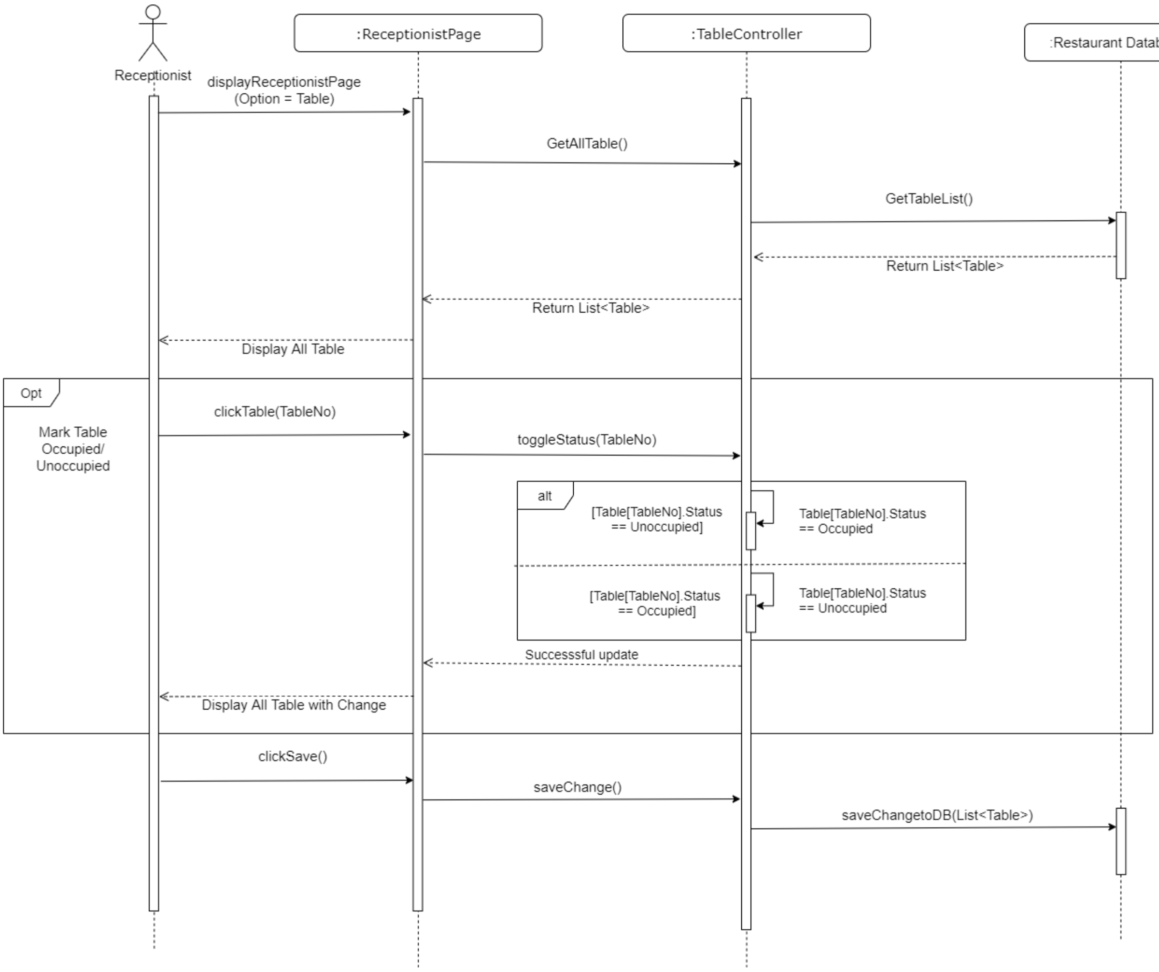
Sequence Diagram

Feature 1: Table Management


Receptionist
:ReceptionistPage
:TableController
:Restaurant Database
displayReceptionistPage (Option = Table)
GetAllTable()
GetTableList()
Return List<Table>
Return List<Table>
Display All Table
Opt
Mark Table Occupied/ Unoccupied
clickTable(TableNo)
toggleStatus(TableNo)
alt
[Table[TableNo].Status == Unoccupied]
Table[TableNo].Status == Occupied
[Table[TableNo].Status == Occupied]
Table[TableNo].Status == Unoccupied
Successsful update
Display All Table with Change
clickSave()
saveChange()
saveChangetoDB(List<Table>)

