | Student: User Name, Academic Year, Email, Department Name, Type | | |  |
|  |  |  |  |  |
| 3 | Teacher: User Name, Email, Department Name , Type | | |  |
|  |  |  |  |  |
| 4 | Other User : User Name, Email, Type | | |  |
|  |  |  |  |  |
| 5 | Infrastructure : establishment year, Description, Location | | |  |
|  |  |  |  |  |
| 6. | Department : Department ID, establishment year, Description, Location ,  Student id , Teacher id , Department name | | |  |
|  |  |  |  |
|  |  |  |  |  |
| 7. | Institute : Institute ID, establishment year, Description, Location ,  Student id , Teacher id , Institute name | | |  |
|  |  | | |  |
| 8. | Office : Office ID , establishment year, Description, Location | | |  |
| 9. | Hall : Hall ID , Student Id ,establishment year, Description, Location | | |  |
|  |  | | |  |
| 10. | Club : Club ID , Student Id , establishment year, Description, Location | | |  |
|  |  | | |  |
| 11. | Governing Body : Type, Name | | |  |
| 12 . | Administrative committee : Type, Name , Rank | | |  |

|  |  |
| --- | --- |
| 13 . | Map : location ID , destination , Area |
| 14 . | Transport : Transport ID , Type |
| 15 . | Bus : Bus ID , Bus Name , destination , Stoppage |
| 16. | General Calendar : Holiday, Events , Date |
| 17 . | Specific Calendar : Holiday, Events , Date , Academic Schedule |
| 18 . | Group Chat : Messenger , Communication ,Chat |
| 19. | Admin : Admin ID, User name , Email , password , website |
| 20 | Editorial committee : Type, Name , Rank |
| 21 | Faculty Dean List : Type, Name , Faculty name |

**Entity Relationship Diagram**



**Figure 50: ER diagram of F-commerce based client Management System**

86

**Relational Schema**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for User** | |
|  |  |  |  |  |  |
|  | **User** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | User\_id | | Number |  | 30 |
|  |  |  |  |  |  |
|  | Email | | Varchar2 |  | 40 |
|  |  | |  |  |  |
|  | Type | | Varchar2 |  | 20 |
|  |  | |  |  |  |

**Table 5: Schema for Student**

**Student**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Attribute** |  | **Type** | **Size** | |
|  |  |  |  |  |  |
|  | Student\_id |  | Number | 30 |  |
|  |  |  |  |  |  |
|  | Academic Year |  | Varchar2 | 20 |  |
|  | Email |  | Varchar2 | 40 |  |
|  |  |  |  |  |  |
|  | User Name |  | Varchar2 | 20 |  |
|  |  |  |  |  |  |
|  | Department Name |  | Varchar2 | 20 |  |
|  |  |  |  |  |  |
|  | Type |  | Varchar2 | 10 |  |

87

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | **Table 6: Schema for Teacher** | | | |
|  |  |  |  |  |  |  |  |  |  |  |
|  | **Teacher** | | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | **Attribute** | | | |  | **Type** |  |  | **Size** | |
|  |  |  | |  |  |  |  |  |  |  |
|  | Teacher\_id | | | |  | Number |  |  | 30 |  |
|  |  |  |  | |  |  |  |  |  |  |
|  | Email |  | | |  | Varchar2 |  |  | 40 |  |
|  | Department Name | | | |  | Varchar2 |  |  | 30 |  |
|  |  |  | | |  |  |  |  |  |  |
|  | Type | | | |  | Varchar2 |  |  | 20 |  |
|  |  |  | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  | **Table 7: Schema for Other User** | | | |
|  |  |  | | |  |  |  |  |  |  |
|  | **Other User** | | | |  |  |  |  |  |  |
|  |  |  | | |  | |  |  | |  |
|  | **Attribute** | | | | **Type** | |  | **Size** | |  |
|  |  | | | |  | |  |  | |  |
|  | User\_id |  | | | Number | |  | 30 | |  |
|  |  | | | |  | |  |  | |  |
|  | Email | | | | Varchaer2 | |  | 30 | |  |
|  |  | | | |  | |  |  | |  |
|  | Type | | | | Varchar2 | |  | 20 | |  |
|  |  | | | |  | |  |  | |  |
|  |  |  |  |  |  |  | 88 |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Infrastructure** | |
|  |  |  |  |  |  |
|  | **Infrastructure** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Infrastructure ID | | Number |  | 30 |
|  |  |  |  |  |  |
|  | Established Year |  | Number |  | 30 |
|  | Description | | Varchar2 |  | 3000 |
|  |  | |  |  |  |
|  | Location | | Number , | Number | 32,32 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Department** | |
|  |  |  |  |  |  |
|  | **Department** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Department ID | | Number |  | 30 |
|  | Student ID |  | Number |  | 30 |
|  | Teacher ID |  | Number |  | 30 |
|  | Established Year |  | Number |  | 30 |
|  | Description | | Varchar2 |  | 3000 |
|  |  | |  |  |  |
|  | Department Name | | Varchar2 |  | 30 |
|  | Location | | Number , | Number | 32,32 |
|  |  | |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  | **Table 4: Schema for Department** | |
|  |  |  | |  |  |  |
|  | **Department** | | |  |  |  |
|  |  |  | |  |  |  |
|  | **Attribute** | | | **Type** |  | **Size** |
|  |  | | |  |  |  |
|  | Department ID | | | Number |  | 30 |
|  |  |  | |  |  |  |
|  | Student ID |  | | Number |  | 30 |
|  | Teacher ID |  | | Number |  | 30 |
|  | Established Year | |  | Number |  | 30 |
|  | Description | | | Varchar2 |  | 3000 |
|  |  | | |  |  |  |
|  | Department Name | | | Varchar2 |  | 30 |
|  | Location | | | Number , | Number | 32,32 |
|  |  | | |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  | **Table 4: Schema for Hall** | |
|  |  |  | |  |  |  |
|  | **Hall** | | |  |  |  |
|  |  |  | |  |  |  |
|  | **Attribute** | | | **Type** |  | **Size** |
|  |  | | |  |  |  |
|  | Hall ID | | | Number |  | 30 |
|  |  |  | |  |  |  |
|  | Student ID |  | | Number |  | 30 |
|  | Established Year | |  | Number |  | 30 |
|  | Description | | | Varchar2 |  | 3000 |
|  |  | | |  |  |  |
|  | Location | | | Number , | Number | 32,32 |
|  |  | | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Club** | |
|  |  |  |  |  |  |
|  | **Club** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Club ID | | Number |  | 30 |
|  |  |  |  |  |  |
|  | Student ID |  | Number |  | 30 |
|  | Established Year |  | Number |  | 30 |
|  | Description | | Varchar2 |  | 3000 |
|  |  | |  |  |  |
|  | Location | | Number , | Number | 32,32 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Office** | |
|  |  |  |  |  |  |
|  | **Office** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Office ID | | Number |  | 30 |
|  | Established Year |  | Number |  | 30 |
|  | Description | | Varchar2 |  | 3000 |
|  |  | |  |  |  |
|  | Location | | Number , | Number | 32,32 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Person View** | |
|  |  |  |  |  |  |
|  | **Person View** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | User ID | | Number |  | 30 |
|  | User Name |  | Varchar2 |  | 30 |
|  | Type | | Varchar2 |  | 30 |
|  |  | |  |  |  |
|  |  |  |  | **Table 4: Schema for Administrative Committee** | |
|  |  |  |  |  |  |
|  | **Administrative Committee** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | User ID | | Number |  | 30 |
|  | User Name |  | Varchar2 |  | 30 |
|  | Rank |  | Varchar2 |  | 30 |
|  | Type | | Varchar2 |  | 30 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Editorial Committee** | |
|  |  |  |  |  |  |
|  | **Editorial Committee** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | User ID | | Number |  | 30 |
|  | User Name |  | Varchar2 |  | 30 |
|  | Rank | | Varchar2 |  | 30 |
|  |  | |  |  |  |
|  | Type | | Varchar2 |  | 30 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Faculty Deans** | |
|  |  |  |  |  |  |
|  | **Faculty Dean** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | User ID | | Number |  | 30 |
|  | User Name |  | Varchar2 |  | 30 |
|  | Faculty Name | | Varchar2 |  | 30 |
|  |  | |  |  |  |
|  | Type | | Varchar2 |  | 30 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Map** | |
|  |  |  |  |  |  |
|  | **Map** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Location ID | | Number , | Number | 30 , 30 |
|  | Destination |  | Float , | Float | 40,40 |
|  | Area | | Varchar2 |  | 50 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Transport** | |
|  |  |  |  |  |  |
|  | **Transport** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Transport ID | | Number |  | 30 |
|  | Type |  | Varchar2 |  | 30 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Bus** | |
|  |  |  |  |  |  |
|  | **Bus** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Bus ID | | Number |  | 30 |
|  | Bus name |  | Varchar2 |  | 30 |
|  | Destination | | Varchar2 |  | 50 |
|  |  | |  |  |  |
|  | Stoppage | | Varchar2 |  | 50 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for General Calendar** | |
|  |  |  |  |  |  |
|  | **General Calendar** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Date | | Varchar2 |  | 30 |
|  | Holiday |  | Varchar2 |  | 100 |
|  | Event | | Varchar2 |  | 100 |
|  |  | |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **Table 4: Schema for Special Calendar** | |
|  |  |  |  |  |  |
|  | **Special Calendar** | |  |  |  |
|  |  |  |  |  |  |
|  | **Attribute** | | **Type** |  | **Size** |
|  |  | |  |  |  |
|  | Date | | Varchar2 |  | 30 |
|  | Holiday |  | Varchar2 |  | 100 |
|  | Academic Schedule |  | Varhcar2 |  | 100 |
|  | Event | | Varchar2 |  | 100 |
|  |  | |  |  |  |

**Chapter 6: Class Based Modeling**

**6.1 Class Based Modeling Concept**

Class-based modeling represents the objects that the system will manipulate, the operations that will applied to the objects, relationships between the objects and the collaborations that occur between the classes that are defined.

**6.2 General Classification**

To identify the potential classes, nouns were selected from the solution space of the story. These were then characterized in seven general classifications. The seven general characteristics are as follows:

1. External entities
2. Things
3. Events
4. Roles
5. Organizational units
6. Places
7. Structures

Following are the specifications of the nouns according to the general classifications.

## Table 2: General Classification of Noun

|  |  |  |  |
| --- | --- | --- | --- |
| **SL NO** | **NOUN** | **P/S** | **General Classification** |
|  |  |  |  |
| 1 | Account | S | - |
| 2 | User | S | 4,5 |
| 3 | Application | P | - |
| 4 | Google play store | S | - |
| 5 | Dhaka University | P | 5,7 |
| 6 | Google account | P | - |
| 7 | Profile | S | - |
| 8 | User Name | S | - |
| 9 | Academic Year | S | - |
| 10 | Email | S | - |
|  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 11 | Type | S |  | - |
| 12 | delete | S |  | 3 |
| 13 | Department | S |  | 5,7 |
| 14 | specified | S |  | - |
| 15 | Sign up | S |  | 3 |
| 16 | Sign in | S |  | 3 |
| 17 | Authentication | S |  | - |
| 18 | Firebase Auth 2 | P |  | 1 |
| 19 | Information | P |  | - |
| 20 | Log out | P |  | 3 |
| 21 | dropdown | P |  | - |
| 22 | Menu | S |  | - |
| 23 | Admin | P |  | 4,5 |
| 24 | features | P |  | - |
| 25 | year | P |  | - |
| 26 | Keyword | S |  | - |
| 27 | exciting | P |  | - |
| 28 | calendar | S |  | 3,5 |
| 29 | group chat | P |  | 5 |
| 30 | messenger | P |  | - |
| 31 | activities | P |  | 2 |
| 32 | update | P |  | 3 |
| 33 | Confirmation | S |  | - |
| 34 | separate | S |  | - |
| 35 | website | P |  | 2 |
| 36 | wish | S |  | - |
| 37 | DU brief history | P |  | - |
| 38 | achievement | S |  | - |
| 39 | Governing Body | S |  | 4,5 |
| 40 | Halls | S |  | 5,7 |
| 41 | Club | S |  | 5,7 |
| 42 | society | S |  | - |
| 43 | places | P |  | - |
| 44 | view | S |  | - |
| 45 | search | S |  | 3 |
| 46 | Transport | S |  | 2,7 |
| 47 | bus | P |  | 2,7 |
| 48 | stoppage | P |  |  |
| 49 | Map | P |  | 3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 50 | current position | S |  | - |
| 51 | destination | P |  | - |
| 52 | route | P |  | - |
| 53 | offices | P |  | 7 |
| 54 | Communication | P |  | - |
| 55 | first | P |  | - |
| 56 | person | P |  | 5 |
| 57 | Database | S |  | 5 |
| 58 | private chat | S |  | - |
| 59 | option | S |  | - |
| 60 | holiday | P |  | 3 |
| 61 | Events | P |  | 3 |
| 62 | Notification | S |  | 3 |
| 63 | Group | S |  |  |
| 64 | Chat | S |  | 3 |
| 65 | academic | S |  | 7 |
| 66 | web interface | S |  | 5 |
| 67 | server | S |  | 2 |
| 68 | data transparency | S |  | - |
| 69 | service | S |  | - |
| 70 | security | S |  | - |
| 71 | Homepage | S |  | - |
| 72 | icons | S |  | - |
| 73 | top | S |  | - |
| 74 | settings | S |  | - |
| 75 | language | S |  | - |
| 76 | layout | S |  | - |
| 77 | tree based | S |  | - |
| 78 | design | S |  | - |
| 79 | structure | S |  | - |
| 80 | database query | S |  | 3,7 |
| 81 | two | S |  | - |
| 82 | restriction | P |  | - |
| 83 | Message | P |  | 3 |
| 84 | link | P |  | - |
| 85 | right | P |  | - |
| 86 | Status | S |  | - |
| 87 | Student | S |  | 4,5,7 |
| 88 | picture | P |  | - |
| 89 | name | P |  | - |
| 90 | rank | P |  | - |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 91 | Administrative committee | S |  | 4,7 |
| 92 | Editorial committee | S |  | 4,7 |
| 93 | Faculty Deans | S |  | 4,7 |
| 94 | faculty name | P |  | - |
| 95 | Dean | P |  | - |
| 96 | Department ID | P |  | - |
| 97 | Institute ID | P |  | - |
| 98 | Hall ID | P |  | - |
| 99 | Office ID | S |  | - |
| 100 | establishment year | S |  | - |
| 101 | Description | S |  | - |
| 102 | Location | P |  | - |
| 103 | Area | P |  | - |
| 104 | Teacher | S |  | 4,5,7 |
| 105 | Department Name | P |  |  |
| 106 | Institute | S |  | 5,7 |
| 107 | Institute Name | P |  |  |
| 108 | Specific Calendar | S |  | 4,5, |
| 109 | Date | P |  | - |
| 110 | Academic Schedule | P |  | 5 |
| 111 | Other user | S |  | 4 |
| 112 | Infrastructure | S |  | 7 |
| 113 | Bus ID | P |  | - |
| 114 | Bus name | P |  | - |
| 115 | password | P |  | - |
|  |  |  |  |  |

**6.3 Selection Criteria**

The potential classes were then selected as classes by six Selection Criteria. A potential class becomes a class when it fulfills all six characteristics.

1. Retained Information
2. Needed Services
3. Multiple Attributes
4. Common attributes
5. Common operations
6. Essential requirements

**Table 17: Selection Criteria of Potential Classes**

|  |  |  |  |
| --- | --- | --- | --- |
| **SL NO** | **NOUN** | **P/S** | **Selection Criteria** |
|  |  |  |  |
| 1 | User | S | 1,2,3,4,5 |
| 2 | Dhaka University | P | - |
| 3 | Delete | S | - |
| 4 | Department | S | 4 |
| 5 | Dhaka University | P | - |
| 6 | Sign up | S | - |
| 7 | Sign in | S | - |
| 8 | Infrastructure | S | 1,3,4,5 |
| 9 | Log out | P | - |
| 10 | Admin | P | 1,2,3,4,5,6 |
| 11 | General Calendar | S | 1,2,4 |
| 12 | Group Chat | P | 2,4,5 |
| 13 | activities | P | - |
| 14 | update | P | - |
| 15 | website | P | - |
| 16 | Governing Body | S | 1,2,4 |
| 17 | Halls | S | 4 |
| 18 | Club | S | 4 |
| 19 | search | S | 2 |
| 20 | Transport | S | 2,3,4,6 |
| 21 | Map | P | 1,2,4,5 |
| 22 | bus | P | - |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 23 | Offices | P | 4 |
| 24 | person | P | - |
| 25 | Database | S | 1,2,6 |
| 26 | Holiday | P | - |
| 27 | Events | P | - |
| 28 | Notification | S | - |
| 29 | Chat | S | - |
| 30 | academic | S | - |
| 31 | web interface | S | - |
| 32 | server | S | - |
| 33 | database query | S | - |
| 34 | Message | P | 2 |
| 35 | Student | S | 1,2,3,4,5 |
| 36 | Editorial committee | S | 2 |
| 37 | Faculty Deans | S | - |
| 38 | Teacher | S | 1,2,3,4 |
| 39 | Institute | S | 4 |
| 40 | Specific Calendar | S | 1,2,4 |
| 41 | Academic Schedule | P | - |
| 42 | Other user | S | 1,2,3,4 |
|  |  |  |  |

**6.4 Potential Classes**

From above table, we have taken all the noun who passed three or more accepted criteria. So these are the candidate classes who are selected primarily:

* User
* Student
* Teacher
* Other User
* Admin
* Infrastructure
* Governing Body
* Transport
* Map
* Database
* Group Chat
* General Calendar
* Specific Calendar

**6.5 Associate Noun and Verb Identification**

The nouns and the verbs associated with the potential classes are identified to find out the attributes and methods of each class.

