***SOFTWARE ANALYSIS SPECIFICATION***

**1.0 Introduction**

**This section provides an overview of the entire requirement document. This document describes all data, functional and behavioral requirements for software.**

**1.1 Goals and objectives**

**Overall goals and software objectives are described.**

The purpose of the new hotel booking system is to replace the old and outdated hotel booking system. The old system is outdated, slow, inefficient and the employees have to put the customers on hold. The employees also have to work with multiple screens to pull up customer information about the reservation in order to cancel or rebook.

The new system will allow for a smooth experience for both the customers and employees. It will allow for both customers and employees the ability to book, modify, & cancel reservations efficiently while allowing for implementation of a rewards system and outside bookings, giving customers more flexibility and value to their stays. The new system will allow for outside bookings and allow the implementation of a reward system. Also employees will be able to view metrics of various customers and manage the various administrative functions of the hotel (Section 1.1 SPMP)

**1.2 Statement of scope**

**A description of the software is presented. Major inputs, processing functionality and outputs are described without regard to implementation detail.**

Inputs

* Login credentials
* Booking Dates, Times
* Hotel Location
* Customer Information (e.g. name, phone, address)
* Reservation number
* Room Preferences

Processing functionality

* Verifying login credentials
* Checking room availability
  + adding to waitlist
* Processing bookings
* Processing payment
* Calculating Customer Rewards
* Calculating metrics (customer tracker)

Outputs

* Customer reward points
* History of Customer activity
  + Login/access account
  + check in/check out
  + reward points gained and spent
* Employees review reservations and/or cancellations
* Response to third party website
* Summary Reports
  + Popular booked locations
  + Revenue vs Expenses
  + Rooms most booked
  + Occupancy Rate

**1.3 Software context**

**The software is placed in a business or product line context. Strategic issues relevant to context are discussed. The intent is for the reader to understand the 'big picture'.**

The purpose of the new hotel booking system is to replace the old and outdated hotel booking system. The old system is outdated, slow, inefficient and the employees have to put the customers on hold. The employees also have to work with multiple screens to pull up customer information about the reservation in order to cancel or rebook.

The new system will allow for a smooth experience for both the customers and employees. The new system will allow for outside bookings and allow the implementation of a reward system in which the customer will be able to view the types of redemptions for their points and be able to see if they will be able to upgrade

The customer will be able to create and login to their account to view hotels, book a room, view existing bookings, and view collected metrics.

The hotel booking system will allow employees to view bookings and metrics, manage the overall operations of the hotel, and be able to make changes to a customer’s reservation. The system will be used to check customer and hotel metrics such as customer’s login information, reservation preferences, hotel room availability, changes to reservations, reward points, and confirm bookings (Use case Summary)

**1.4 Major constraints**

**Any business or product line constraints that will impact the manner in which the software is to be specified, designed, implemented or tested are noted here.**

Not Applicable

**2.0 Usage scenario**

**This section provides a usage scenario for the software. It organized information collected during requirements elicitation into use-cases.**

**2.1 User profiles**

|  |  |
| --- | --- |
| Employee | Primary Actor: interacts with system to book, modify, and cancel, and review reservations, create accounts for customers, view summary reports and hotel informatics, and manage the hotel and other various functions |
| Customer | Primary Actor: interacts with the system to login, book, modify, and cancel stays reservations, view and manage individual reward points, edit name and location. Also interacts with Employee to modify the reservation |

**2.2 Major software functionality**

* Access the system
  + Create a new user account / ID
  + Login/Log-out the user
  + Check account information (separate views for users and admins)
* Check-in Check-out
  + When a customer arrives or leaves check in or out
* Reservation
  + Display Rooms and check room availability
  + Create, make changes to, or cancel reservations
  + Create a waitlist if all the rooms are reserved
* Rewards
  + History of Rewards activity
  + Manage and view rewards
  + View and book upgrades using rewards obtained based on type of room and length of stay
* Hotel Administration
  + Manage major physical functions of a hotel
  + Track metrics/Summary Reports (for bookings and number of visits to the site)
    - Most booked Rooms
    - Most booked hotels
    - Track number of reward points
    - Most visited Customers.
    - Revenue from each stay

