SE2 Course Project

**Main Report**

for

**Online Shopping Management System**

[name of your system company]

*Submitted to*

*by*

[student name – ID]

October 2020

**Abstract**

Covid-19 pandemic has isolated world to their houses limiting them to fulfil their necessities through online shopping. This software requirement report introduces the secure online shopping management system which will be a secure database management system identifying its key stakeholders, the UML design, the scope of system along with functional and non-functional requirements as well as the system design and implementation outline.

The intended audience of this document are stakeholders who want to expand their business online and wanted to develop a trusted and secure shopping management system with seamless process of payment transfer. This report will also intended for students who want to develop the professional level UML requirement identification and development steps of a software development.

This report includes all phases of UML requirement specification, design and implementation for online shopping management system. The requirement specification phase will include the Entity relationship modeling to Relational Database modelling as well as design and implementation details in different steps.

Keywords*: Online shopping, management system, software development, functional requirements, non-functional requirements*.

**Table of Contents**

1. Chapter 1. Document Software Requirements

1.1 System Overall Description

1.2 System Use Case Diagram

1.3 System Specific Requirements

1.4 List of classes

1.5 Non-Functional Requirements

2. Chapter 2. System Design and Architecture

2.1 Application Architecture

2.2 Inter-Package Dependencies

2.3 Data Decomposition

2.4 Packages Detailed Design

3. Chapter 3. System Implementation and Testing

3.1 Implementation and Coding Standards

3.2 Software Testing

4. References

**Chapter 1. Document Software Requirements for Online Shopping Management System**

**1.1 System Overall Description**

COVID-19 pandemic has led world to new type of shopping behavior, statistics shows that during pandemic the online shopping has raised by many folds. The proposed online shopping management system offers the solution for shopping system for online website. The online shopping modules have stakeholders of user/customer, admin, and database to store information. The system has major modules of user, product they want to purchase, and adding product to cart, add payment card as well as payment for the product.

The online shopping management system will provide security to user for their online payments and how they will be using the system. The major functional requirements for this system include customer and his personal information including address, billing address and product to buy. The other functional requirements include about product and its attributes and cart information and make payment. The non-functional requirements include security, performance, design, quality and reliability.

**1.2 System Use Case Diagram**

The use case diagram for user/customer entity for online shopping management system is shown in figure 1.

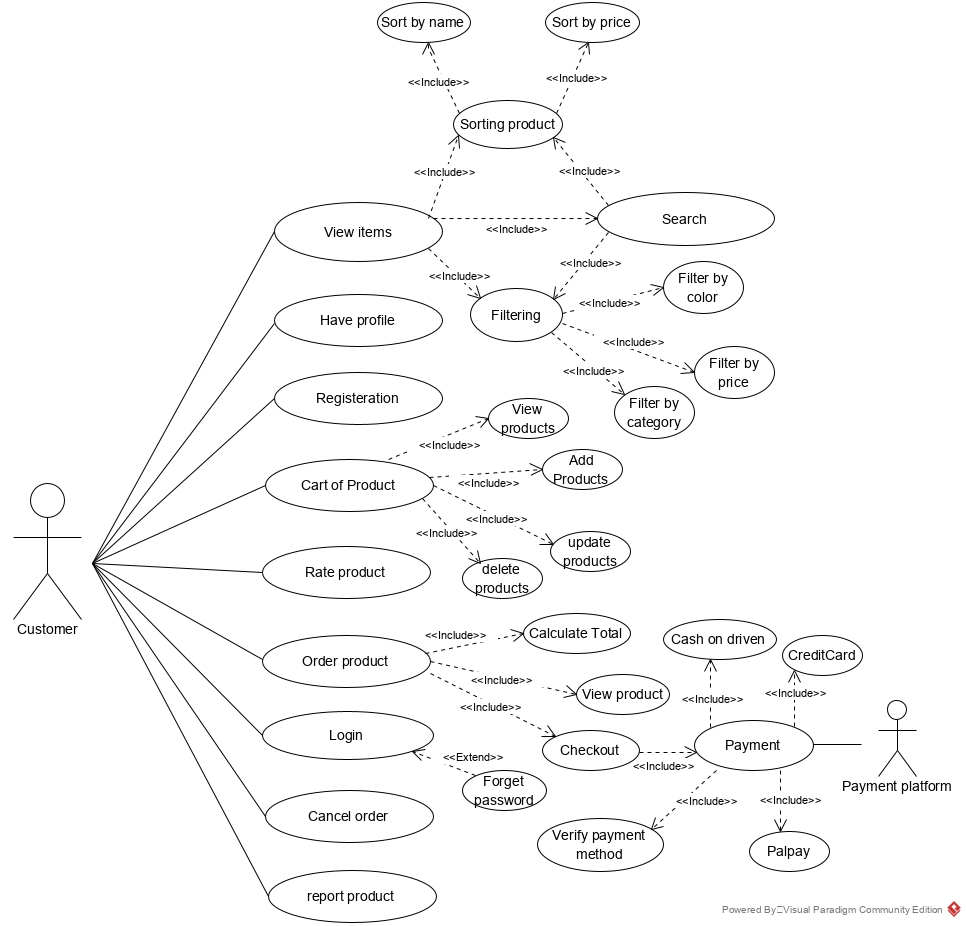


Figure : use case diagram for online shopping management system

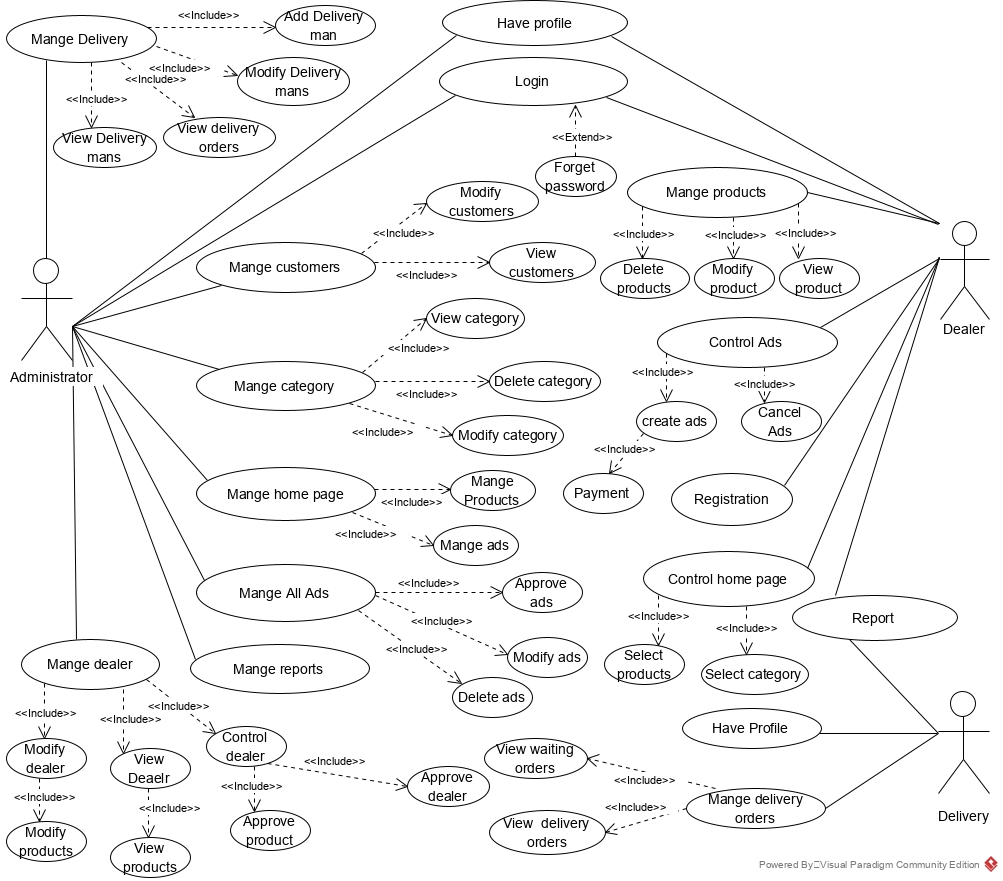


Figure : use case diagram for administrator and dealer in online shopping management system

**1.3 System Specific Requirements**

**1.3.1 Function Scenario for User Login**

|  |  |  |
| --- | --- | --- |
| ID | 1 |  |
| Name | | LOGIN |
| Description | | The user enter his credentials to login to the online shopping management system |
| **Priority** | | User must have already sign-up to the system |
| **Actors** | | System user, customer, admin |
| **Pre-Conditions** | | * User must have sign-up for system * User must have confirmed his email id |
| **Flow of Events** | | * User enter his user name or email * User enter his password * User hit the login button |
| Normal flow | | 1. User opens the login page 2. User enter his username 3. User enter his password 4. User hit LOGIN 5. If Login==Database 6. User sees the home page 7. Else LOGIN!=Database 8. User will see either wrong email or password error message |
| Alternative flow | | 1. When LOGIN != database match 2. User will see wrong email or password message 3. He will reenter correct information or sign-up to get membership for the system |
| **Post-Conditions** | | * User end up on the Home page of online shopping management system |

Table : Use case Scenario for User Login

**1.3.1.1 Sequence Diagram for User LOGIN**

The sequence diagram for user login is presented in figure 2

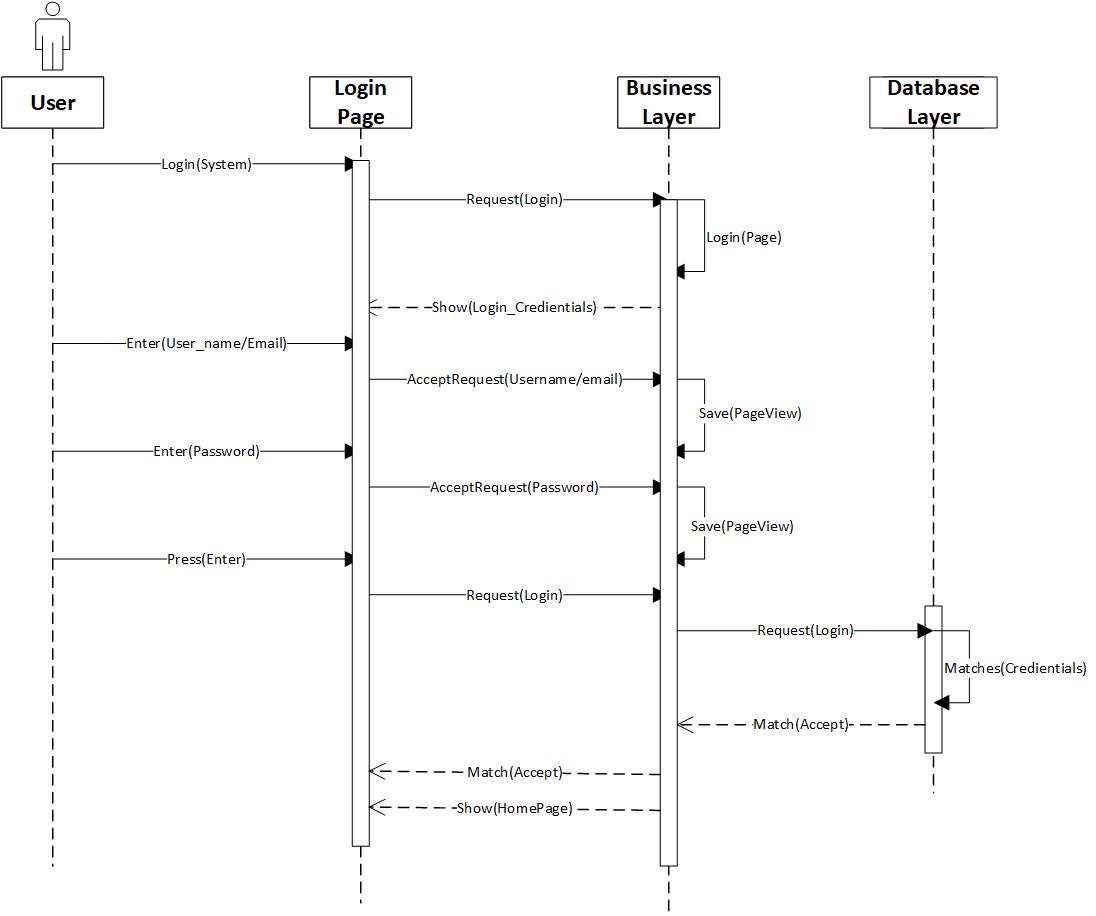


Figure : Sequence Diagram for User Login

**1.3.2 Function Scenario for User Sign-up**

|  |  |  |
| --- | --- | --- |
| ID | 2 |  |
| Name | | LOGIN |
| Description | | The user enter his credentials to Sign-up to the online shopping management system |
| **Priority** | |  |
| **Actors** | | System user, customer, admin |
| **Pre-Conditions** | | * User know how to sign-up and know to use the technology |
| **Flow of Events** | | * User enter his preferred username and email * User enter his Preferred password * User hit the sign-up button |
| Normal flow | | 1. User opens the sign-up page 2. User enter his preferred username 3. User enter his preferred email 4. User enter his password 5. User hit sign-up 6. System shows email alert to confirm 7. User check his email and confirm email to complete sign-up process 8. System ask for enter first name 9. User enter first name 10. System ask for enter last name 11. User enter last name 12. User enter his home address as follow     1. User enter his house no     2. User enter his street no     3. User enter his area code/zip code     4. User enter his city     5. User enter his state     6. User enter his country 13. User complete the sign-up and hit sign-up button |
| Alternative flow | | 1. System may ask for down of server or show error 2. Errors can be user already exist, with same user id and email match |
| **Post-Conditions** | | * User end up complete of sign-up process and see sign-in screen |

