Project documentation

Software Engineering 2

**Cover Sheet**

# Project Title: Restaurant Reservation system

# Project Description:

# Reservation system helps customer to reserve table online , customer can view the menu to see what he can order at the restaurant

# Admin of the system can add/update/delete another admin

# He can add/update/delete meal

# He can confirm reservation

# Main functions:

- MakeReservation(): The customer can make the reservation and waits the approval from the admin.

- UpdateReservation(): customer can update his reservation data .

- addEvent() : the customer can win in restaurant a special code that customer can with it make a special event like birthday party ,he could enter his code and description of the event with reservation data to reserve the event .

-ViewMenu(): Customer can see the menu of the restaurant .

-ConfirmReservation(): admin can confirm reservation which the customer has made.

-AddMeal(): Admin can add new meal to the menu.

-AddAdmin():Admin can add another admin to the system to control it.

|  |  |  |
| --- | --- | --- |
| Requirement | points |  |
| Backend project with RESTful APIs | 6 |  |
| DB architecture (schema and data) | 2 |  |
| SOLID principles | 3 |  |
| Clean Code | 2 |  |
| UML diagrams | 5 |  |
| OCL constraints | 2 |  |
| Total | 20 |  |

# Solid principles :

# Principles have been applied in our project:

# Single Responsibility Principle

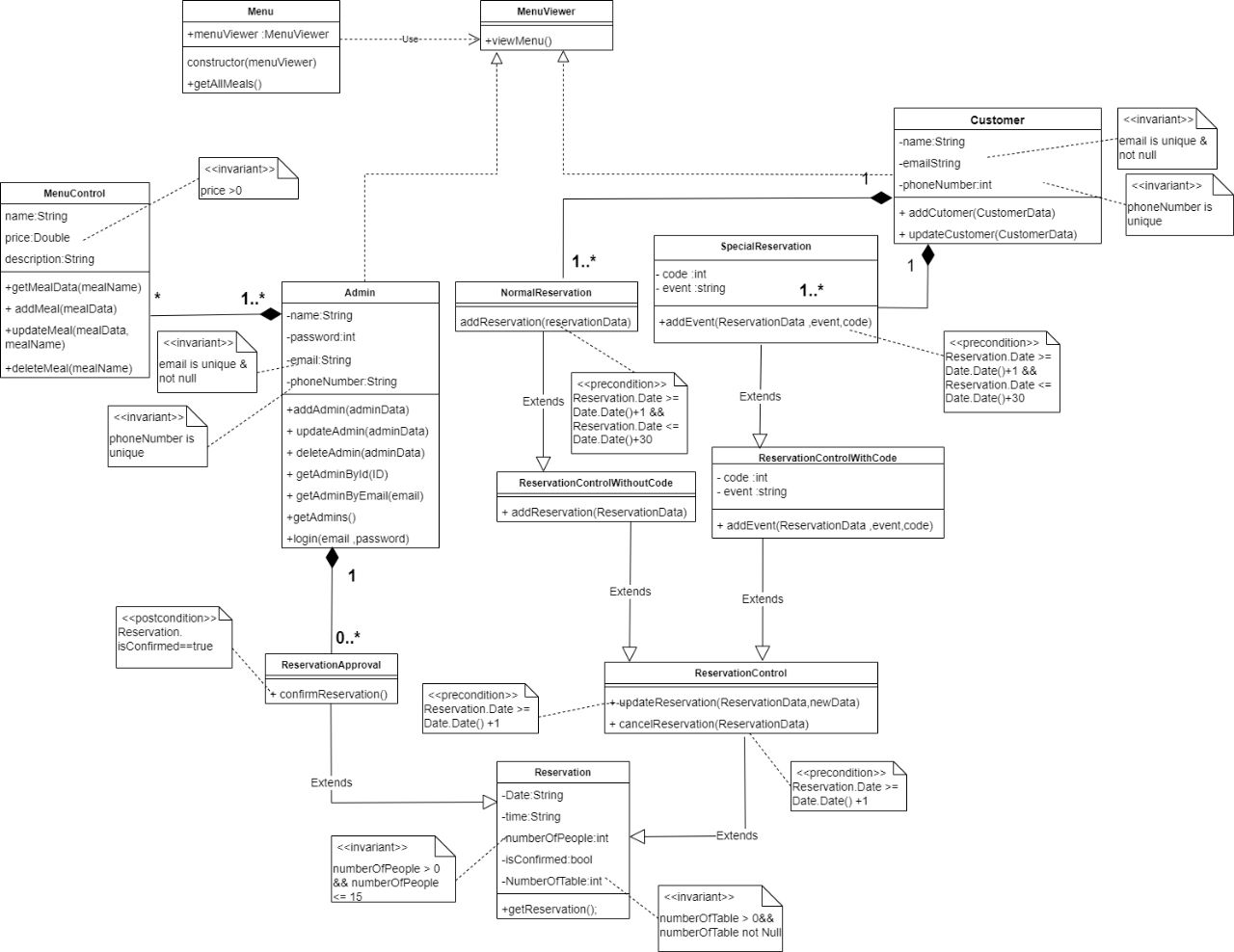
# Open Closed Principle

# Liskov Substitution Principle

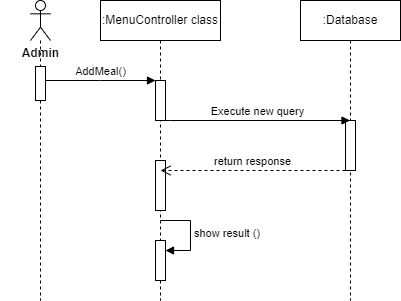
# Dependency Inversion Principle

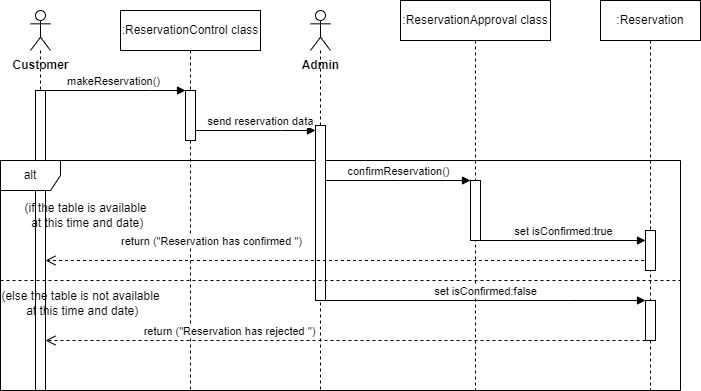
UML Diagrams

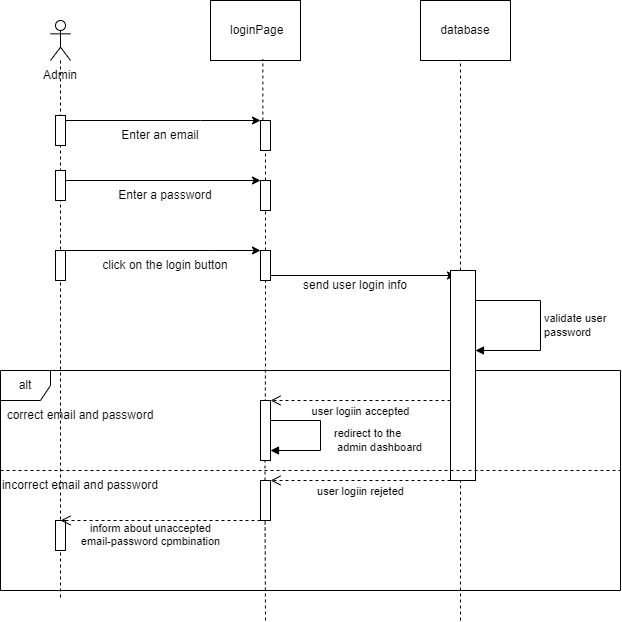
1-class diagram:



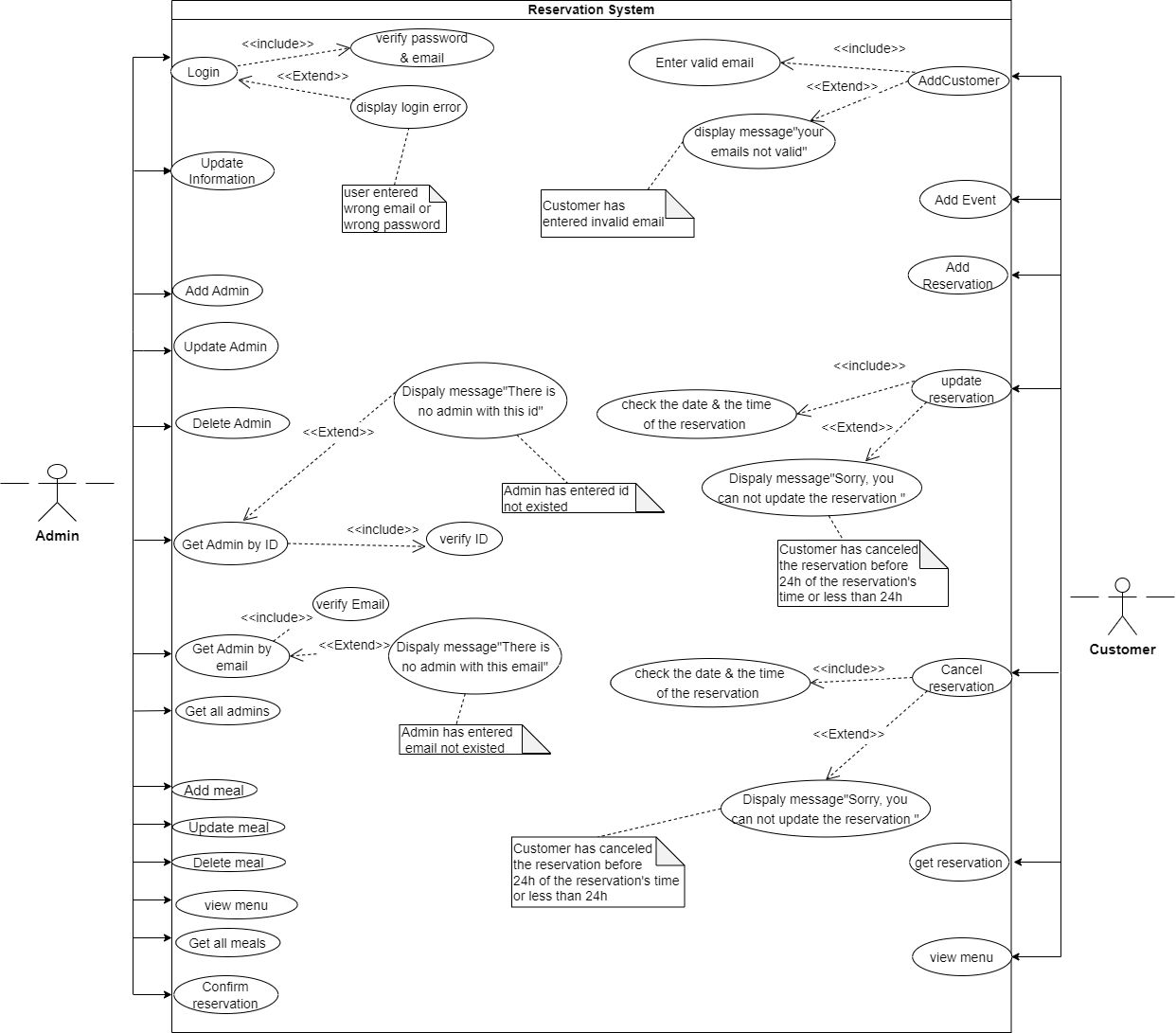
2-sequence diagram:



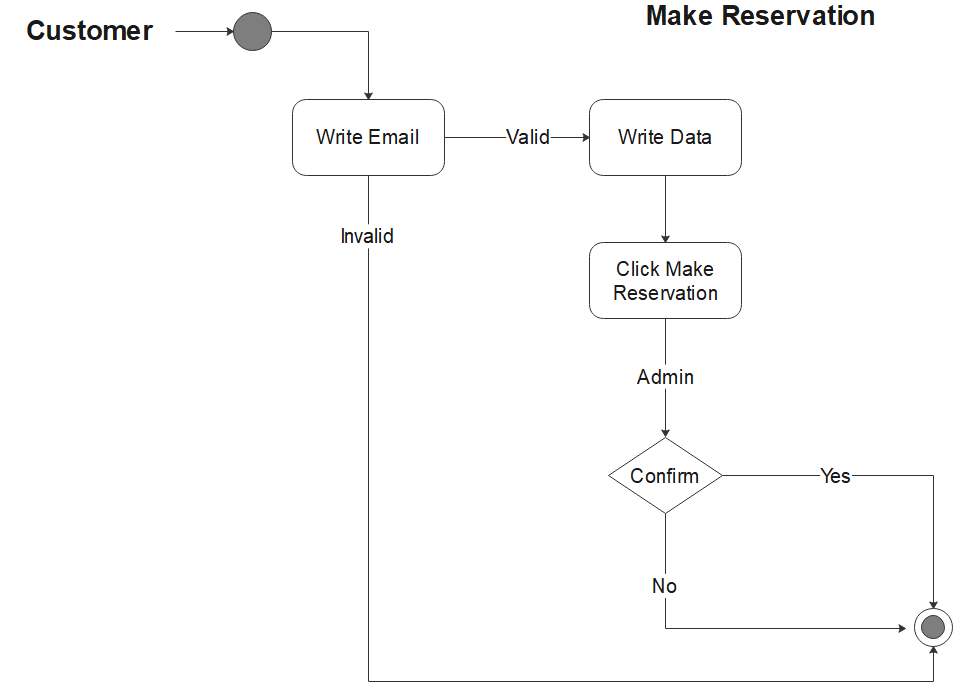


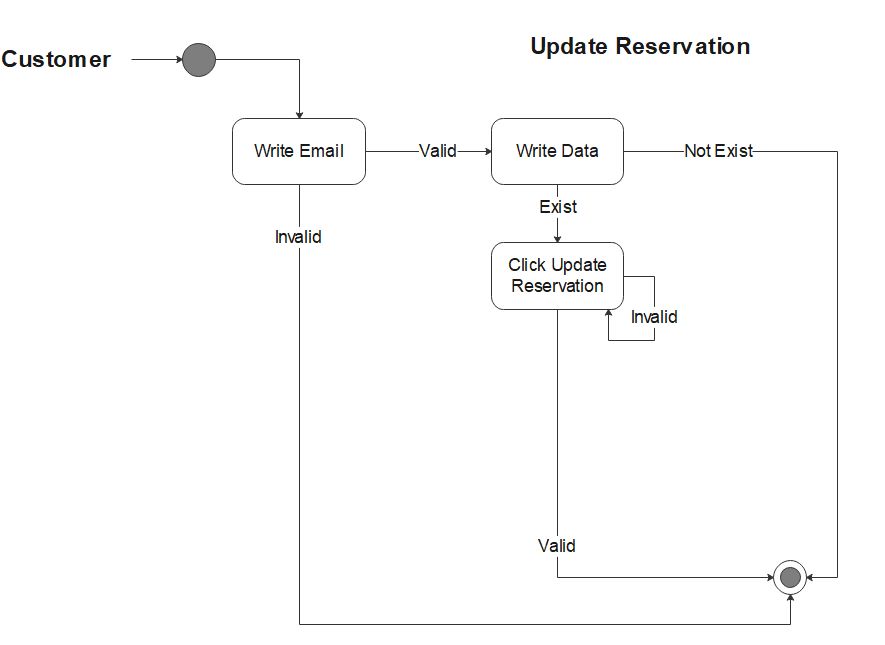


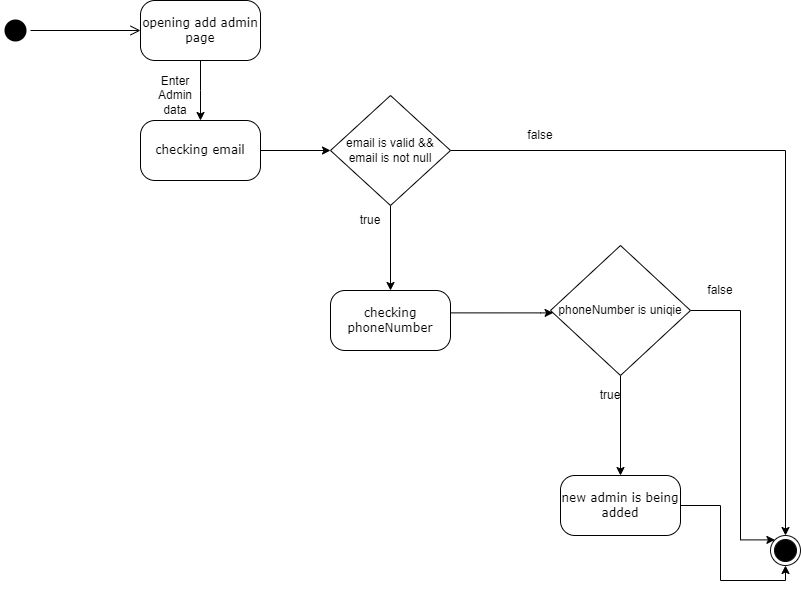
3-useCase diagram:



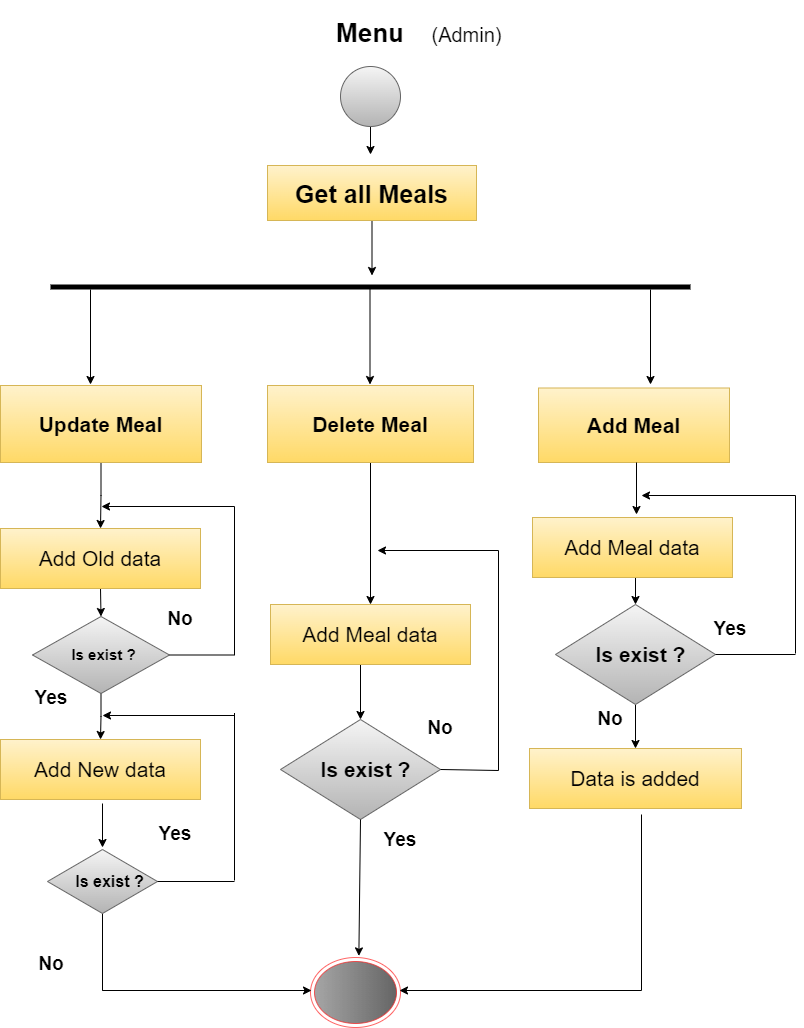
4-state diagram

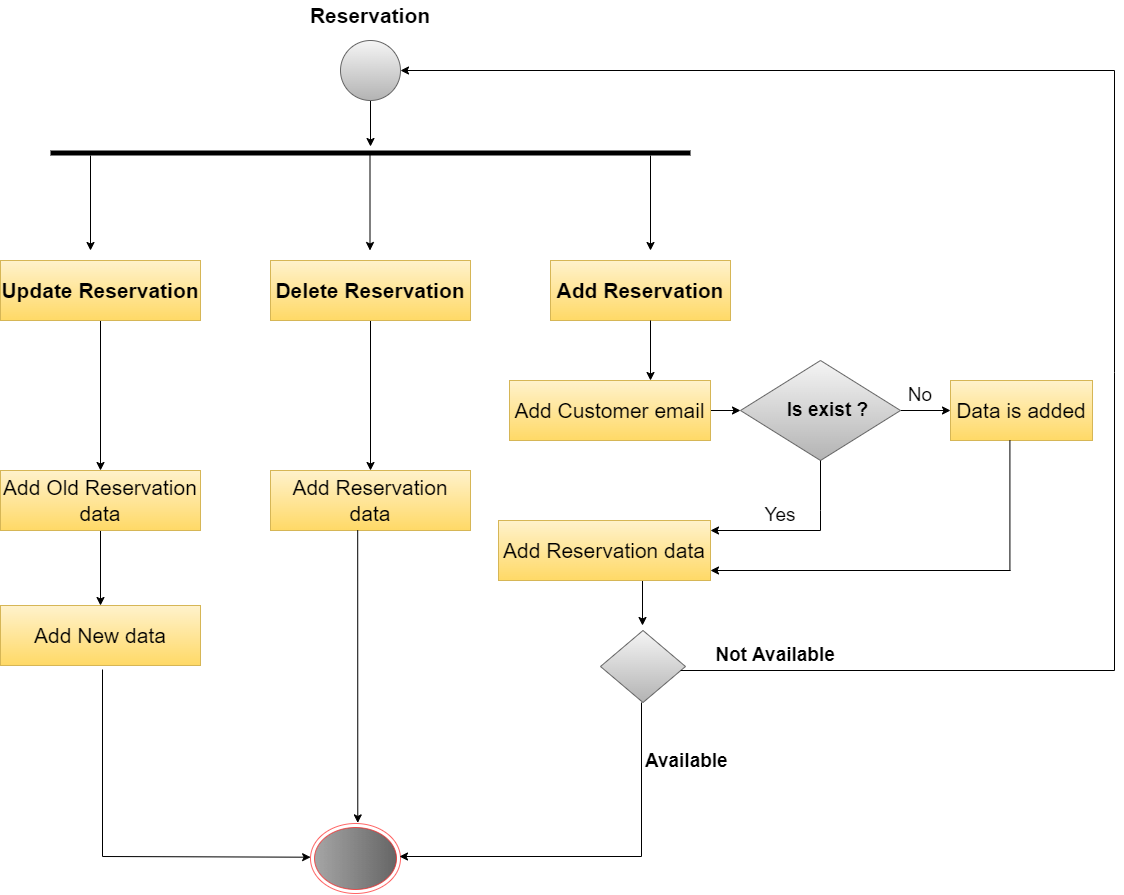