**2.3 Special usage considerations**

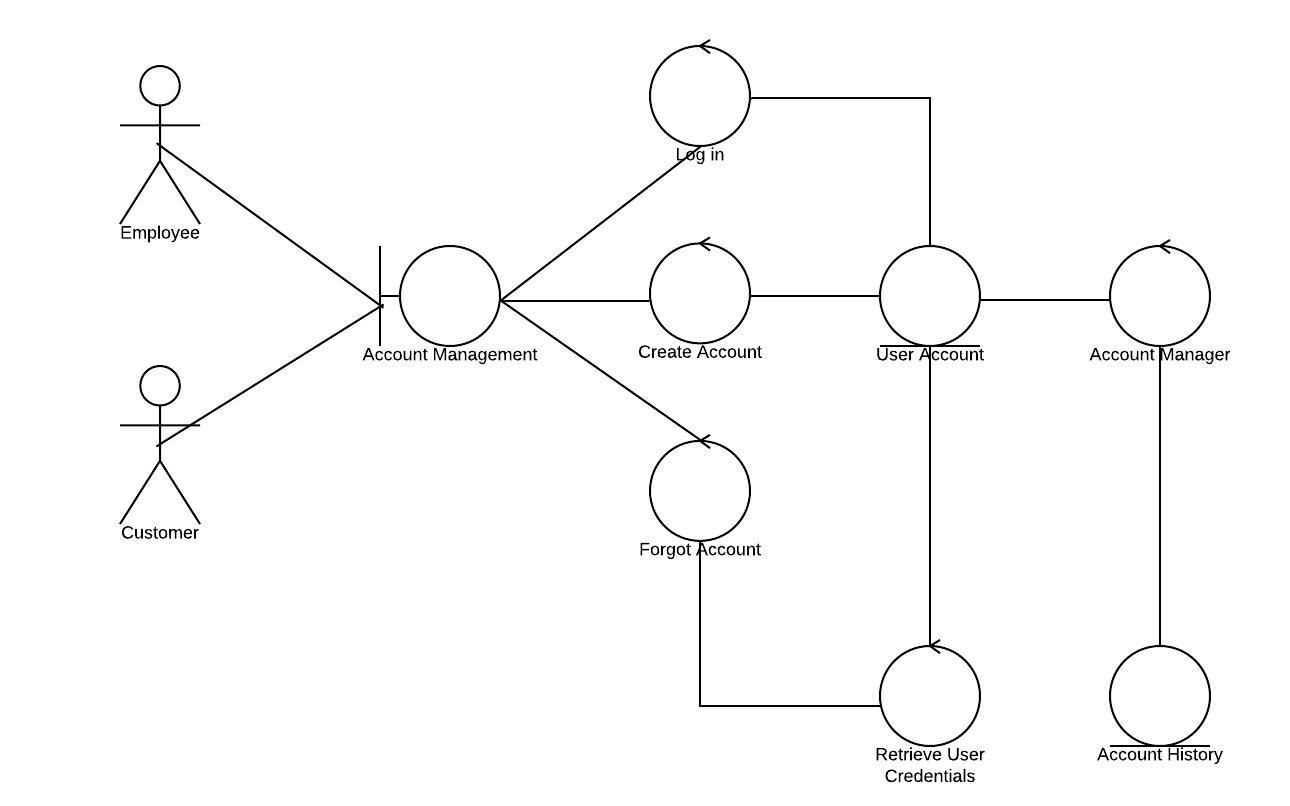
Not Applicable

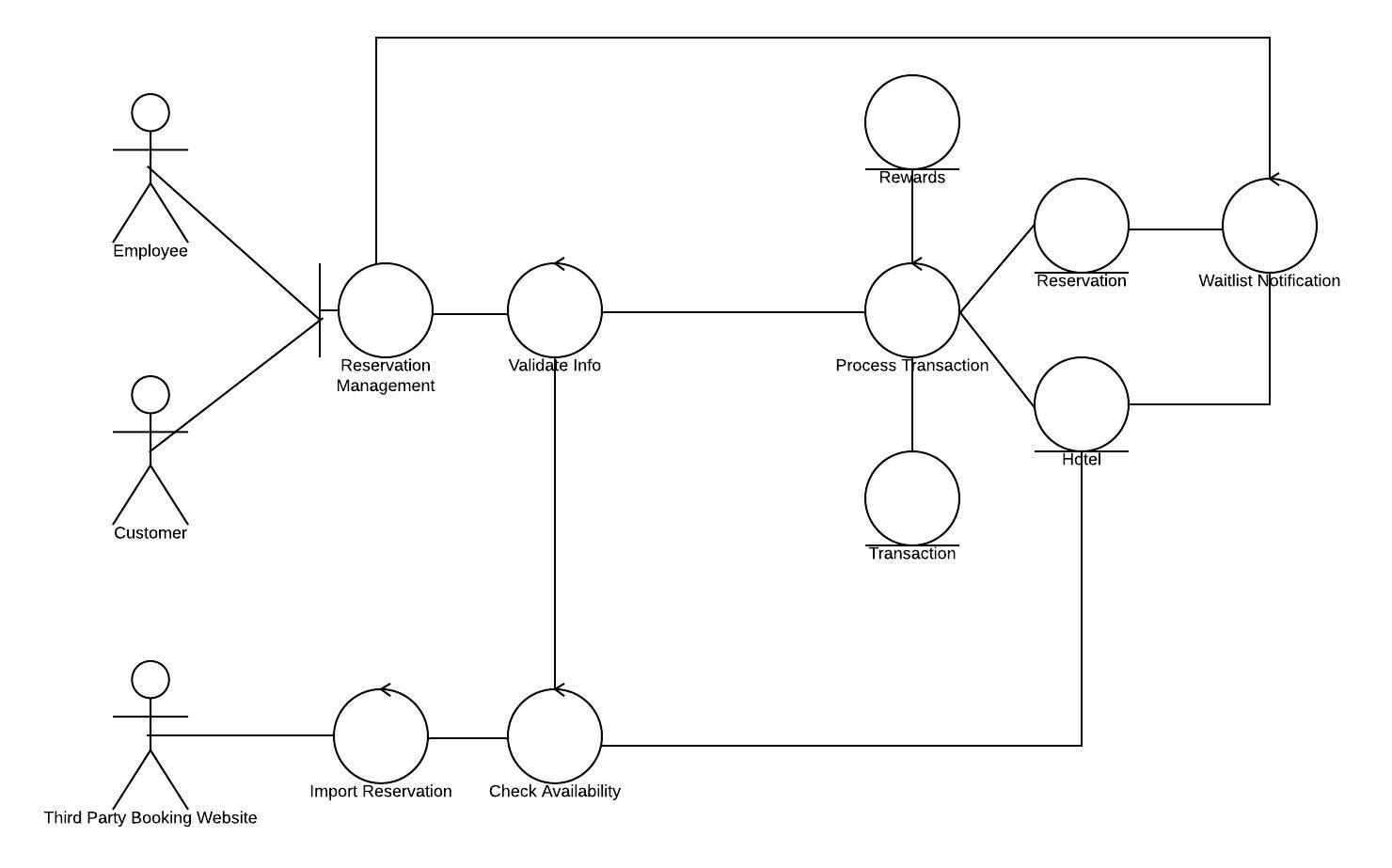
**3.0 Data Model and Description**

**This section describes