Table : use case scenario for sign-up process of user

**1.3.1.2 Sequence Diagram for User SIGN-UP**

The sequence diagram for user Sign-up is presented in figure 3

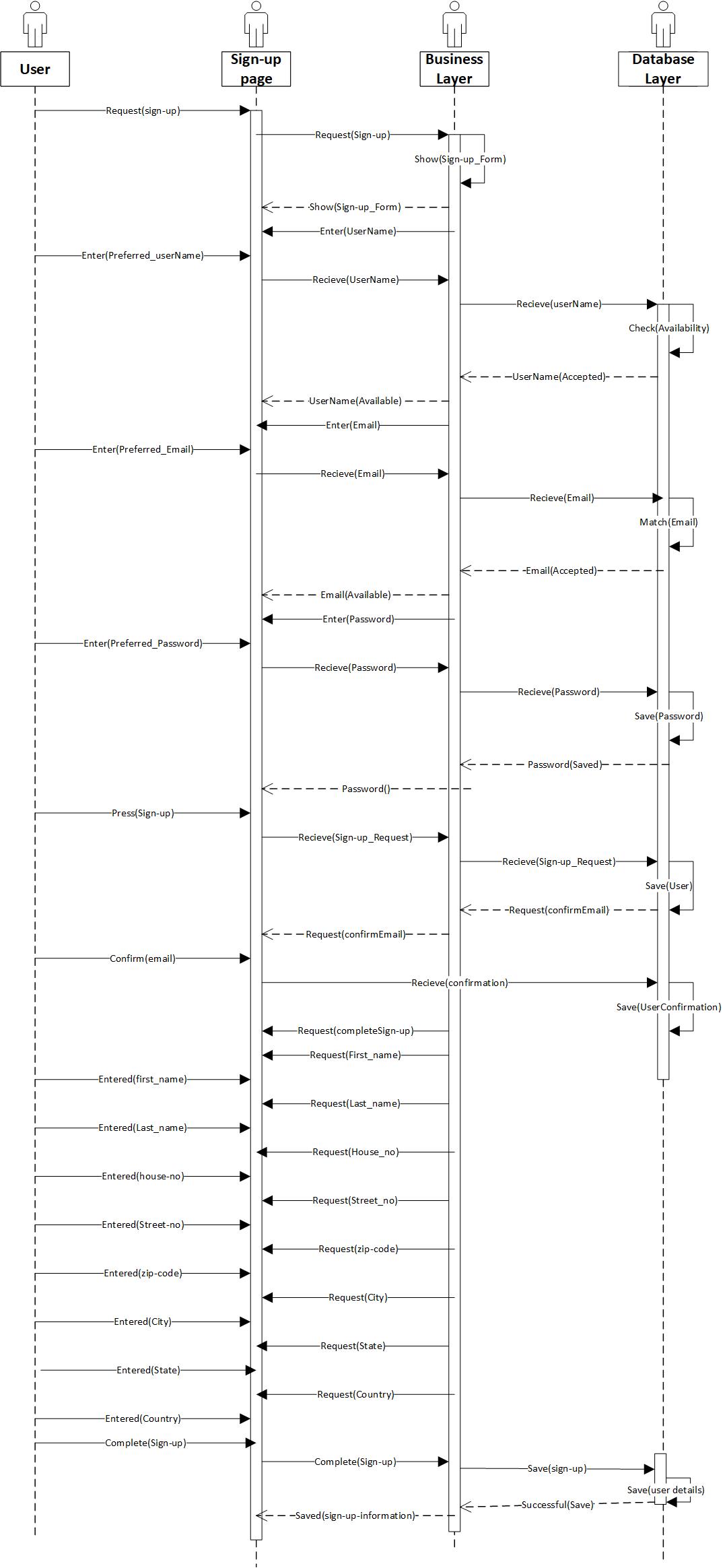


Figure : sequence diagram for sign-up process

**1.3.3 Function Scenario for View Products**

|  |  |  |
| --- | --- | --- |
| ID | 3 |  |
| Name | | View Product |
| Description | | User view different products and browse through them |
| **Priority** | | User can view similar products on the same page |
| **Actors** | | System user, customer |
| **Pre-Conditions** | | * User is member of system or simple guest user |
| **Flow of Events** | | * User browse through categories of products * User knows the products information |
| Normal flow | | 1. User search for the product he want to browse 2. User can browse through list of products with their little description 3. User can select and browse some specific product 4. User can view product details 5. User can add product to his cart 6. User can remove product from his cart 7. User can proceed to check-out process |
| Alternative flow | | 1. System confirms user is a member or not of the system at the time of check-out |
| **Post-Conditions** | | * User can browse through different products |

Table : use case scenario for view products

**1.3.1.3 Sequence Diagram for view Products**

The sequence diagram for view products is presented in figure 4

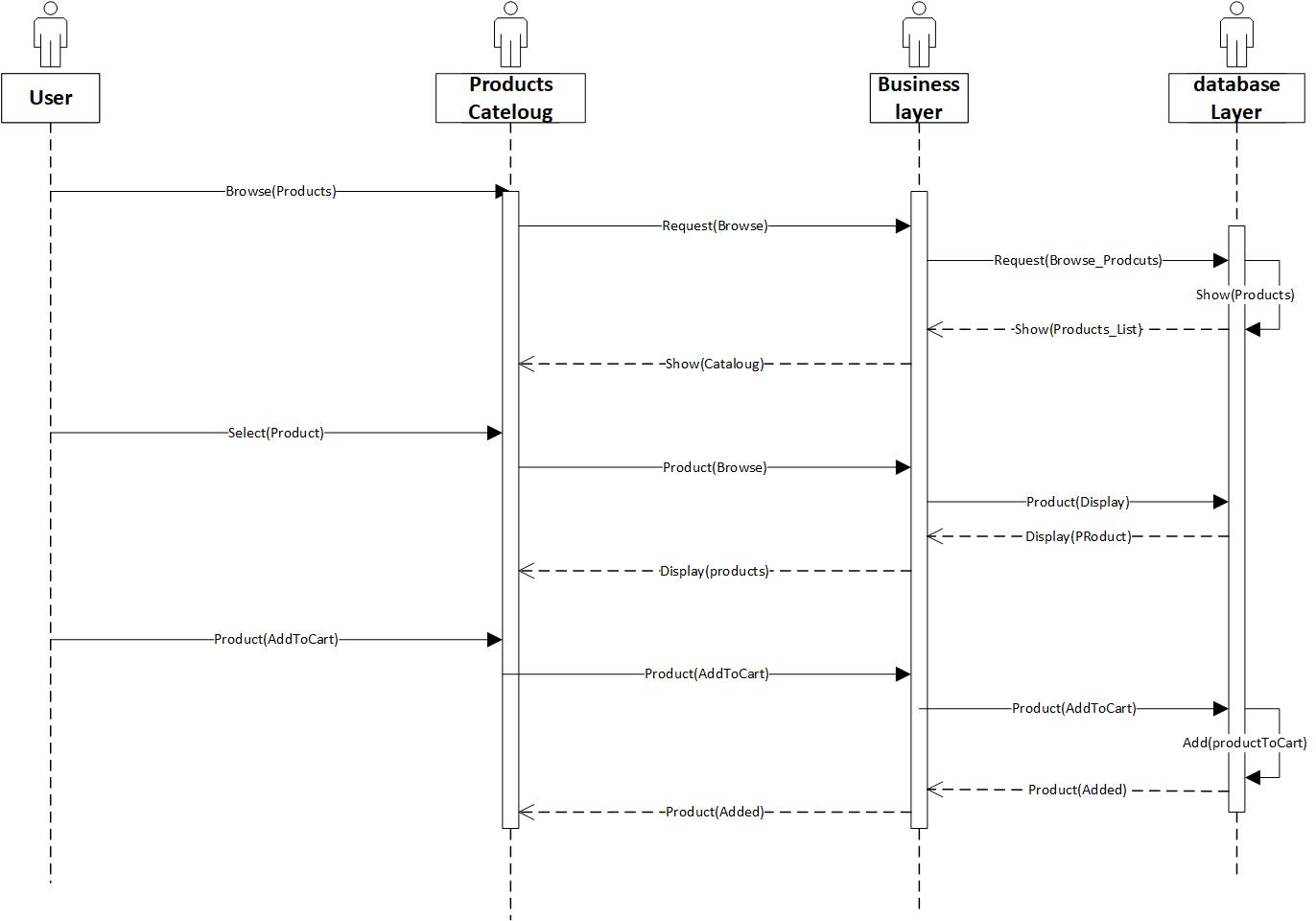


Figure : sequence diagram for Product view/browsing

**1.3.4 Function Scenario for order Products**

|  |  |  |
| --- | --- | --- |
| ID | 4 |  |
| Name | | order Product |
| Description | | User order different products and browse through them |
| **Priority** | | User can view similar products on the same page |
| **Actors** | | System user, customer |
| **Pre-Conditions** | | * User is member of system or simple guest user |
| **Flow of Events** | | * User browse through categories of products * User knows the products information * Order product to be delivered at home address |
| Normal flow | | 1. User search for the product he want to browse 2. User can browse through list of products with their little description 3. User can select and browse some specific product 4. User can view product details 5. User can add product to his cart 6. User can remove product from his cart 7. User can order product by clicking the checkout now 8. User Add card information 9. User add the address information 10. User can proceed to check-out process |
| Alternative flow | | 1. System confirms user is a member or not of the system at the time of check-out |
| **Post-Conditions** | | * User can browse through different products |

