

Object Oriented System Development Lab

Software Requirements System for Online Restaurant System

Submitted by:

Jahnavi (160115733125)

Jyothi (160115733126)

Komal (160115733127)

Krishna Sri (160115733128)



**Department of Computer Science and Engineering
Chaitanya Bharathi Institute of Technology
Gandipet, Hyderabad – 500075**

Table of Contents

1. Introduction

- 1.1 Purpose
- 1.2 Scope

2. Interface Requirements

- 2.1 Hardware interfaces
- 2.2 Software interfaces

3. Intended Users

- 3.1 Customers
- 3.2 Staff

4. Functionality

- 4.1 Description
- 4.2 Security features
- 4.3 Constraints

5. Diagrams

- 5.1 Class Diagram
- 5.2 Object Diagram
- 5.3 Use case Diagram
- 5.4 Sequence Diagram
- 5.5 Collaboration Diagram
- 5.6 Activity Diagram
- 5.7 State Chart Diagram
- 5.8 Component Diagram
- 5.9 Deployment Diagram

1. Introduction

1.1 Purpose:

This project aims at making it easier for restaurants to manage their orders and reservations while also enabling the customers to place their requests well ahead of time, so that there would not be the hassle of waiting in queues or staying on hold on the phone lines.

1.2 Scope:

This project will let customers place their take-away orders online and to make their reservations. The system will ensure that their requests are processed in meaningful time and that the customers stay updated throughout the process. The hotel staff also interact with the tool to keep up to date with their orders and reservations to manage, removing the need for manual logging.

2. Interface Requirements

2.1 Hardware Interfaces:

Either of the following devices are needed:

- Smartphones with iOS 6.1.6+ or Android 5.0+
- PCs/Laptops that can run Chrome 32.0.1700+

2.2 Software Interfaces:

- Project .apk/.ipa installed on the smartphone
- Chrome 32.0.1700+ with JavaScript enabled

3. Intended Users

3.1 Customers:

A customer will need a valid email address to register and will be asked to set-up a password, both of which will have to be entered while logging in. The portal can be accessed either through the mobile application or through the web application. Once logged in, customers can browse the restaurant menu and place their take-away order. They will be updated at each step: on order confirmation\rejection, food preparation status, packaging status etc. Customers can also place their reservations on this portal.

3.2 Staff:

The restaurant staff will need to access the portal to manage the various orders and reservations. Each staff member with access will be assigned a role, keeping privacy issues in mind. They can be uniquely identified using their employee ID which is used while signing up and logging in. The staff will need to confirm/reject requests and orders, update current status of customers' orders, coordinate with offline staff, etc.

4. Functionality

4.1 Description:

The project will be developed using the MERN stack owing to its scalability and using React-Native would make the project platform independent. Although we have specified a particular minimum version of the Chrome browser, most other browsers are supported too, but all the features will have to be thoroughly tested.

The customers can use the portal for: 1. ordering take-away, 2. placing reservations. They get to browse the menu and choose what they would like to order.

The staff receive an alert when an order/request is made and with every order/request, one staff member is assigned to look over its successful completion. They will have to coordinate with the offline staff and records, for instance, to ensure that a table is not already reserved through an offline request.