information domain for the software**

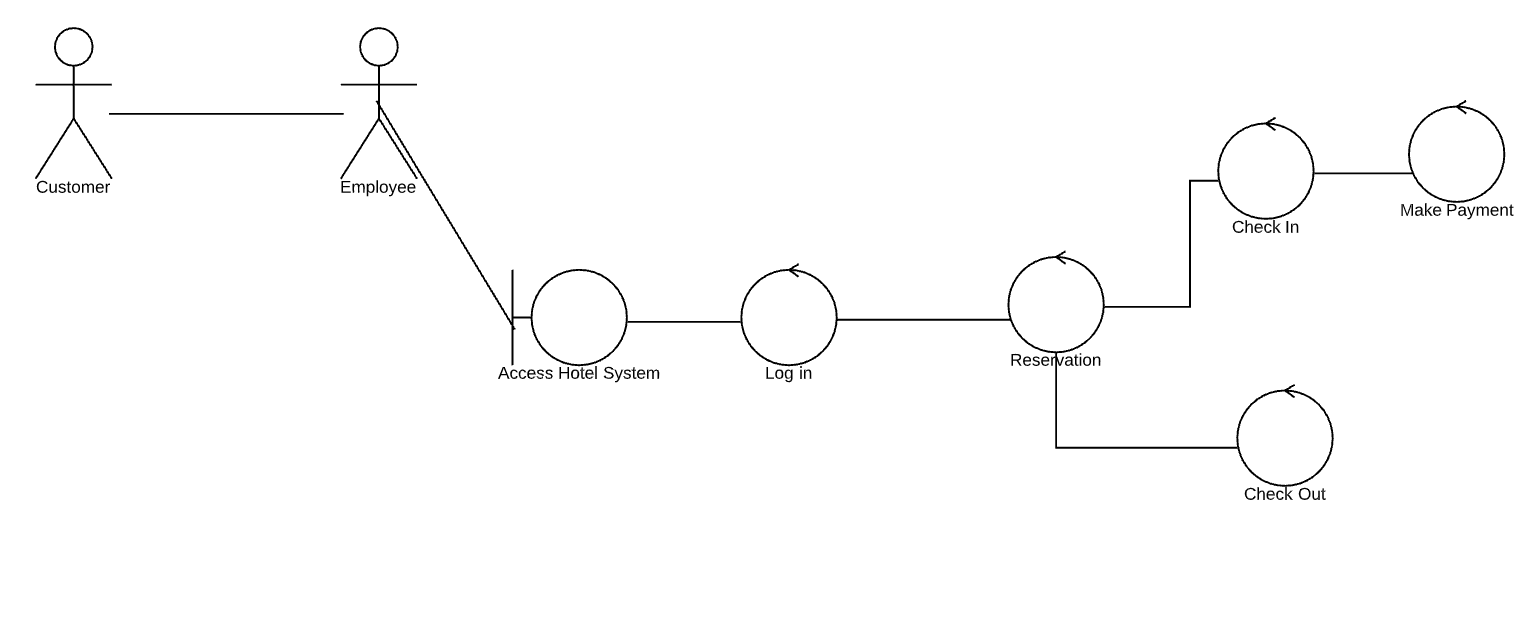
**3.1 Data Description**

**Robustness Diagrams - data objects that will be managed/manipulated by the software