**Table 18: Associate Noun and Verbs**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Potential** | **Noun** |  | **Verb** |  |
|  | **Class** |  |  |  |  |
|  |  |  |  |  |  |
| 1 | Account | Verification |  | Log in, sign up, log out, lock account, send |  |
|  |  |  |  |
|  |  | code, user |  | verification code |  |
|  |  |  |  |  |  |
| 2 | User | Id, first name, | last | Requires an account, can log in to the system, |  |
|  |  |  |  |  |
|  |  | name, email, |  | sends message, receives message, search, update |  |
|  |  |  |  |  |
|  |  | phone number, |  | user information, recover his/her password |  |
|  |  |  |  |  |
|  |  |  |  |  |  |
| 3 | Admin | address, |  | Approves request, can add product, can edit |  |
|  |  |  |  |
|  |  |  |  | product, remove product, order product, receive |  |
|  |  | message, |  | product, advertise product, remove advertise |  |
|  |  |  |  |  |
|  |  |  |  | product, block customer, change type, inputs type |  |
|  |  | status(av |  | of shop management, identifying the needed |  |
|  |  |  |  |  |
|  |  | ailable or |  | products, can monitor order, can manage discount |  |
|  |  |  |  |  |
|  |  |  |  | and discounted products, change discount, manage |  |
|  |  | not) |  | social media |  |
|  |  |  |  |  |  |

100

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 4 | Customer | Type, amount paid | | Can search product, View or buy products online, | | |  |
|  |  | can rate and comment products, | share products | |  |
|  |  | online, | total |  |
|  |  | with friends, can chat with admin and edit own | | |  |
|  |  |  |  |  |
|  |  | amount | paid, | information |  |  |  |
|  |  | message |  |  |  |  |  |
|  |  |  |  |  | |  |  |
| 5 | System | Product, user. |  | Show popular product(s), show viewing options, | |  |  |
|  |  |  |  | show searched items, can generate SMS and send | |  |  |
|  |  |  |  | SMS, set connection for chat, set items to add to | |  |  |
|  |  |  |  | cart. |  |  |  |
|  |  |  |  |  |  |  |  |
| 6 | Product | Id, name, quantity, | | Quantity will be updated, |  |  |  |
|  |  | editing, product details, product color and price will | |  |  |
|  |  | price,color, |  |  |  |
|  |  |  | be updated, discount will be set | with each pro |  |  |
|  |  | discount, |  |  |  |
|  |  |  | advertisement and rating by users will also be updat | |  |  |
|  |  | insufficient |  |  |  |
|  |  |  |  |  |  |  |
|  |  | quantity, | rate, |  |  |  |  |
|  |  | advertisement |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| 7 | Notificati | Id, receiver, type | | Will be generated and sent |  |  |  |
| on |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | | |  |
| 8 | Database | User, |  | Will be stored, will be updated,will be removed | | |  |
|  |  |  |  |  |  |
|  |  | product |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

101

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | Search | User, |  | Search by keyword, search by popular, search type | | | | |  |
|  |  | wise, search recommended, search by price and | | | | |  |
|  |  | product, date. | |  |
|  |  | discounted price | |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | Keyword, price | |  |  |  |  |  |  |
|  |  |  |  |  | | | | |  |
| 10 | Order | Quantity,Product, | | calculate total price of product(s),make order | | | | |  |
|  | summary, set an order date and ID | | | |  |  |
|  |  | Price,Customer | |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  | | | | |  |
| 11 | Activity | Product,Admin,Cu | | Store authentication information,store ratings and | | | | |  |
|  | comments | of | product(s), | update | product |  |
|  |  | stomer |  |  |
|  |  |  | information, store advertisement details and store | | | | |  |
|  |  |  |  |  |
|  |  |  |  | order information | |  |  |  |  |
|  |  |  |  |  | | | | |  |
| 12 | Delivery | Destination, date, | | Sets destination, sets date, provide delivery service | | | | |  |
| Service |  |  |  |  |  |  |
|  | delivery | service |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  | info |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

102

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 13 | Payment | Payment method, | Selects payment method, stored payment info |  |
|  |  |  |
|  |  | payment info |  |  |
|  |  |  |  |  |
| 14 | Social | Connection status | Establishes connection with social media, stores |  |
| Media | social media information |  |
|  |  |  |
|  |  |  |  |  |

**6.5 Attribute Selection**

**Table 19: Attribute Selection of Classes**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Potential Class | Noun |  |
|  |  |  |  |
| 1 | Account | Verification code |  |
|  |  |  |
|  |  | User |  |
|  |  |  |  |
| 2 | User | Id |  |
|  |  | first name |  |
|  |  | last name |  |
|  |  | email |  |
|  |  | phone number |  |
|  |  |  |  |

103

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3 | Admin |  | Address |  |
|  |  |  |  |
|  |  |  | message |  |
|  |  |  | status(available or not) |  |
|  |  |  |  |  |
| 4 | Customer |  | Type |  |
|  |  |  |  |
|  |  |  | amount paid online |  |
|  |  |  | total amount paid |  |
|  |  |  | message |  |
|  |  |  |  |  |
| 5 | System |  | Product |  |
|  |  |  | User |  |
|  |  |  |  |  |
| 6 | Product |  | Id |  |
|  |  |  |  |
|  |  |  | Name |  |
|  |  |  | Quantity |  |
|  |  |  | Price |  |
|  |  |  | Color |  |
|  |  |  | Discount |  |
|  |  |  | Insufficient quantity |  |
|  |  |  | Rate |  |
|  |  |  | Advertisement |  |
|  |  |  |  |  |
| 7 | Notification |  | ID |  |
|  |  |  |
|  |  |  | Receiver |  |
|  |  |  | Type |  |
|  |  |  |  |  |
| 8 | Database |  | User |  |
|  |  |  |
|  |  |  | Product |  |
|  |  |  |  |  |
|  |  | 104 | |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | Search | User |  |
|  |  |
|  |  | Product |  |
|  |  | Date |  |
|  |  | Keyword |  |
|  |  | Price |  |
|  |  |  |  |
| 10 | Order | Quantity |  |
|  |  |
|  |  | Product |  |
|  |  | Price |  |
|  |  | Customer |  |
|  |  |  |  |
| 11 | Activity | Product |  |
|  |  |
|  |  | Admin |  |
|  |  | Customer |  |
|  |  |  |  |
| 12 | Delivery Service | Destination |  |
|  |  |
|  |  | Date |  |
|  |  | Delivery service info |  |
|  |  |  |  |

105

|  |  |  |  |
| --- | --- | --- | --- |
| 13 | Payment | Payment method |  |
|  |  |
|  |  | Payment info |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 14 | Social Media | Connection status |  |
|  |  |

**6.6 Methods Identification:**

**Table 20: Methods of Class**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Class | Methods | |
|  |  |  |  |
| 1. | Account |  | signUp() |
|  |  |  | login() |
|  |  |  | signOut() |
|  |  |  | lockAccount() |
|  |  |  | sendVerificationCode() |
|  |  |  | recoverPassword() |
|  |  |  | verifyEmail() |
|  |  |  | verifyUser() |
|  |  |  | manageForgotPassword() |
|  |  |  |  |
| 2 | User |  | setCustomerID() |
|  |  |  | getCustomerID() |
|  |  |  | setFirstName() |
|  |  |  | getFirstName() |
|  |  |  | setLastName() |
|  |  |  |  |
|  |  | 106 |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | getLastName() |
|  |  |  | setEmail() |
|  |  |  | getEmail() |
|  |  |  | setPhoneNo() |
|  |  |  | getPhoneNo() |
|  |  |  |  |
| 3 | Admin |  | approveRequest() |
|  |  |  | chat() |
|  |  |  | addProduct() |
|  |  |  | removeProduct() |
|  |  |  | editProduct() |
|  |  |  | advertiseProduct() |
|  |  |  | monitorOrder() |
|  |  |  | manageDiscount() |
|  |  |  | editProfile() |
|  |  |  | manageDeliveryService() |
|  |  |  | manageSocialMedia() |
|  |  |  |  |
| 4 | Customer |  | setCustomerID() |
|  |  |  | getCustomerID() |
|  |  |  | setFirstName() |
|  |  |  | getFirstName() |
|  |  |  | setLastName() |
|  |  |  | getLastName() |
|  |  |  | setEmail() |
|  |  |  | getEmail() |
|  |  |  | setPhoneNo() |
|  |  |  | getPhoneNo() |
|  |  |  | setAddress() |
|  |  |  | getAddress() |
|  |  |  | receiveMesseges() |
|  |  |  | chat() |
|  |  |  | orderProduct() |
|  |  |  | purchase() |
|  |  |  |  |
|  |  | 107 |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | editProfile() |
|  |  |  | comment() |
|  |  |  | rateProduct() |
|  |  |  | shareWithFriends() |
|  |  |  |  |
| 5 | System |  | viewOptions() |
|  |  |  | showSearchedItem() |
|  |  |  | generateSMS() |
|  |  |  | sendSMS() |
|  |  |  | setConnection() |
|  |  |  | addToCart() |
|  |  |  |  |
| 6 | Product |  | setProductID() |
|  |  |  | getProductID() |
|  |  |  | getProductName() |
|  |  |  | setProductName() |
|  |  |  | setQuantity() |
|  |  |  | getQuantity() |
|  |  |  | setColor() |
|  |  |  | getColor() |
|  |  |  | setPrice() |
|  |  |  | getPrice() |
|  |  |  | setDiscount() |
|  |  |  | getDiscount() |
|  |  |  | attachAdvertisement() |
|  |  |  | calculateDiscountedPrice() |
|  |  |  | setRating() |
|  |  |  | getRating() |
|  |  |  |  |
| 7 | Notification |  | setNotificationID() |
|  |  |  | getNotificationID() |
|  |  |  | getSender() |
|  |  |  | setSender() |
|  |  |  | setReceiver() |
|  |  |  | getReceiver() |
|  |  |  |  |
|  |  | 108 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | getDescription() | |
|  |  |  | setDescription() | |
|  |  |  | sendNotification() | |
|  |  |  | generateNotification() | |
|  |  |  |  | |
| 8 | Database |  | insertInfo() | |
|  |  |  | updateInfo() | |
|  |  |  | deleteInfo() | |
|  |  |  | retrieveInfo() | |
|  |  |  |  | |
| 9 | Search |  | searchbykeywords() | |
|  |  |  | searchPopular() | |
|  |  |  | searchTypewise() | |
|  |  |  | searchRecommended() | |
|  |  |  | searchbyprice() | |
|  |  |  | searchDiscountedPrice() | |
|  |  |  |  |  |
| 10 | Order |  |  | setOrderID() |
|  |  |  |  | getOrderID() |
|  |  |  |  | makeOrderSummery() |
|  |  |  |  | getOrderDate() |
|  |  |  |  | setOrderDate() |
|  |  |  |  | manageCancelOrder() |
|  |  |  |  | |
| 11 | Activity |  | storeAuthenticationInfo() | |
|  |  |  | storeRatingAndComment() | |
|  |  |  | storeProductModificationInfo() | |
|  |  |  | storeAdvertisementRecord() | |
|  |  |  | storeOrderInfo() | |
|  |  |  |  | |
| 12 | Delivery Service |  | setDeliveryDestination() | |
|  |  |  | getDeliveryDestination() | |
|  |  |  | manageDeliveryDate() | |
|  |  |  | setServiceID() | |
|  |  |  | getServiceID() | |
|  |  |  |  |  |
|  |  | 109 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | storePathaoDeliveryInfo() |
|  |  |  | storeCourierDeliveryInfo() |
|  |  |  |  |
| 13 | Payment |  | payViaCreditCard() |
|  |  |  | payViaBKash() |
|  |  |  | payViaDBBL() |
|  |  |  | storePaymentInfo() |
|  |  |  | storeCashOnDeliveryInfo() |
|  |  |  |  |
| 14 | Social Media |  | establishConnection() |
|  |  |  | connectWithFacebook() |
|  |  |  | connectWithTwitter() |
|  |  |  |  |

**6.7 Finalizing Classes**

To identify the final classes, it was required to check if there can be any hierarchies, merges, additional attributes, methods or classes. These identifications are given below:

1. There are two types of users in the system. So the user class could be the parent class of the admin class and customer class. But, as the customer class and the admin class has different attributes and methods there is no need of user class.
2. System and User classes will have an additional method (viewOptions()) so that it can be used to get to other methods.

**6.8 Class Cards**

**Table 21: Class Card of Customer**

Customer

|  |  |  |  |
| --- | --- | --- | --- |
| Attributes | Methods | |  |
|  |  |  |  |
| Type |  | setCustomerID() |  |
| amount paid online |  | getCustomerID() |  |
|  | setFirstName() |  |
|  |  |
|  |  |  |  |
|  |  | 110 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| total amount paid | |  | getFirstName() |  |
| message |  |  | setLastName() |  |
|  |  | getLastName() |  |
|  |  |  |
|  |  |  | setEmail() |  |
|  |  |  | getEmail() |  |
|  |  |  | setPhoneNo() |  |
|  |  |  | getPhoneNo() |  |
|  |  |  | setAddress() |  |
|  |  |  | getAddress() |  |
|  |  |  | receiveMesseges() |  |
|  |  |  | chat() |  |
|  |  |  | signOut() |  |
|  |  |  | orderProduct() |  |
|  |  |  | browseProduct() |  |
|  |  |  | showOrderList() |  |
|  |  |  | addTocart() |  |
|  |  |  | SearchProduct() |  |
|  |  |  | purchase() |  |
|  |  |  | editProfile() |  |
|  |  |  | comment() |  |
|  |  |  | rateProduct() |  |
|  |  |  | shareWithFriends() |  |
|  |  |  | |  |
| Responsibilities |  | Collaborator | |  |
|  |  |  | |  |
| Customer | Information | Database | |  |
| Storing |  |  |  |  |
|  | |  | |  |
| Customer authentication | | Account | |  |
|  | |  | |  |
| Connection for sharing | | Social Media | |  |
|  |  |  |  |  |
| Shopping |  | Order |  |  |
|  | |  | |  |
| Viewing product | | Searching | |  |
|  |  |  |  |  |

111

Table 22: Class Card of Admin

Administrator

|  |  |  |  |
| --- | --- | --- | --- |
| Attributes | Methods | |  |
|  |  |  |  |
| Address |  | approveRequest() |  |
|  | chat() |  |
|  |  |
| Message |  | signOut() |  |
|  | addProduct() |  |
|  |  |
|  |  | removeProduct() |  |
| status(available |  | editProduct() |  |
|  |  | advertiseProduct() |  |
| or not) |  | monitorOrder() |  |
|  |  | manageDiscount() |  |
|  |  | editProfile() |  |
|  |  | manageDeliveryService() |  |
|  |  | manageSocialMedia() |  |
|  |  | approveCancelOrder() |  |
|  |  | |  |
| Responsibilities | Collaborator | |  |
|  |  | |  |
| Admin Authentication | Account | |  |
|  |  | |  |
| Edit product Info | Product | |  |
|  |  | |  |
| Delivery of Product | Delivery Service | |  |
|  |  |  |  |

**Table 23: Class Card of Database**

Database

|  |  |  |  |
| --- | --- | --- | --- |
| Attributes | Methods | |  |
|  |  |  |  |
| User |  | insertInfo() |  |
| Product |  | updateInfo() |  |
|  | deleteInfo() |  |
|  |  |
|  |  | retrieveInfo() |  |
|  |  | sendConfirmation() |  |
|  |  | |  |
| Responsibilities | Collaborator | |  |
|  |  |  |  |
|  |  | 112 |  |

|  |  |
| --- | --- |
| Store product Info | Product |
|  |  |
| Store admin Information | Admin |
|  |  |
| Store customer Information | Customer |
|  |  |
| Product Selected | Order |
|  |  |

**Table 24: Class Card of Product**

Product

|  |  |  |  |
| --- | --- | --- | --- |
| Attributes | Methods | |  |
|  |  |  |  |
| Id |  | setProductID() |  |
| Name |  | getProductID() |  |
|  | getProductName() |  |
|  |  |
| Quantity |  | setProductName() |  |
| Price |  | setQuantity() |  |
|  | getQuantity() |  |
|  |  |
| Color |  | setColor() |  |
| Discount |  | getColor() |  |
|  | setPrice() |  |
|  |  |
| Insufficient quantity |  | getPrice() |  |
|  | setDiscount() |  |
| Rate |  |
|  | getDiscount() |  |
| Advertisement |  | attachAdvertisement() |  |
|  | calculateDiscountedPrice() |  |
|  |  |
|  |  | setRating() |  |
|  |  | getRating() |  |
|  |  | showPopularProduct() |  |
|  |  | showRecommendedProduct() |  |
|  |  | |  |
| Responsibilities | Collaborator | |  |
|  |  | |  |
| Stores Product Info | Database | |  |
|  |  |  |  |
| Edit Product Info | Admin |  |  |
|  |  | |  |
| Manage activity | Activity | |  |
|  |  |  |  |

113

**Table 25: Class Card of System**

System

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Attributes | |  | Methods | |  |  |
|  | |  |  |  |  |  |
| Product | |  |  | viewOptions() |  |  |
| User | |  |  | showSearchedItem() |  |  |
|  |  | generateSMS() |  |  |
|  |  |  |  |  |
|  |  |  |  | sendSMS() |  |  |
|  |  |  |  | setConnection() |  |  |
|  |  |  |  | manageCart() |  |  |
|  |  |  |  | manageNotification() |  |  |
|  |  |  |  | checkCancellationAvailability() |  |  |
|  | |  |  | |  |  |
| Responsibilities | |  | Collaborator | |  |  |
|  | |  |  | |  |  |
| Login | |  | Social Media | |  |  |
|  | |  |  | |  |  |
| Check Destination | |  | Delivery service | |  |  |
|  | |  |  | |  |  |
| Get searched Items | |  | Searching | |  |  |
|  | |  |  | |  |  |
| Generate Notifications | |  | Notification | |  |  |
|  |  |  |  | |  |  |
|  |  | **Table 26: Class Card of Account** | | |  |  |
|  | |  |  |  |  |  |
| Account | |  |  |  |  |  |
|  | |  |  | |  |  |
| Attributes | |  | Methods | |  |  |
|  |  |  |  |  |  |  |
|  | User |  |  | signUp() |  |  |
|  | Verification Code |  |  | login() |  |  |
|  |  |  |  | signOut() |  |  |
|  |  |  |  | lockAccount() |  |  |
|  |  |  |  | sendVerificationCode() |  |  |
|  |  |  |  | recoverPassword() |  |  |
|  |  |  |  | verifyEmail() |  |  |
|  |  |  |  | verifyUser() |  |  |
|  |  |  |  | manageForgotPassword() |  |  |
|  | |  |  | |  |  |
| Responsibilities | |  | Collaborator | |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  | 114 |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Authenticate Admin | Admin |  |  |  |
|  |  | |  |  |
| Authenticate Customer | Customer | |  |  |
|  |  | |  |  |
| Manage activity | Activity | |  |  |
|  |  | |  |  |
|  | **Table 27: Class Card of Activity** | |  |  |
|  |  |  |  |  |
| Activity |  |  |  |  |
|  |  | |  |  |
| Attributes | Methods | |  |  |
|  |  |  |  |  |
| Product |  | storeAuthenticationInfo() |  |  |
| Admin |  | storeRatingAndComment() |  |  |
|  | storeProductModificationInfo() |  |  |
|  |  |  |
| Customer |  | storeAdvertisementRecord() |  |  |
|  |  | storeOrderInfo() |  |  |
|  |  | |  |  |
| Responsibilities | Collaborator | |  |  |
|  |  | |  |  |
| Take info from user | Account | |  |  |
|  |  | |  |  |
| Manage product activity | Product | |  |  |
|  |  | |  |  |
|  | **Table 28: Class Card of Searching** | |  |  |
|  |  |  |  |  |
| Searching |  |  |  |  |
|  |  | |  |  |
| Attributes | Methods | |  |  |
|  |  |  |  |  |
| User |  | searchbykeywords() |  |  |
| Product |  | searchPopular() |  |  |
|  | searchTypewise() |  |  |
|  |  |  |
| Date |  | searchRecommended() |  |  |
| Keyword |  | searchbyprice() |  |  |
|  | searchDiscountedPrice() |  |  |
|  |  |  |
| Price |  |  |  |  |
|  |  | |  |  |
| Responsibilities | Collaborator | |  |  |
|  |  |  |  |  |
| Generating Search | System |  |  |  |
|  |  | |  |  |
| Search for Product | Customer | |  |  |
|  |  | |  |  |
| Get Products | Product | |  |  |
|  |  |  |  |  |
|  |  | 115 |  |  |

**Table 29: Class Card of Payment**

Payment

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attributes |  | Methods | |  |
|  |  |  |  |  |
| Payment method |  |  | payViaCreditCard() |  |
| Payment info |  |  | payViaBKash() |  |
|  |  | payViaDBBL() |  |
|  |  |  |
|  |  |  | storePaymentInfo() |  |
|  |  |  | storeCashOnDeliveryInfo() |  |
|  |  |  | |  |
| Responsibilities |  | Collaborator | |  |
|  | |  |  |  |
| Calculate payment as per | | Order |  |  |
| Order |  |  |  |  |
|  |  |  | |  |
|  | **Table 30: Class Card of Notification** | | |  |
|  |  |  |  |  |
| Notification |  |  |  |  |
|  |  |  | |  |
| Attributes |  | Methods | |  |
|  |  |  |  |  |
| ID |  |  | setNotificationID() |  |
| Receiver |  |  | getNotificationID() |  |
|  |  | getSender() |  |
|  |  |  |
| Type |  |  | setSender() |  |
|  |  |  | setReceiver() |  |
|  |  |  | getReceiver() |  |
|  |  |  | getDescription() |  |
|  |  |  | setDescription() |  |
|  |  |  | sendNotification() |  |
|  |  |  | generateNotification() |  |
|  |  |  | |  |
| Responsibilities |  | Collaborator | |  |
|  |  |  |  |  |
| Generate Notification |  | System |  |  |
|  |  |  |  |  |
| Gets Notification |  | User |  |  |
|  |  |  |  |  |

116

**Table 31: Class Card of SocialMedia**

SocialMedia

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attributes |  | Methods |  |  |  |
|  |  |  |  |  |  |
| Connection status |  |  | establishConnection() |  |  |
|  |  |  | connectWithFacebook() |  |  |
|  |  |  | connectWithTwitter() |  |  |
|  |  |  |  |  |  |
| Responsibilities |  | Collaborator |  |  |  |
|  |  |  |  |  |  |
| Verifies info |  | Admin |  |  |  |
|  |  |  | |  |  |
| Login to Social Media |  | System, Customer | |  |  |
|  |  |  | |  |  |
|  | **Table 32: Class Card of DeliveryService** | | |  |  |
|  |  |  |  |  |  |
| DeliveryService |  |  |  |  |  |
|  |  |  |  |  |  |
| Attributes |  | Methods |  |  |  |
|  |  |  |  |  |  |
| Destination |  |  | setDeliveryDestination() |  |  |
| Date |  |  | getDeliveryDestination() |  |  |
|  |  | manageDeliveryDate() |  |  |
|  |  |  |  |
| Delivery service info |  |  | setServiceID() |  |  |
|  |  |  | getServiceID() |  |  |
|  |  |  | storePathaoDeliveryInfo() |  |  |
|  |  |  | storeCourierDeliveryInfo() |  |  |
|  |  |  |  |  |  |
| Responsibilities |  | Collaborator |  |  |  |
|  |  |  |  |  |  |
| Set Delivery Service |  | Admin |  |  |  |
|  |  |  |  |  |  |
| Get Delivered Products |  | Customer |  |  |  |
|  |  |  |  |  |  |
| Set Destination |  | Order |  |  |  |
|  |  |  |  |  |  |