4.2 Security Features:

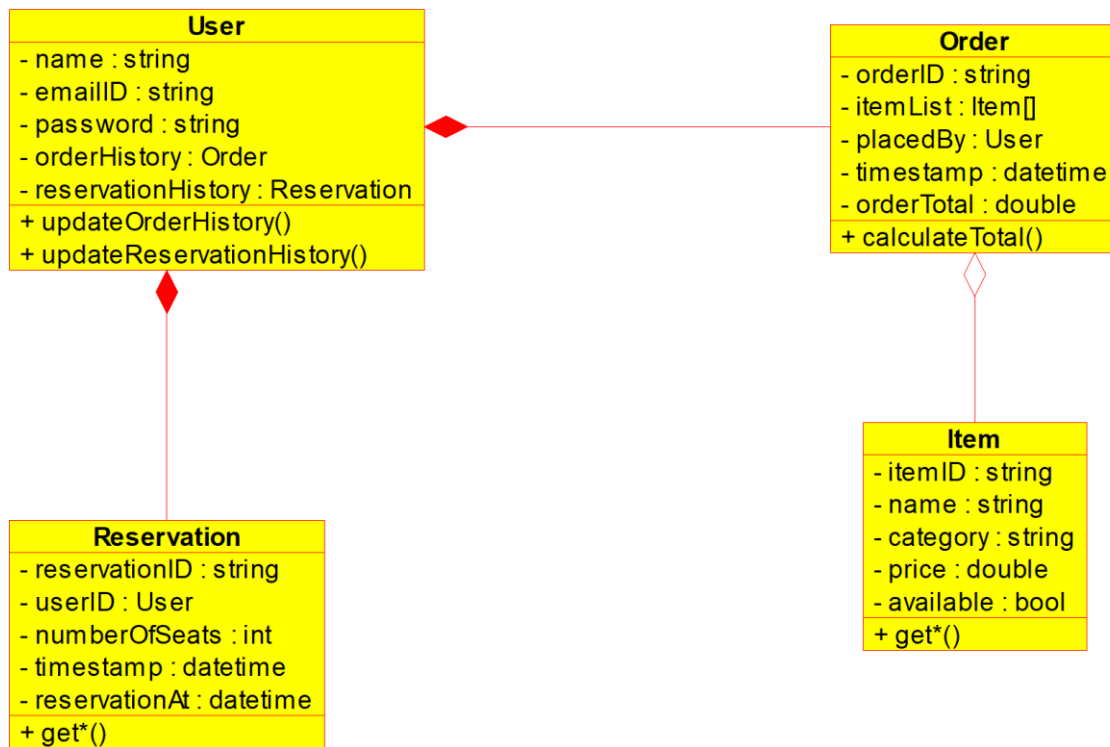
Using frameworks help with abstracting away the security issues with the Web to the framework implementation. Another issue to keep in mind is the customers' privacy. For instance, a staff member could misuse their house address, which is why each staff member is assigned a role, according to which the information regarding a user is visible to them.

4.3 Constraints:

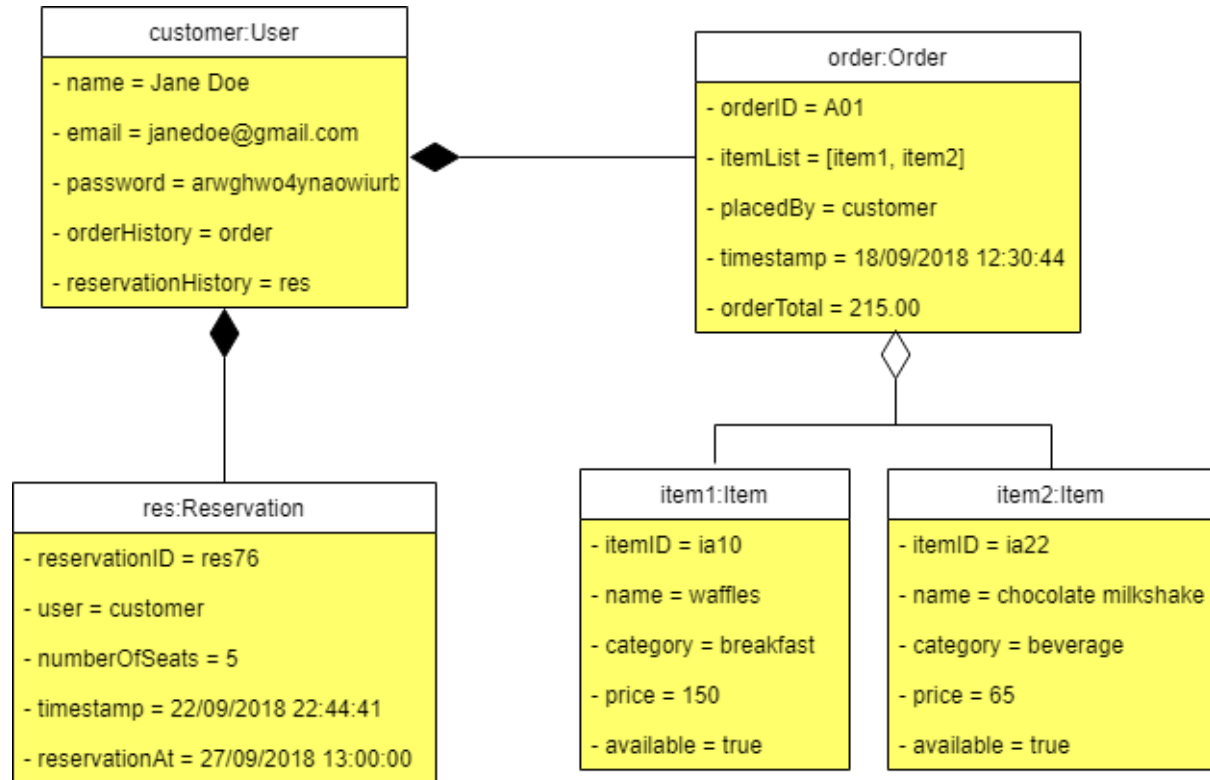
Although the system makes ordering take-away easy, the problem is with the non-availability of delivery service by the restaurant. A potential solution would be hiring a third party service such as Swiggy, Zomato, etc.