Visit this url to see the diagram better:

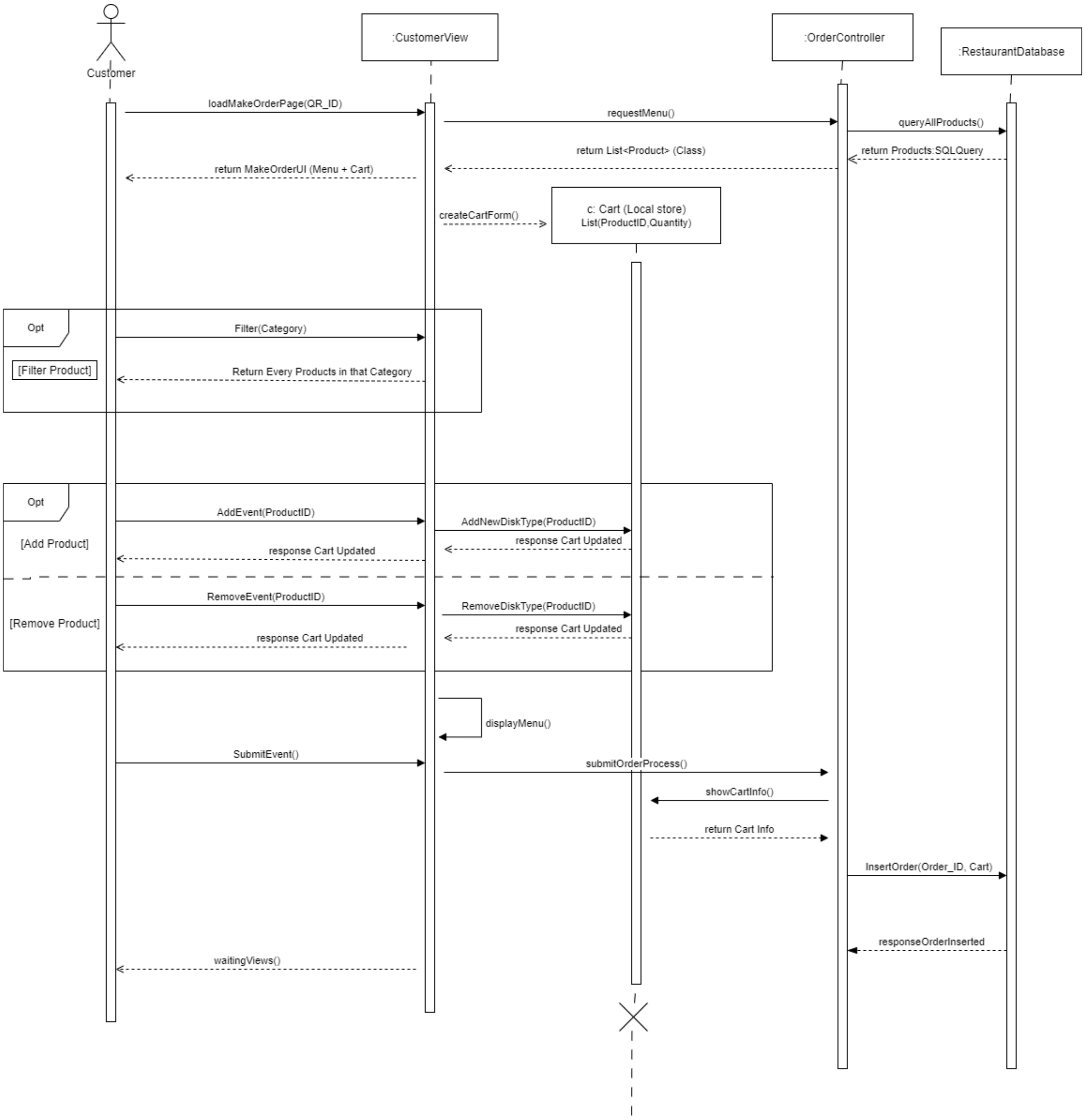
https://drive.google.com/file/d/1xDFNwIk9sSTJ1ehrfkcpwO0hmEh_2-Wk/view?usp=sharing