117

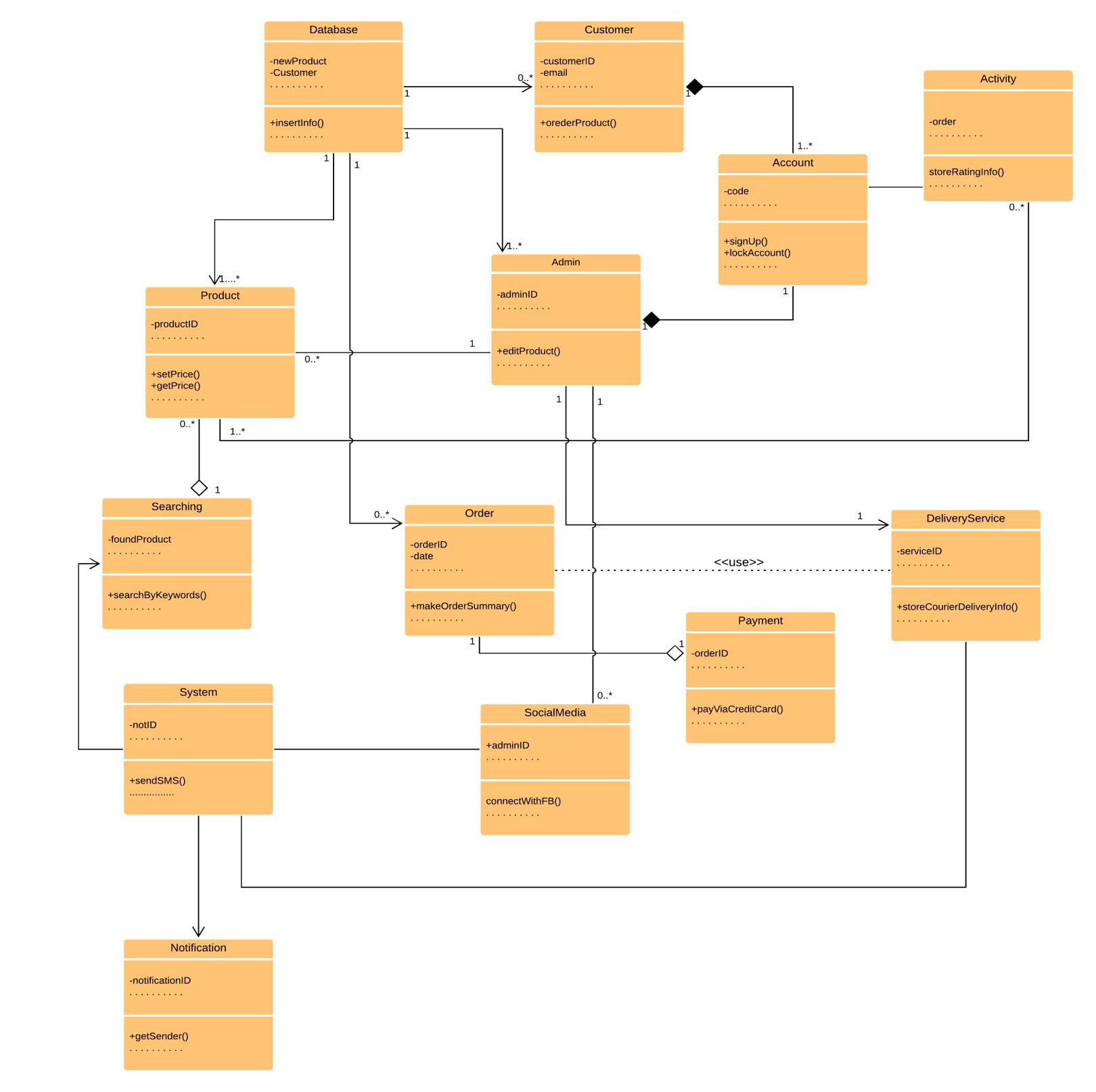
**Table 33: Class Card of Order**

Order

|  |  |  |  |
| --- | --- | --- | --- |
| Attributes | Methods |  |  |
|  |  |  |  |
| Quantity |  | setOrderID() |  |
| Product |  | getOrderID() |  |
|  | makeOrderSummery() |  |
|  |  |
| Price |  | getOrderDate() |  |
| Customer |  | setOrderDate() |  |
|  | manageCancelOrder() |  |
|  |  |
|  |  | calculateOrderPrice() |  |
|  |  |  |  |
| Responsibilities | Collaborator |  |  |
|  |  | |  |
| Order for product | Customer, Product | |  |
|  |  |  |  |
| Calculate Total Price to pay | Payment |  |  |
|  |  | |  |
| Selected via Order | Delivery service | |  |
|  |  |  |  |

118

**6.9 UML Diagram**



**Figure** 51**: UML Diagram of F-Commerce Based Client Management System**

119

**Chapter 7: Flow-Oriented Model**

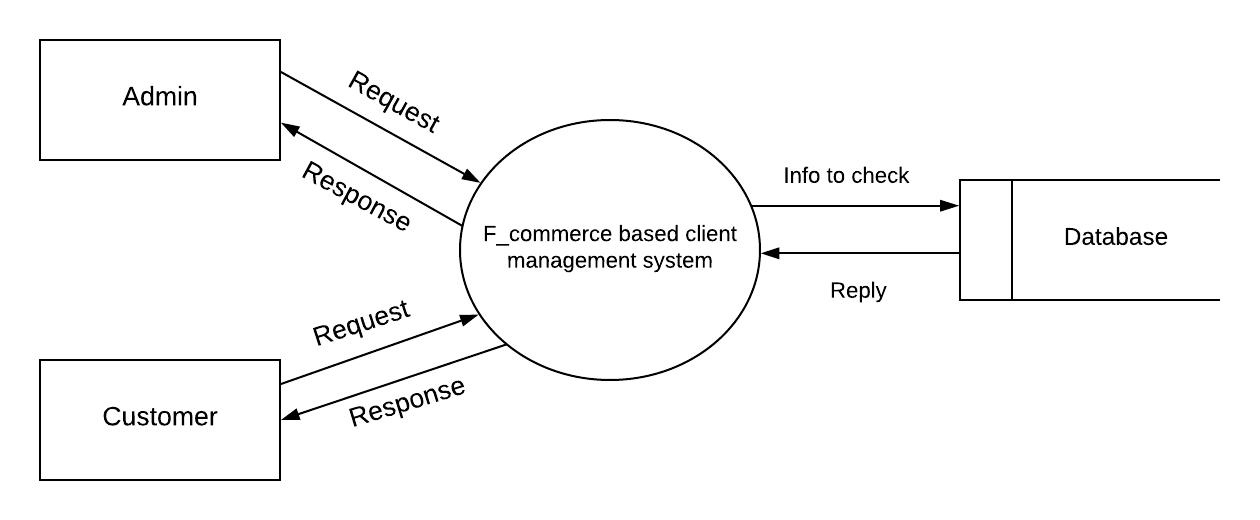
This chapter focuses on the flow oriented modeling.

**7.1 Introduction**

Although data flow-oriented modeling is perceived as an outdated technique by some software engineers, it continues to be one of the most widely used requirements analysis notations in use today. It provides additional insight into system requirements and flow.

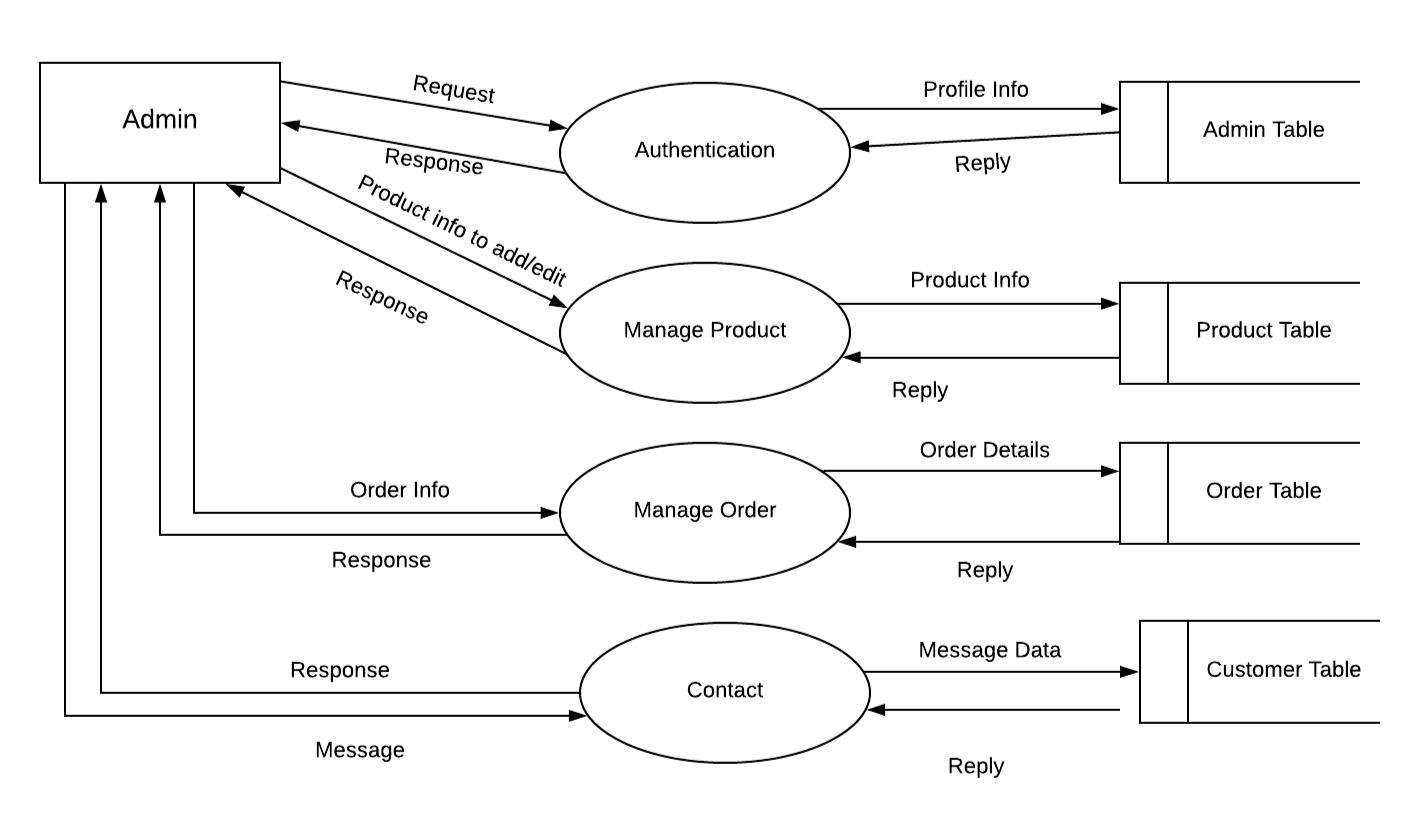
**7.2 Data Flow Diagram (DFD)**

The DFD takes an input-process-output view of a system. In the figures, data objects are represented by labeled arrows and transformations are represented by circles.



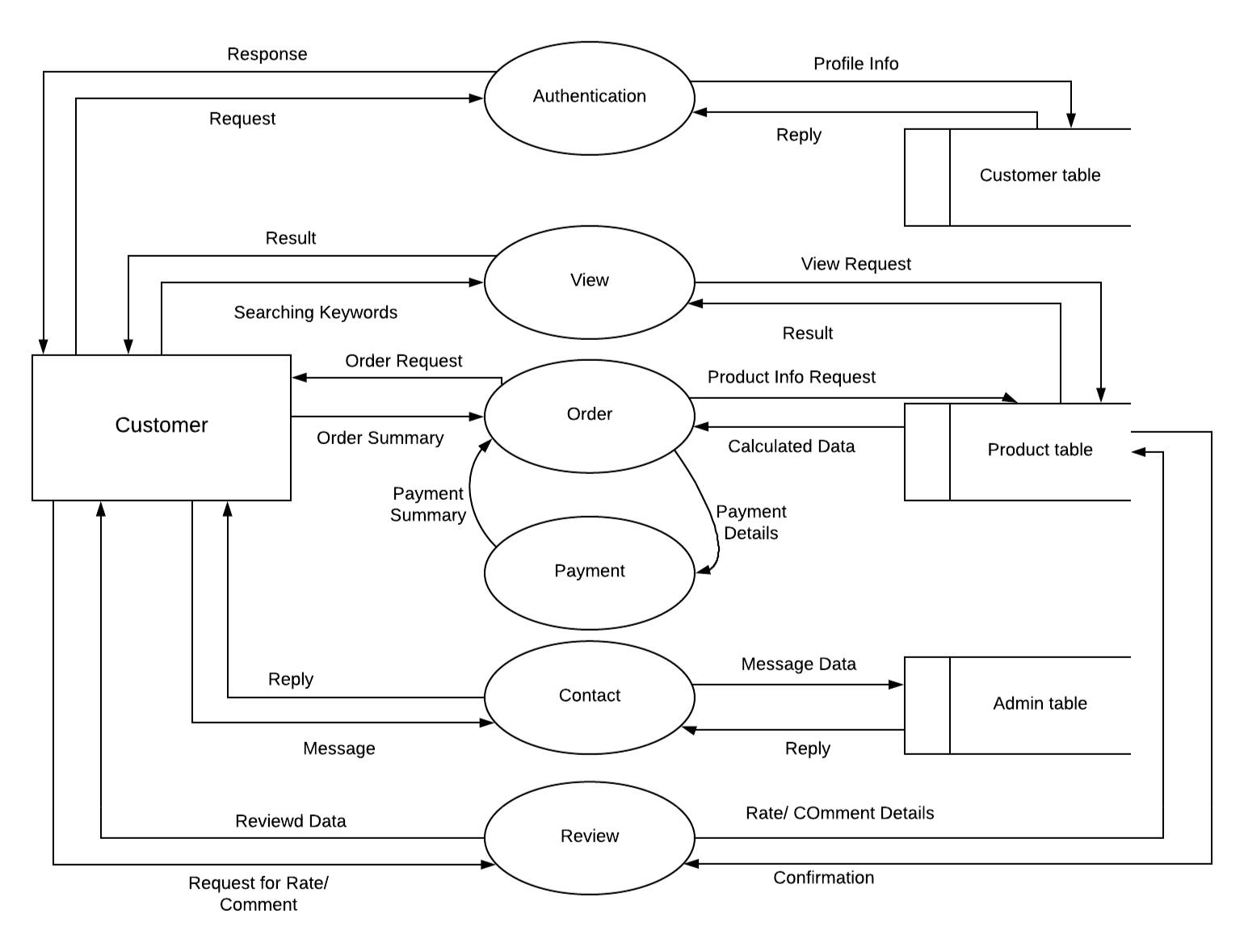
**Figure 52: Level 0 for F-commerce based client management system**