5. Diagrams

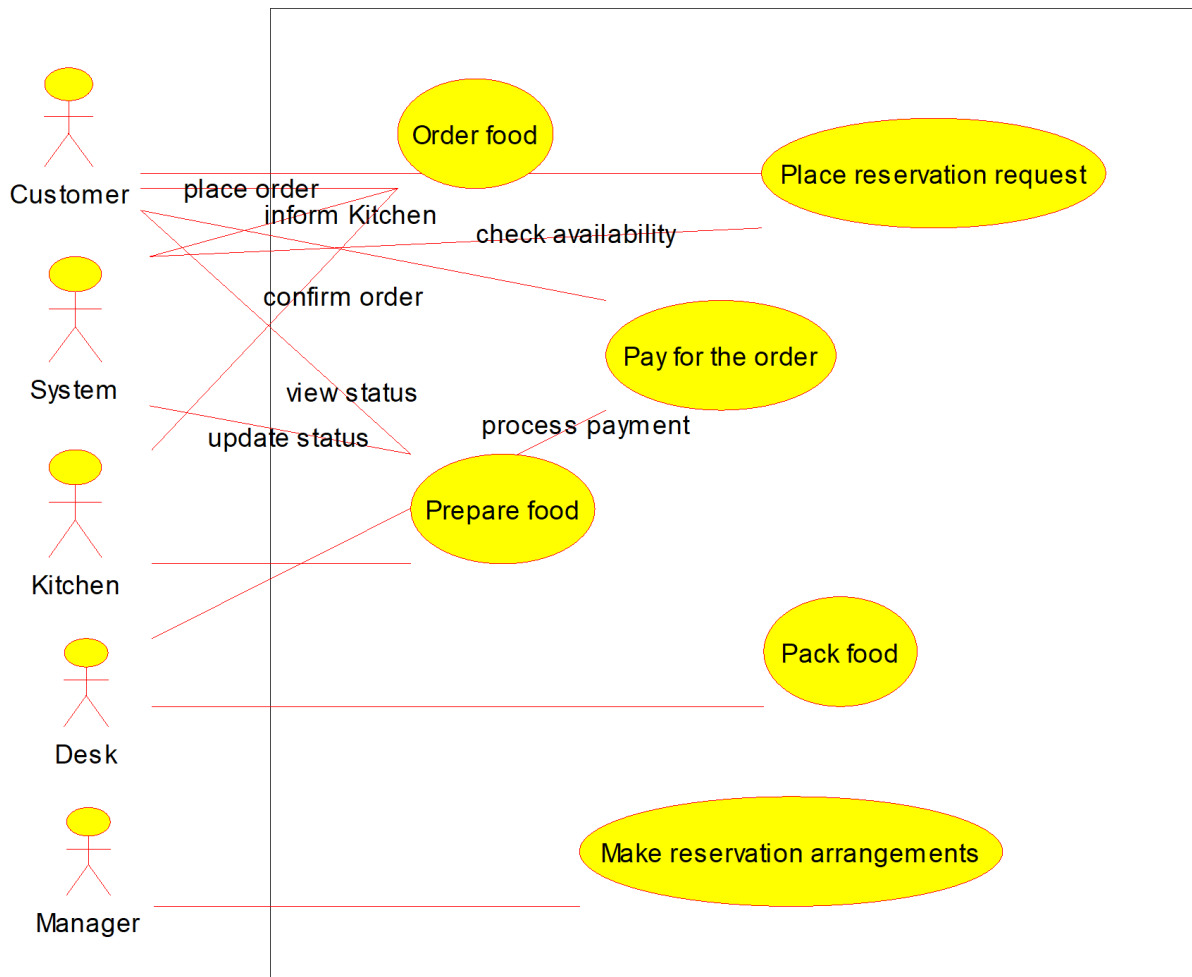
5.1 Class Diagram



