

OPTIONAL LAB

COMP8117-1-R-2021S

SUBMITTED TO:

Dr. Aznam Yacoub

21-08-2021



SUBMITTED BY:

ANUBHA SHARMA

110037181

**SOFTWARE:** Ferns and Petals [1]

**Need**: It is a system used in India whichhelps in booking and gifting cakes, flowers and some personalized items like cards, photo mugs etc., for various occasions and in various cities in the India. If a user is stationed out of their home city and wants to surprise their loved ones, they will be able to use this platform to do so. The option of customizing the item and selecting delivery date and time will be provided.

**Scope of this report**: here I have only defined how the core logic of the system will be organized, i.e., the application engine.

**5 features and their specifications.**

* **Search Item:**

The user should be able to search the item they want from the search bar and a list of related items will be displayed to the user. If no related items are found, the error message will be displayed.

* **Customize Item:**

Once the item is selected, user can use the customize option, if available (depends on the item being ordered)

* **Place Order:**

For placing the order, the user has to select a delivery location and date and time when the order has to be delivered. The system will access if the delivery can be made. Based on that the order proceeds to payment page or the error notification pos up.

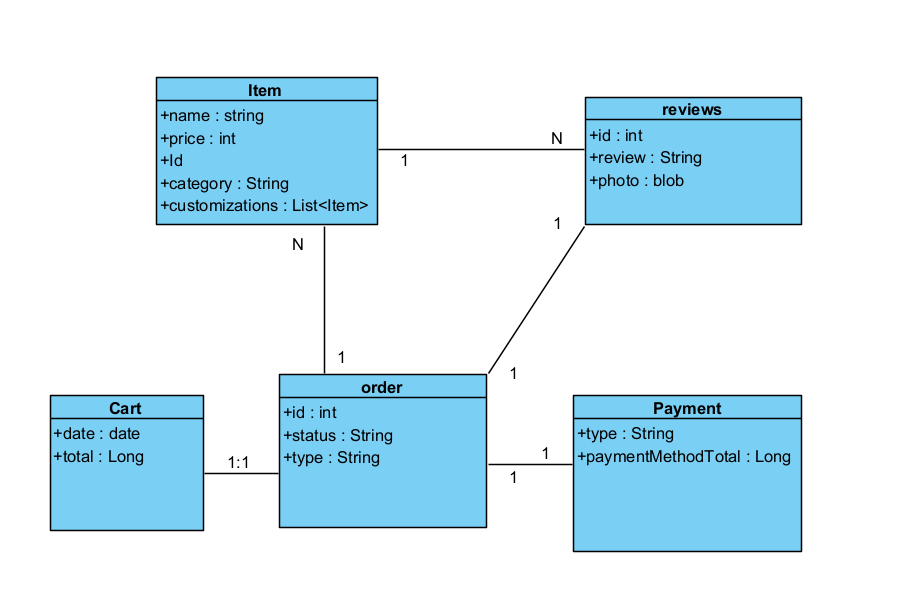
* **Payment:**

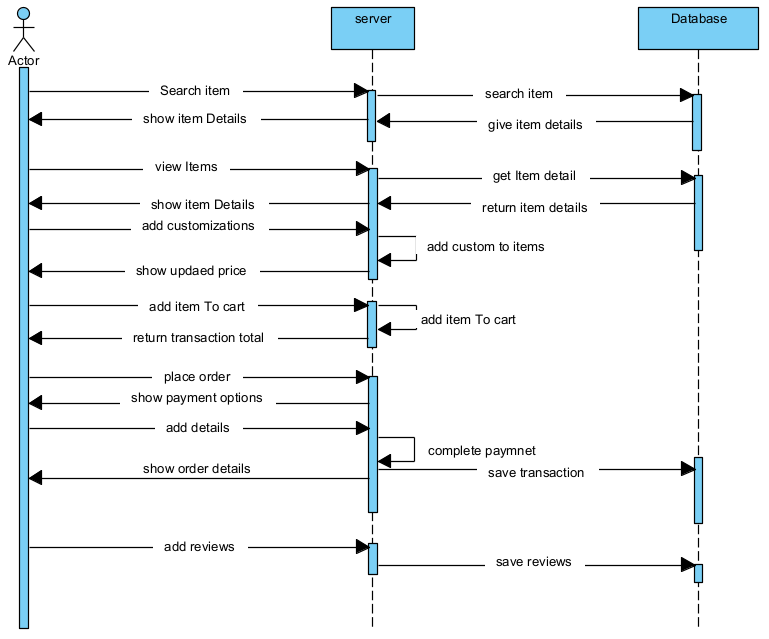
The customer should be able to make the online payment through debit/credit card PayPal. We should be able to add the payment methods in the future as well