120

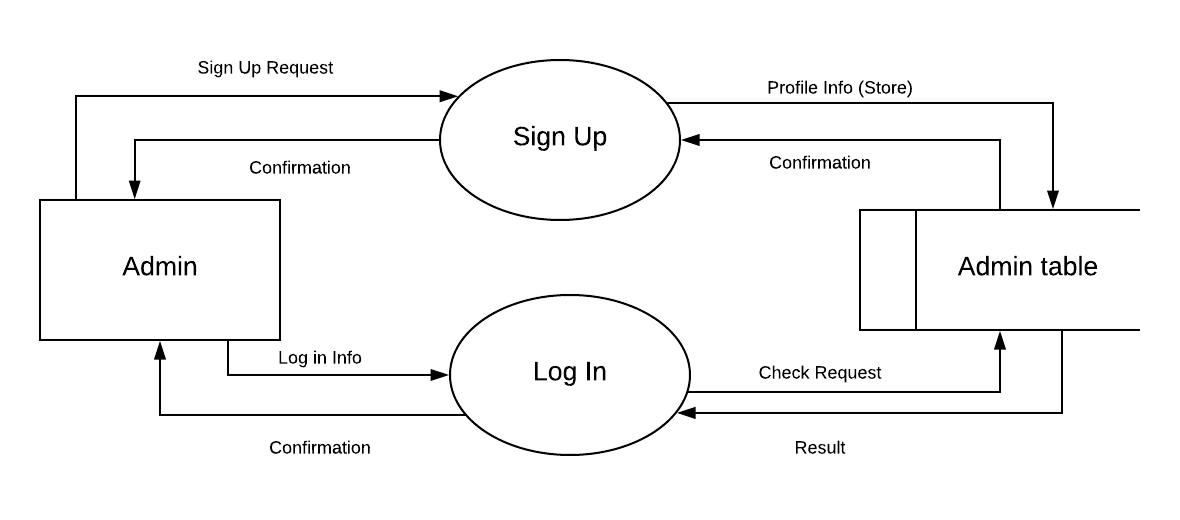


**Figure 53: Level 1.1 for F-commerce based client management system**

121

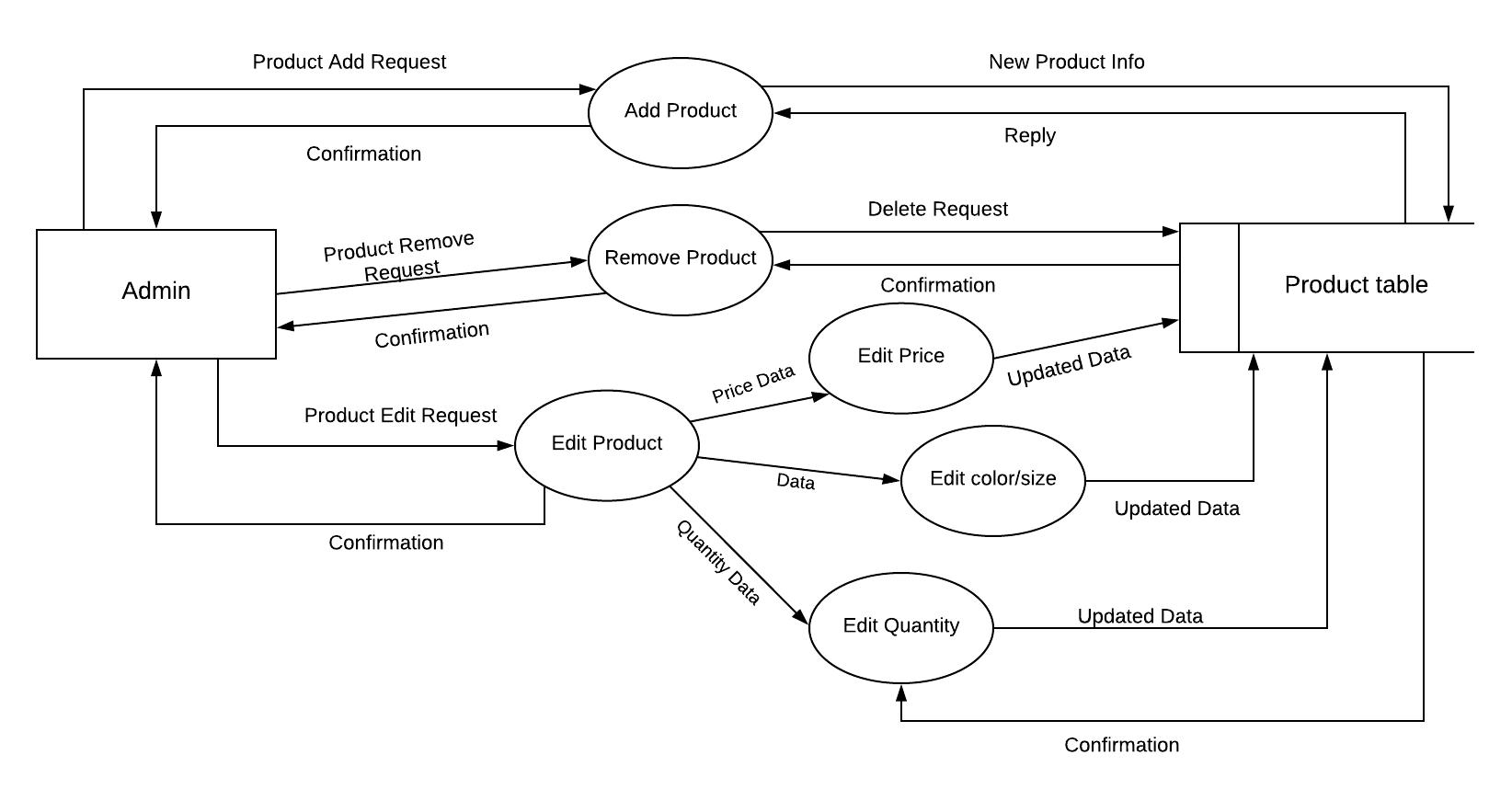


**Figure 54: Level 1.2 for F-commerce based client management system**



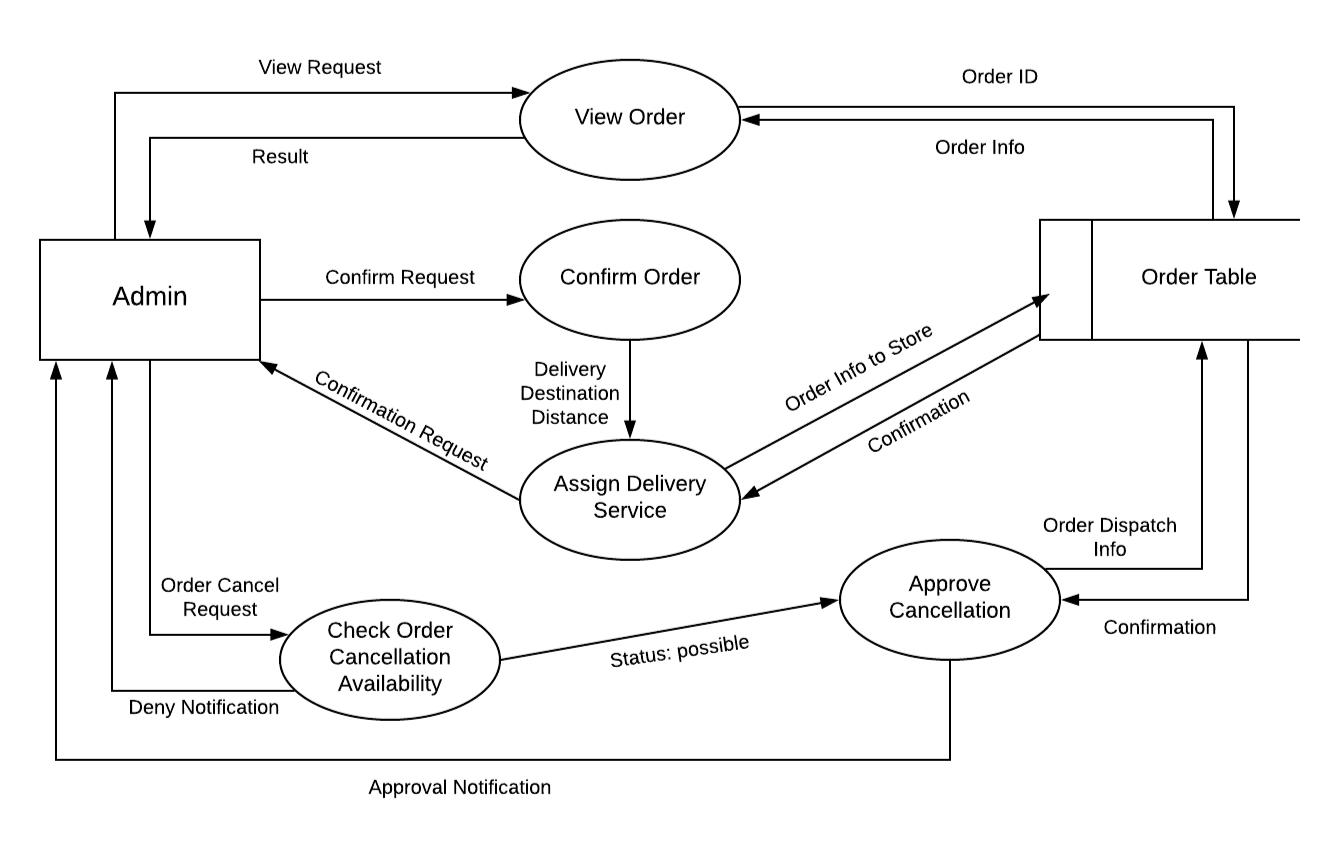
**Figure 55: Level 2.1 for F-commerce based client management system**

122

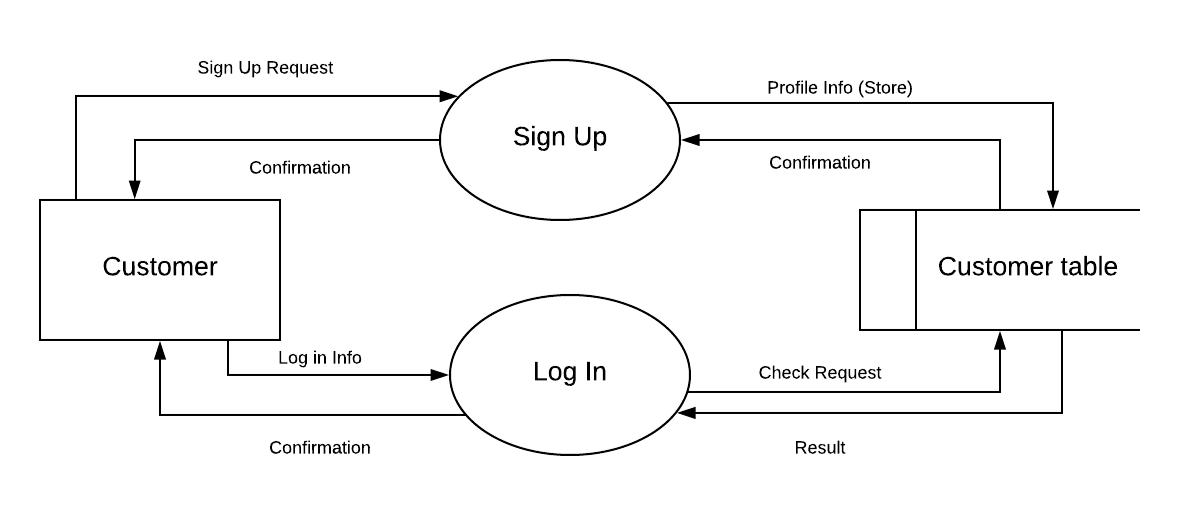


**Figure 56: Level 2.2 for F-commerce based client management system**

123

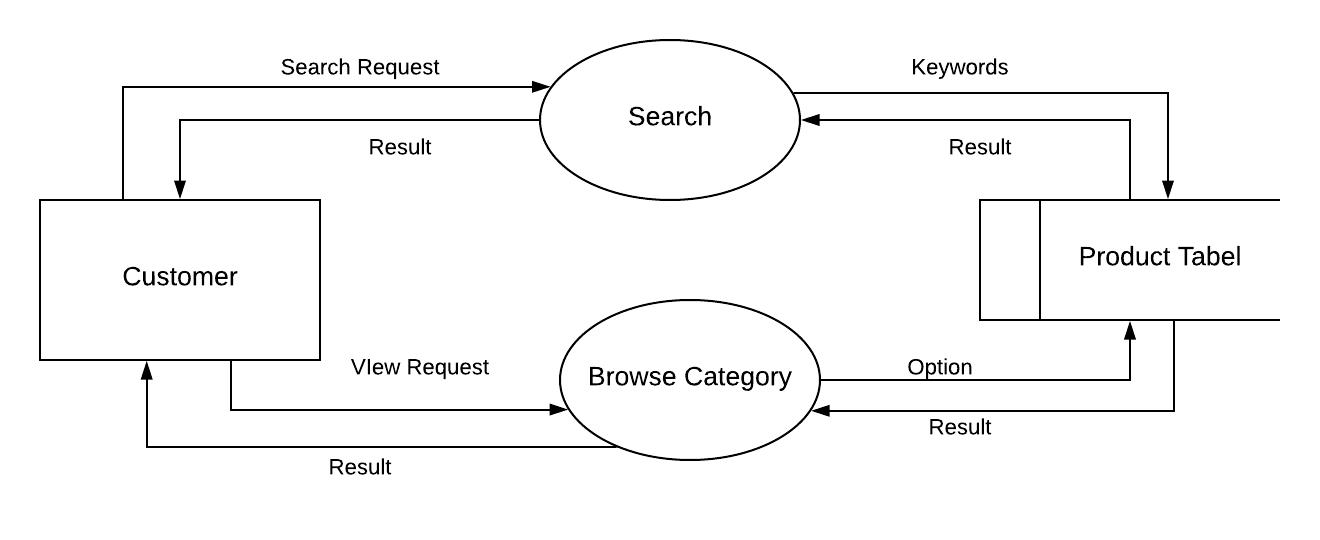


**Figure 57: Level 2.3 for F-commerce based client management system**



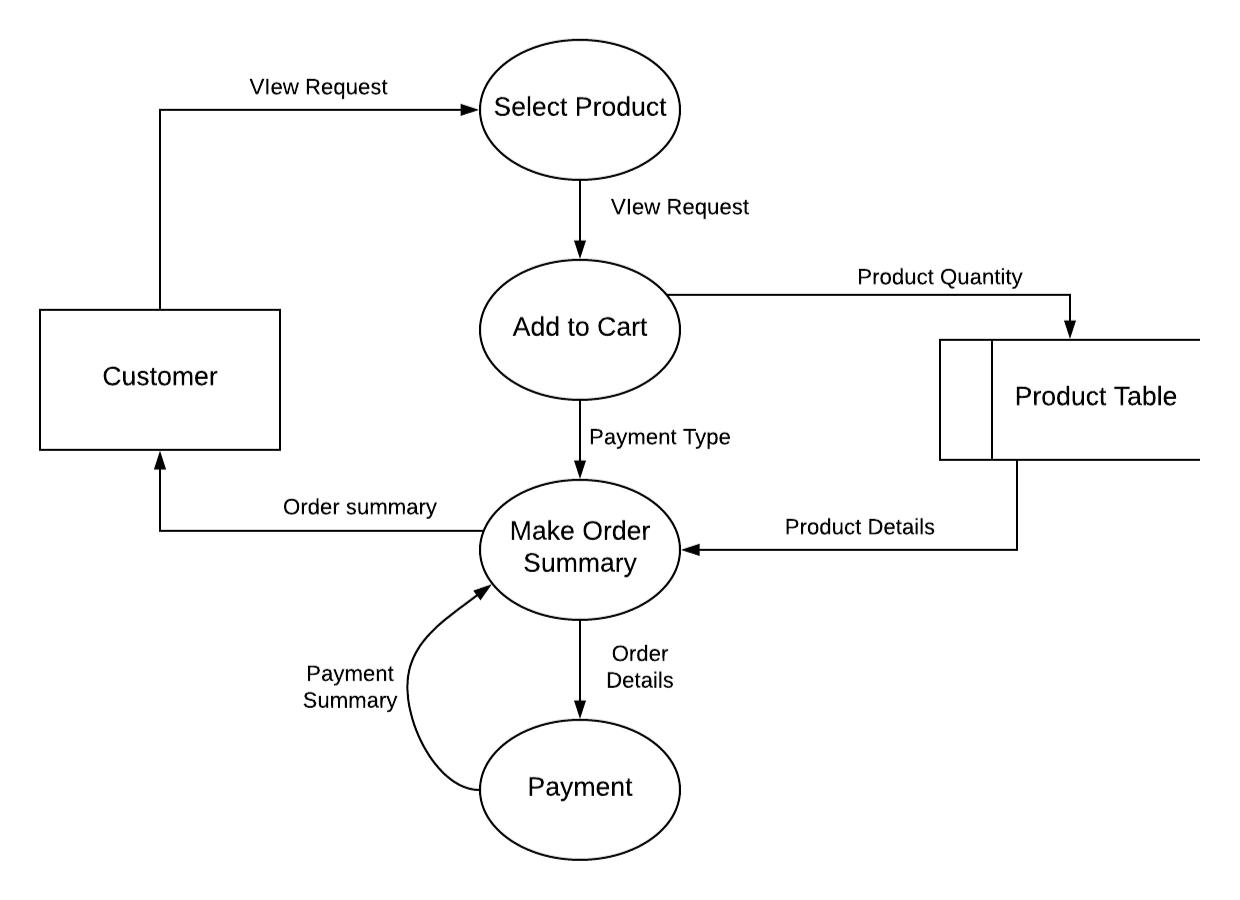
**Figure 58: Level 2.4 for F-commerce based client management system**

124

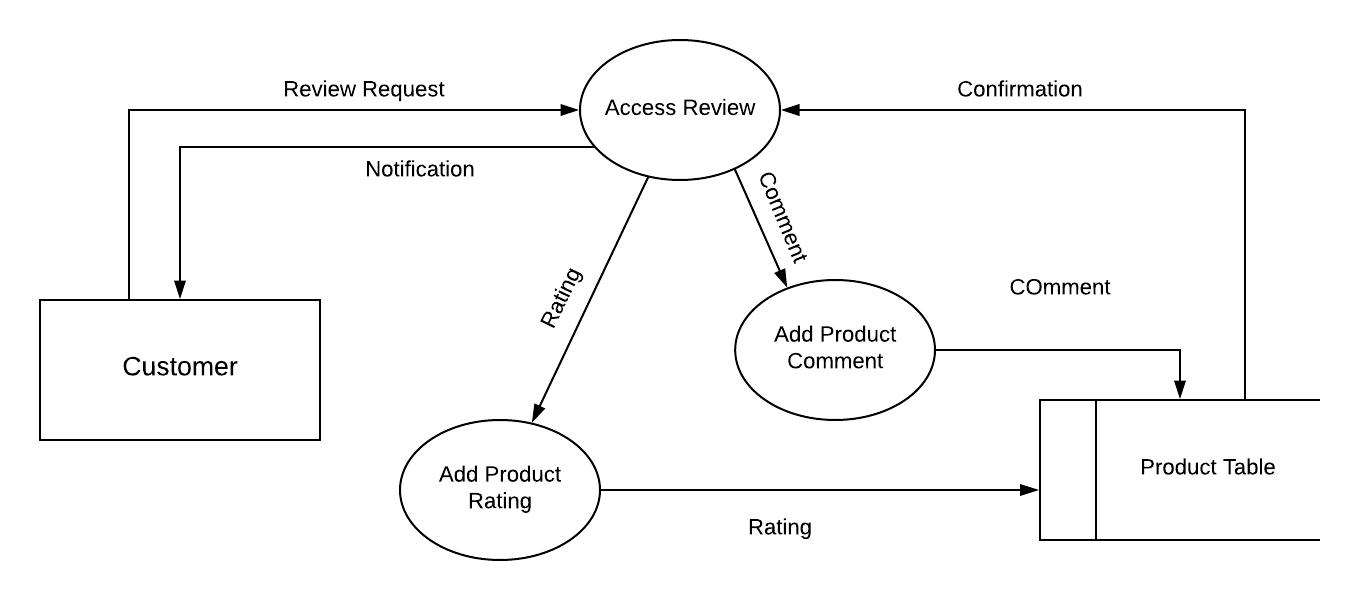


**Figure 59: Level 2.5 for F-commerce based client management system**

125



**Figure 60: Level 2.6 for F-commerce based client management system**



**Figure 61: Level 2.7 for F-commerce based client management system**

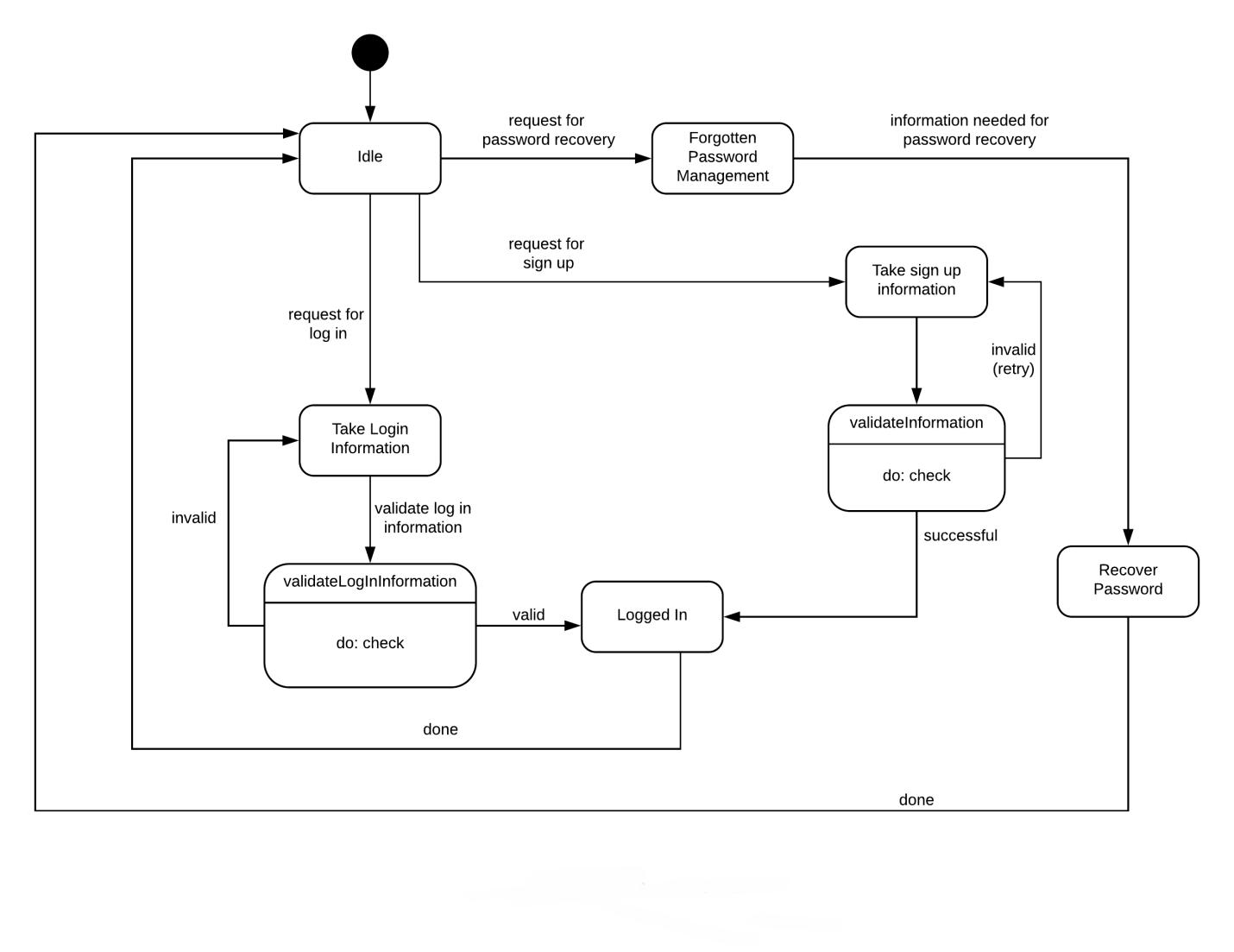
126

**Chapter 8: Behavioral Model**

The behavioral model indicates how software will respond to external events.