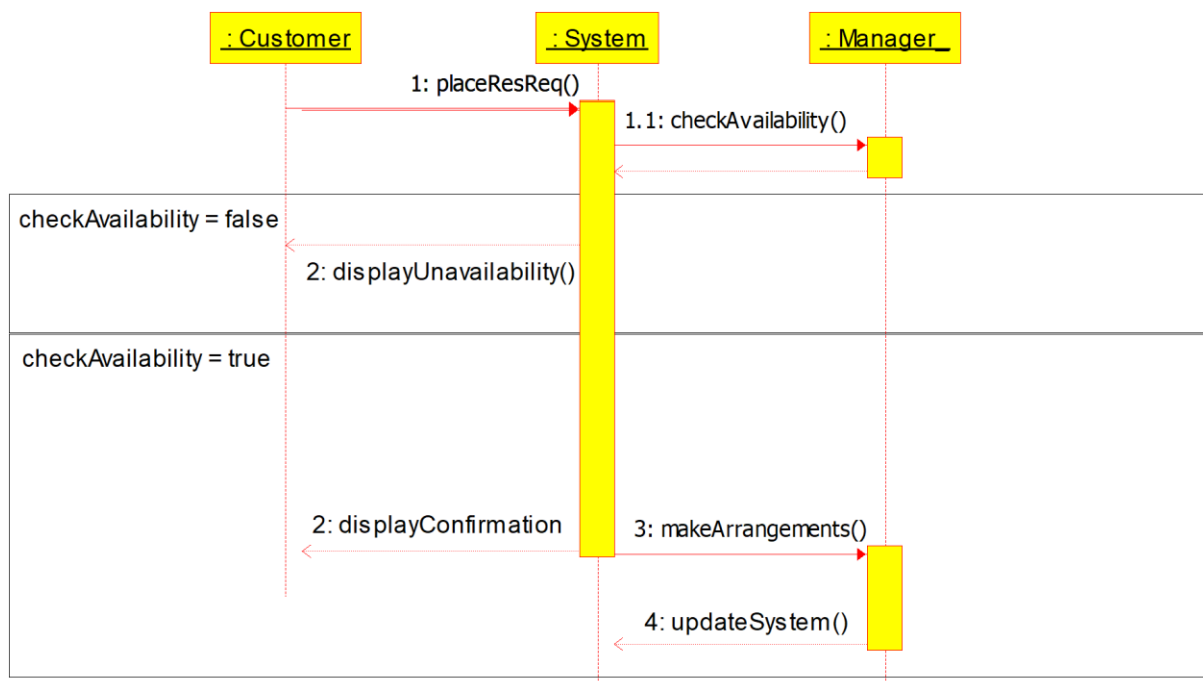
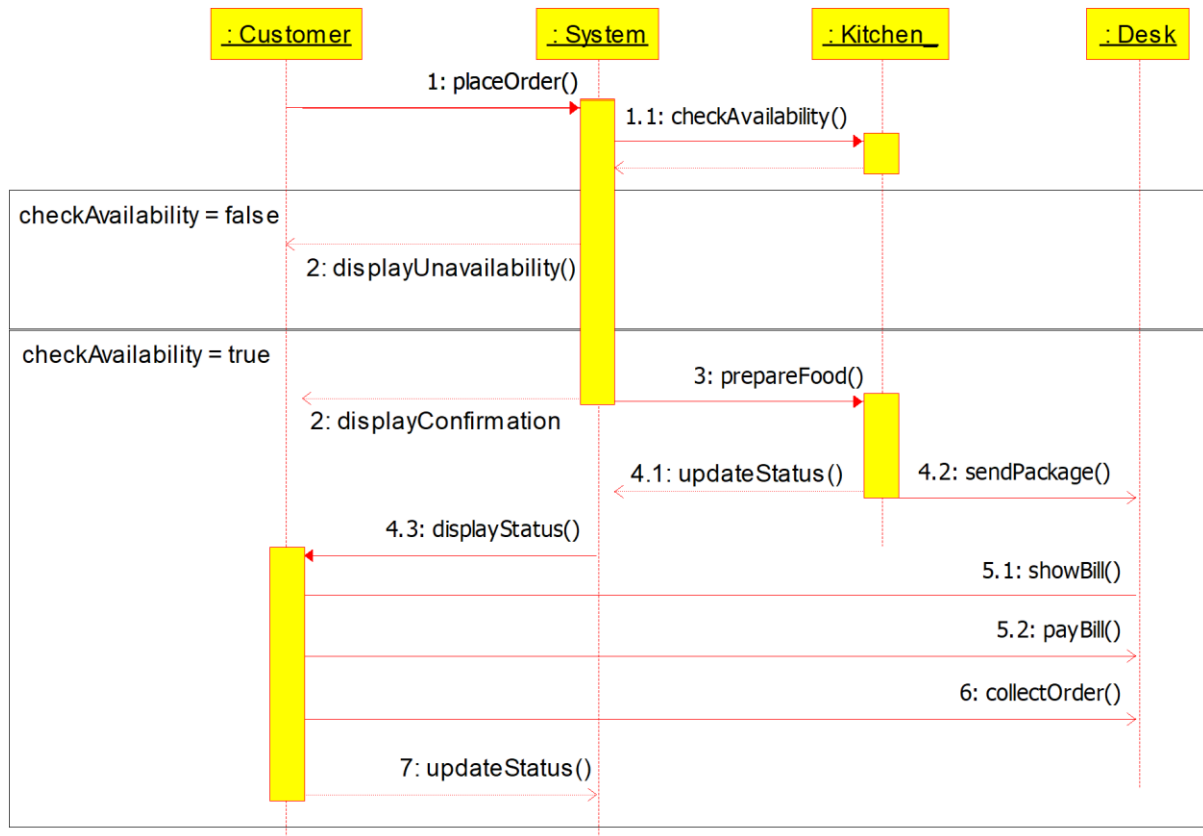
5.2 Object Diagram