1. The receptionist opens the table option in the Receptionist Tab see all the table status. System sends a request to get the status of all table lists from Database by GetAllTable() and the system returns a table list (List<table>)).
2. A table status list will display on the screen to show all the tables with it's status.
3. Now, the receptionist can do as following:

   3.1 The receptionist presses the update button below a table, it's status will change from Occupied to Unoccupied or vice versa. After updating the table status successfully, the system will show the table list again with changes. After that, the receptionist clicks the save button (clickSave()) to save changes to the Database system or not. The receptionist can continue to update another table or can choose to exit.

   3.2 The receptionist just views all the table status and exit.

Feature 2: Order System (Make Order - Customer View)



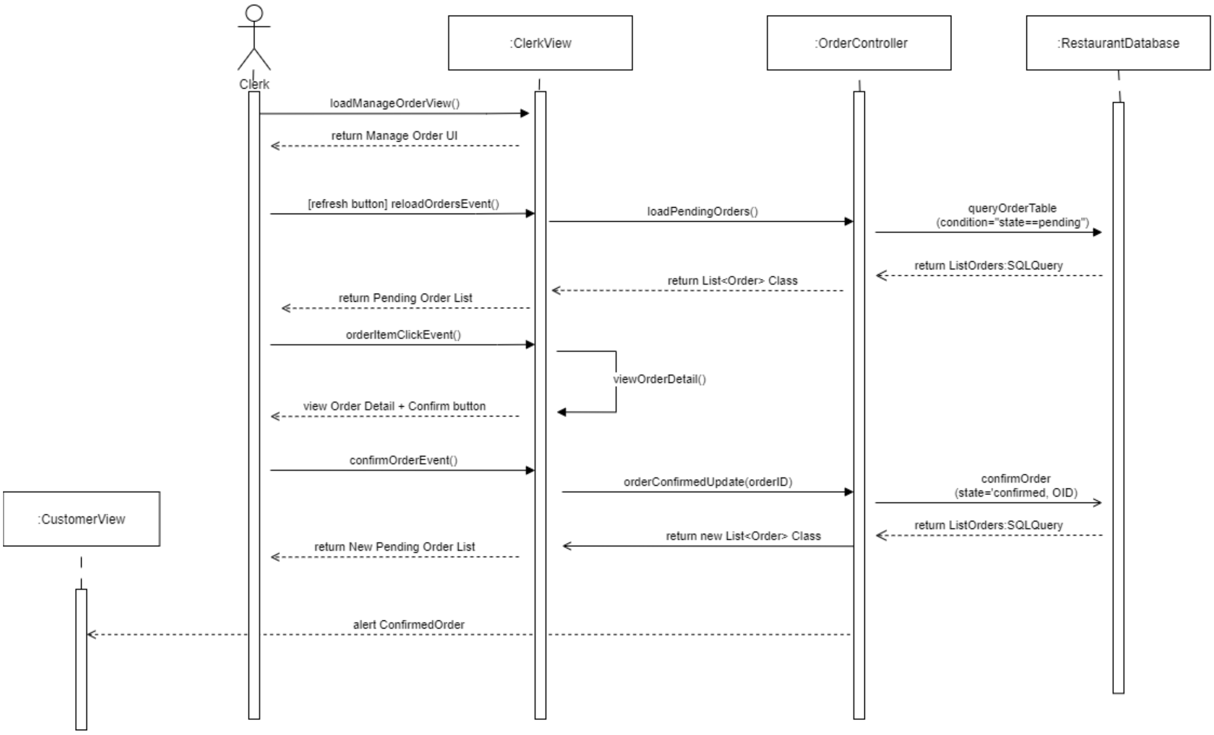
Visit this url, click on tab “Main Make Order” to see the diagram better:

https://drive.google.com/file/d/1c7ADQQkPwCfp_yUE6oIjMJ_xwM7F5mpg/view?usp=sharing

The sequence diagram above present first phase of make order process:

1. After the customer accesses the table through QR_code, MakeOrderView UI will appear at the customer screen. Before displaying, the view object will ask for Menu Info from the database.
2. A Cart object will be created and stored at customer local device to save the info about current order (disk added)
3. Customer views the menu, filters dishes according to categories, clicks the product to view its details. Customers can set the quantities and click the add button to add the disk into the cart or remove added disk from the cart.
4. After getting the wanted cart, the customer clicks the submit button and an Order object will be created with information of Date time, Cart, QR of the table that made this order, state as ‘pending’ and inserted into the database.

Feature 2: Order System (Manage Order - Clerk View)



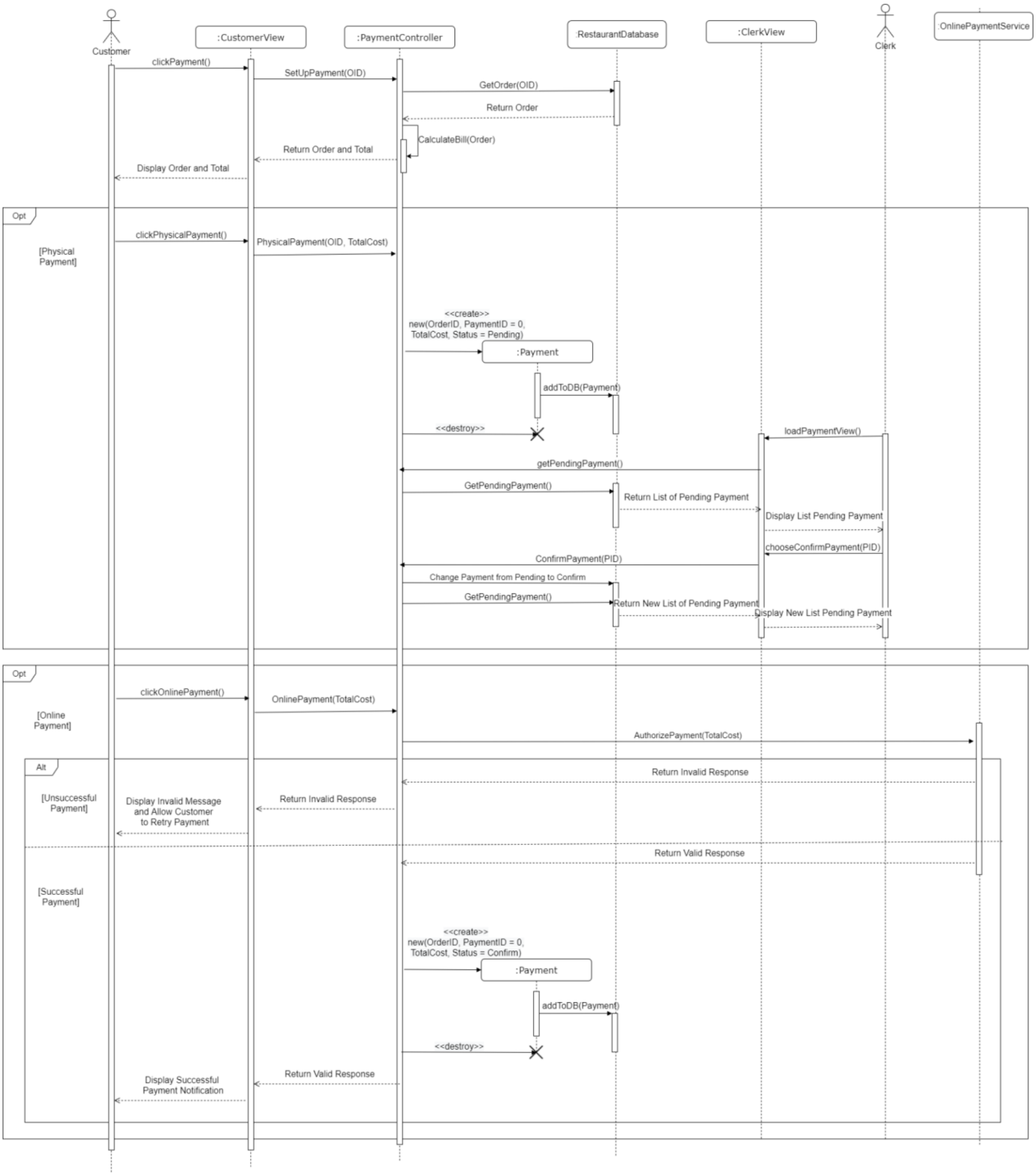
Visit this url, click on tab “Main Manage Order” to see the diagram better:

https://drive.google.com/file/d/1cpehbG53OUSMxbfNWHnr81AZmXNcrOMO/view?usp=sharing

The sequence diagram above present second phase of the make order process:

1. Clerk starts the special machine that is assigned to manage the order.
2. Clerk click the refresh button to load new pending orders from the database
3. Clerk view the detail the earliest order and click confirm, set the order’s state to ‘confirmed’
4. The customer accessing table that submits the confirmed order will be directed to the payment page, and the clerk will refresh to get a new list of pending orders.

Feature 3: Making Payment

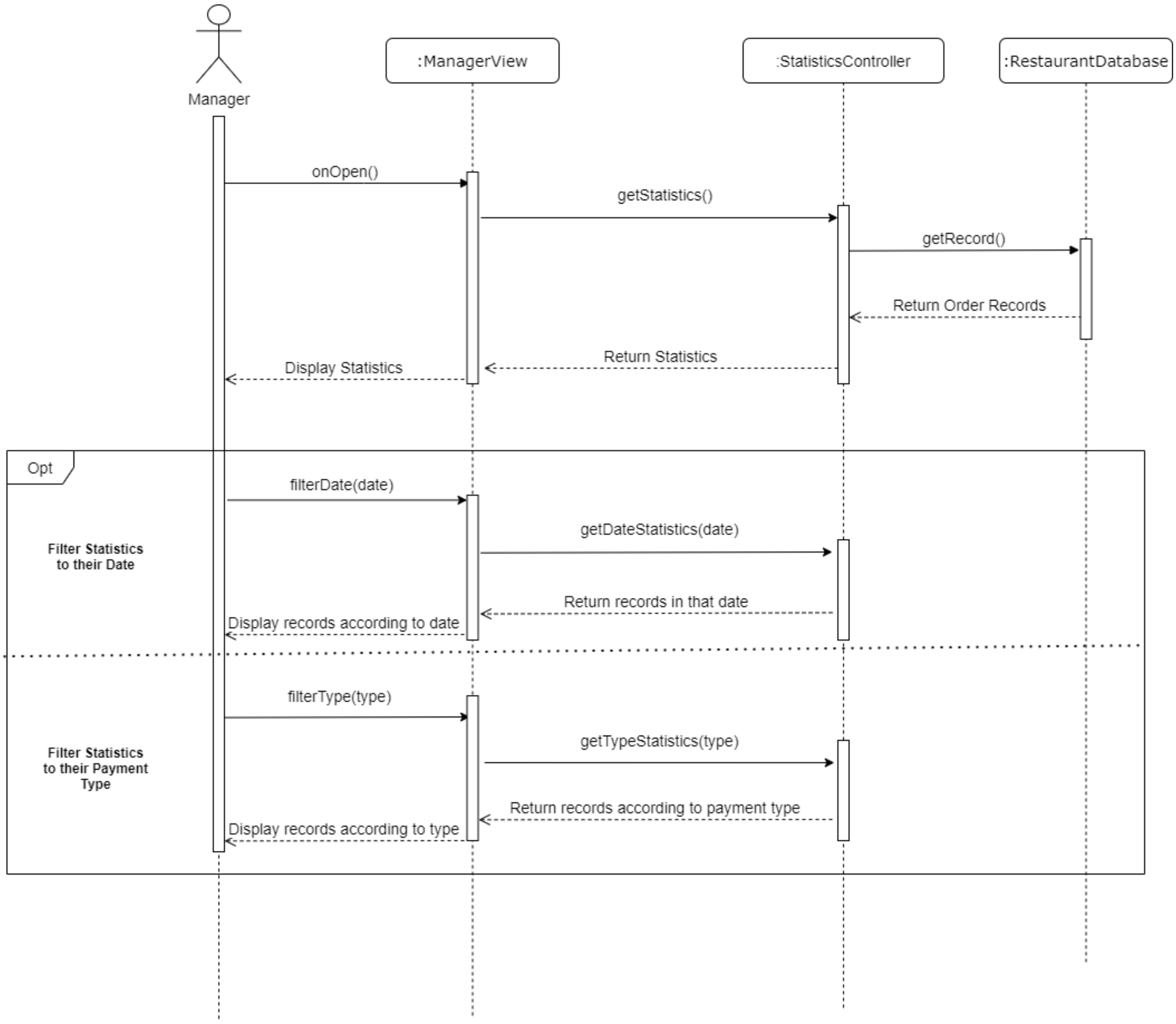


Visit this url to see the diagram better:

https://drive.google.com/file/d/1yNlu978lmTCuTrYxkh3oboYnmG2memGA/view?usp=sharing

1. Customers choose to make their payment, (clickPayment( ) is called and navigate them to the payment page) after receiving the request, payment controller retrieves the order information and calculates their bills then displays it.
2. Customer can choose two payment methods:
    - 2.1. If customers choose to make physical payment, "clickPhysicalPayment()" is called, , their payment request is sent to the payment controller to handle (PhysicalPayment(TotalCost)). Payment controller creates a payment object storing payment information and stores it to the database.
        - 2.1.1. Every time clerk view is refreshed by clerk, the view display payment pending list through method "getPendingPayment()" and return it to clerk view
        - 2.1.2. Clerk chooses to confirm a payment in the pending list, the payment status is changed from pending to confirmed. After that, a new pending payment list is displayed.
    - 2.2. If customers choose to make online payment, "clickOnlinelPayment()" is called, their payment request is sent to payment controller to handle (OnlinePayment(TotalCost)) Payment controller then send payment request to an online payment service (AuthorizePayment(TotalCost)), the service handles the transaction and returns back information to the system.
        - 2.2.1. If payment is invalid, customer view will navigate to the payment page, customer can choose to make online payment again or change to physical payment.
        - 2.2.2. If payment is valid, payment object will be created and store their payment information to the database and customer view will display the successful payment notification

Feature 4: View Statistic



Visit this url to see the diagram better:

https://drive.google.com/file/d/1RteZkRCepFY0qhjA46RbMybvvXh4qLgQ/view?usp=sharing

1. The manager gets access to the manager view.
2. The system loads the data (getStatistics()) to the manager view.
3. The Statistics Controller handles that request by sending a request to the Database.
4. The Database returns statistics to the Statistics Controller and Statistics Controller will send statistics to the manager view.
5. If the manager chooses to filter some statistics (filterDate(date), filterType(type)), the filter request is handled by the Statistics Controller and displays the manager view.