Table : use case scenario for order products

**1.3.1.4 Sequence Diagram for order Products**

The sequence diagram for order products is presented in figure 5

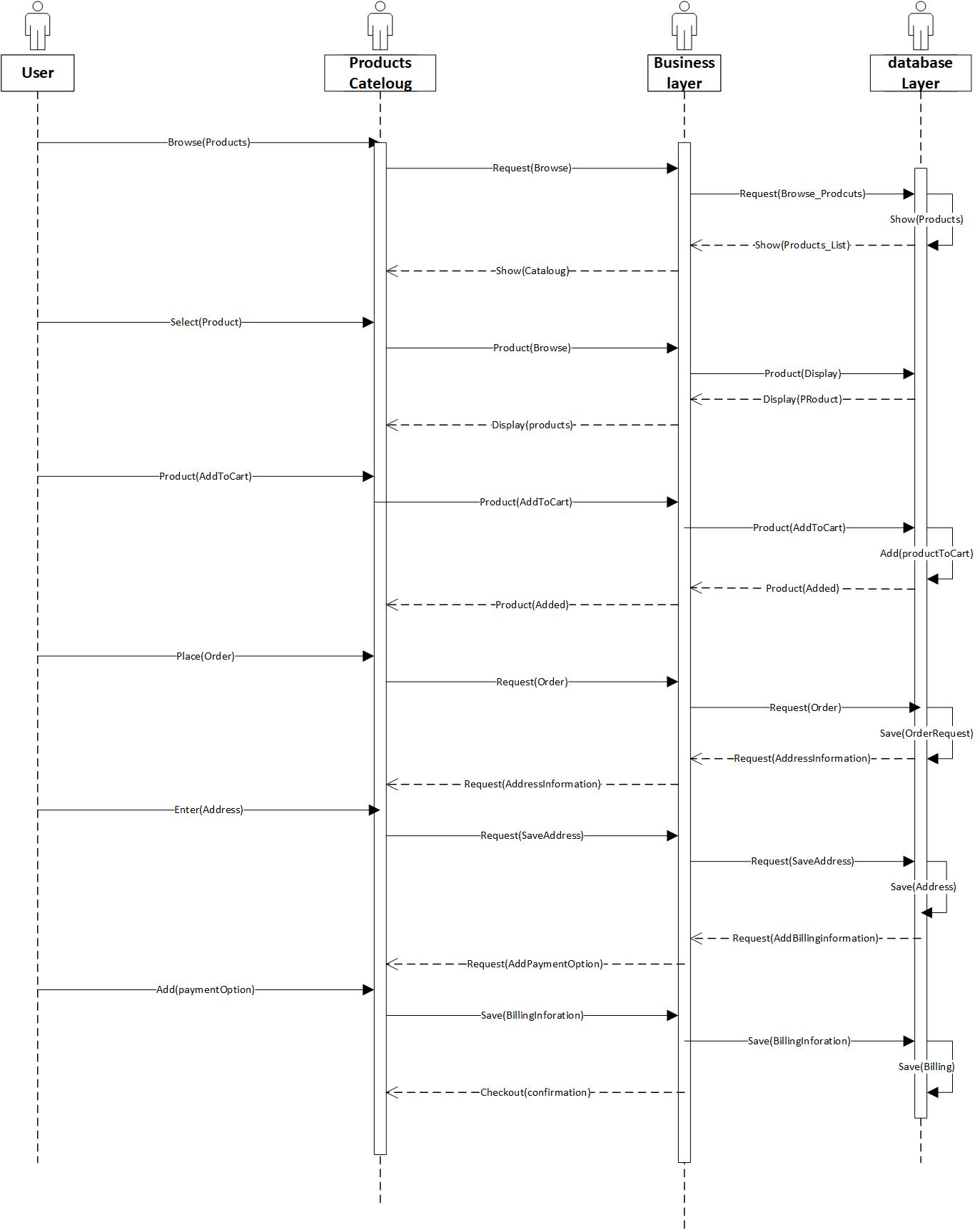


Figure : Sequence diagram for user products order

**1.3.5 Function Scenario for Payment**

|  |  |  |
| --- | --- | --- |
| ID | 5 |  |
| Name | | Payment |
| Description | | User Adds the payment method to pay for the products he bought |
| **Priority** | | User can add multiple payment methods |
| **Actors** | | System user, customer |
| **Pre-Conditions** | | * User is member of system |
| **Flow of Events** | | * User browse through categories of products * User knows the products information * Order product to be delivered at home address |
| Normal flow | | 1. User want to check-out 2. System ask for the delivery address 3. User add delivery address 4. System ask for payment method (i.e. giving options) 5. User select one option 6. User add card to pay for the products he bought 7. User complete the card attaching process 8. The card is deducted for products 9. The card value is checked for validity 10. The card payment email is sent to customer email 11. Customer can proceed to check-out complete 12. System shows the final receipt of address and item and ask for anything to change 13. Customer confirm the changes or go without change 14. System shows the thank you for shopping message |
| Alternative flow | | 1. System is unable to detect the card 2. System sees the card is not working properly |
| **Post-Conditions** | | * User is able to make attachment of his card to pay for bills. |

Table : User make payment through card case scenario

**1.3.1.5 Sequence Diagram for Payment for Products**

The sequence diagram for payment for products is presented in figure 6

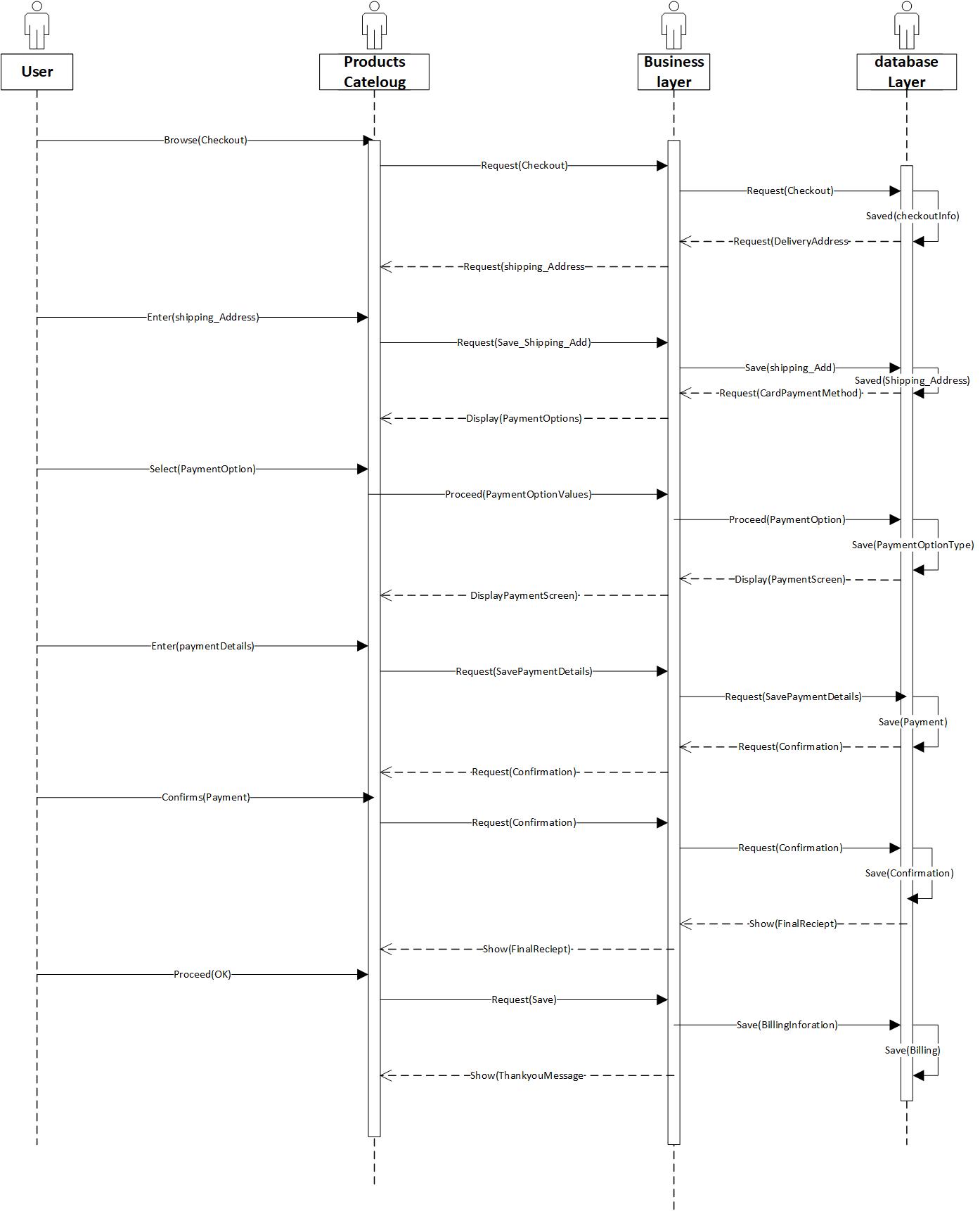


Figure : sequence diagram for adding payment options

**1.3.6 Function Scenario for Manage Customer**

|  |  |  |
| --- | --- | --- |
| ID | 6 |  |
| Name | | Manage Customer |
| Description | | Admin can manage different customers |
| **Priority** | | System admin is able to view customer status about product purchases, in-cart or proceed to check-out |
| **Actors** | | System admin |
| **Pre-Conditions** | | * User is member of system |
| **Flow of Events** | | * System user is able to see customer activity on the shopping website * Admin is able to see each customer cart * Admin can see customer card information to work with banks * Admin can see checked-out customer orders |
| Normal flow | | 1. Admin see customer directory 2. Admin can see different categories of customers including checked-out, checking-in and who hold the shopping cart 3. Admin can view the payment methods customer fulfilled 4. Admin can check on card information to send to bank for processing 5. Admin is able to see the customer history of shopping on the website 6. Admin can see the customer billing addresses to confirm the order to be delivered. |
| Alternative flow | | 1. System is unable to detect the customer card information 2. System sees the card is not working properly |
| **Post-Conditions** | | * Admin is able to view and evaluate the information of all customers |

Table : use case scenario for admin to manage the customers

**1.3.1.6 Sequence Diagram for view Products**

The sequence diagram for admin to manage the customers is presented in figure 7

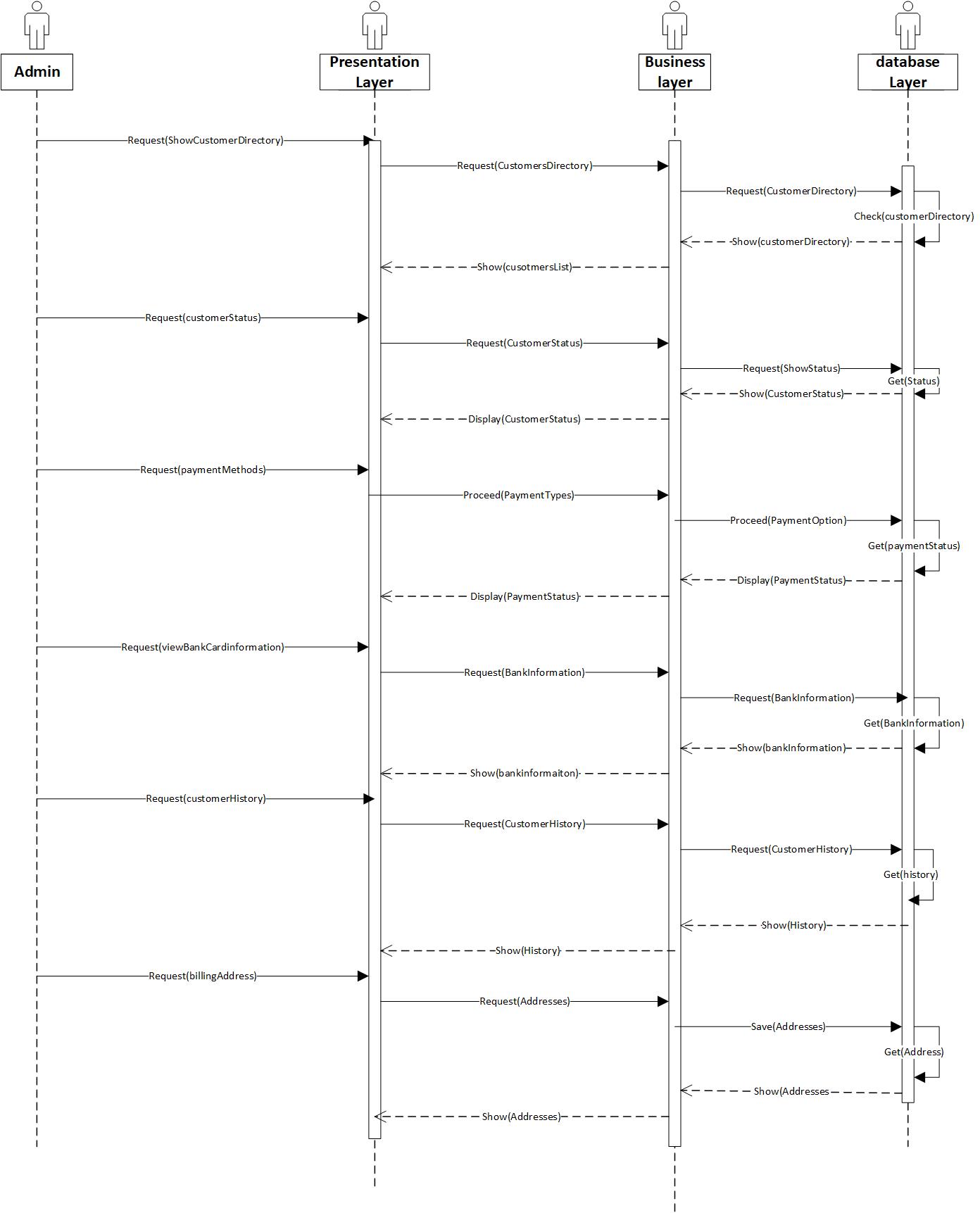


Figure : Admin managing customer information

**1.3.7 Function Scenario for Fulfil Customer Order**

|  |  |  |
| --- | --- | --- |
| ID | 7 |  |
| Name | | Fulfill Customer orders |
| Description | | Admin can manage different customers and fulfill their ordres |
| **Priority** | | System admin is able to view customer status about product purchases, in-cart or proceed to check-out |
| **Actors** | | System admin |
| **Pre-Conditions** | | * User is member of system |
| **Flow of Events** | | * System user is able to see customer activity on the shopping website * Admin is able to see each customer cart * Admin can see customer card information to work with banks * Admin can see checked-out customer orders |
| Normal flow | | 1. Admin see customer directory 2. Admin view the new order to be delivered in his directory 3. Admin check the customer card has paid the payment for product 4. Admin than check the status of order for being fulfilled 5. Admin confirms the order being fulfilled and generate email stating the expected delivery arrival to customer |
| Alternative flow | | 1. System is unable to detect the customer card information 2. System sees the card is not working properly |
| **Post-Conditions** | | * Admin is able to fulfil orders of all customers |