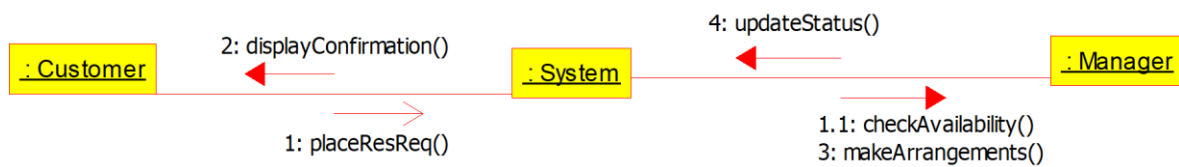
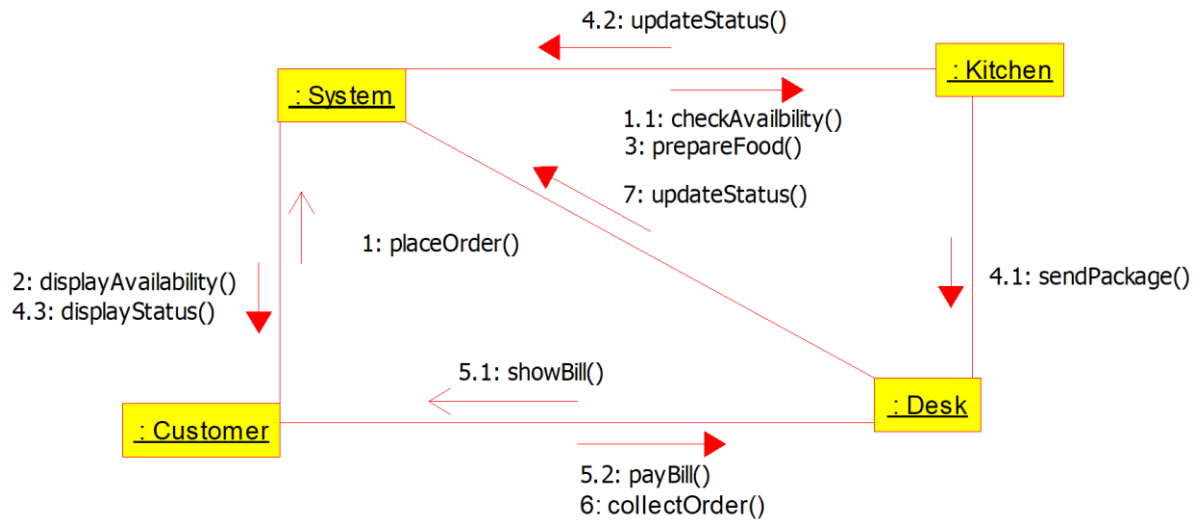
5.3 Use case Diagram