are described in this section.**

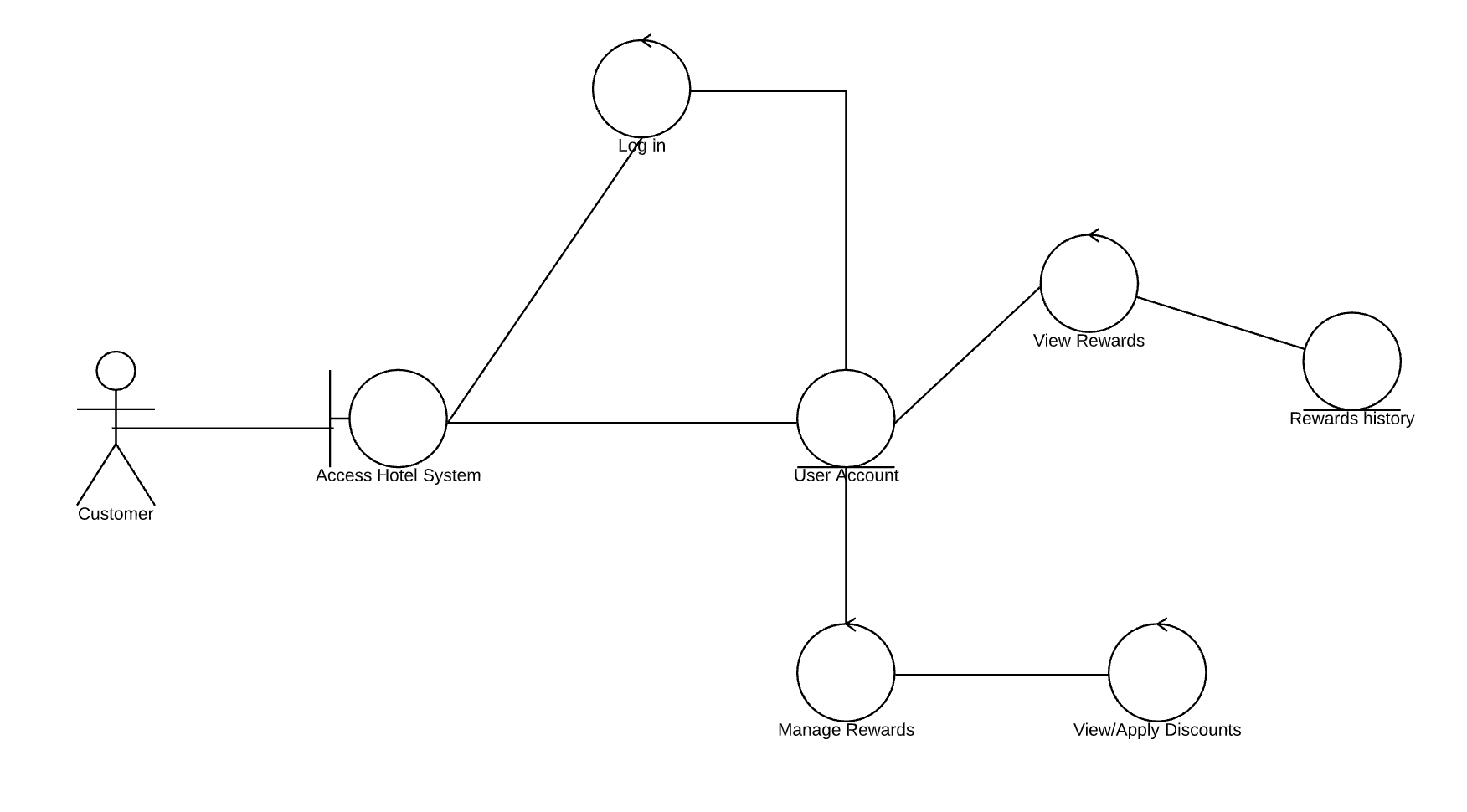
Access Hotel System****

Reservations:****

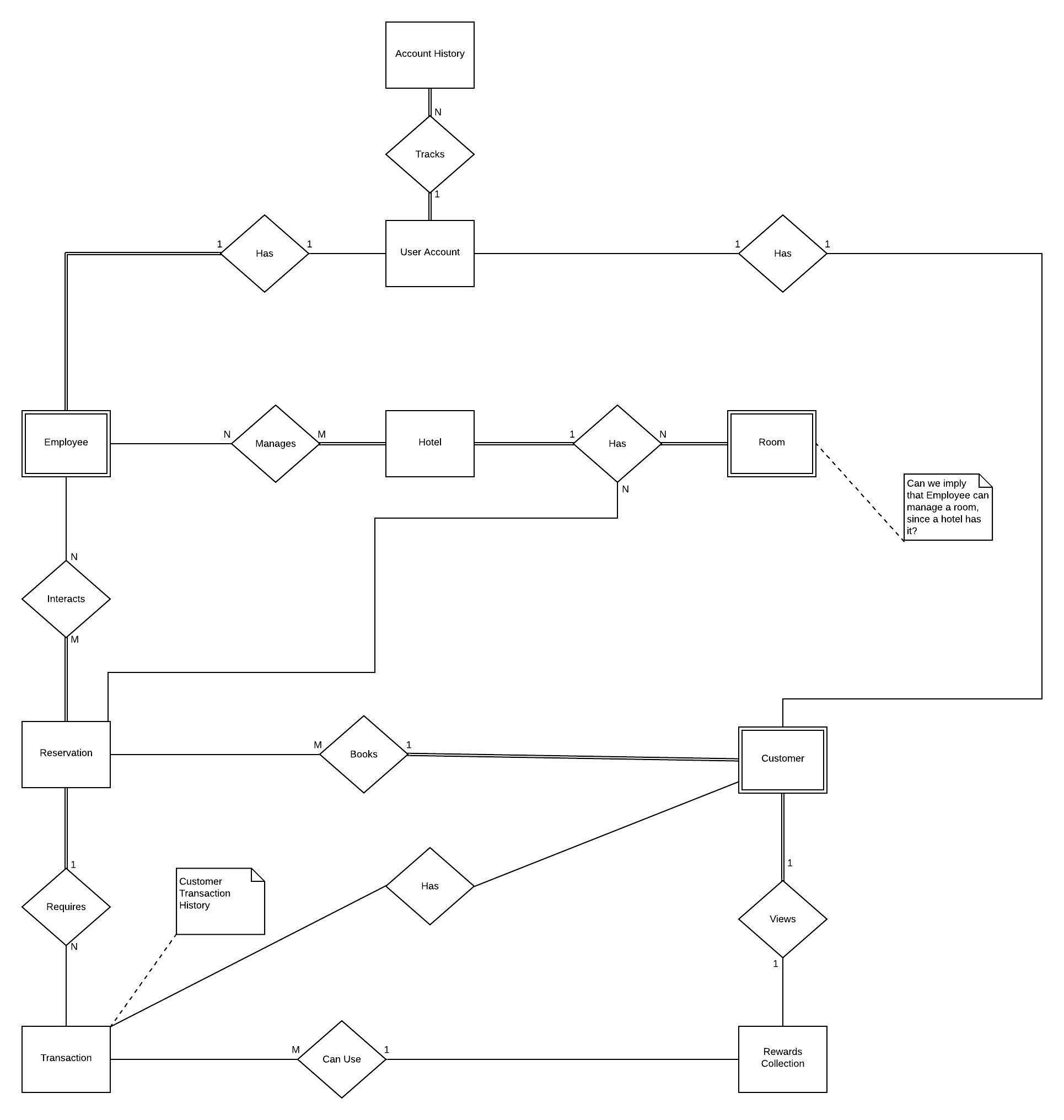
Check In/Check Out:



Rewards:



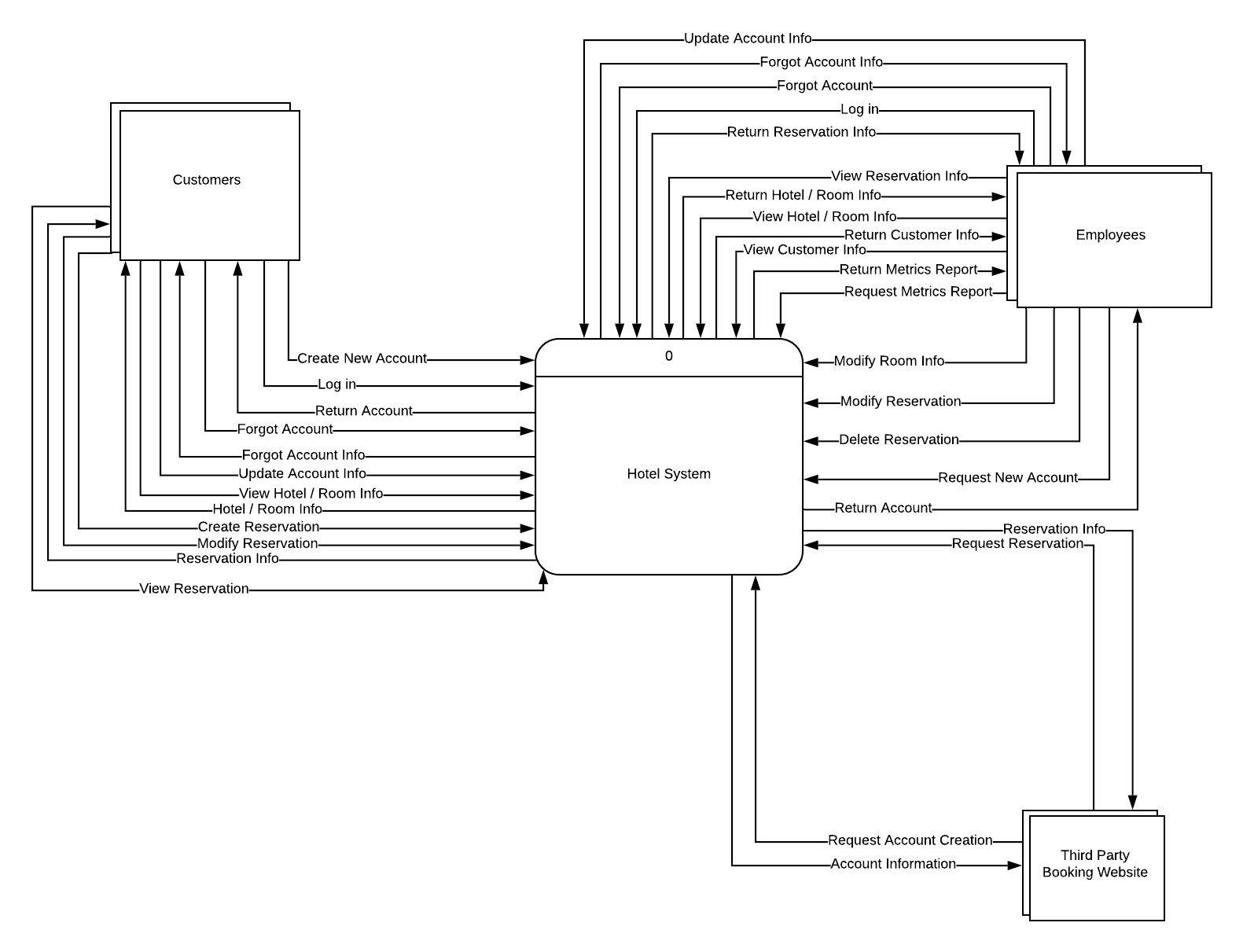
**3.1.1 Entity-Relationship Diagram**

**Each data object in 3.1 must appear as part of the BUILD ERD.**

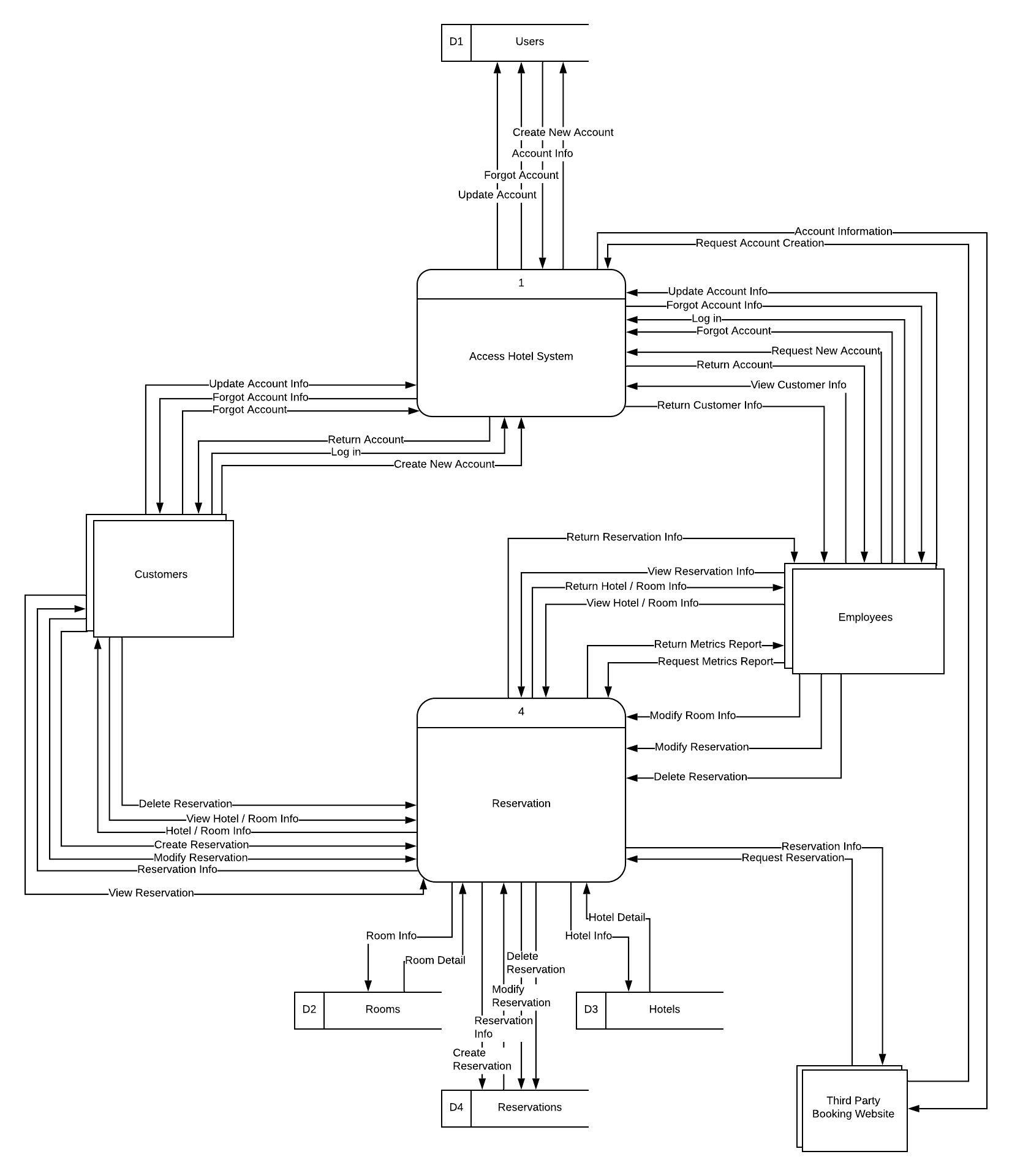
**3.1.2 Data Flow Diagram**

**Describe flow of data into/out of application – processes match to use-cases**

**Context:**

****

**0 DFD:**

****

**3.1.3 Object Relationships**

Not applicable

**3.1.4 Complete data model**

**An UML Class model (class diagram) for the software is developed – through attributes and actions (not data typing, method signatures, access)**