Table : use case scenario for the customers orders fulfilling by admin

**1.3.1.7 Sequence Diagram for Fulfil orders**

The sequence diagram for admin to fulfil customer orders is presented in figure 8

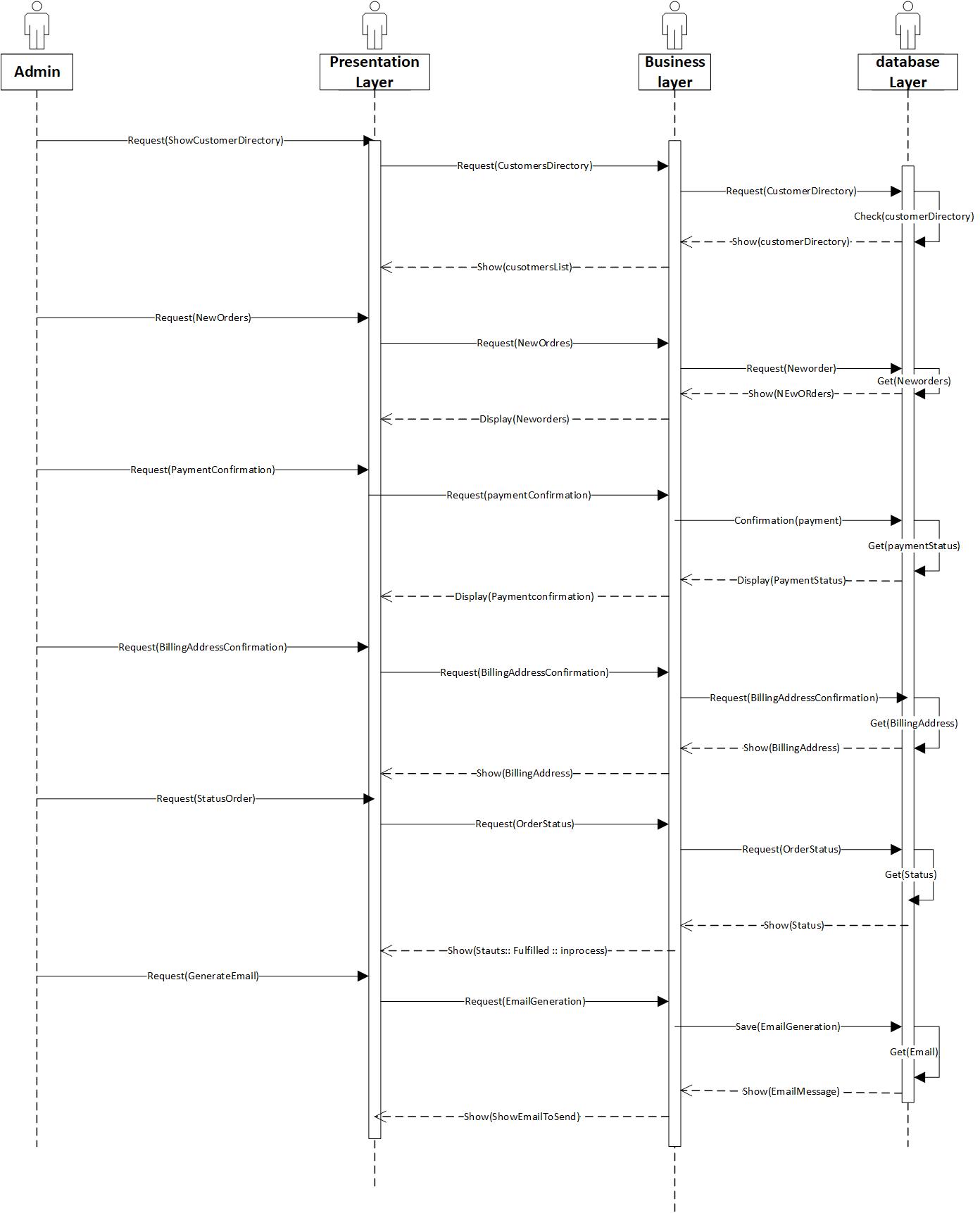


Figure : admin managing the fulfilling of orders

**1.3.8. Function Scenario for [Mange Product]**

|  |  |  |
| --- | --- | --- |
| ID | 8 |  |
| Name | | Manage products dealer |
| Description | | Is full control of the product provided by the dealer and it give the dealer ability to modify , delete ,draft or publish an product . |
| **Priority** | | High Priority |
| **Actors** | | dealer |
| **Pre-Conditions** | | The dealer must be logged in the site. |
| **Flow of Events** | | * Dealer select Product Management from the nav bar * The dealer select add Product to add an product , the product will be publish after it approved by the administrator . * The dealer select Update Product to update an product. * The dealer select Delete Product to delete an product. |
| Normal flow | | 1. The dealer open site and open the login page. 2. The dealer writes his email and password and click log in button. 3. The server check if the entered data = data in database. 4. Server will direct the dealer to the home page of the store. 5. The dealer select dashboard through an list appear when he click on his photo icon in the top of the page. 6. The server will direct the dealer to dashboard. 7. The dealer selects Product Management from the nav bar. 8. The dealer can add new product buy click on the ADD PRODUCT. 9. The product will be store in the data base and waiting to approve from the Administrator to be publish. 10. The dealer can update an product buy click on the UPDATE PRODUCT. 11. The dealer can Delete an product buy click on the DELETE PRODUCT. 12. The dealer can draft an product or publish |
| Alternative flow | | 1. The size of the images is not suitable for viewing 2. Products that violate company policy |
| **Post-Conditions** | | The product will be published on the dealer product page and shown to customers |

Table : use case scenario for manage product

**1.3.8.2 Sequence Diagram for [Mange Product]**

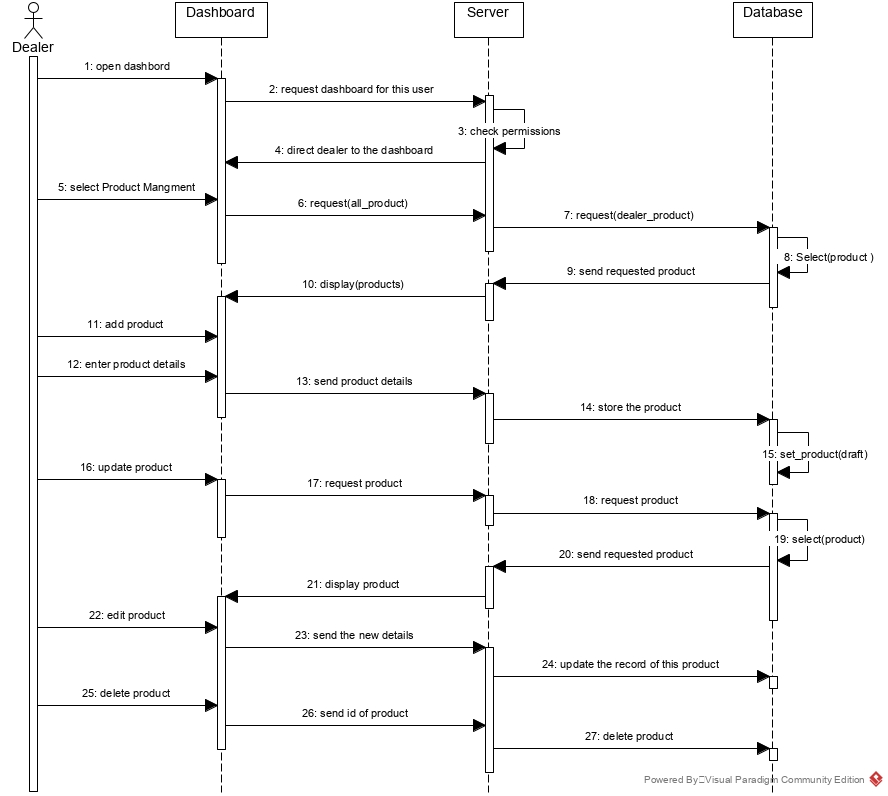
****

Figure : sequence diagram for manage products

**1.3.9 Function Scenario for [Control Ads ]**

|  |  |  |
| --- | --- | --- |
| ID | 9 |  |
| Name | | Control Ads dealer |
| Description | | The ads is an way to increases the viewer of an product , it controlled by the dealer, such as adding or deleting ads where they are not violating company policy and it published after the administrator approval it and completion of the payment process. |
| **Prioriny** | | Normal Priority |
| **Actors** | | dealer |
| **Pre-Conditions** | | * The dealer must be logged in the site. * The advertisement must be appropriate to the company’s policy, suitable for the target age groups, and be clearly attractive and effective. |
| **Flow of Events** | | * Dealer select an Ads Management from the nav bar * The dealer select start ads * The dealer select the product which he want to be in the ads . * The select the plane of the ads. * The dealer add payment method . * The user click start ads. |
| Normal flow | | 1. The dealer open store site and open the login page. 2. The dealer writes his email and password and click log in button. 3. The server check if the entered data = data in database. 4. Server will direct the dealer to the home page of the store. 5. The dealer select dashboard through an list appear when he click on his photo icon in the top of the page. 6. The server will direct the dealer to dashboard. 7. The dealer select Ads Management from the nav bar. 8. The dealer select start ads 9. The dealer select the product which he want to be in the ads . 10. The select the plane of the ads. 11. The dealer add payment method to complete the payment process after   The ads be approved from the Administrator.   1. The server verify he payment method and the ads , the ads must agree to the privacy policy of the company or administrator. 2. The ads will store in the database and appear in the administrator 3. After the administrator approve the ads , the payment process will happen ,and the ads will start display in different site pages . 4. The dealer can stop an ads after it approved from the administrator. |
| Alternative flow | | * The ads is not appropriate to the company policy. * Problem in the payment method. * The server does not respond. |
| **Post-Conditions** | |  |