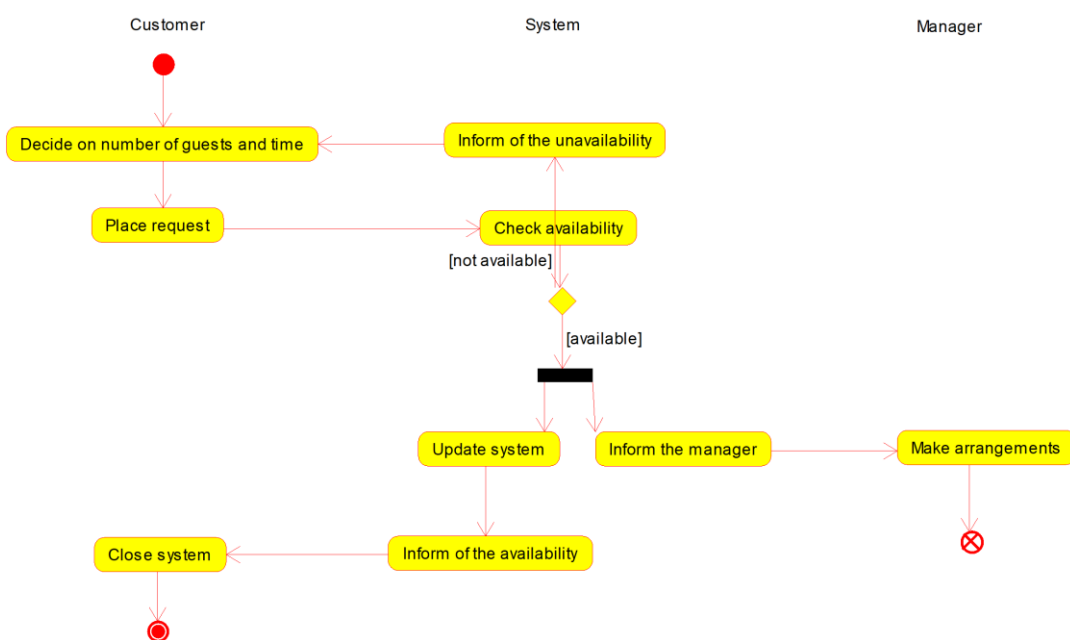
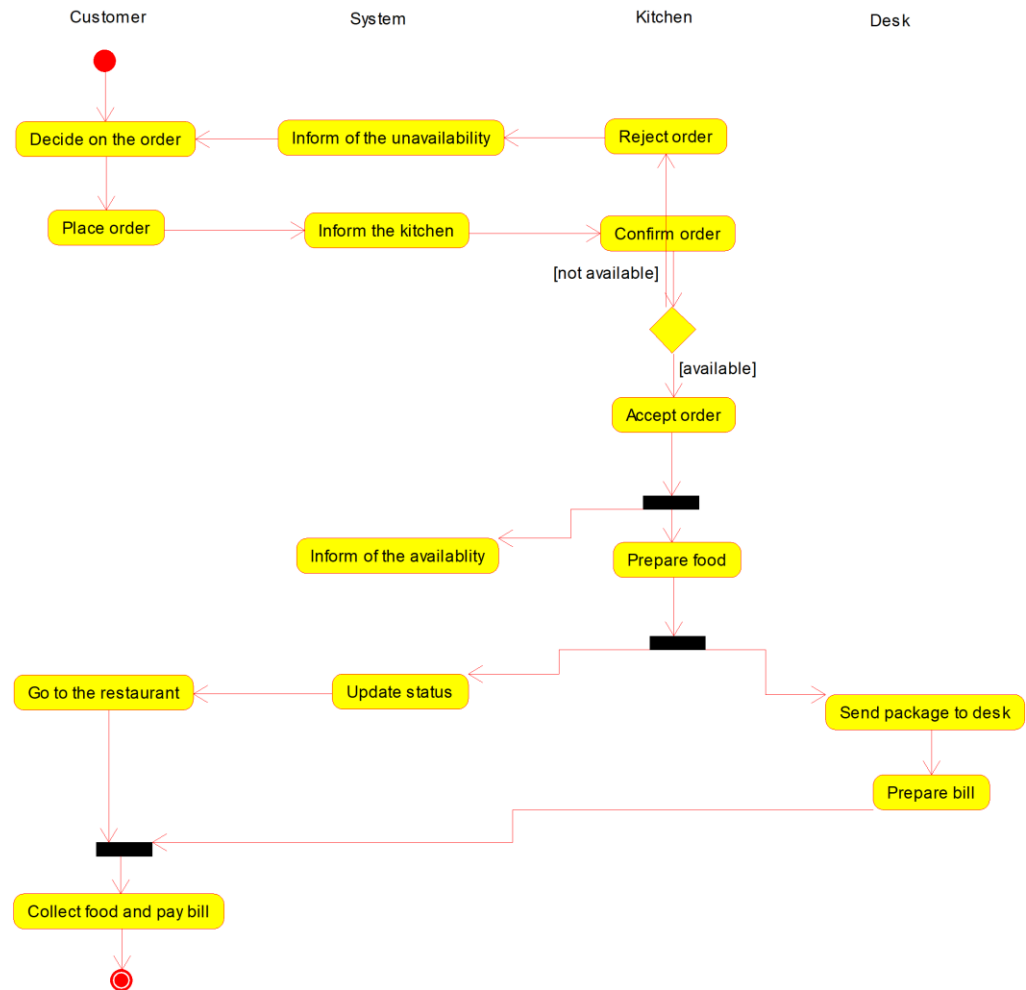
5.4 Sequence Diagram