**3.1.5 Data dictionary**

**A reference to the data dictionary is provided. The dictionary is maintained in electronic form.**

**4.0 Functional Model and Description**

**Description of major software functions along with UML Use Case, sequence, and communication diagrams.**

**4.1 Use cases**

**A detailed description of each software function is presented by completing the use case template.**

**Cross reference this document with file name of use case summary document**

**LIST all of the use cases cross-listed with the file names of actual document**

**4.2 Software Interface Description**

**The software interface(s)to the outside world is(are) described.**

**4.2.1 External machine interfaces**

Not Applicable

**4.2.2 External system interfaces**

Not Applicable

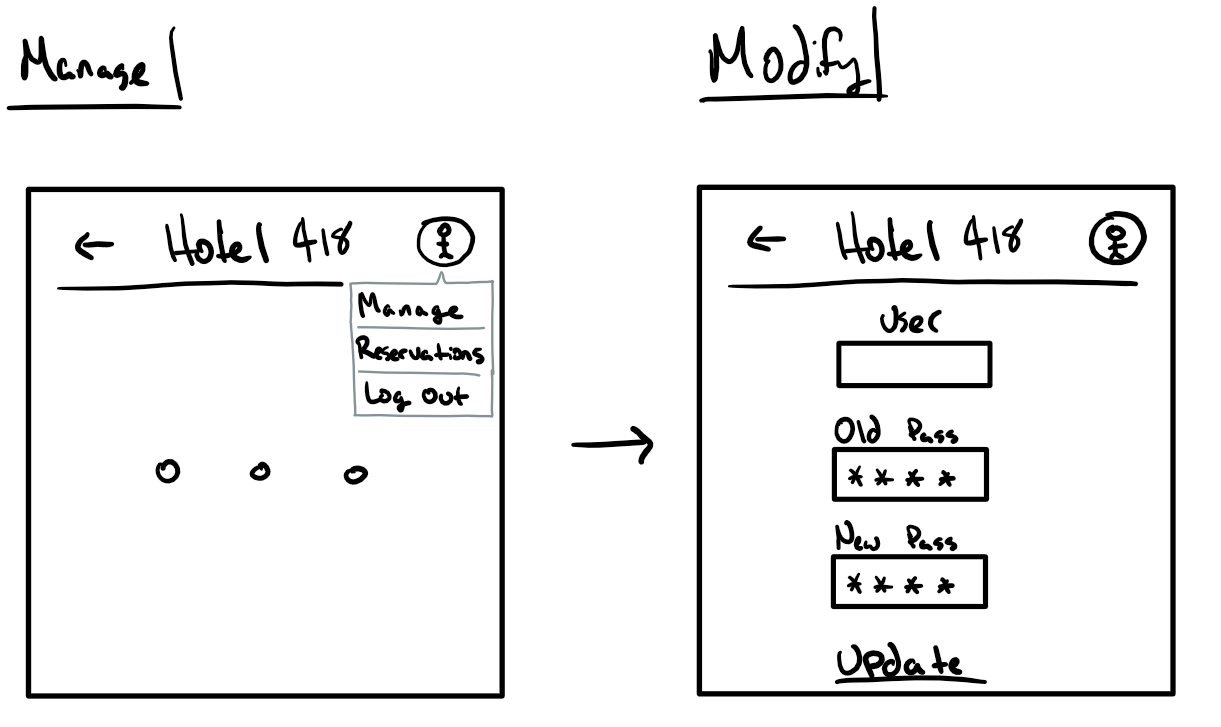
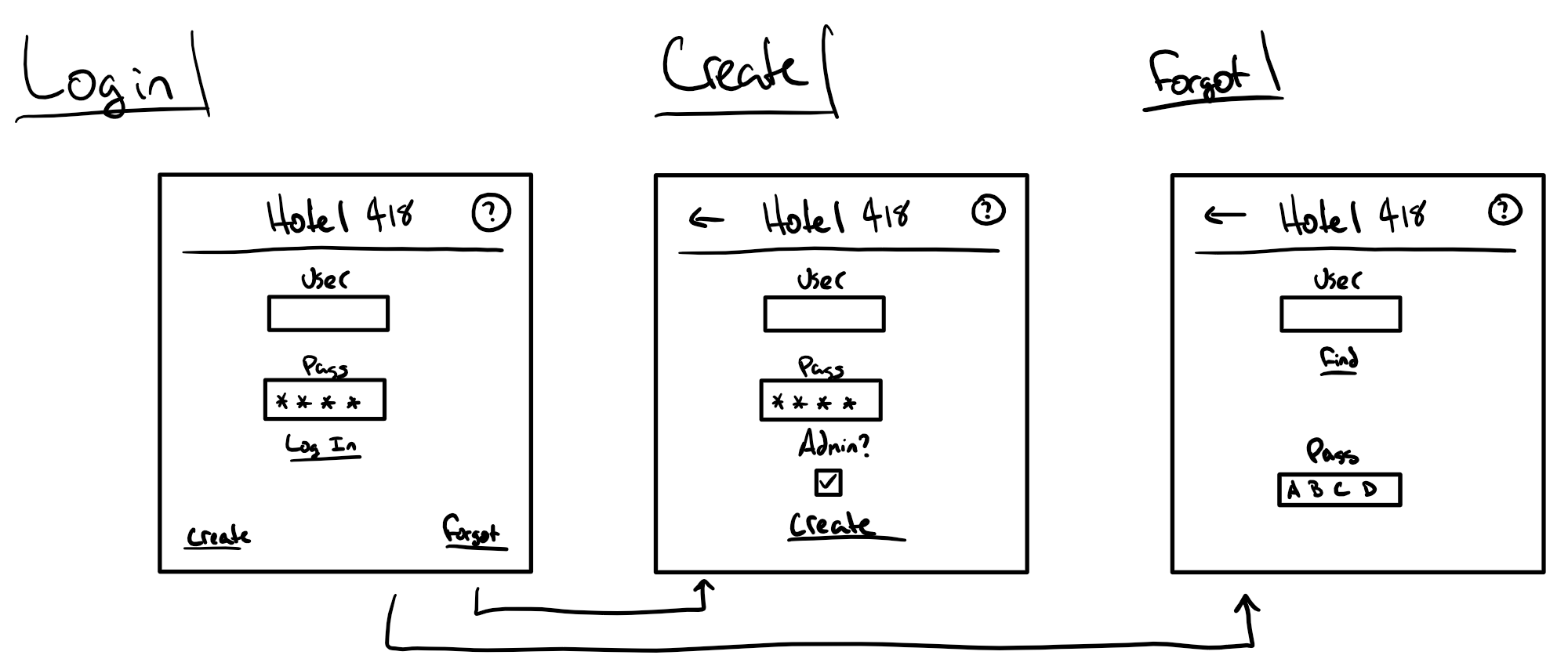
**4.2.3 Human interface**