* **Reviews:**

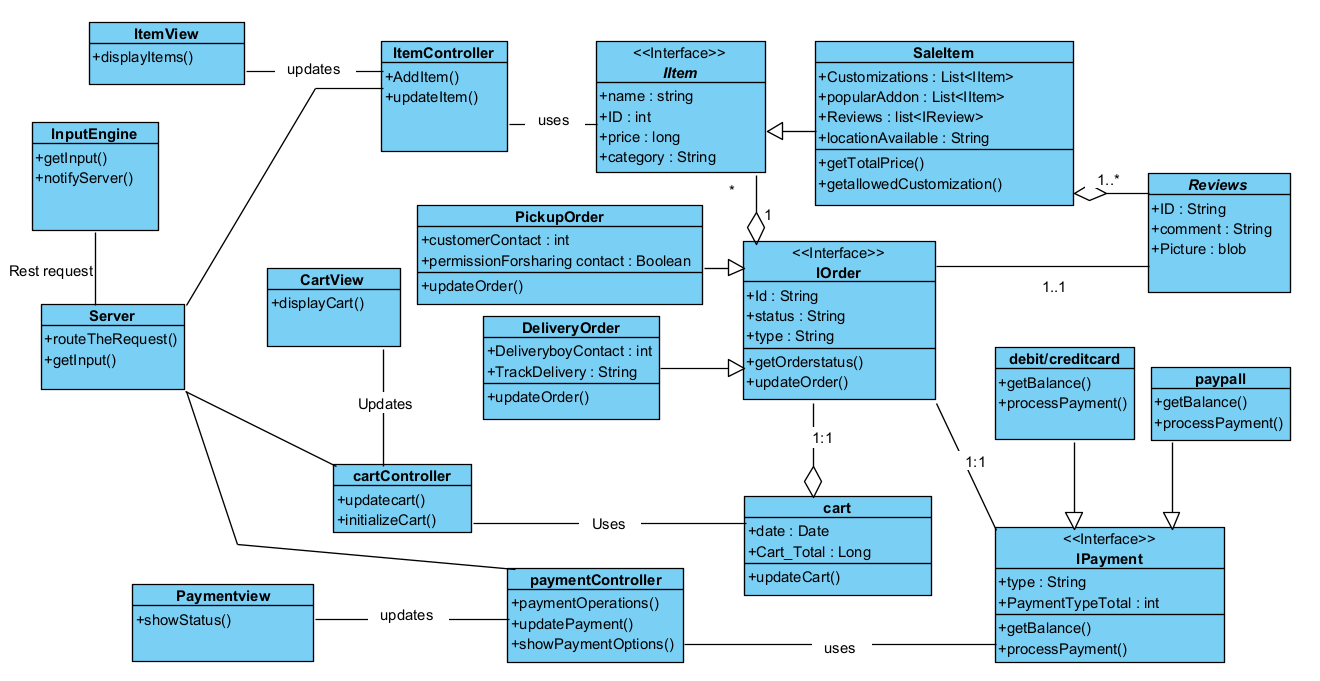
The customer should be able to review the and Order items received

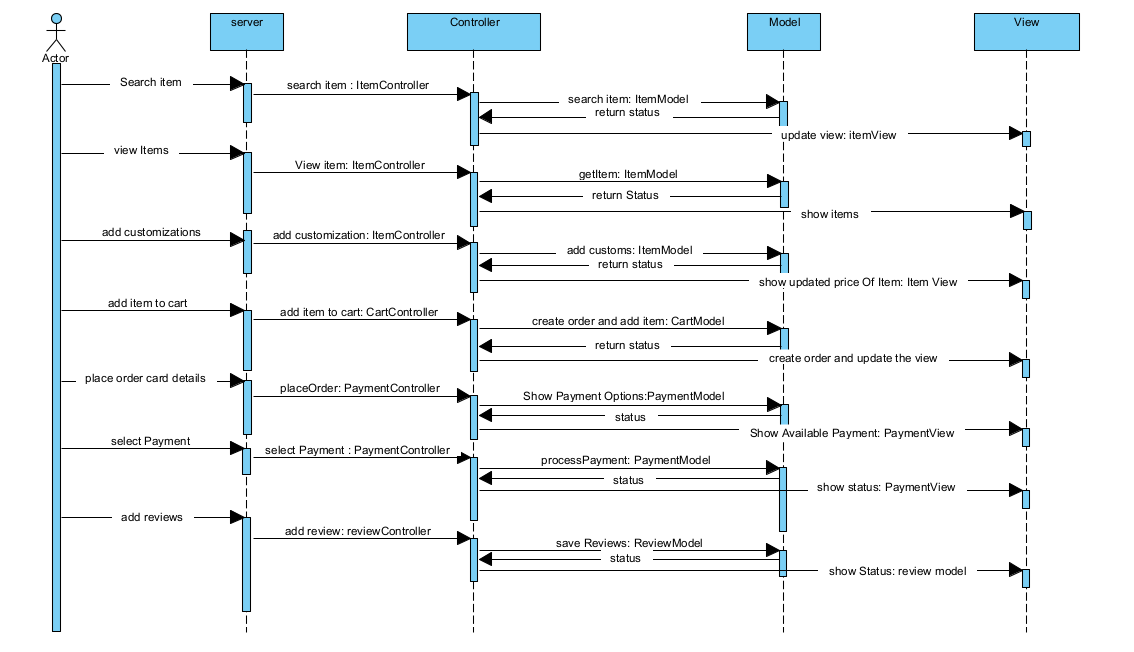
The specifications are expressed in formal manner as below:





**High Level Diagram:** I have discussed in detail the arrangement of only the model part





**TEST CASE SPECIFICATION:**

Below are the test cases we will be executing to ensure that our application logic is working as expected. Since we are working with the business logic layer of the code, we will be doing white box testing to be performed by the developer. We will use Postman to test our server and controllers and see if they are correctly working and the request is getting handled to the correct controller.

**Case Search Item:**

**Test Case 1:**

1. Search an item category “CAKE” or “cake” or “CaKe”.
2. All the items related to only cake should be returned by the API.

**Test Case 2:**

1. Search an item category “FLOWER” or “flower” or “FloWer”
2. All the items related to only cake should be returned by the API.

**Test Case 3:**

1. Search an item category “GIFTS” or “gifts” or “GiFts”
2. All the items related to only cake should be returned by the API.

**Test Case 4:**

1. Search a category by inputting wrong values “acbde” or any other wrong values.
2. Error message should be returned “Please input right values”.

**Case Customize Item:**

**Test Case 1:**

1. Input an item
2. See that the API return correct item details like photo, description, price, previous reviews and customization option details to be used by the View.

**Case Place Order:**

**Test Case 1:**

1. placing an order by giving an item Id delivery date, delivery time and delivery location, customization details.
2. We should get an Order number and order details should be correctly saved in the table.

**Test Case 2:**

1. Place an order by do not give a delivery date.
2. API should return an error message saying that delivery date is missing.

**Test Case 3:**

1. Place and order by not giving a delivery time.
2. API should return an error message saying that delivery time is missing.

**Test Case 4:**

1. Place an order by do not give any customization details.
2. We should get an order number; the order should be placed with the base item.

**Test Case 5:**

1. Place and order by not giving a delivery location.
2. API should return an error message saying that delivery time is location.

**Case Payment:**

**Test Case 1:**

1. To test the payment, miss the CVV from the card details of the credit Card.
2. The system should not proceed to the payment gateway if all the mandatory fields are not present

**Test case 2:**

1. To test the payment, miss the card from the card details of the credit Card.
2. The system should not proceed to the payment gateway if all the mandatory fields are not present

**Test Case 3:**

1. To test the payment, miss the expiry date from the card details of the credit Card.
2. The system should not proceed to the payment gateway if all the mandatory fields are not present

**Test Case 4:**

1. Test The card with valid card number, valid CVV, valid expiry.
2. The system should proceed to the payment gateway and the amount should be deducted

**Test Case 5:**

1. Test the payment gateway with Valid card, valid CVV, invalid expiry date
2. The system should return us the error.

**Test Case 6:**

1. Test the payment gateway with Valid card, invalid CVV, valid expiry date
2. The system should return us the error.

**Test Case 7:**

1. Test the payment gateway with invalid card, valid CVV, valid expiry date
2. The system should return us the error.

**Test Case 8:**

1. Test the payment gateway with a blocked card information.
2. The system should return us the error

**Case Reviews:**

**Test Case 1:**

1. Using post man, hit the API with the message and Item Id.
2. The system should recognize that review is for the item Id, The Item Id begins with ITM. Review should be saved for that item

**Test Case 2:**

1. Using postman, Hit the API with message and order Id.
2. The system should recognize that review is for the total order. The order Id begins with OD. The system should recognize that review is for the total order and save it for that order.

**NOTE:** I don’t know the code of this system, hence I am not sure how to dissemble the code and to confirm and deny the assumptions.

# References

|  |  |
| --- | --- |
| [1] | A. Sharma, "Training Lab 1". |