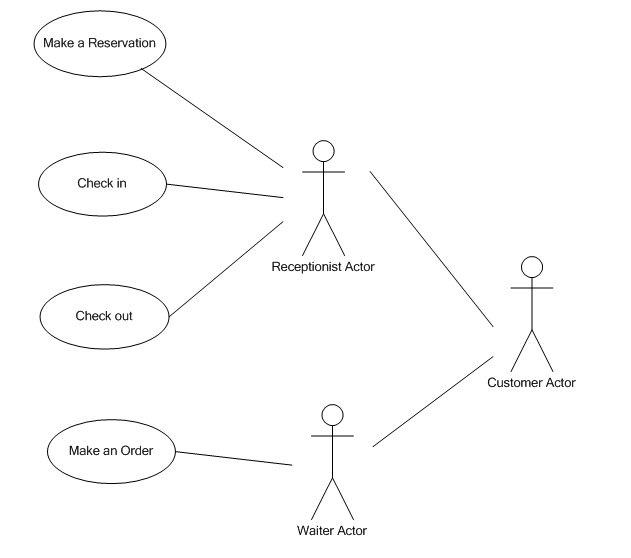
**Restaurant Management System**

**Part I**

**Rui Da, Yi Yang, Kun Chen**

**Use Case Diagram**



**Use Case: Make a Reservation (Author: Yi Yang)**

**Actor:** Receptionist

**Goals:** Help customer make a reservation for a specific date and time.

**Precondition:**

**Summary:** Receptionist finds available table for the specific date and time, record contact information, reserves the table for customer.

**Related use case: None**

**Steps:**

|  |  |
| --- | --- |
| **Actor actions** | **System responses** |
| 1.Select “Reservation” command | 2. Pop up reservation window |
| 3. Select date and time, then search the system | 4.Display all available tables |
| 5. Select table size, type into customer’s name and cell phone, submit reservation.  7. Return to main window | 6. Pop up message indicates making reservation successfully |

**Postconditions: table reserved for customer, record inserted into database, available table number decreases**

**Scenario**

**Make a reservation (Author: Yi Yang)**

Steps:

|  |  |
| --- | --- |
| Actor actions | System responses |
| Select “Reservation” command | Pop up reservation window |
| Set October 20 for date and dinner for time, then search the system | Display all available tables |
| Type customer name “David”, cell phone “336-334-1234”, reserve the table | Pop up message indicates making reservation successfully |
| Return to main window |  |

**Make a reservation (Author: Yi Yang)**

Steps:

|  |  |
| --- | --- |
| Actor actions | System responses |
| Select “Reservation” command | Pop up reservation window |
| Set October 20 for date and dinner for time, then search the system | No table is available |
| Exit reservation window, return to main window |  |
|  |  |

**Requirements**

**Functional**

* Inputs: date time and group size.
* Outputs: display all available tables, pop up message indicating reserving successfully.
* Data store: a record of reservation will be inserted into database, including customer name, cell phone and group size.

**Quality**

* Response time: results for actions should be completed in 2 seconds.
* Recovery from failure: data should recover to the last time when a guest’s order has made.
* Allowance for reusability: at least 50% measured in terms of lines of code, must be designed generically so that it can be reused.

**Platform**

* Operating system: Windows, Mac OS

**Process**

* Programming language: Java
* Delivery date: Nov, 2014