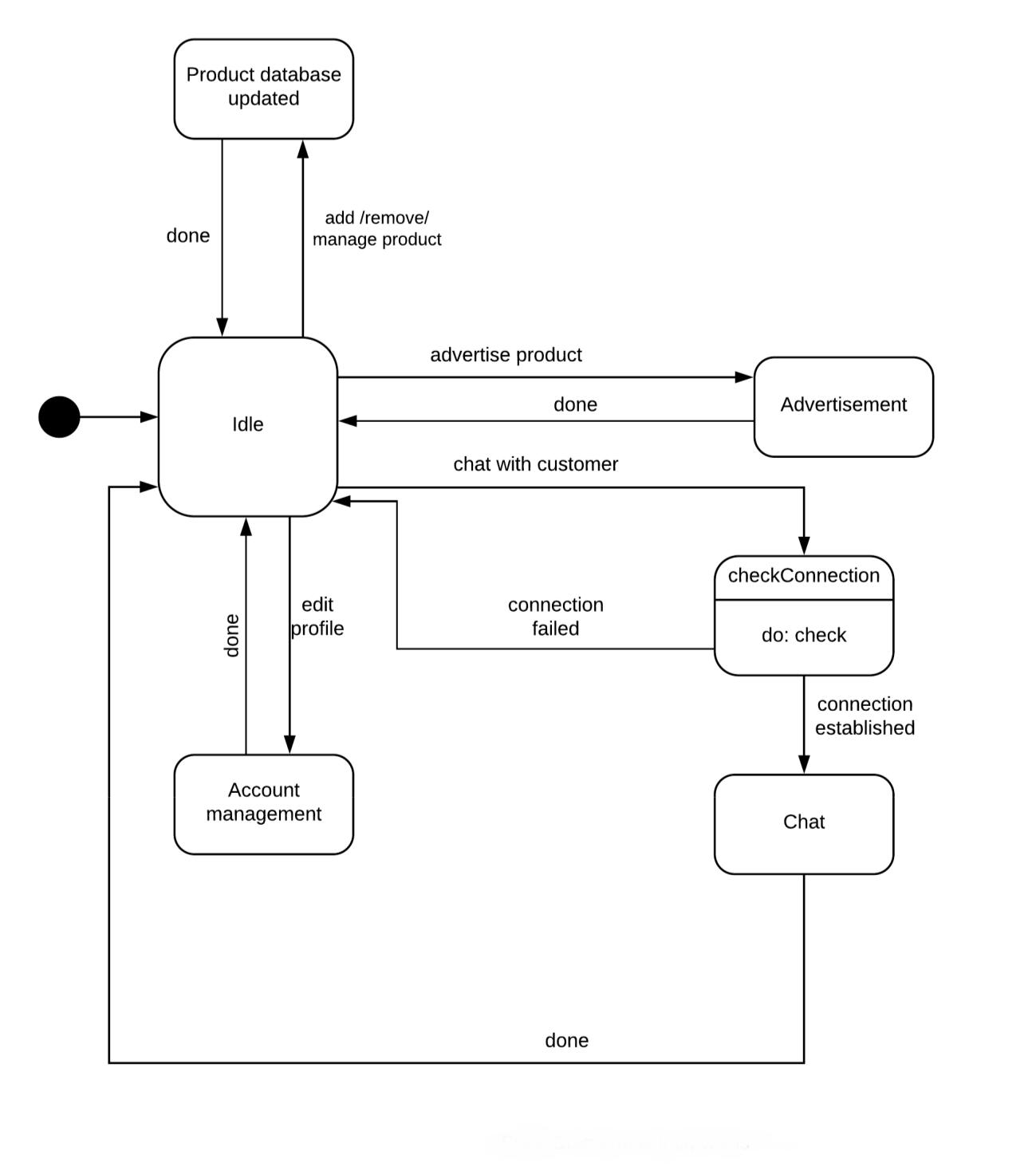
**8.1 State Diagram**

State diagram represents active states for each class the events (triggers).



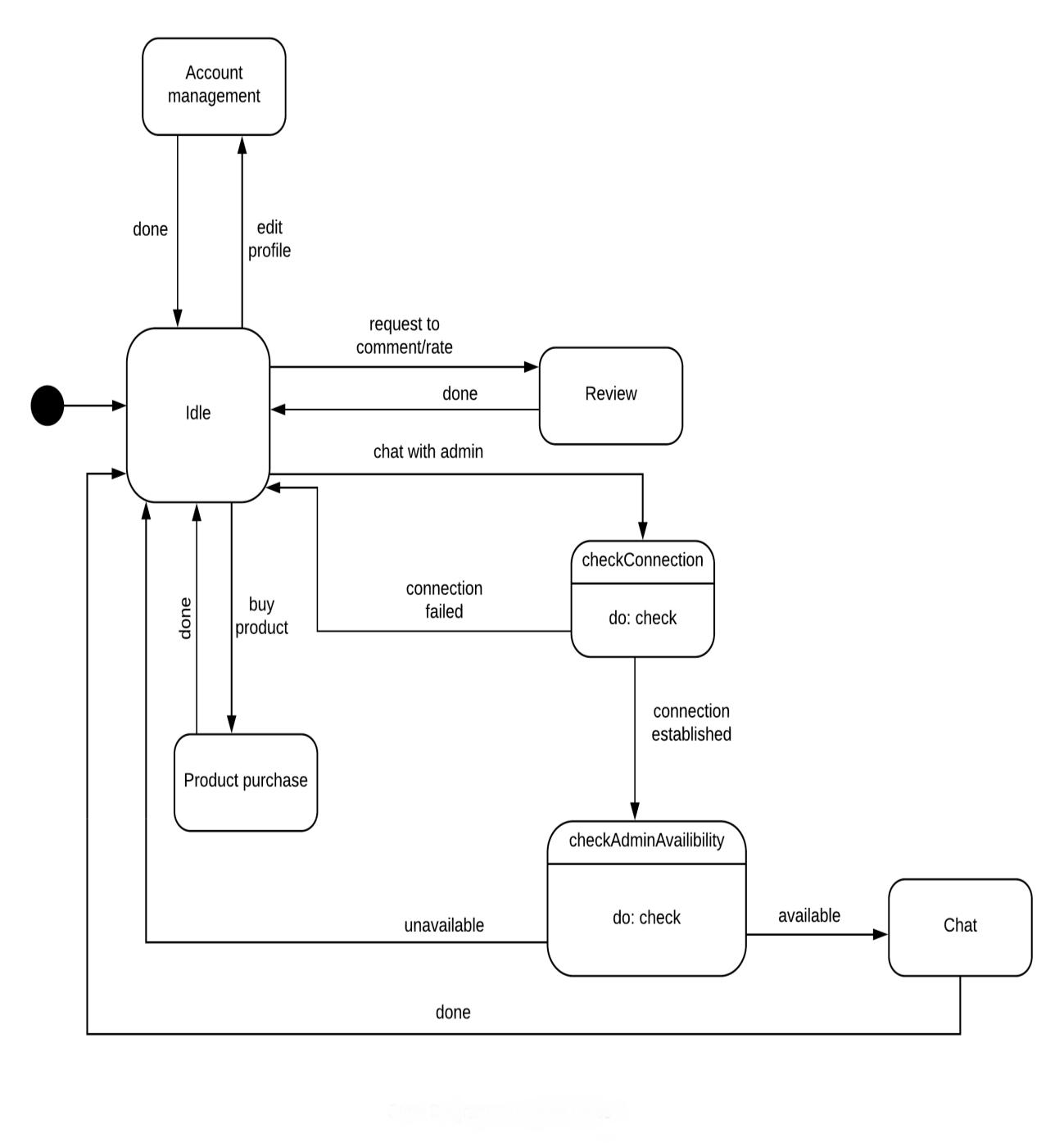
**Figure 62: State Diagram (Account Class)**

127



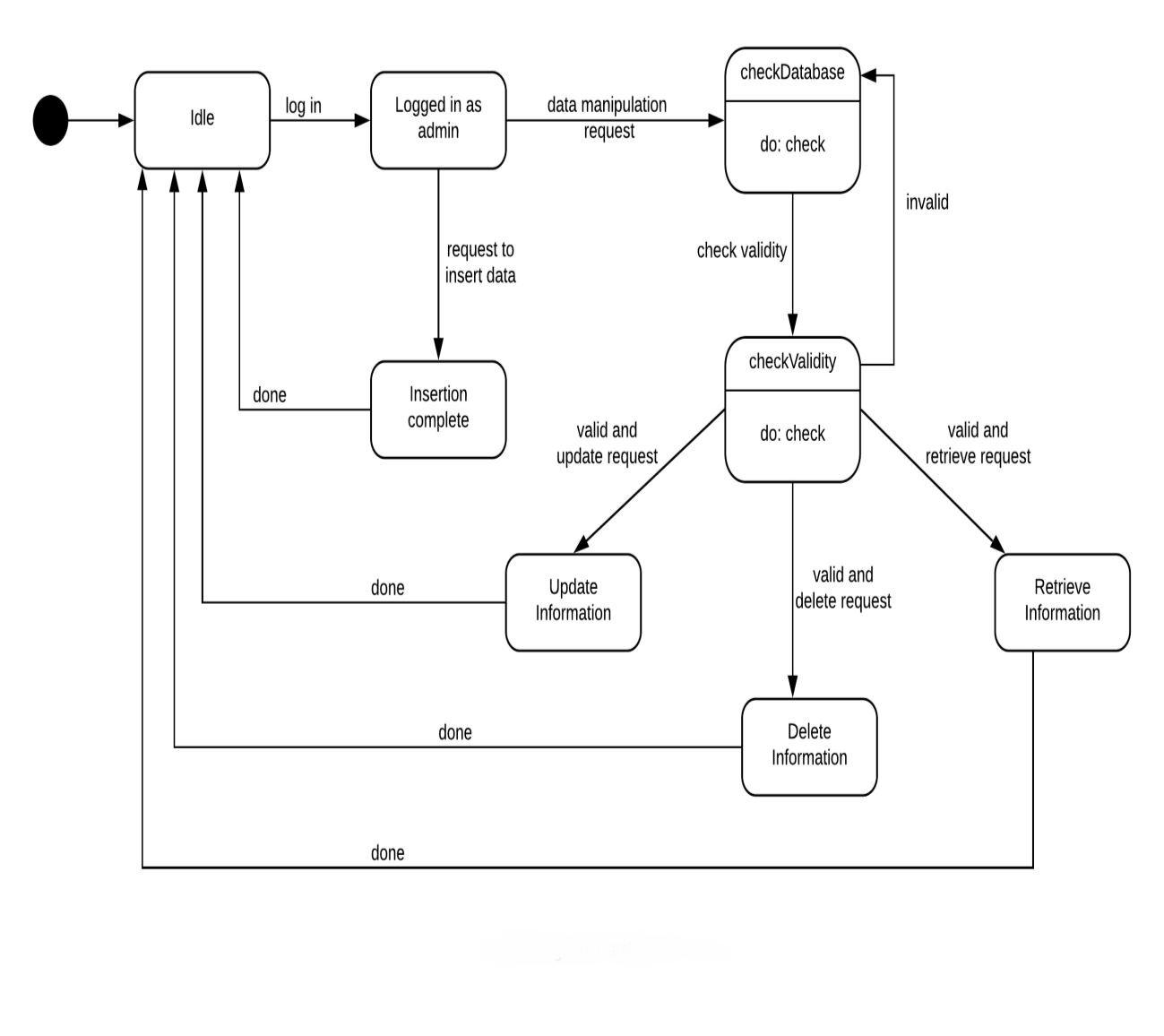
**Figure 63: State Diagram (Admin Class)**

128



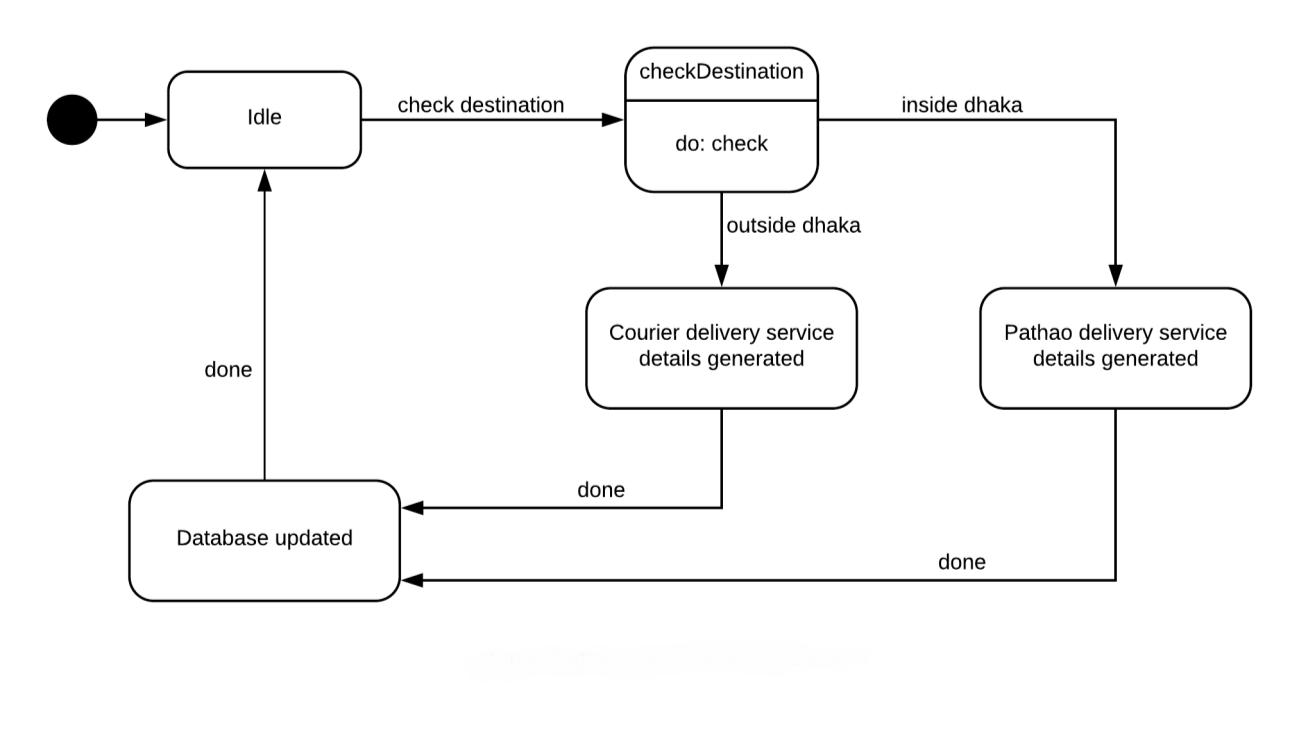
**Figure 64: State Diagram (Customer Class)**

129

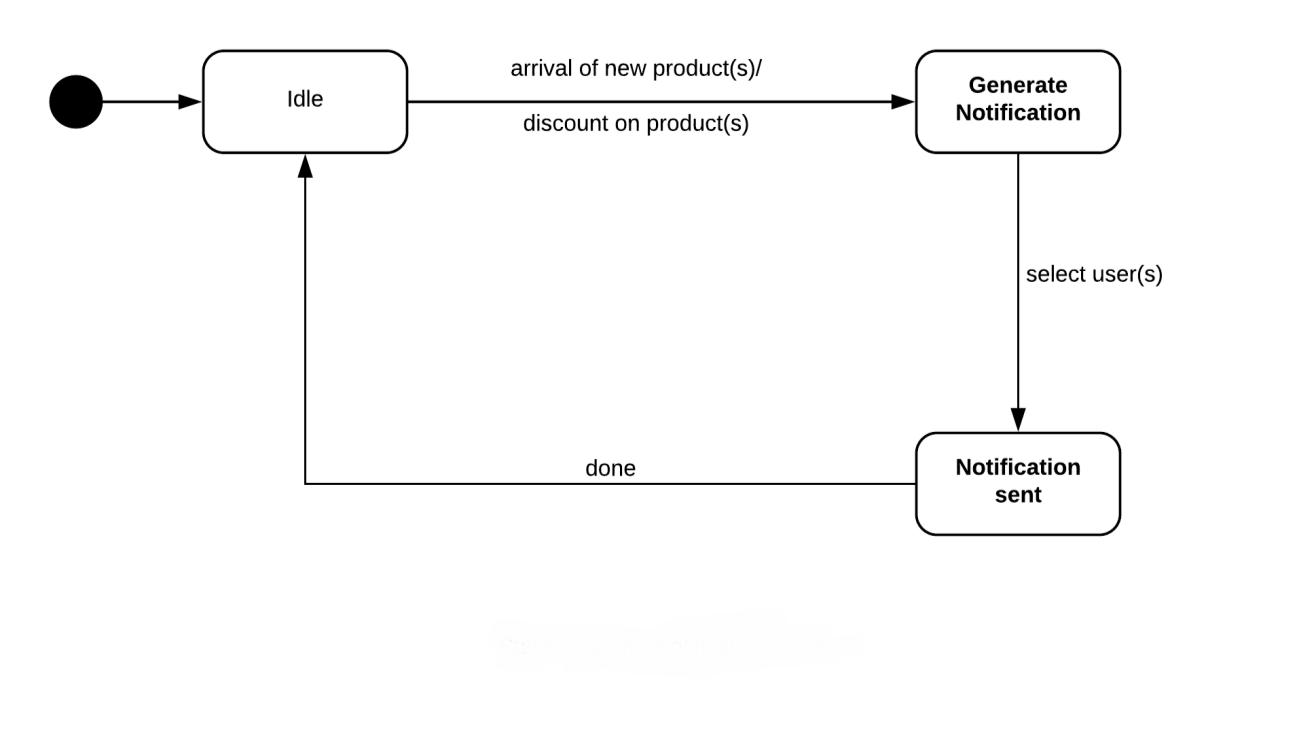


**Figure 65: State Diagram (Database Class)**

130

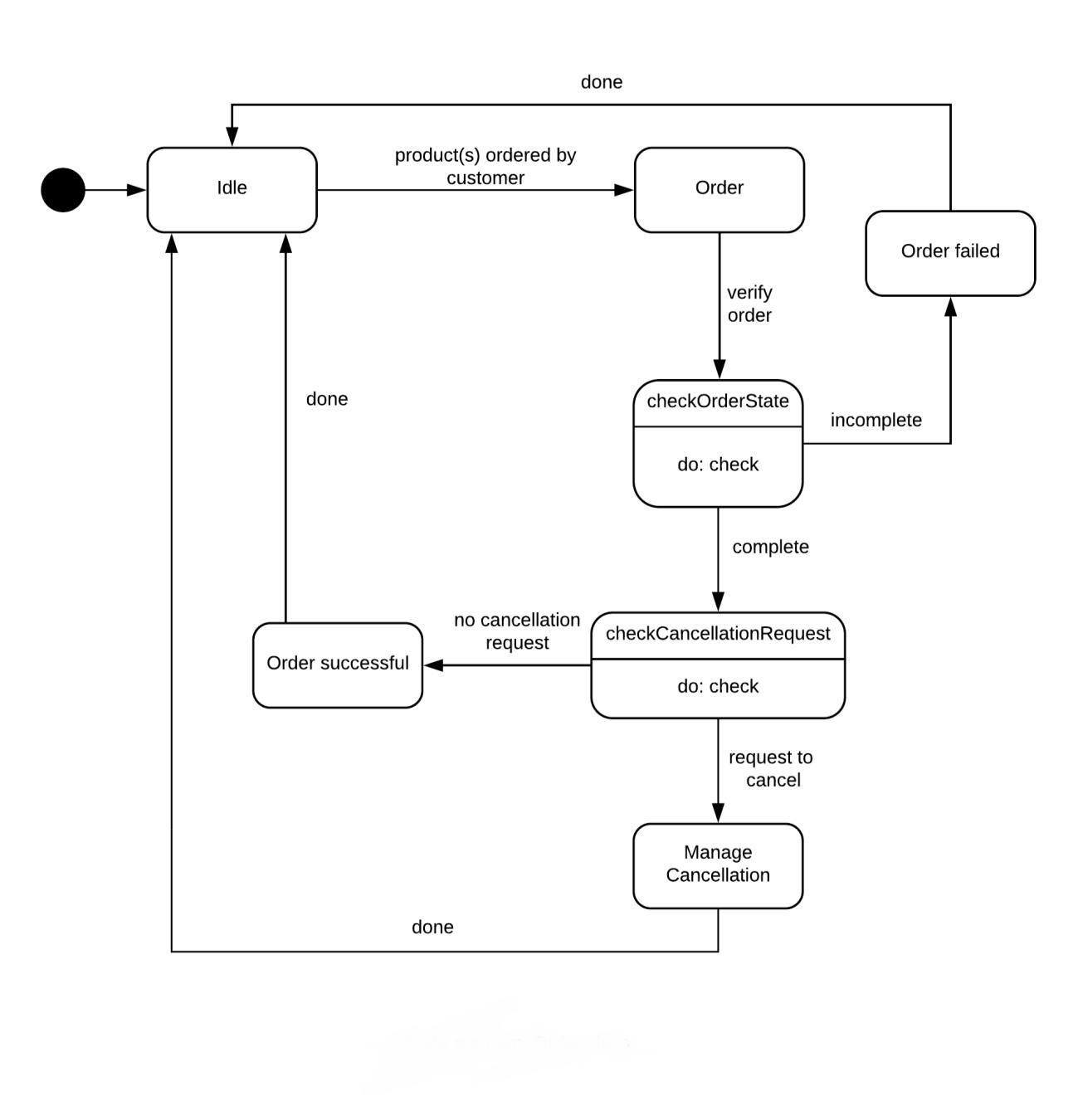


**Figure 66: State Diagram (Delivery Service Class)**



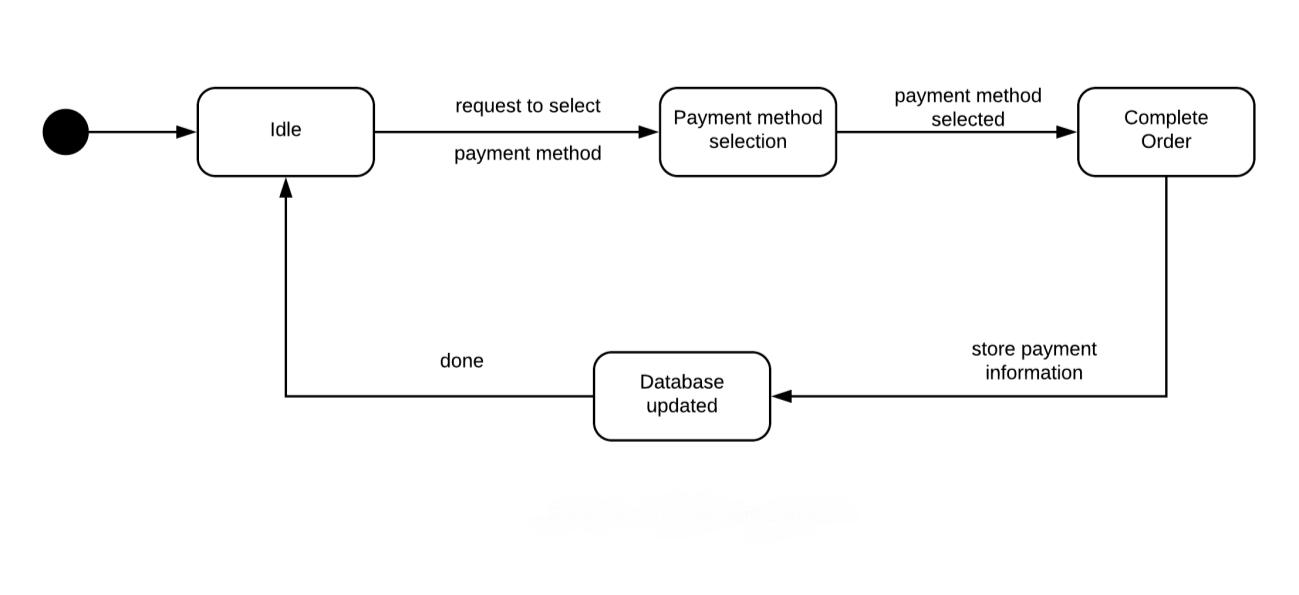
**Figure 67: State Diagram (Notification Class)**