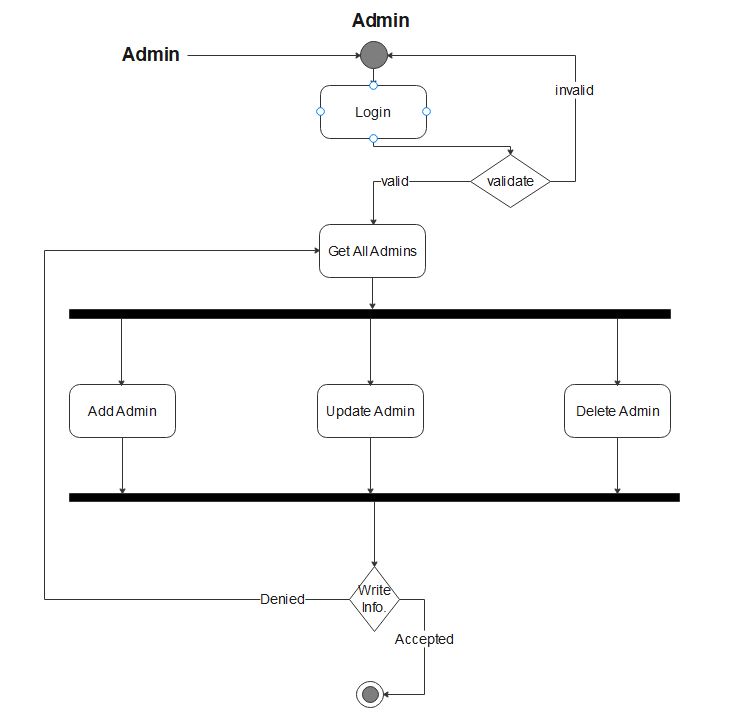




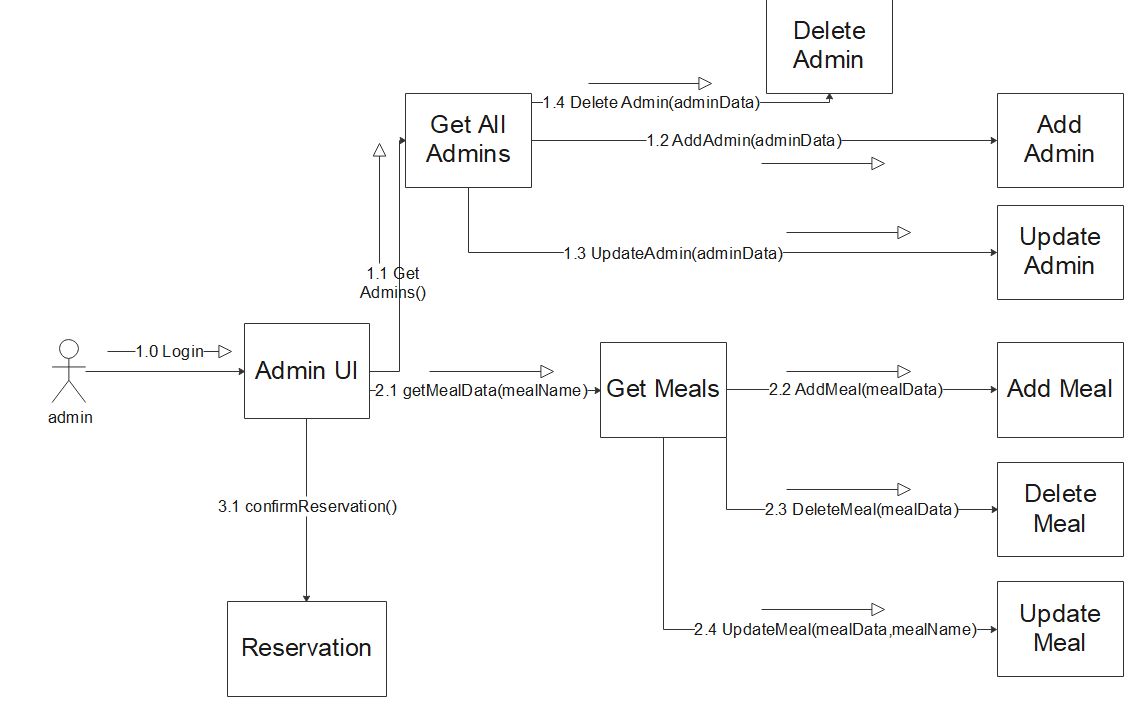
5-Activity Diagram:

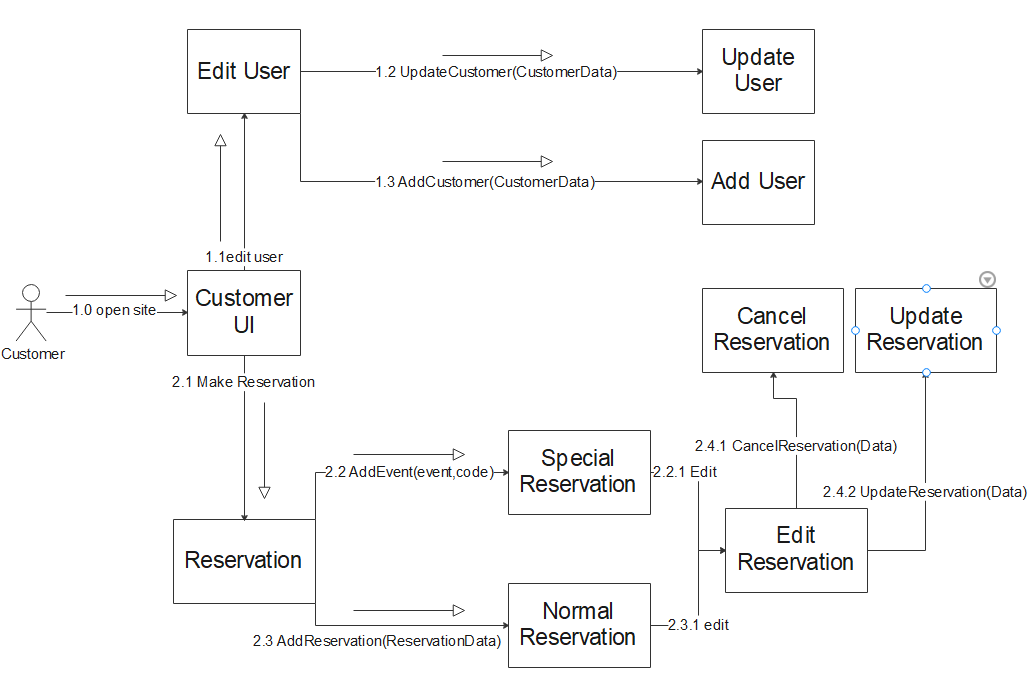






6-communication diagram





7-dataBase schema (diagram)

