**CLASS Responsibility Collaborators (CRC)**

1. ISubject

|  |  |
| --- | --- |
| **ISubject** | |
| Subject for waitlisthandler maintains ordered list of table. | TableList |
| It notify Observers after a table becomes available |  |
| Removes the customer from the waitlist table | CustomerList |

1. WaitListHandler

|  |  |
| --- | --- |
| **WaitListHandler** | |
| Handles the queue that record all the waitlisted customers | Customerlist |
| Matches the appropriate table for customer on the basis of size | TableList |
| Removes the customer from the waitlist table | CustomerList queue |

1. CustomerList

|  |  |
| --- | --- |
| **CustomerList** | |
| Maintains a queue that record all the waitlisted customers | CustomerList queue |
| It will store Name, cellphone and email ids of all the users |  |

1. IObserver

|  |  |
| --- | --- |
| **IObserver** | |
| Give customer info to the CustomerList | Customerlist queue |
| It is responsible to confirm the reservation | Customerlist queue |
| It is responsible to cancel the reservation | CustomerList queue |

1. TableList

|  |  |
| --- | --- |
| **TableList** | |
| Maintains a queue that record all the tables | WaitList Handler |

**Explanation for design patter usage**

1. Observer Design Pattern:

It have been implemented as each time when a new table or customer is available, the subject will notify the Observer. Each time customer will update and take decision whether to confirm or cancel the reservation.

1. Chain of responsibility Design Pattern:

WishListHandler will handle all the tableList and CustomerList. It will match the customers and tablelist according to the table and customer party size.