131

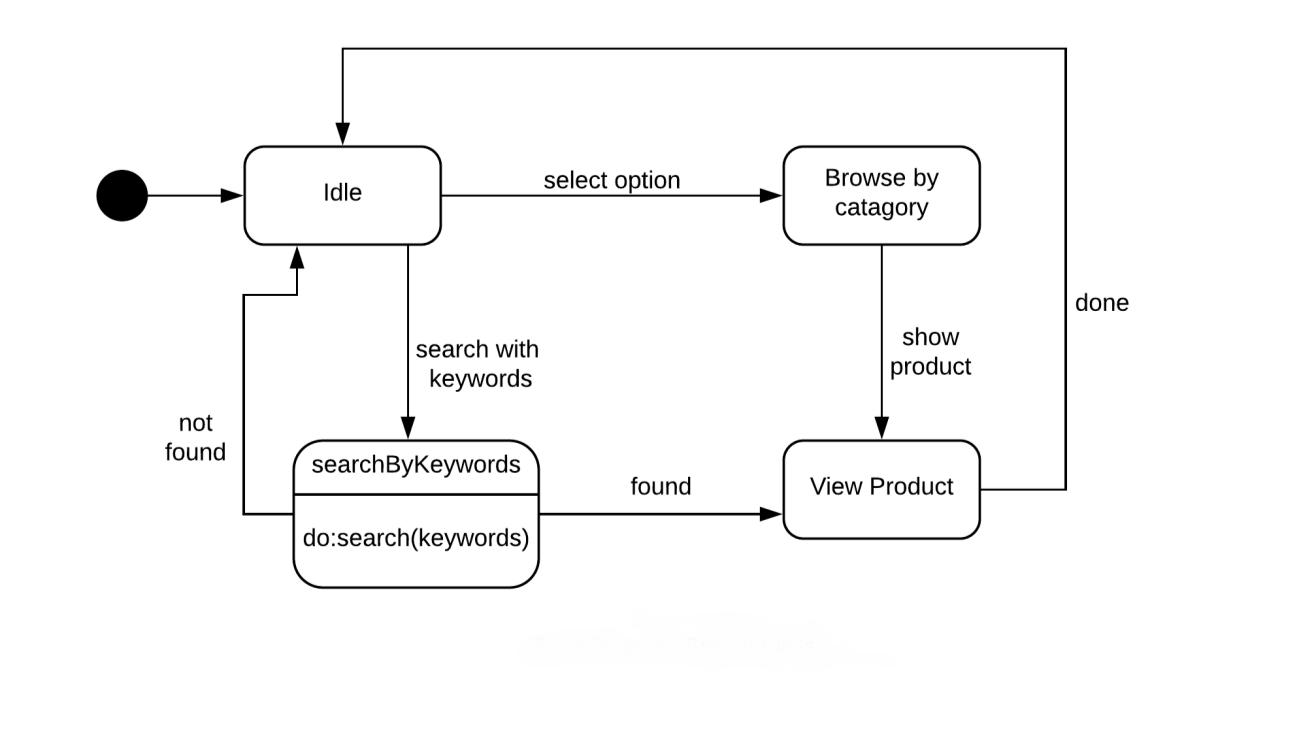


**Figure 68: State Diagram (Order Class)**

132

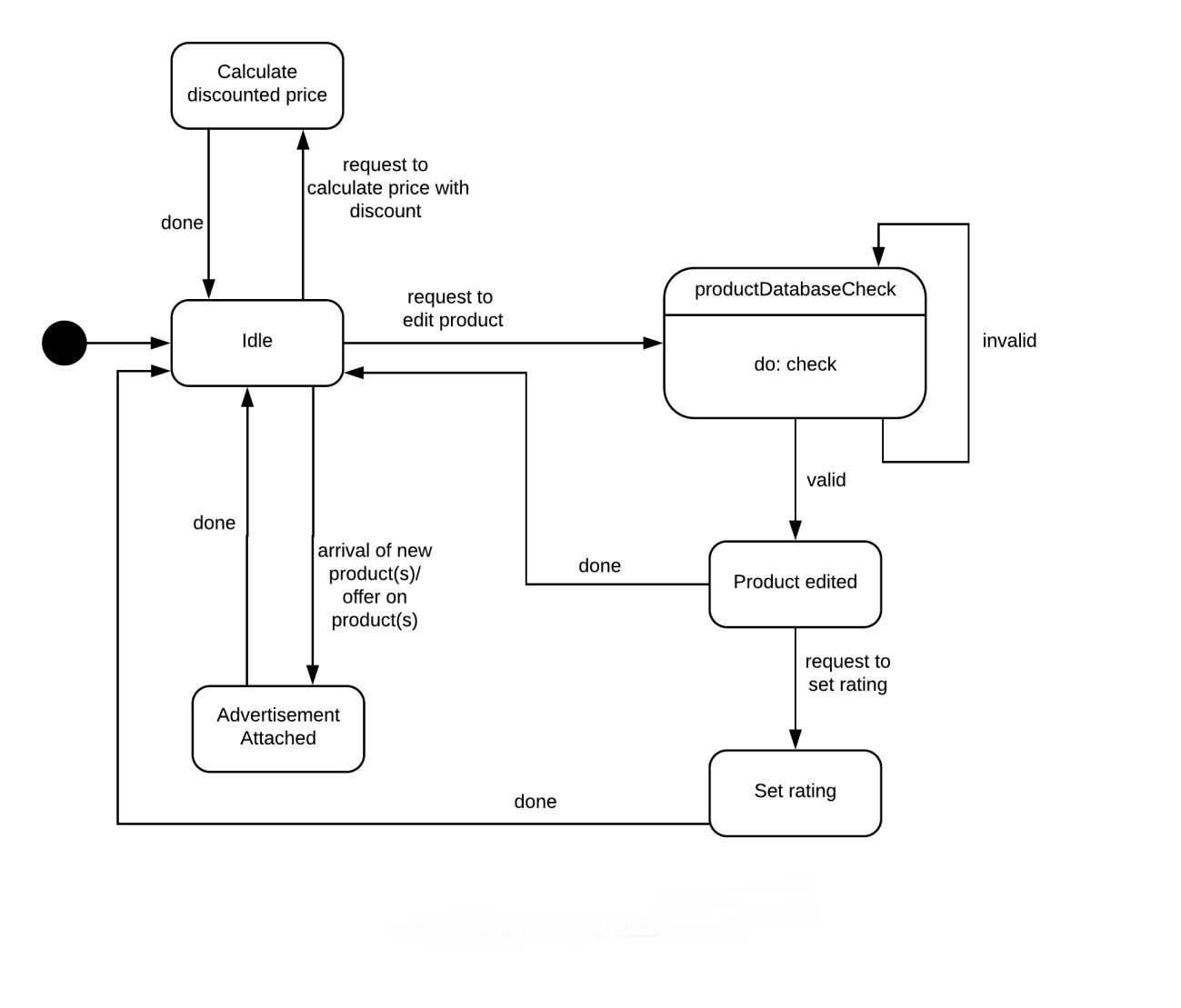


**Figure 69: State Diagram (Payment Class)**



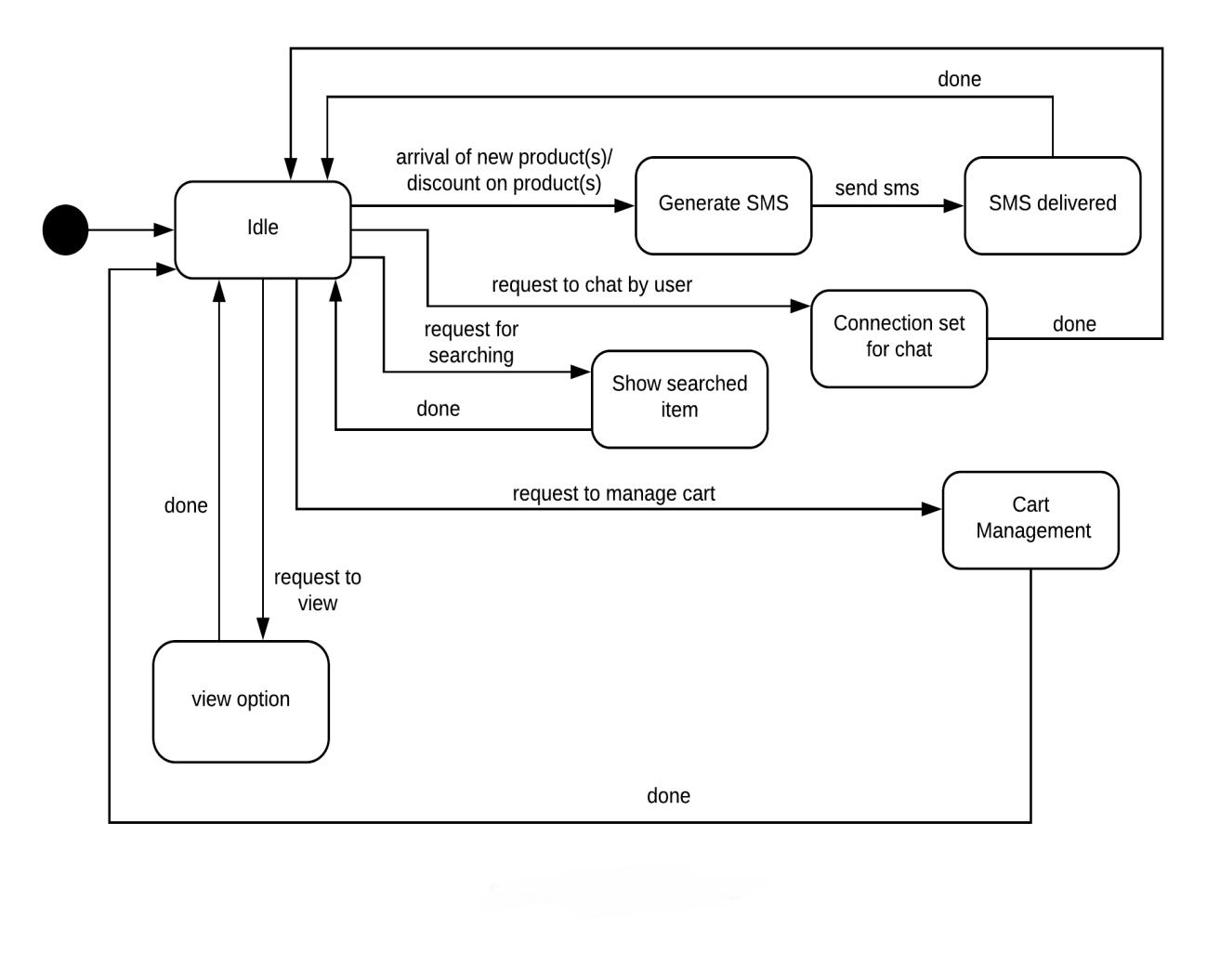
**Figure 70: State Diagram (Search Class)**

133



**Figure 71: State Diagram (Product Class)**

134

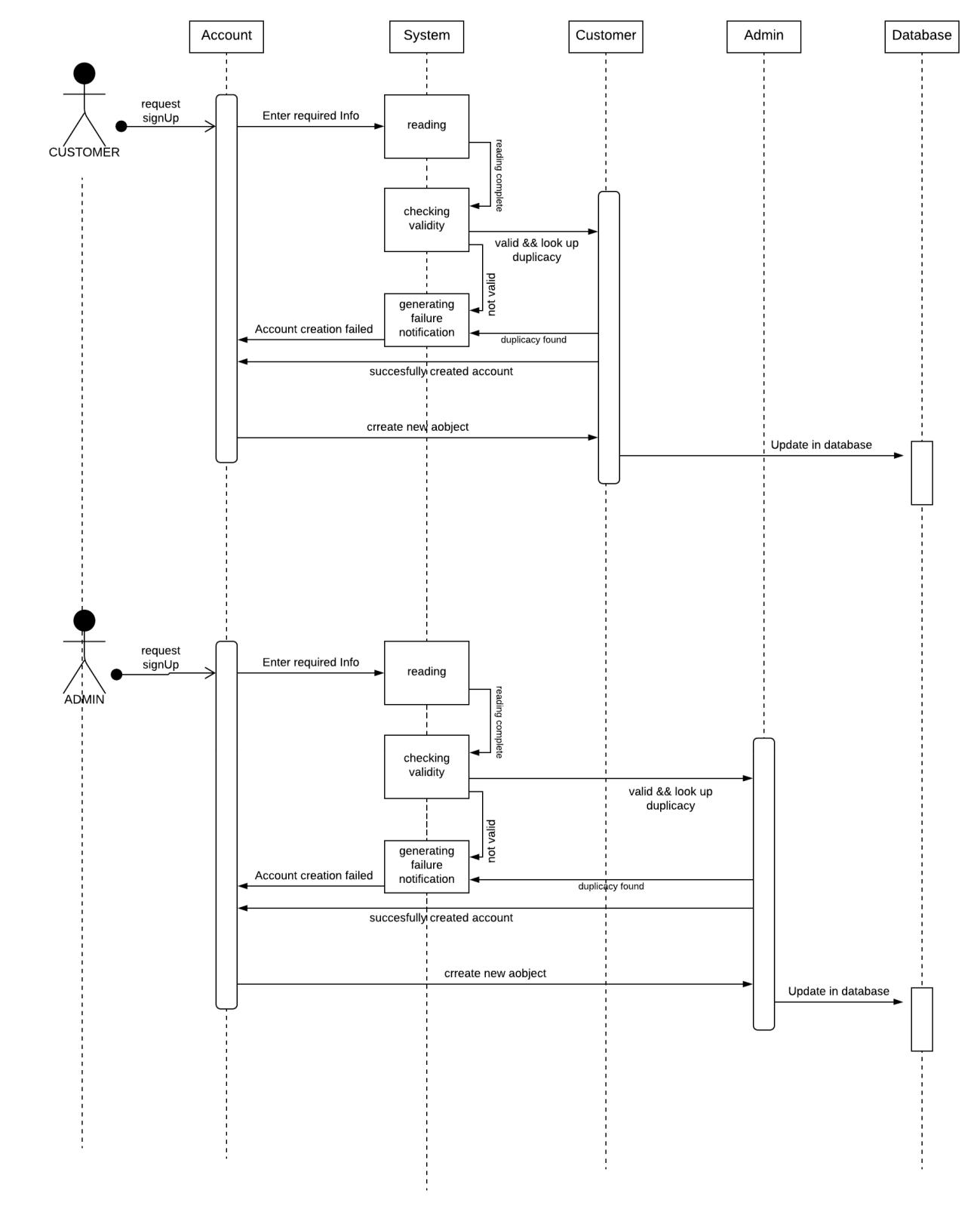


**Figure 72: State Diagram (System Class)**

135

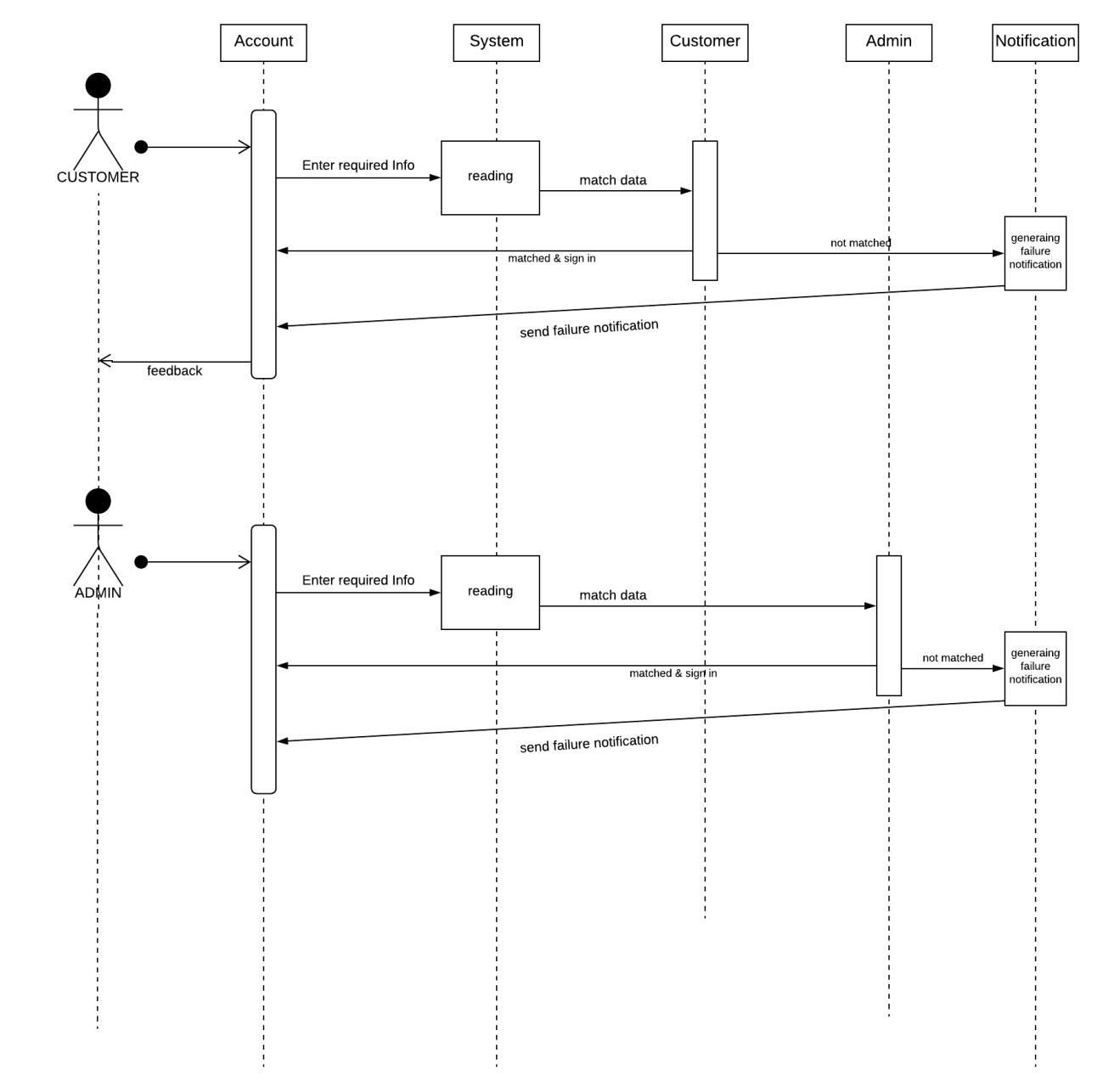
**8.2 Sequence Diagram**

Sequence diagram indicates how events cause transitions from object to object.