Table : use case scenario for control ads

**1.3.9.1 Sequence Diagram for [Control Ads]**

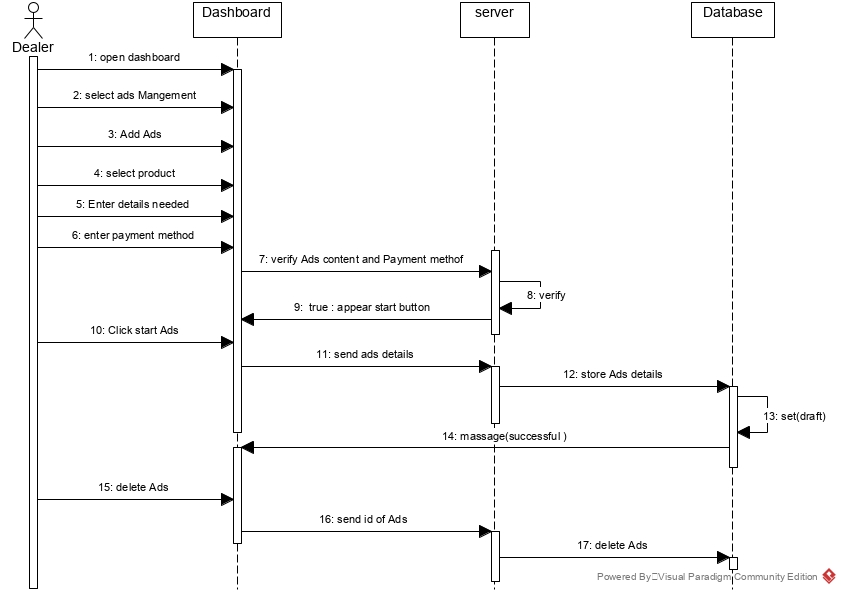
****

Figure : sequence diagram for control ads

**1.3.10 Function Scenario for [Mange Ads]**

|  |  |  |
| --- | --- | --- |
| ID | 10 |  |
| Name | | Mange Ads |
| Description | | is the control and approval of the ads that created by dealers . |
| **Priority** | | Normal priority |
| **Actors** | | Administrator |
| **Pre-Conditions** | | * Administrator must be logged in the system . |
| **Flow of Events** | | * Administrator log in to the store . * The Administrator open the dashboard . * The Administrator select Ads Managements from nav bar. * The Administrator start do his job of approving ads and censorship on it. |
| Normal flow | | 1. The Administrator store site and open the login page. 2. The Administrator write his email and password and click log in button. 3. Server will direct the Administrator to the home page of the store. 4. The Administrator select dashboard through an list appear when he click on his photo icon in the top of the page . 5. The server will direct the administrator to dashboard . 6. The Administrator select Ads Managements from the nav bar. 7. The Administrator click on Approval of ads and check the ads if it Violation of the conditions or not . 8. The Administrator approve the ads. 9. The price of the ads will be transfer from the dealer bank account to the administrator bank account . 10. The server send a massage contain that the ads of the dealer will start publishing , and the payment process don successfully. 11. The server start displaying and publishing the ads on the different page of the store. 12. The Administrator can stop or delete an ads if the ads poses any risks and problems to the store . 13. The Administrator can view all ads that running or waiting in the queue . 14. Admin can select which ads appear on the home page . |
| Alternative flow | | * The Administrator can’t login to the system. * The server does not respond * The ads Violation of the conditions so it will not be approved . * Problem in payment process . |
| **Post-Conditions** | |  |

Table : use case scenario for manage ads

**1.3.10.1 Sequence Diagram for [Mange Ads]**

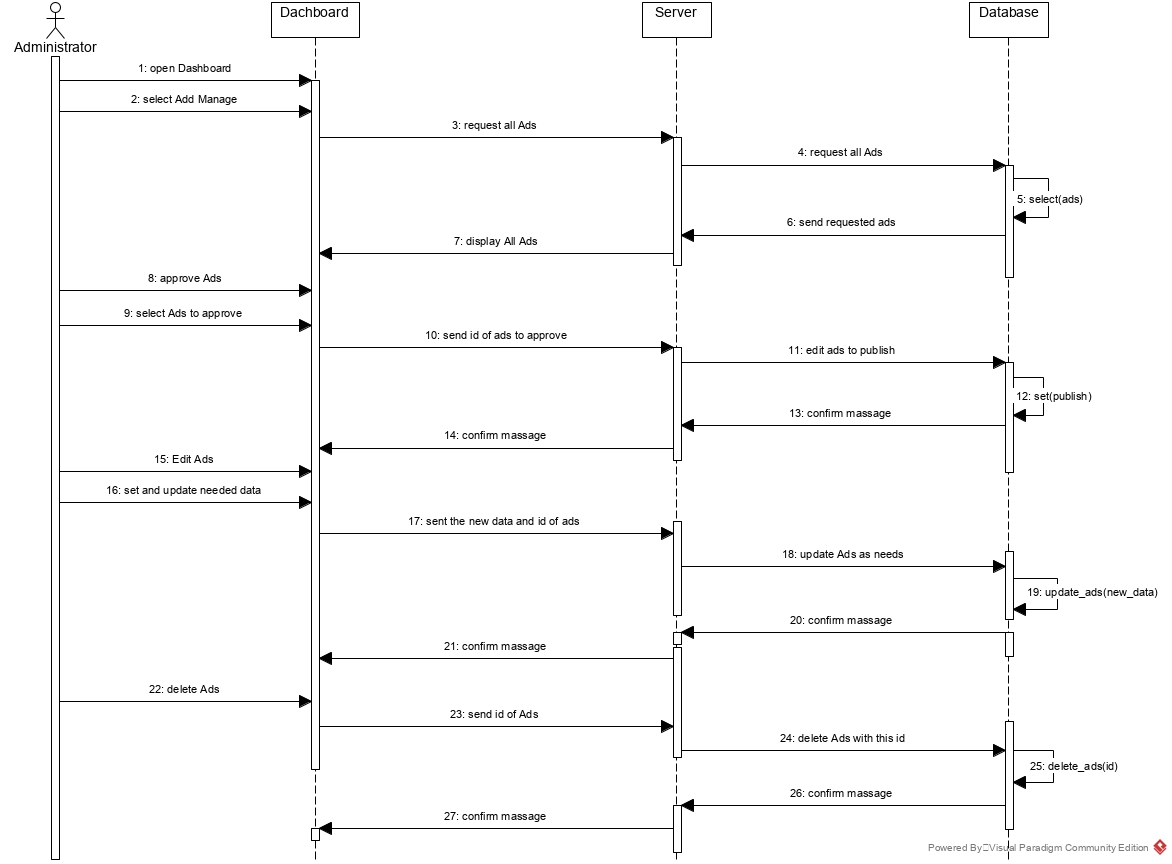
****

Figure : sequence diagram for manage ads

**1.3.11 Function Scenario for [Mange Dealers]**

|  |  |  |
| --- | --- | --- |
| ID | 11 |  |
| Name | | Mange Dealers |
| Description | | is the control and approval of everything related to the data of dealers and the products provided by the dealer and the dealer approval , to give him the ability to publish his products. |
| **Priority** | | High Priority . |
| **Actors** | | Administrator. |
| **Pre-Conditions** | | * Administrator should be logged in . |
| **Flow of Events** | | * Administrator log in to the store . * The Administrator open the dashboard . * The Administrator select Dealers Managements from nav bar. * The Administrator start do his job of checks and Censorship. |
| Normal flow | | 1. The Administrator store site and open the login page. 2. The Administrator write his email and password and click log in button. 3. Server will direct the Administrator to the home page of the store. 4. The Administrator select dashboard through an list appear when he click on his photo icon in the top of the page . 5. The server will direct the administrator to dashboard . 6. The Administrator select Dealers Managements from the nav bar . 7. The Administrator check the new dealers registered to the store . 8. The Administrator make set of verification to approve the dealer. 9. The server will send an email to the dealer with a message stating that the registration was successful. 10. The Administrator check the Approval of products and verify of the new products sent from the dealer to the administrator to approve on it   And make read all info about the product then if it meeting all condition , it will be approved .   1. The Server will publish the product on the store. 2. The Administrator can View all dealer data and products . 3. The Administrator can edit as want on the dealer information . 4. The Administrator can block an dealer if the dealer has violated the usage policy. 5. The Administrator can block an product if the product violating conditions . |
| Alternative flow | | -The Administrator can’t login to the system.  -The server does not respond. |
| **Post-Conditions** | |  |

Table : use case scenario for manage dealers

**1.3.7.2 Sequence Diagram for [Mange Dealers]**

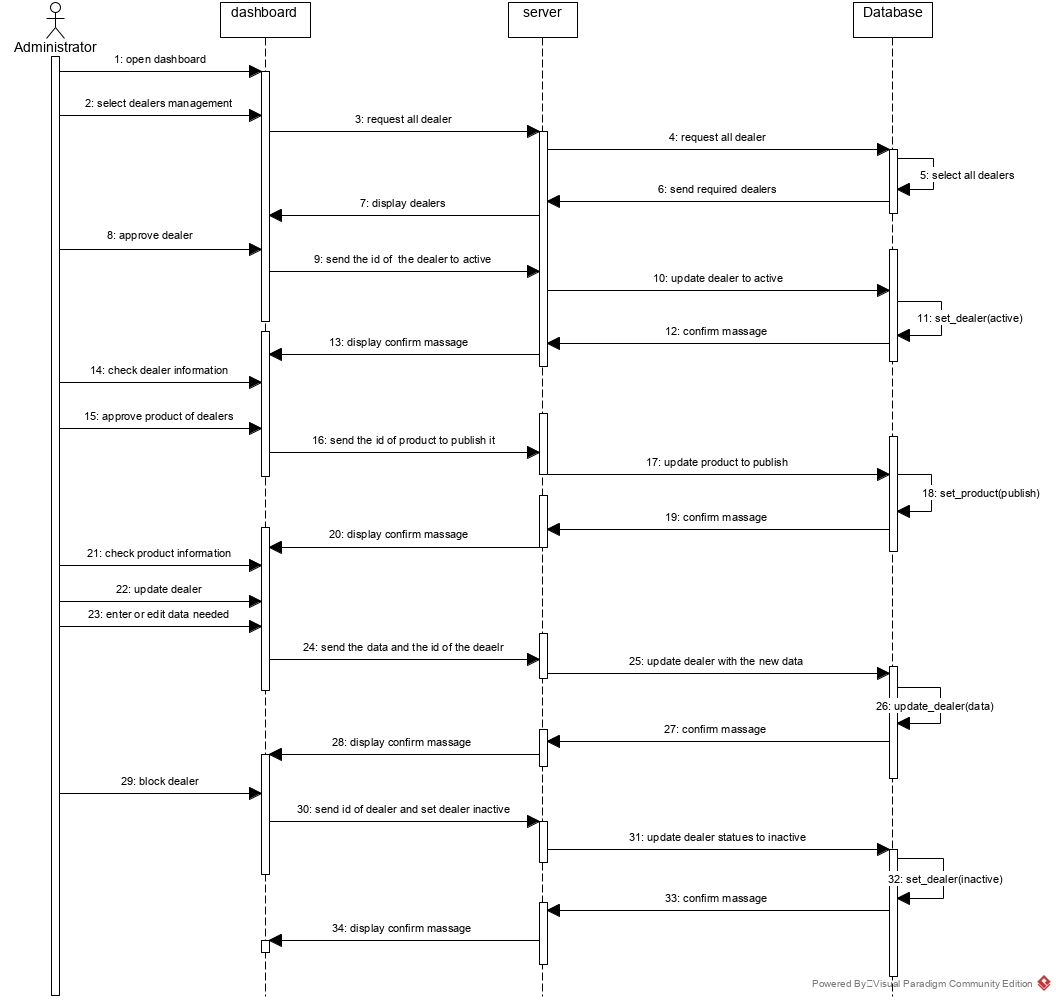
****

Figure : sequence diagram for manage dealers

**1.3.12 Function Scenario for [Reports]**

|  |  |  |
| --- | --- | --- |
| ID | 12 |  |
| Name | | Report |
| Description | | It is the contact way between the users and administrator in case of problem and bugs or reporting any user who violated the site policy. |
| **Priority** | | High Priority |
| **Actors** | | System users |
| **Pre-Conditions** | |  |
| **Flow of Events** | | * The user must add his full personal name * The user must add his email address * The user should write the problem that you want to report * The user must click on the Submit button |
| Normal flow | | 1. The user browsing the store . 2. The user faces a problem. 3. The user clicks the Report Problem button 4. The user fills in the personal name field completely 5. The user fills in the email field correctly 6. The user writes the problem clearly and accurately to find out the intended defect 7. The user click submit button to send the problem to the administrator. 8. The server will store the problem in the database . |
| Alternative flow | | -The system does not accept the entered email because it was written in a wrong way.  - The server does not respond. |
| **Post-Conditions** | |  |