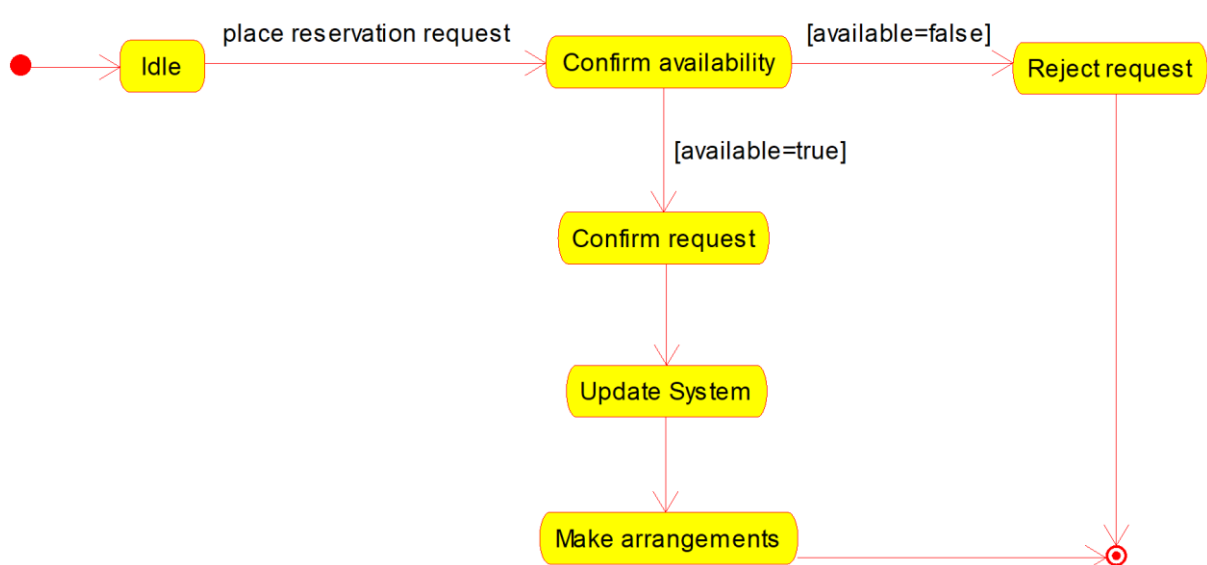
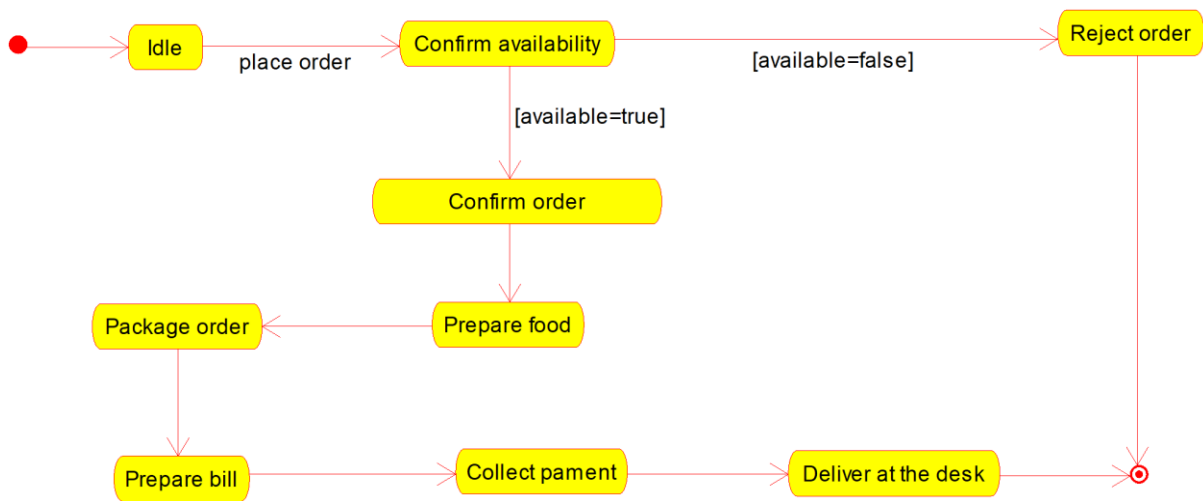
5.5 Collaboration Diagram