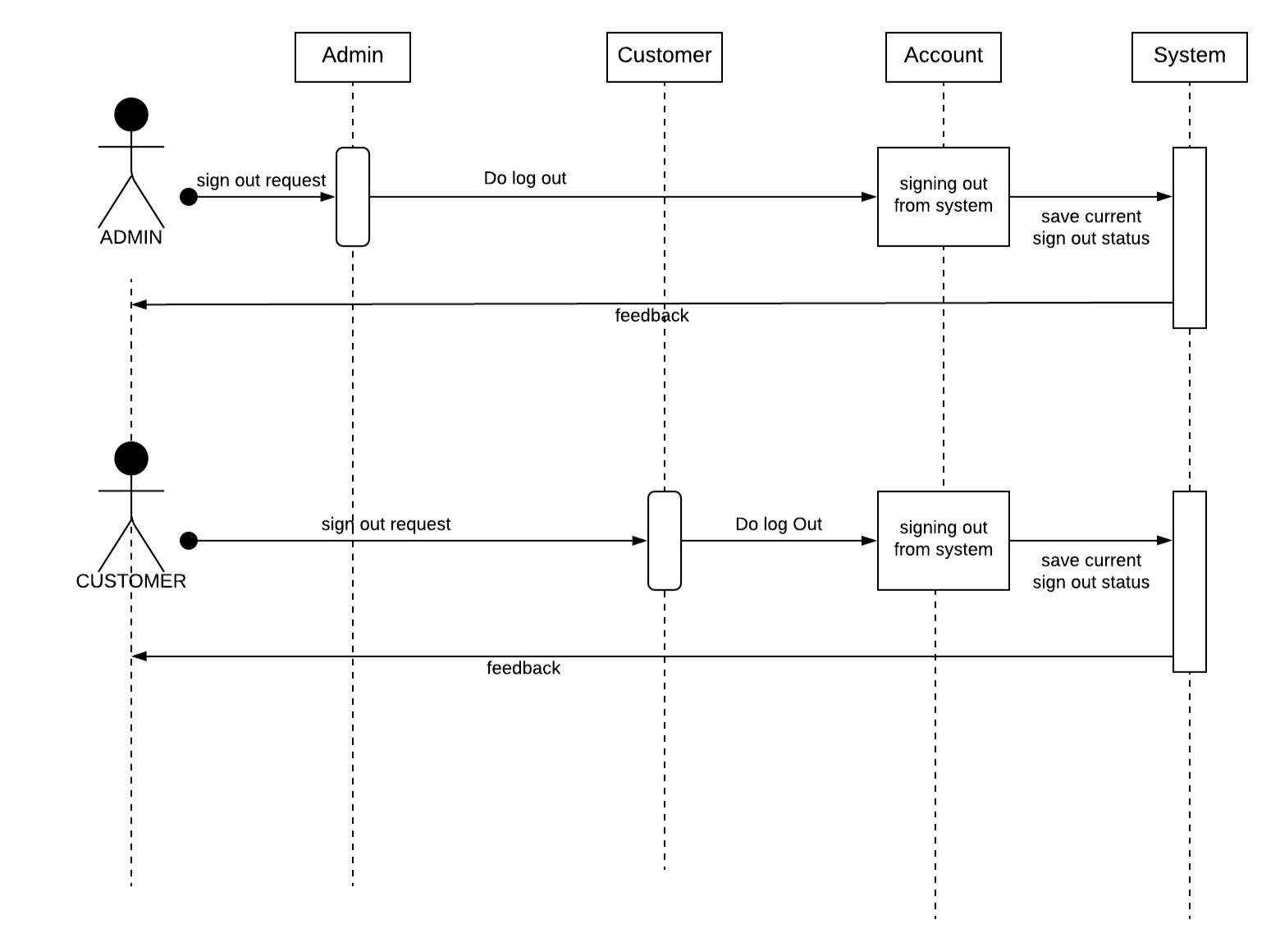
**Figure 73: Sequence diagram (Sign Up)**

136



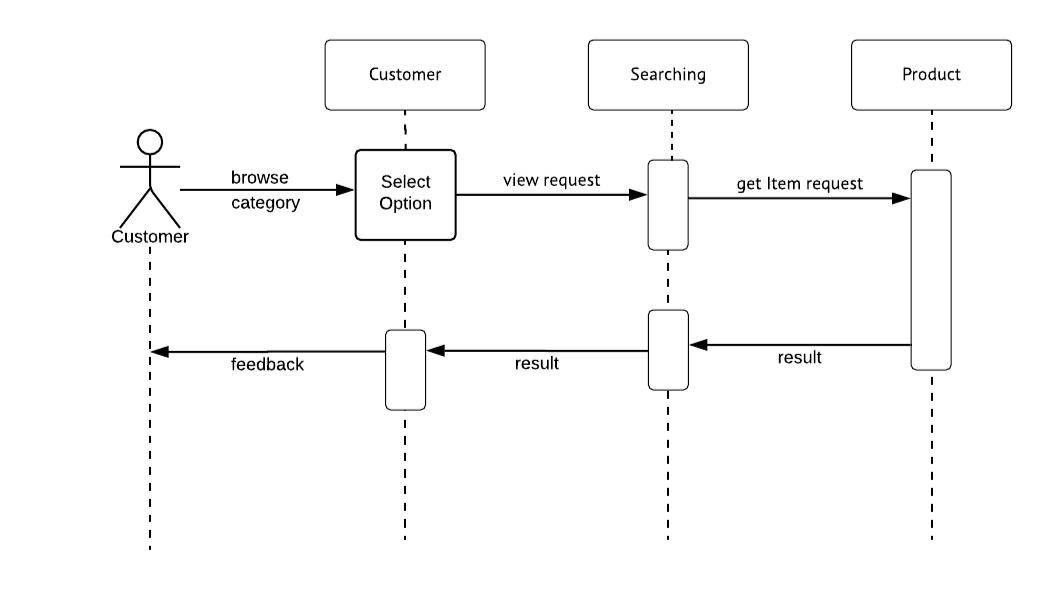
**Figure 74: Sequence diagram (Sign In)**