Table : use case scenario for reports

**1.3.12.1 Sequence Diagram for [Report]**

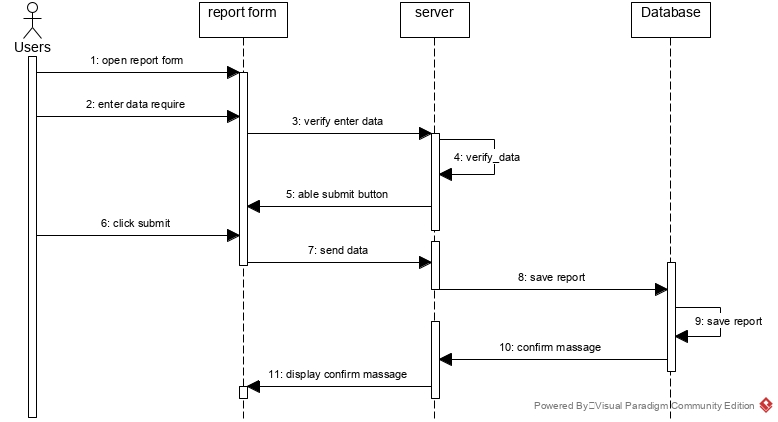
****

Figure : sequence diagram for reports

**1.3.13 Function Scenario for [Mange Reports ]**

|  |  |  |
| --- | --- | --- |
| ID | 13 |  |
| Name | | Mange Reports |
| Description | | Is checking the report that sent from the user of the store and reply on it, taking into consideration all store policies . |
| **Priority** | | High Priority |
| **Actors** | | Administrator . |
| **Pre-Conditions** | | * Administrator must be logged in the system |
| **Flow of Events** | | * Administrator log in to the store . * The Administrator open the dashboard . * The Administrator select Reports Managements from nav bar. * The Administrator start check the reports and replay on it . * The Administrator take prohibition measures against violators |
| Normal flow | | 1. The Administrator store site and open the login page. 2. The Administrator write his email and password and click log in button. 3. Server will direct the Administrator to the home page of the store. 4. The Administrator select dashboard through an list appear when he click on his photo icon in the top of the page . 5. The server will direct the administrator to dashboard . 6. The Administrator select Reports Managements from the nav bar. 7. The server direct the Administrator to the page contain all report from the users . 8. The administrator start reading the repots carefully . 9. The administrator replay to the user that we received your problem. 10. The Server send massage to the user contain that we received your problem and we are fixing the issue . 11. The Administrate start fixing the problem . 12. The Administrator take prohibition measures against violators. 13. The Administer send a massage to the report sender contain that the problem fixed successfully . |
| Alternative flow | | * The Administrator can’t login to the system . * The server does not respond . * The problem nor fixed . * False reports received . |
| **Post-Conditions** | |  |

Table : use case scenario for manage reports

**1.3.13.1 Sequence Diagram for [Mange Reports]**

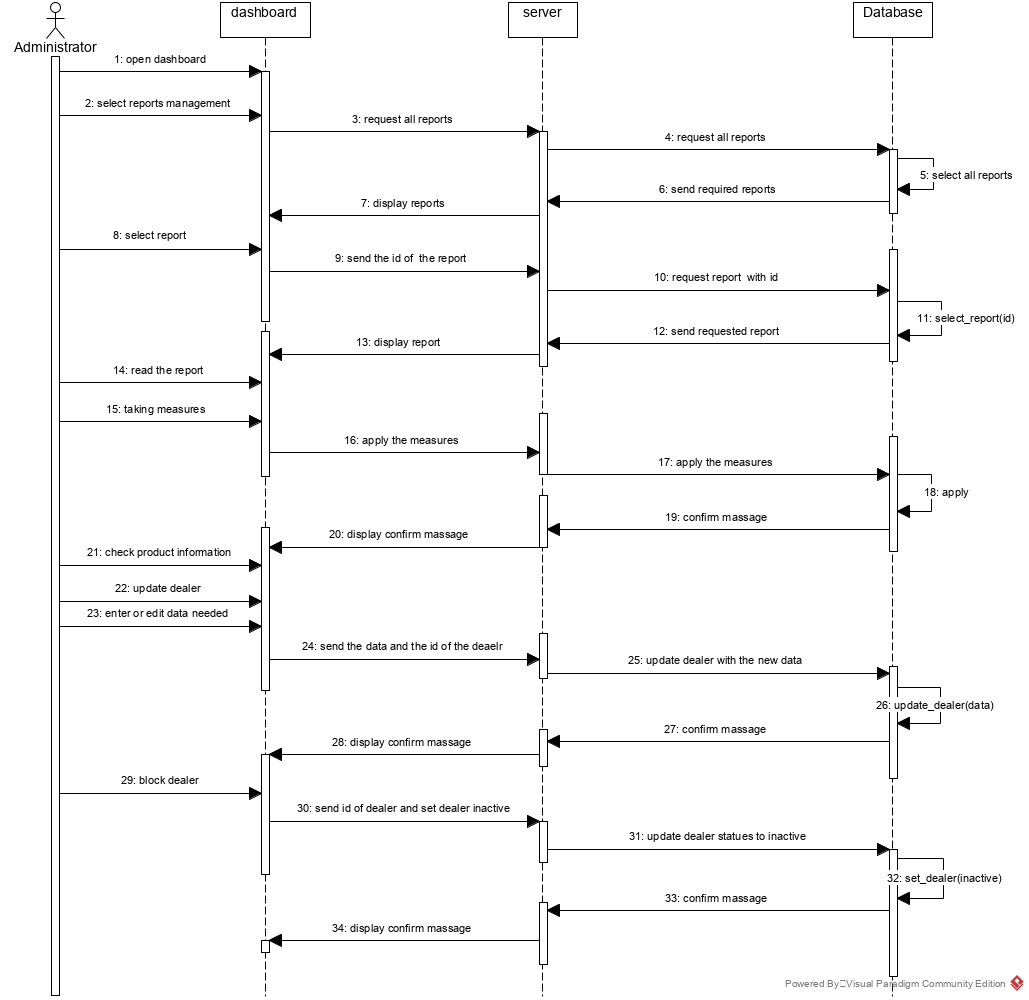
****

Figure : sequence diagram for manage reports

**1.4 List of classes**

The major list of classes which are derived from the use case scenario, and sequence diagrams are:

1. customer,
2. admin/manager
3. products
4. cart
5. payment
6. shipping cart
7. dealer
8. Search
9. Deliveryman
10. Report
11. Ads
12. Profile
13. Category
14. Product
15. Login
16. Registration
17. View products
18. View single product
19. Order

**1.4.1 Interface (Boundary) classes**

The major interface (boundary) classes for the online shopping management system are:

1. Customer<< interface>>
2. Admin <<Interface>>
3. DatabaseConnection
4. Home page
5. Confirm massage
6. Dashboard
7. Profile
8. Report
9. Login/registration
10. Search
11. Category
12. Order product

**1.4.2 Entity classes**

The entity classes for the online shopping management system are:

1. Customer
2. Dealer
3. Administrator
4. Deliveryman
5. Payment

**1.4.3 Control (Manager) classes**

The control (manager) classes for the online shopping management system are:

1. AddDelivery
2. UpdateDelivery
3. updatedealer
4. Adddealer
5. Addpayment
6. Updatepayment
7. ApproveProduct
8. ApproveDealer
9. Blockdealer
10. BlockProduct
11. BlockCustomer

**1.5 Non-Functional Requirements**

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior or functions.

*Requirement-1:* The system shall be a web-based application that will enable users to access the system using latest versions of Internet Explorer, Safari, or Firefox browsers.

*Requirement-2:* The system shall support at least 10,000 simultaneous connections/users.

*Requirement-3:* The system shall be running on a Windows Operating System based server.

*Requirement-4:* The system shall be developed using Php language.

*Requirement-5:* The system shall be available for 99.9% of time.

*Requirement-6:* The system shall be secure so that the user profiles will be encrypted.

*Requirement-7:* The system shall be robust and in case of the system failure the user data will not be lost. Due to database backup

*Requirement-8:* The system should accurately provide real time information taking into consideration various concurrency issues.

**Chapter 2. System Design and Architecture for [name of your system]**

* 1. **Application Architecture**

A application architecture describes the overall flow between application different layers and well as how they will interact with each other.

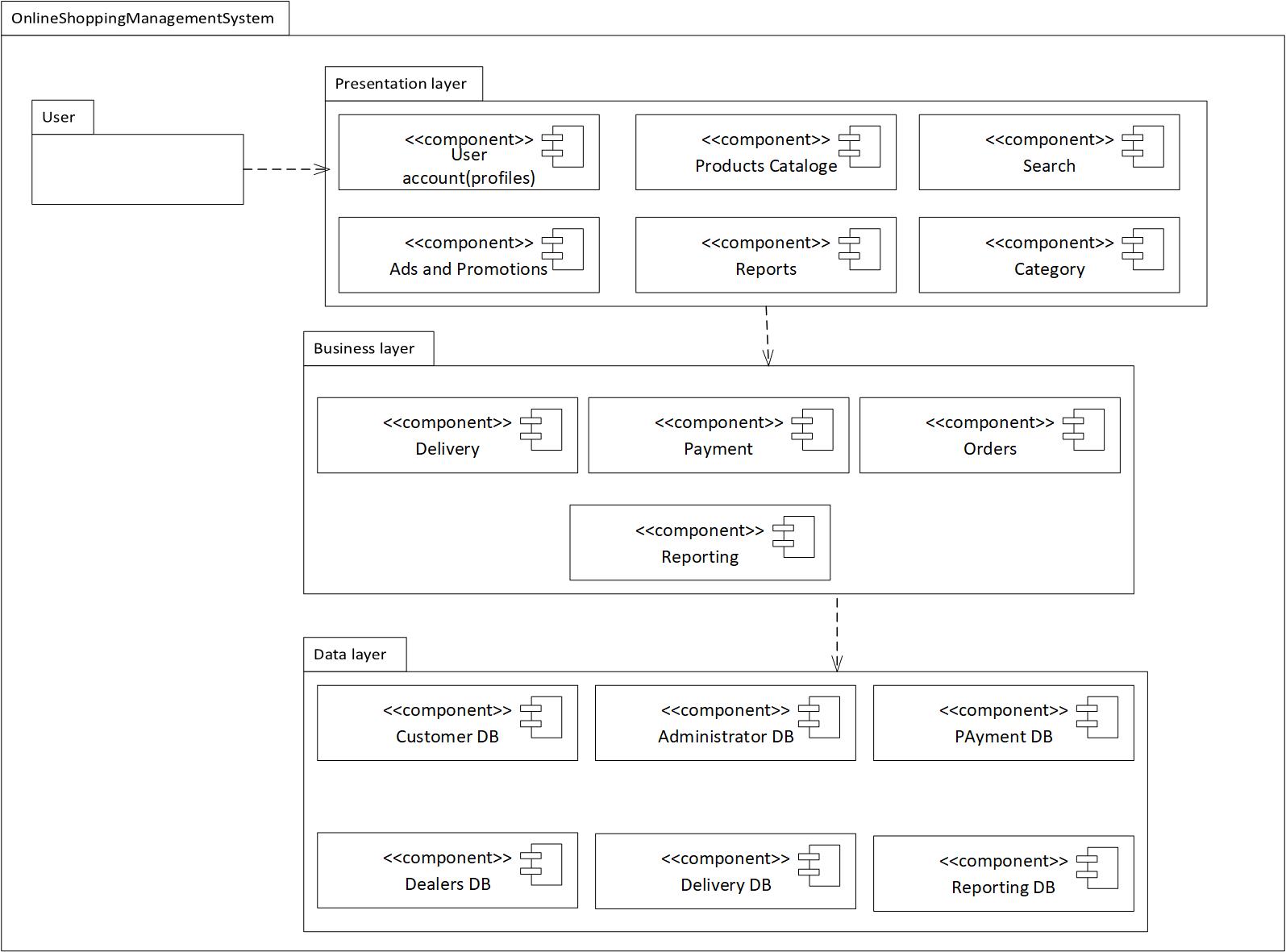


Figure : online shopping management system package diagram

* + 1. **User Profile Package**

This is a presentation layer package which will include the users like customer, admin, delivery, and dealer who are the key stakeholders of the system. The purpose of this package is to register and provide login details of customer, admin, delivery, and dealer. The major actions this package will perform includes sign-in, sign-up/register for the system. The class diagram for this module is given in figure below