**An overview of any human interfaces to be designed for the software is presented.**

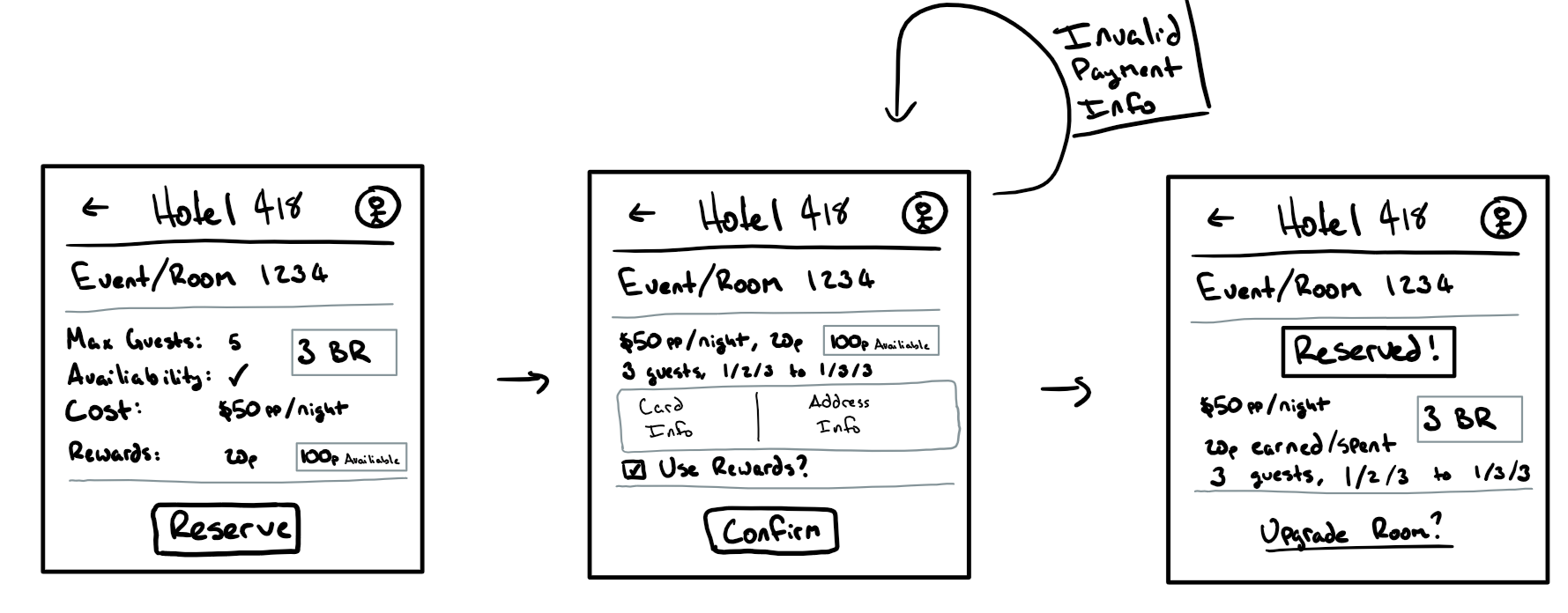
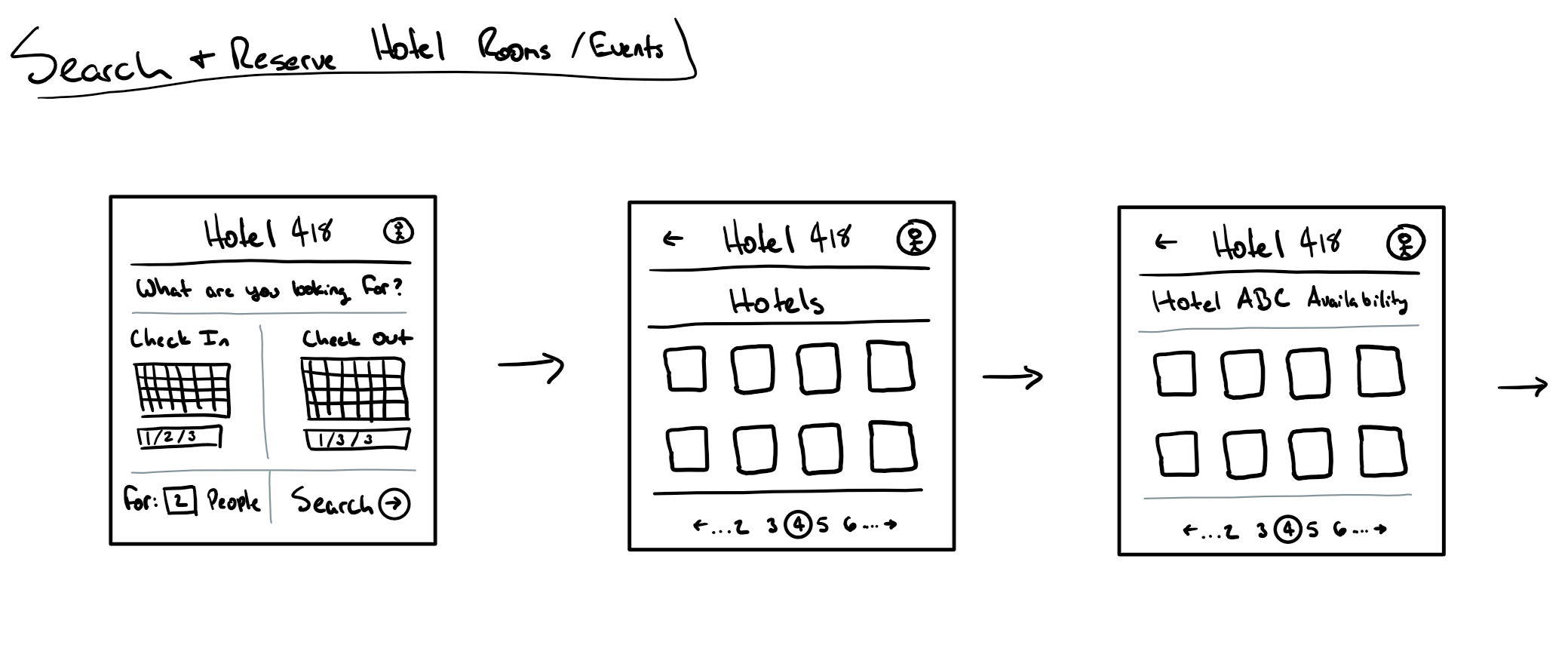
**4.2.3.1 User screen interface layouts**