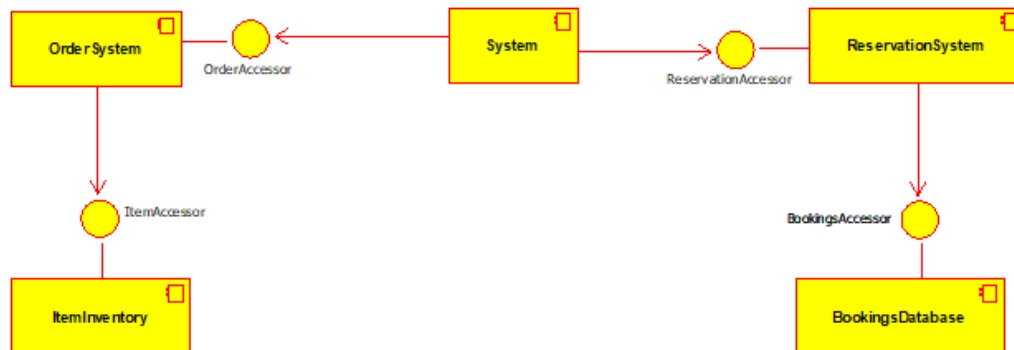
5.6 Activity Diagram



5.7 State Chart Diagram



5.8 Component Diagram



5.9 Deployment Diagram