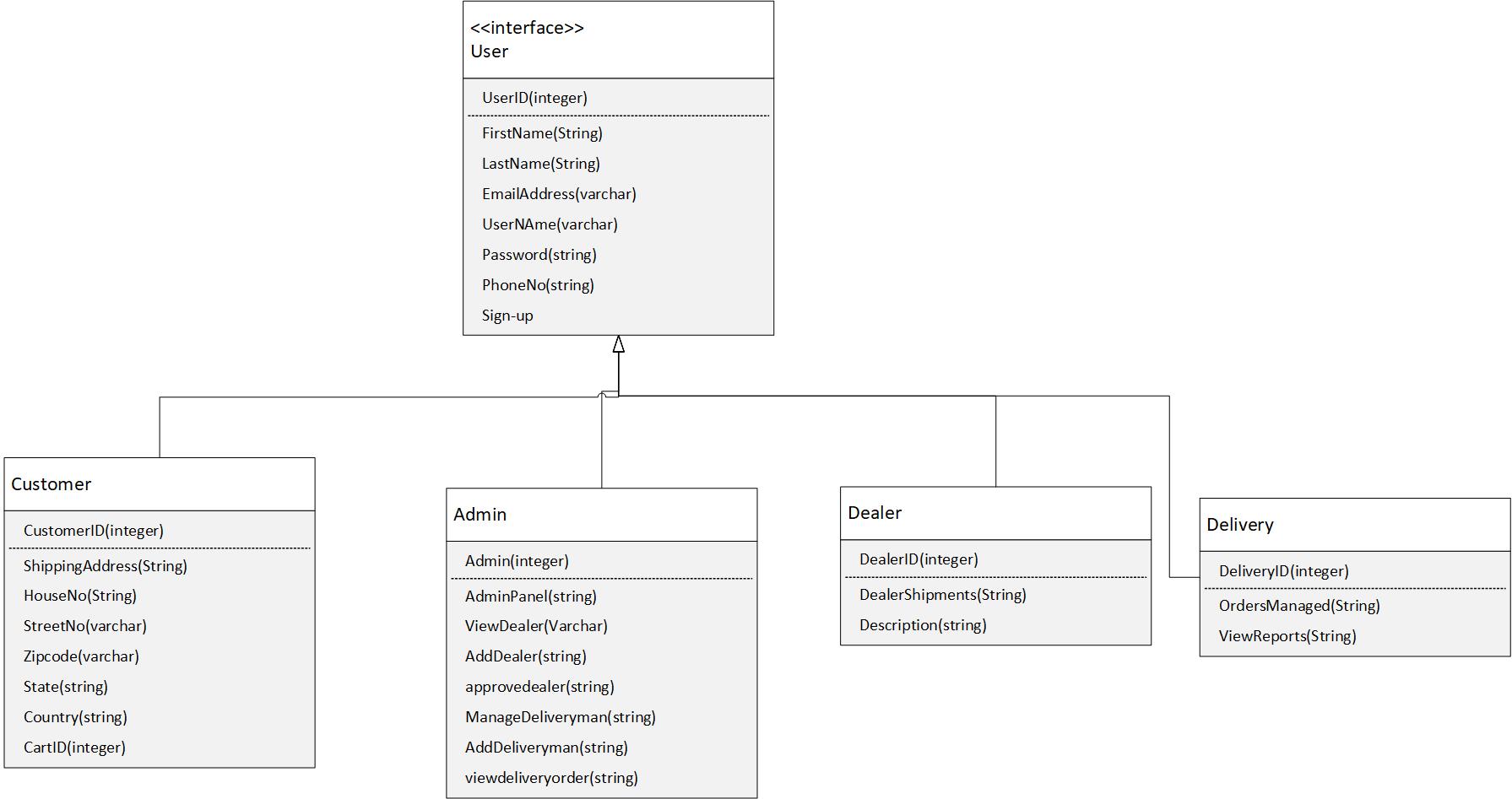


Figure : class diagram for user package

* + 1. **Product Package**

This package will include the classes and state information about products, catalogue and other information about product. The major functions for the product features includes the product descriptions, searching product, sorting product, filtering products, view products, browse products, add them to the cart, and management of different product types. The class diagram for product package is shown in following figure.

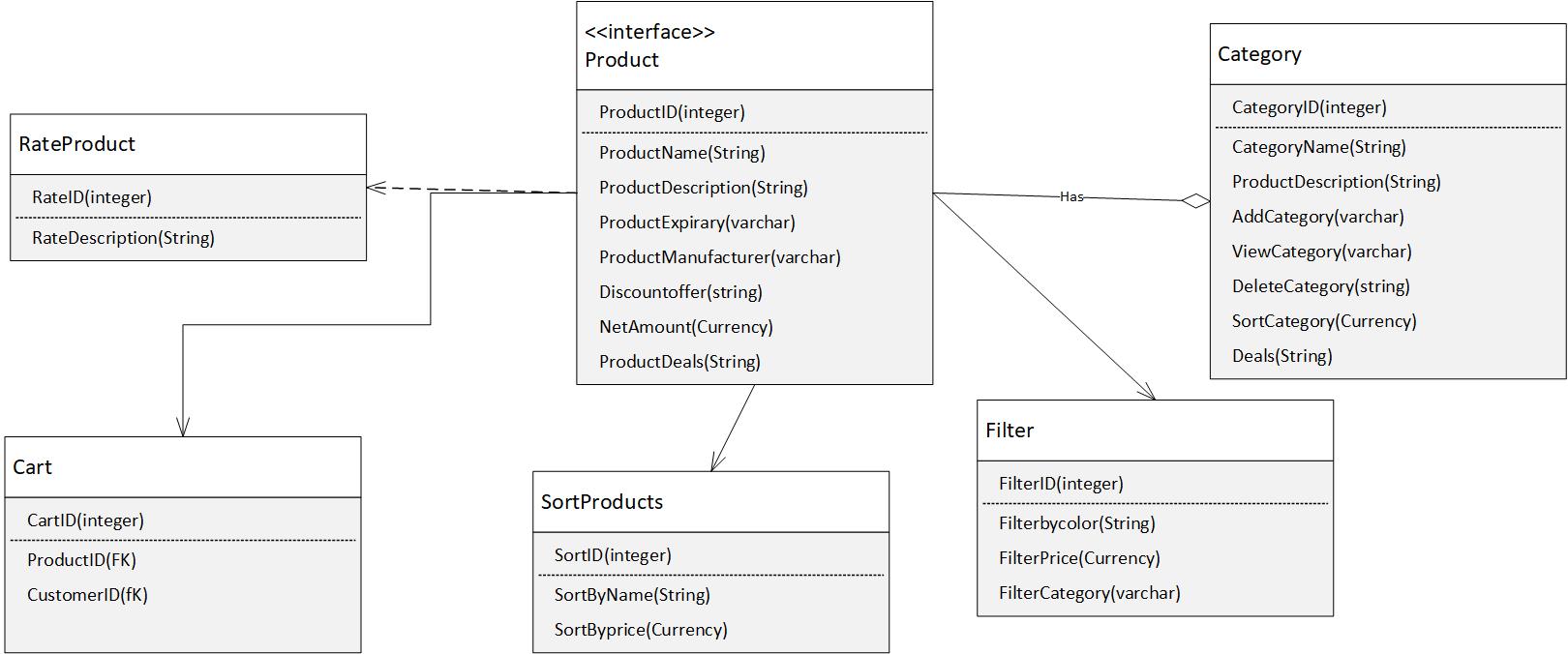


Figure : product package class diagram

* + 1. **Search Package**

Search package will provide a lot of information about the website to user, admin, dealer as well as delivery man. The users can search according to their desire for the category, reports, and many other functionalities available in the website. The class diagram for the search package is given in following diagram.

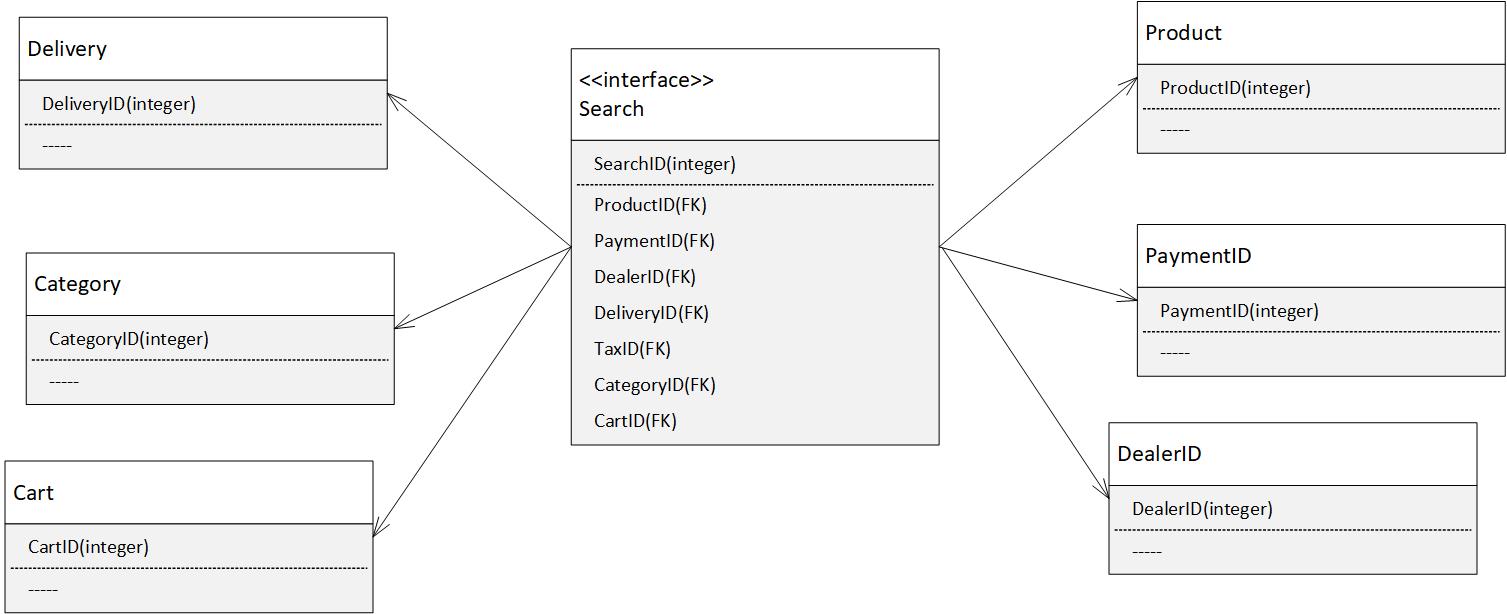


Figure : class diagram for search package

* + 1. **Ads and Promotions Package**

The ads and promotion package is important for the admin and dealer to promote their products in order to make sales on the website. The major functions of this package includes creating ads, cancel ads, approving ads, modify ads or delete them. The class diagram for ads and promotion package includes.

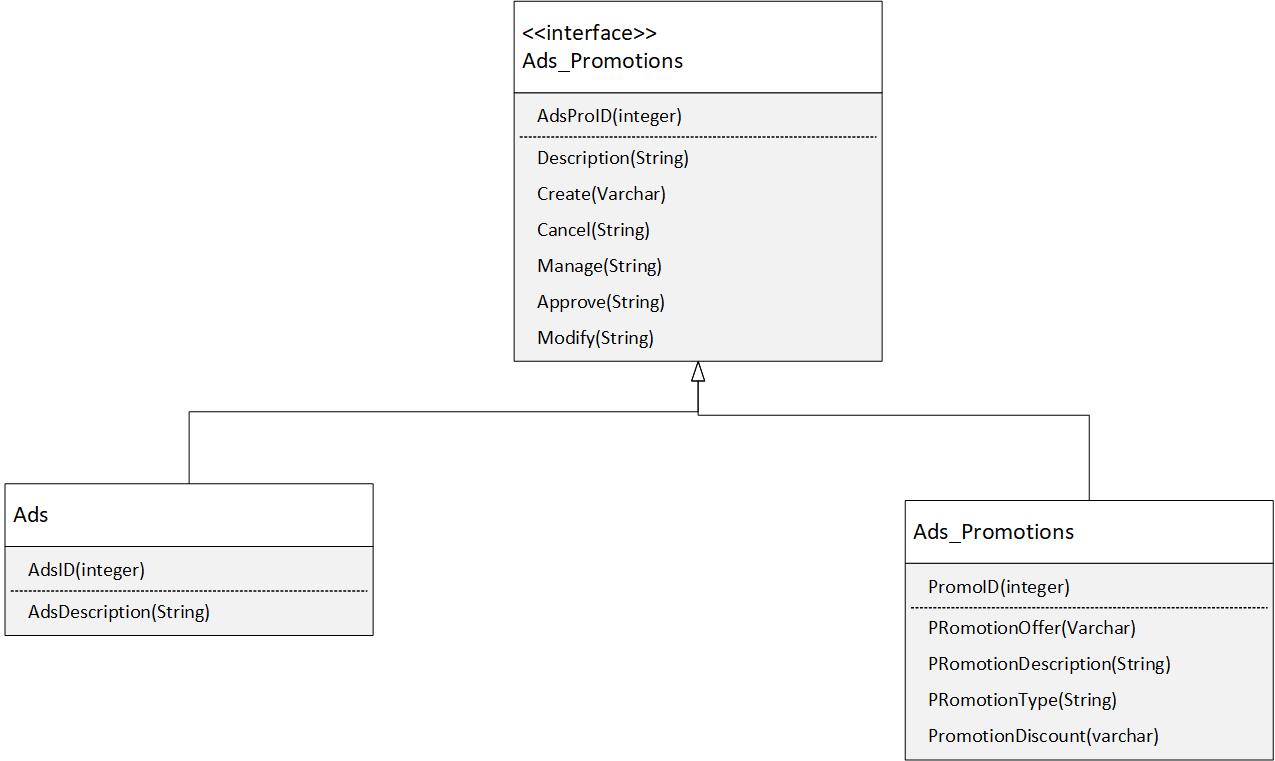


Figure : class diagram for promotion and ads package

* + 1. **Reports Package**

This is a very important package because all the reports are generated in this module and different types of reports for admin, dealer and delivery man can be generated using this package. The class diagram for this package is given in the figure below.