137

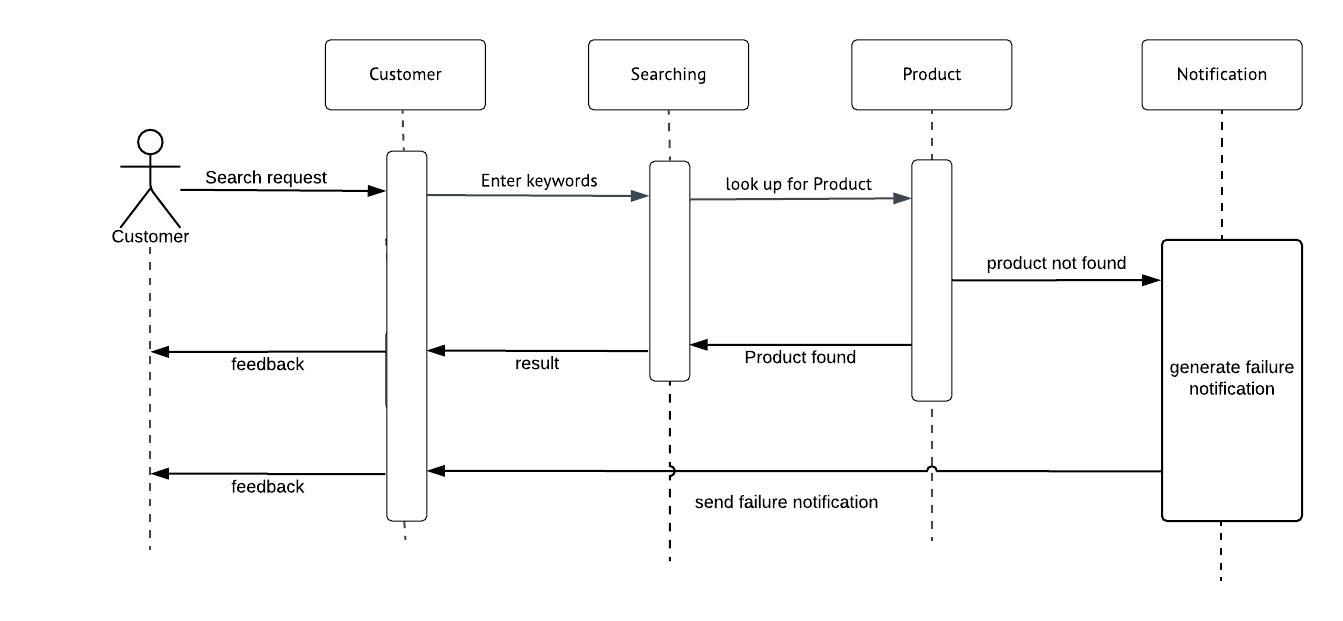


**Figure 75: Sequence diagram (Sign Out)**

138

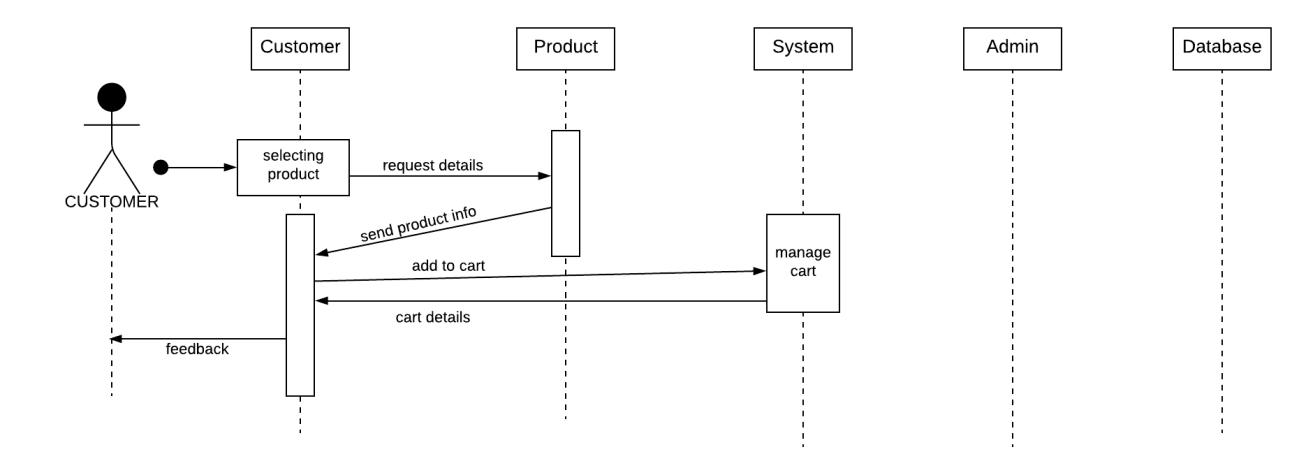


**Figure 76: Sequence diagram (Browse Category)**

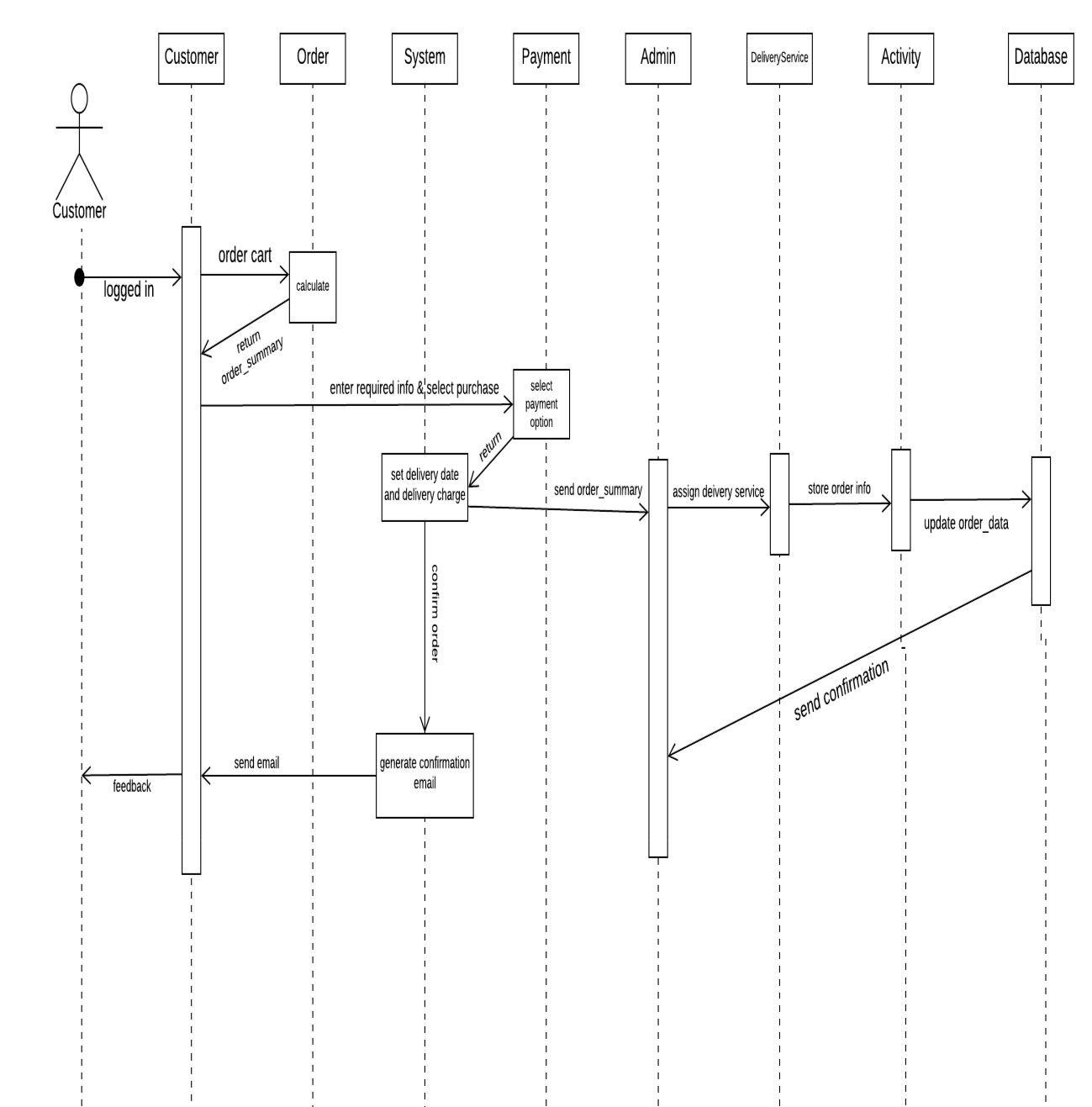


**Figure 77: Sequence diagram (Search)**

139

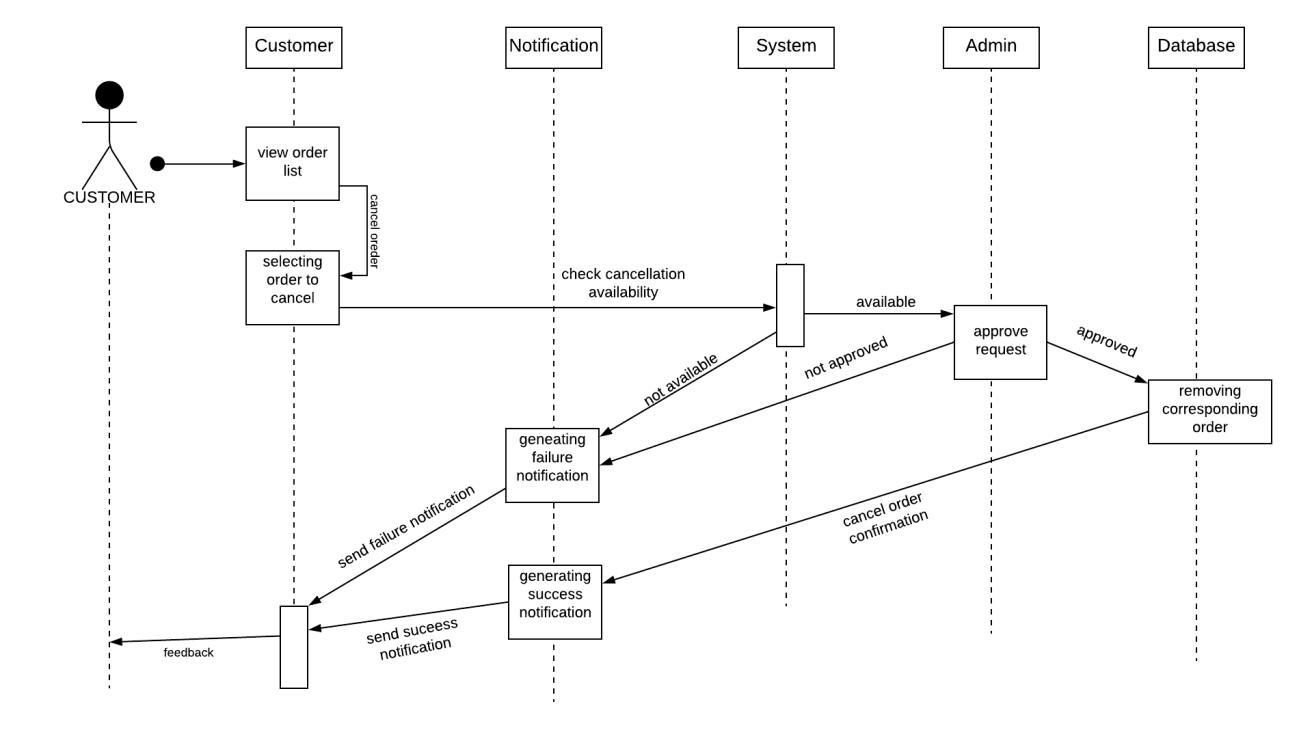


**Figure 78: Sequence diagram (Add to Cart)**

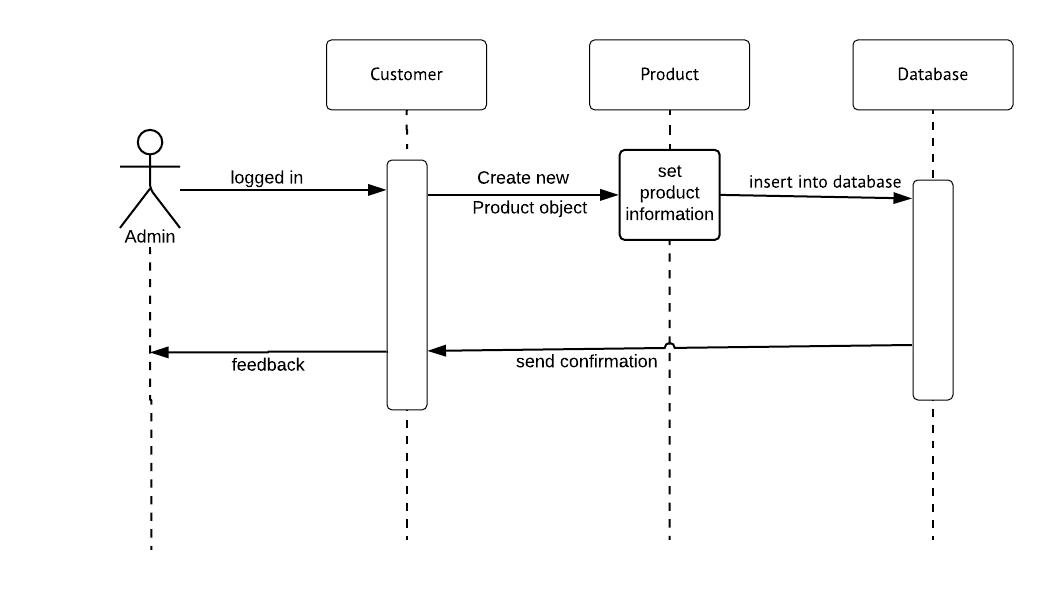


**Figure 79: Sequence diagram (Purchase)**

140

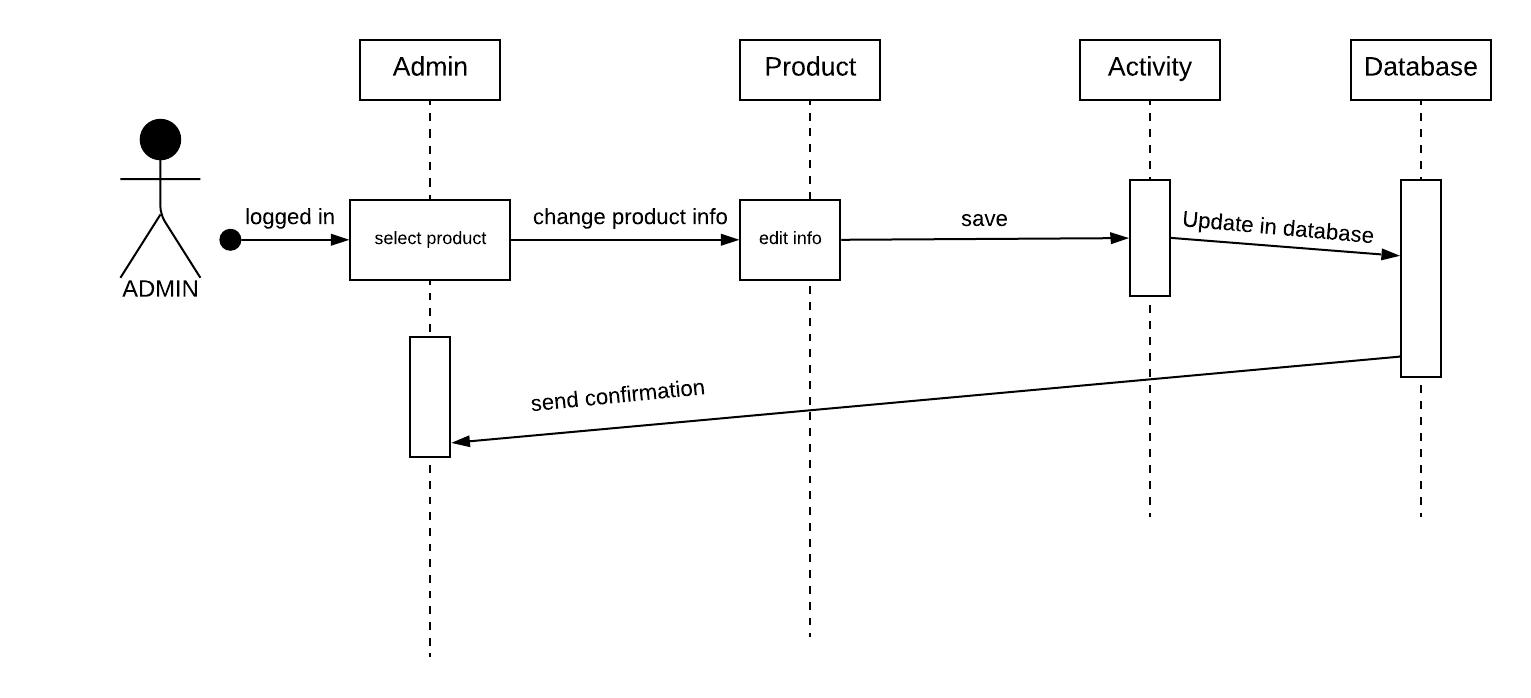


**Figure 80: Sequence diagram (Cancel Order)**

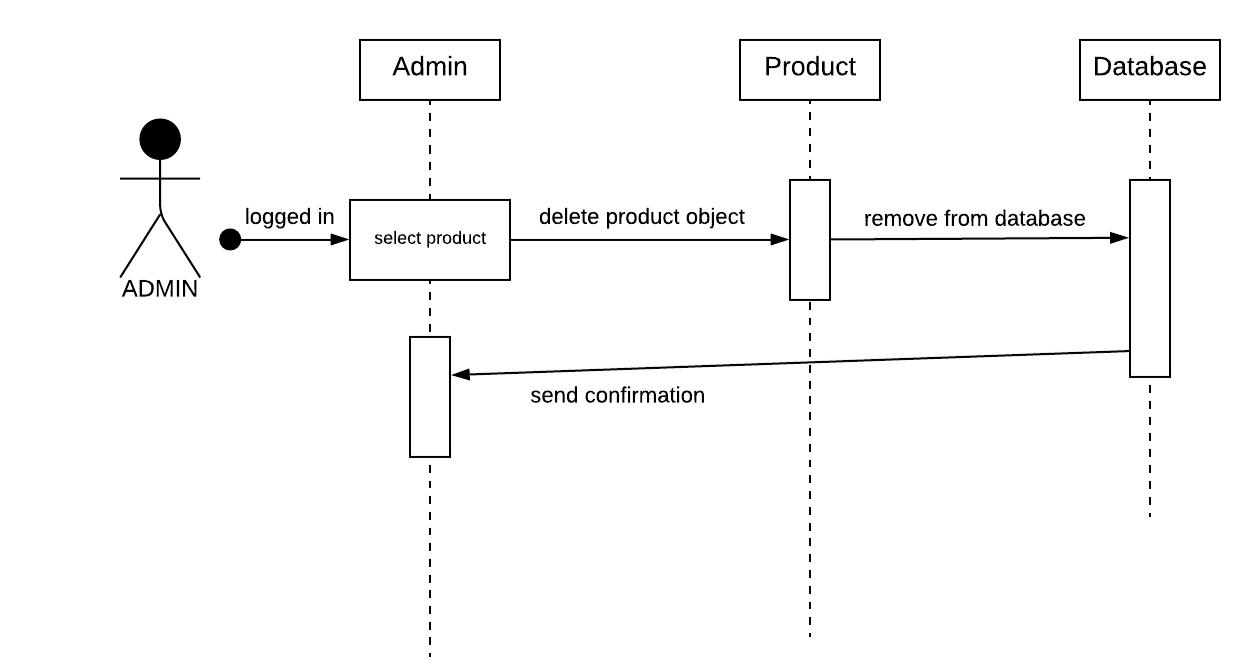


**Figure 81: Sequence diagram (Add Product)**

141

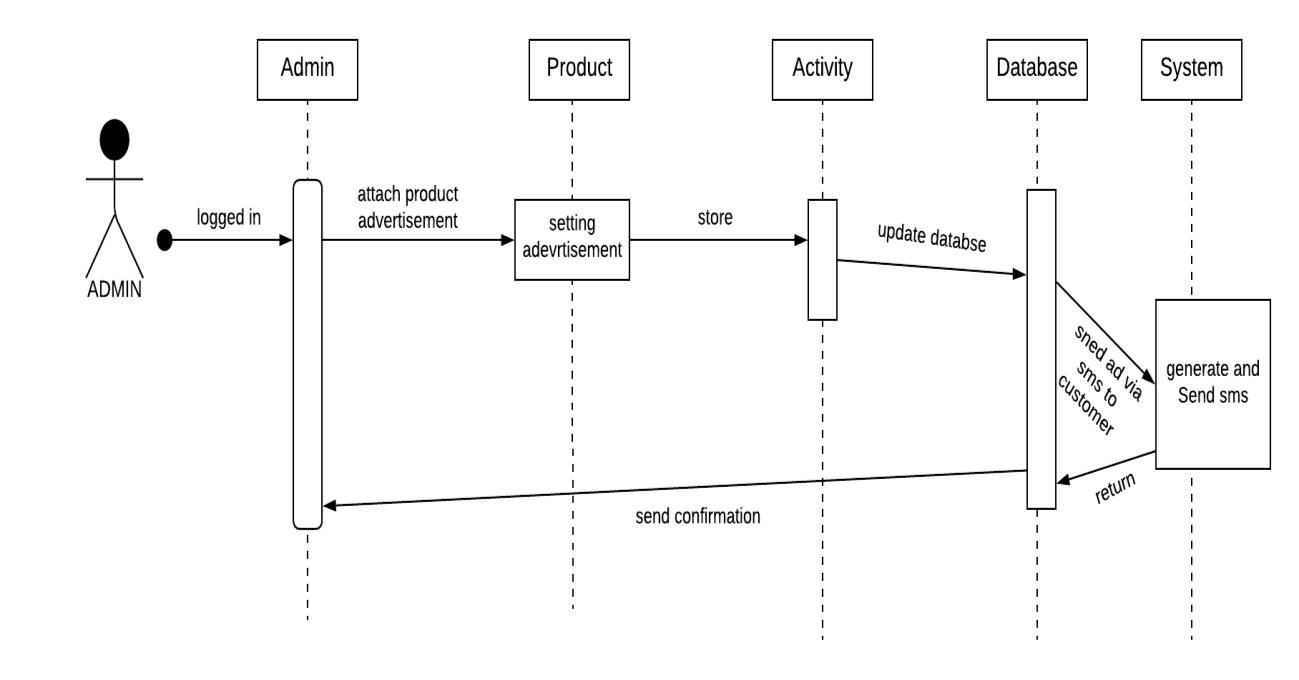


**Figure 82: Sequence diagram (Edit Product)**

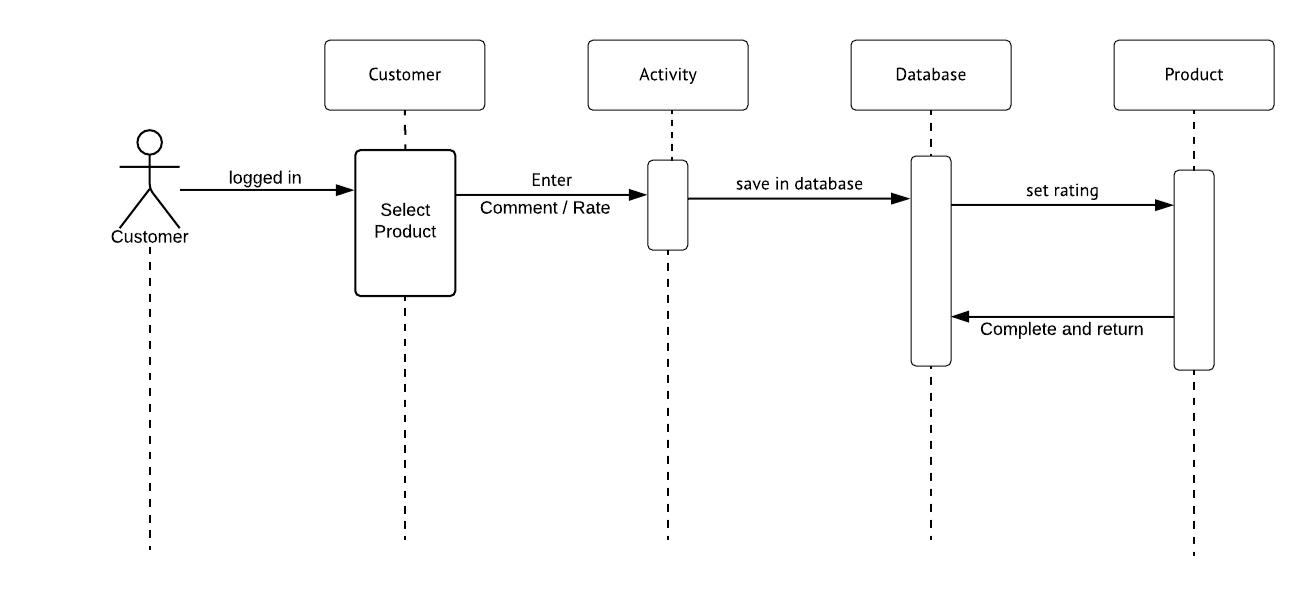


**Figure 83: Sequence diagram (Remove Product)**

142

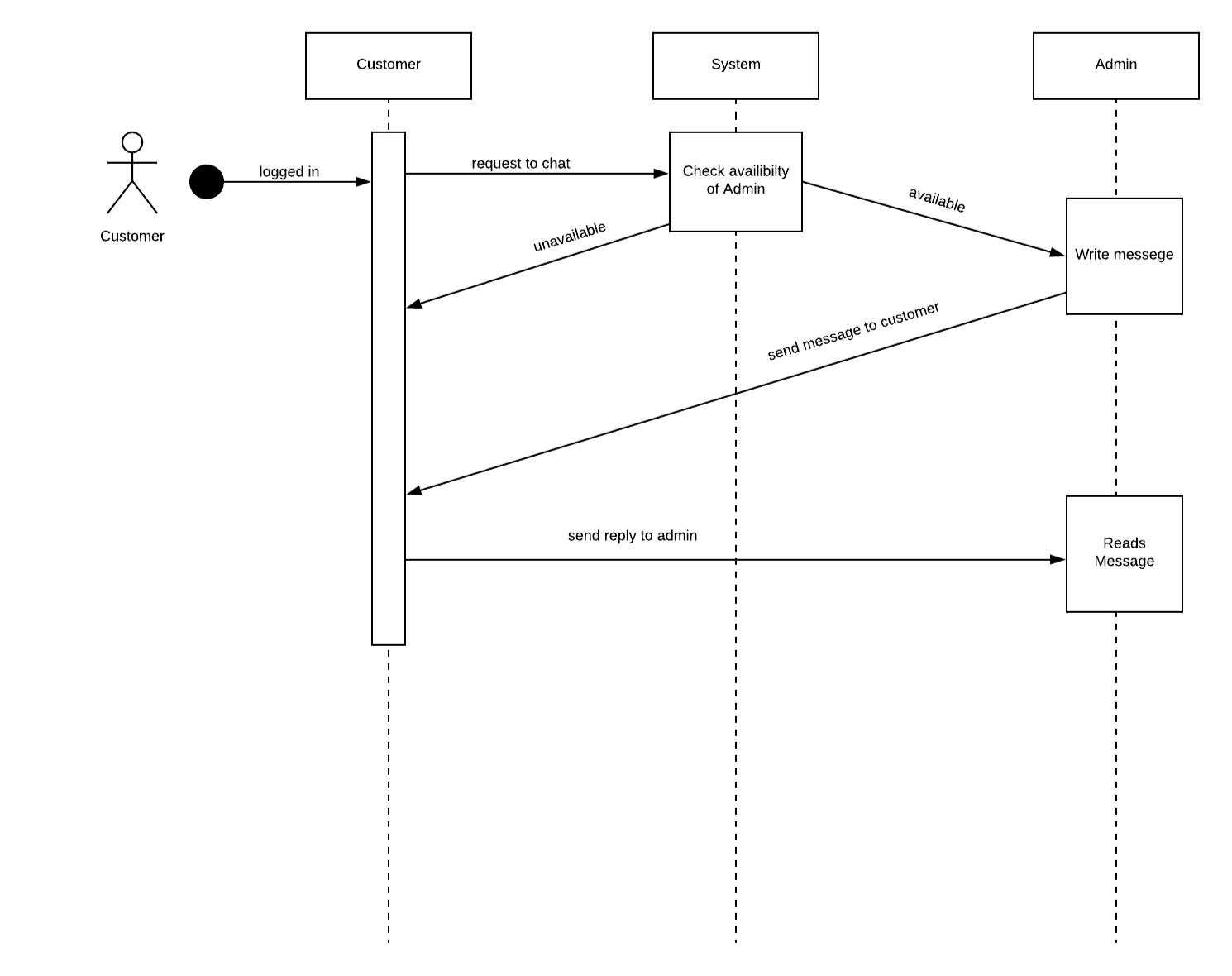


**Figure 84: Sequence diagram (Manage Product Advertisement)**



**Figure 85: Sequence diagram (Review and Comment Product)**

143



**Figure 86: Sequence diagram (Chat)**

144



**Figure 87: Sequence diagram (Email)**

145

**Chapter 9: Conclusion**

From this SRS report on F-commerce Based Client Management System, the readers will get a clear and easy view of the overall system of management system of a Facebook based online shop. This SRS document can be used effectively to maintain the software development cycle. It will be very easy to conduct the whole project using SRS. Hopefully, this document can also help the junior BSSE students. We tried best to remove all dependencies and make an effective and fully designed SRS

146

**References**

1. Pressman, Roger S. Software Engineering: A practitioner’s Approach (7th Edition)

**Appendix**

**Meeting Schedule**

**1.**

**Date:** 22.07.2018

**Location:** IIT, DU

**Discussion:** Preparing the steps for stakeholder identification

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

2.

**Date:** 25.07.2018

**Location:** IIT, DU

**Discussion:** Discussing first visit with the owner of the facebook page

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

BSSE0902

BSSE0924

BSSSE0925

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0926

BSSE0942

3.

**Date:** 29.07.2018

**Location:** IIT, DU

**Discussion:** Discussing visits with the owner of the facebook page

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

4.

**Date:** 01.08.2018

**Location:** IIT, DU

**Discussion:** Identifying Viewpoints of Stakeholders

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

5.

**Date:** 04.08.2018

**Location:**

IIT, DU

1

**Discussion:**

Defining QFD

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

6.

**Date:** 05.08.2018

**Location:** IIT, DU

**Discussion:** Defining QFD

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

7.

**Date:** 08.08.2018

**Location:** IIT, DU

**Discussion:** Preparing the usage scenario

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

BSSE0902

BSSE0924

BSSSE0925

2

A.T.M. Fazlay Rabbi

Niraj Chaudhary

8.

**Date:** 12.08.2018

**Location:** IIT, DU

**Discussion:** Correcting the usage scenario

BSSE0926

BSSE0942

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

9.

**Date:** 18.08.2018

**Location:** IIT, DU

**Discussion:** Correcting the usage scenario

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

10.

**Date:** 19.08.2018

3

**Location:** IIT, DU

**Discussion:** Preparing activity diagram

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

11.

**Date:** 01.09.2018

**Location:** IIT, DU

**Discussion:** Correcting Activity Diagram

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

12.

**Date:** 05.09.2018

**Location:** IIT, DU

**Discussion:** Preparing swimlane diagram

**Members:**

4

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

13.

**Date:** 19.09.2018

**Location:** IIT, DU

**Discussion:** Discussing about data modeling

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

14.

**Date:** 23.09.2018

**Location:** IIT, DU

**Discussion:** Preparing data modeling

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

BSSE0902

BSSE0924

BSSSE0925

5

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0926

BSSE0942

15.

**Date:** 26.09.2018

**Location:** IIT, DU

**Discussion:** Preparing the ER diagram

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

16.

**Date:**

**Location:**

**Discussion:**

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

6

17.

**Date:** 30.09.2018

**Location:** IIT, DU

**Discussion:** Discussing about class based modeling

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

18.

**Date:** 03.10.2018

**Location:** IIT, DU

**Discussion:** Defining the data object relation

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

19.

**Date:** 06.10.2018

**Location:** IIT, DU

7

**Discussion:** Preparing the data schema

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

20.

**Date:** 07.10.2018

**Location:** IIT,DU

**Discussion:** Discussing about class based modeling

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

21.

**Date:** 10.10.2018

**Location:** IIT, DU

**Discussion:** Preparing class based modeling

**Members:**

S.M. Shahriyar

BSSE0902

8

Saif Kamal Chowdhury Anika Tabassum A.T.M. Fazlay Rabbi Niraj Chaudhary

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

22.

**Date:** 28.10.2018

**Location:** IIT,DU

**Discussion:** Preparing class card

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

23.

**Date:** 31.10.2018

**Location:** IIT, DU

**Discussion:** Preparing class card

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

9

Niraj Chaudhary BSSE0942

24.

**Date:** 03.11.2018

**Location:** IIT, DU

**Discussion:** Correcting Class Card

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

25.

**Date:** 04.11.2018

**Location:** IIT, DU

**Discussion:** Preparing data flow modeling

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

26.

10

**Date:** 07.11.2018

**Location:** IIT, DU

**Discussion:** Preparing state transition diagram

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

27.

**Date:** 10.11.2018

**Location:** IIT,DU

**Discussion:** Preparing state transition diagram

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

28.

**Date:** 11.11.2018

**Location:** IIT, DU

**Discussion:** Preparing Sequence diagram

11

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

29.

**Date:** 14. 11. 2018

**Location:** IIT, DU

**Discussion:** Discussing about the report

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

Anika Tabassum

A.T.M. Fazlay Rabbi

Niraj Chaudhary

BSSE0902

BSSE0924

BSSSE0925

BSSE0926

BSSE0942

30.

**Date:** 17.11.2018

**Location:** IIT, DU

**Discussion:** Finalizing the report

**Members:**

S.M. Shahriyar

Saif Kamal Chowdhury

BSSE0902

BSSE0924

12

Anika Tabassum

BSSSE0925

A.T.M. Fazlay Rabbi

BSSE0926

Niraj Chaudhary

BSSE0942

13