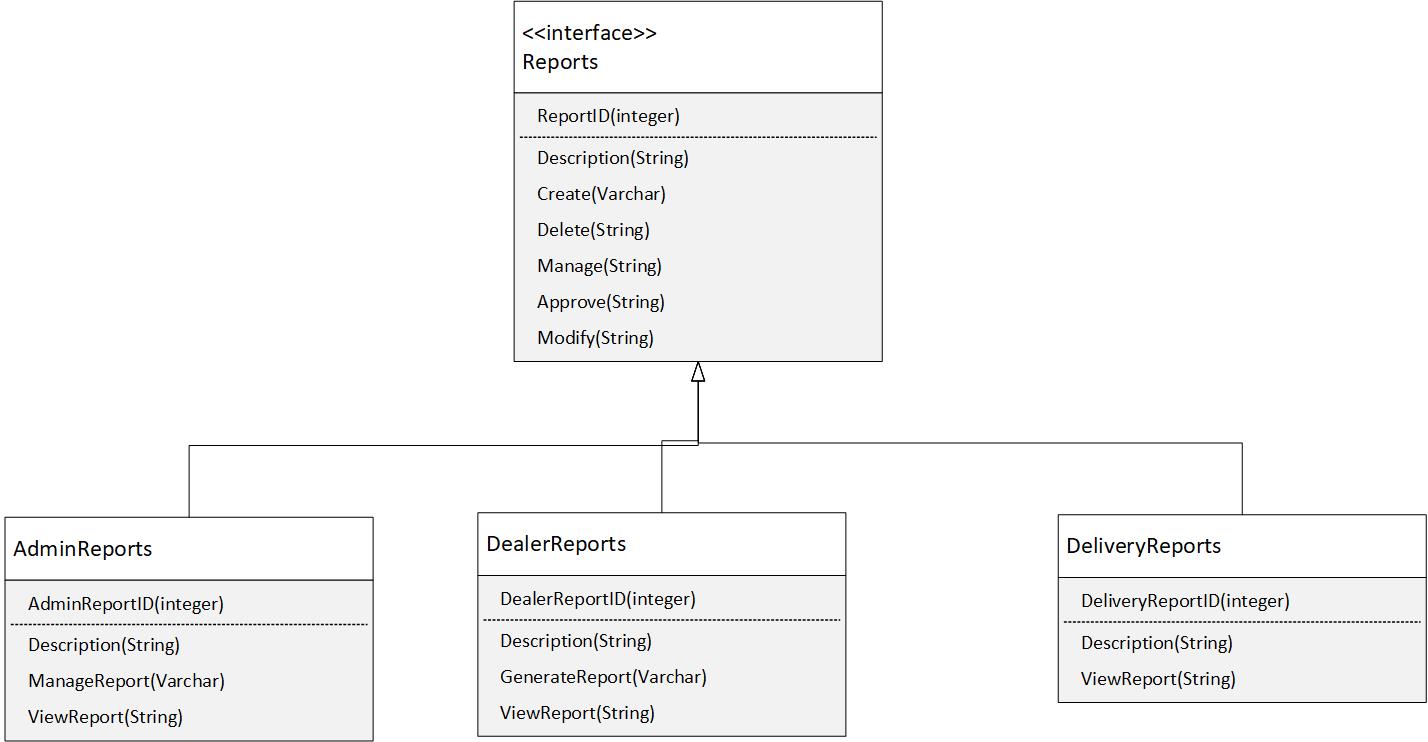


Figure : reporting classes for report package

* + 1. **Payment Package**

The payment package is important package because this has to deal with the payment options as well as how the payment methods will be securely held on the website. The payment module will include the options to be paid to website manager/admin, dealer, and delivery man who are providing services to the website. The major functions will include the payment type at checkout, and cash on driven, credit card, bank card, or PayPal as well as the payment verification option will be given by the system. The class diagram for payment package in given in following.

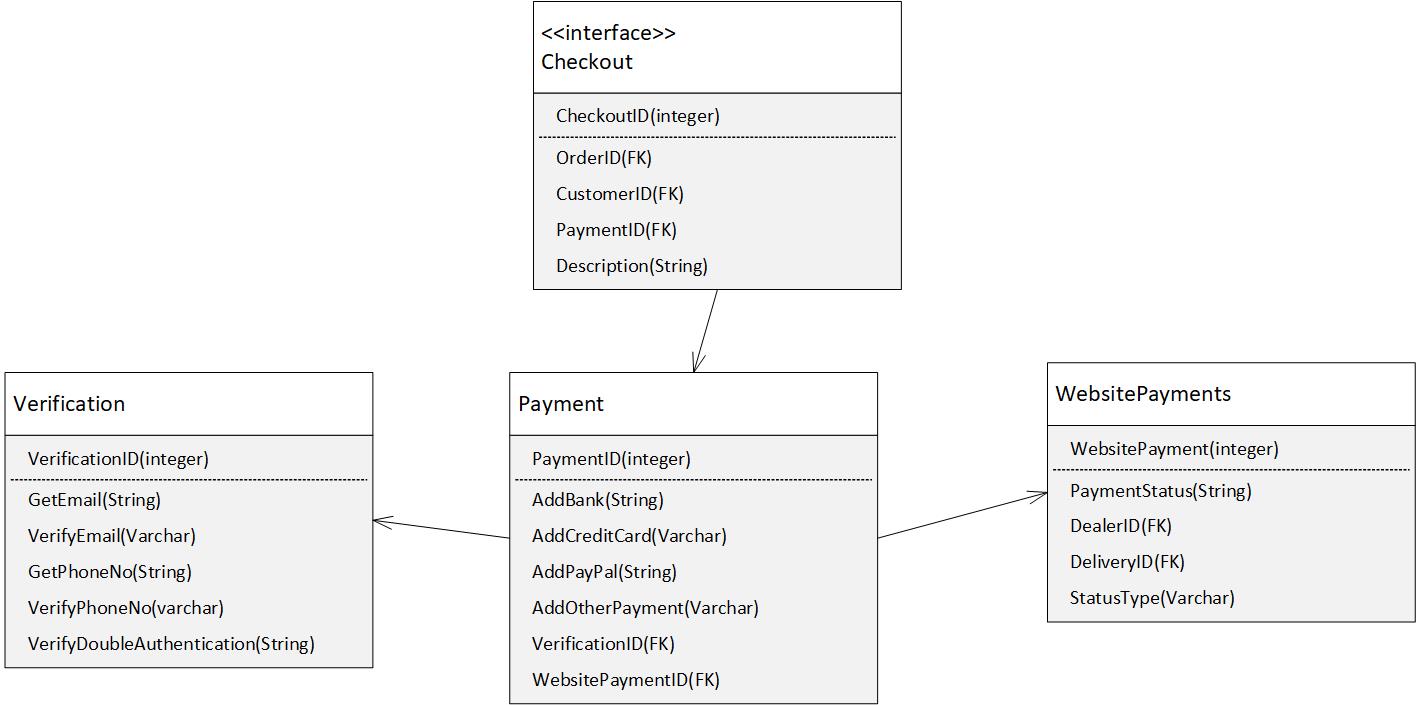


Figure : payment package class diagram

* + 1. **Orders package**

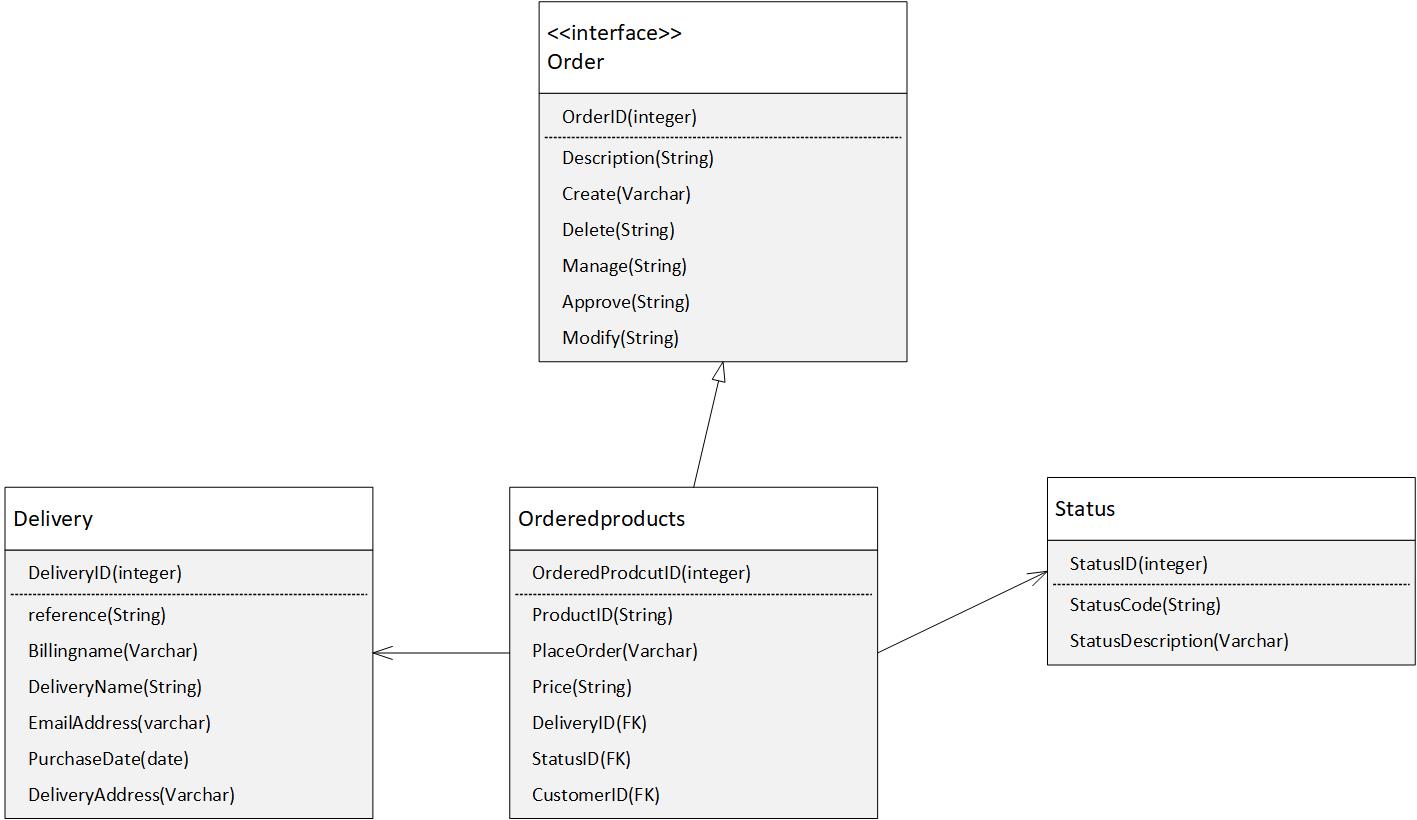
The order package has major stakeholders of customer, manager, dealer and delivery man. The customer places the order, manager manages it by telling dealer about new order and delivery man can easily deliver it to the required shipping address. The major functions of order package are place order, checkout, notify orders, clear order status, view order status, and deliver order. The class diagram for order package is given in following diagram. 

Figure : Order package class diagram

* 1. **Inter- Package Dependencies**

There are many inter-package dependencies which are already shown through façade design pattern in previous section. In this section we will summarize the inter-package dependencies in a detailed manner.

1. Package user profile is dependent on rest of package and according to the role of each user, for example the role of customer is limited as compare to role of manager, and dealer and delivery man. The inter-package dependency for this package are shown as:



Figure : inter-package dependency for user profile