**Be sure to include “exception” screens, if any**

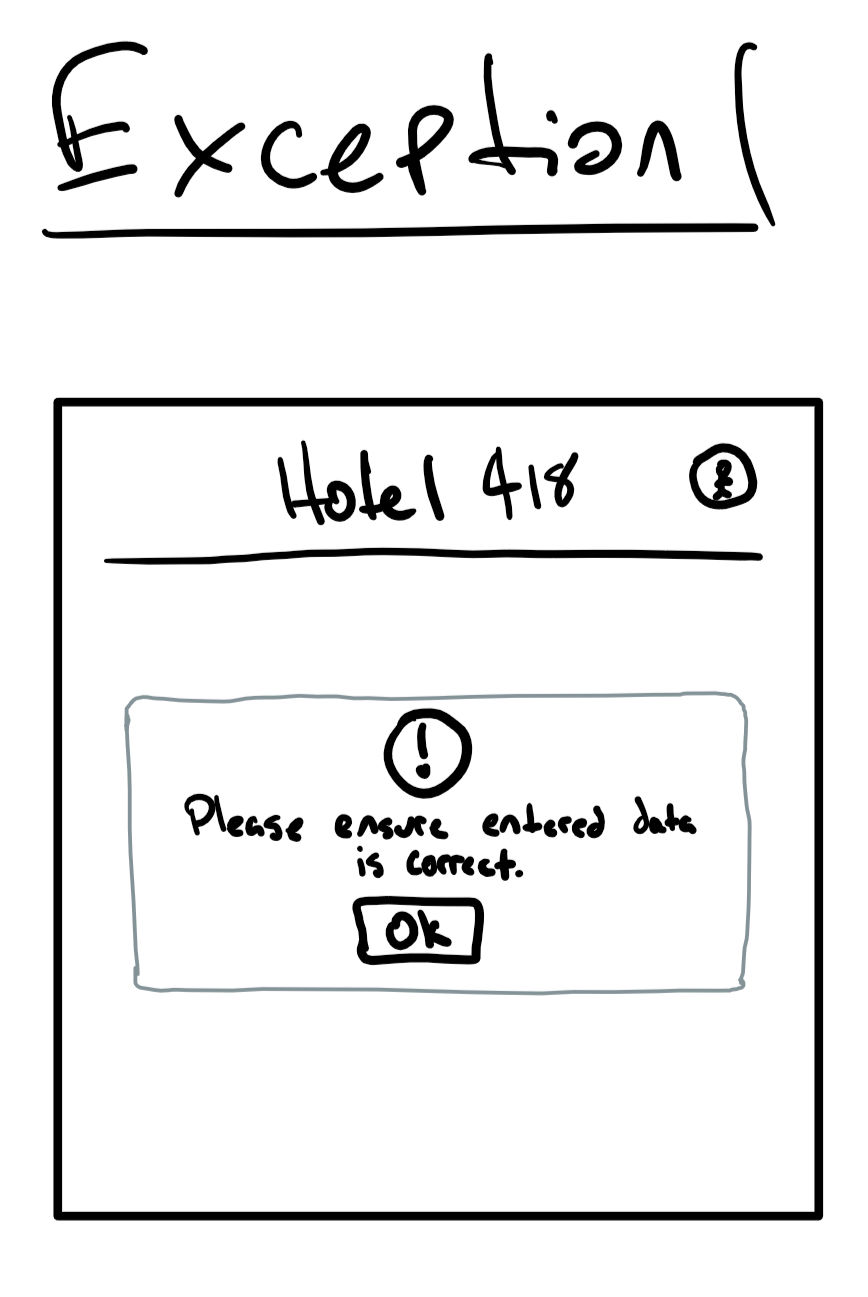
**Account:**

****

**Reservations:**

****

**Exceptions:**

****

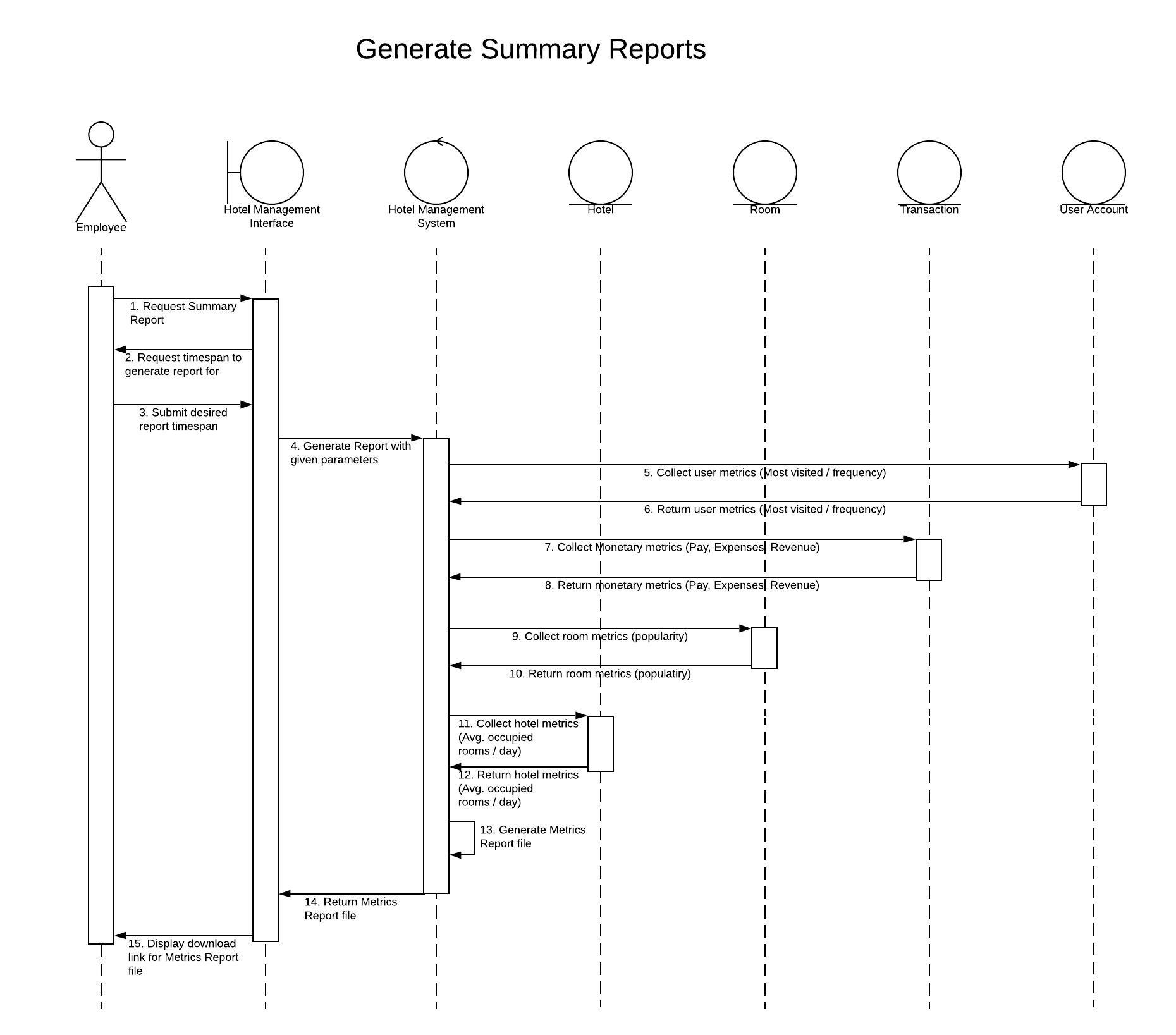
**4.2.3.2 Report layouts**

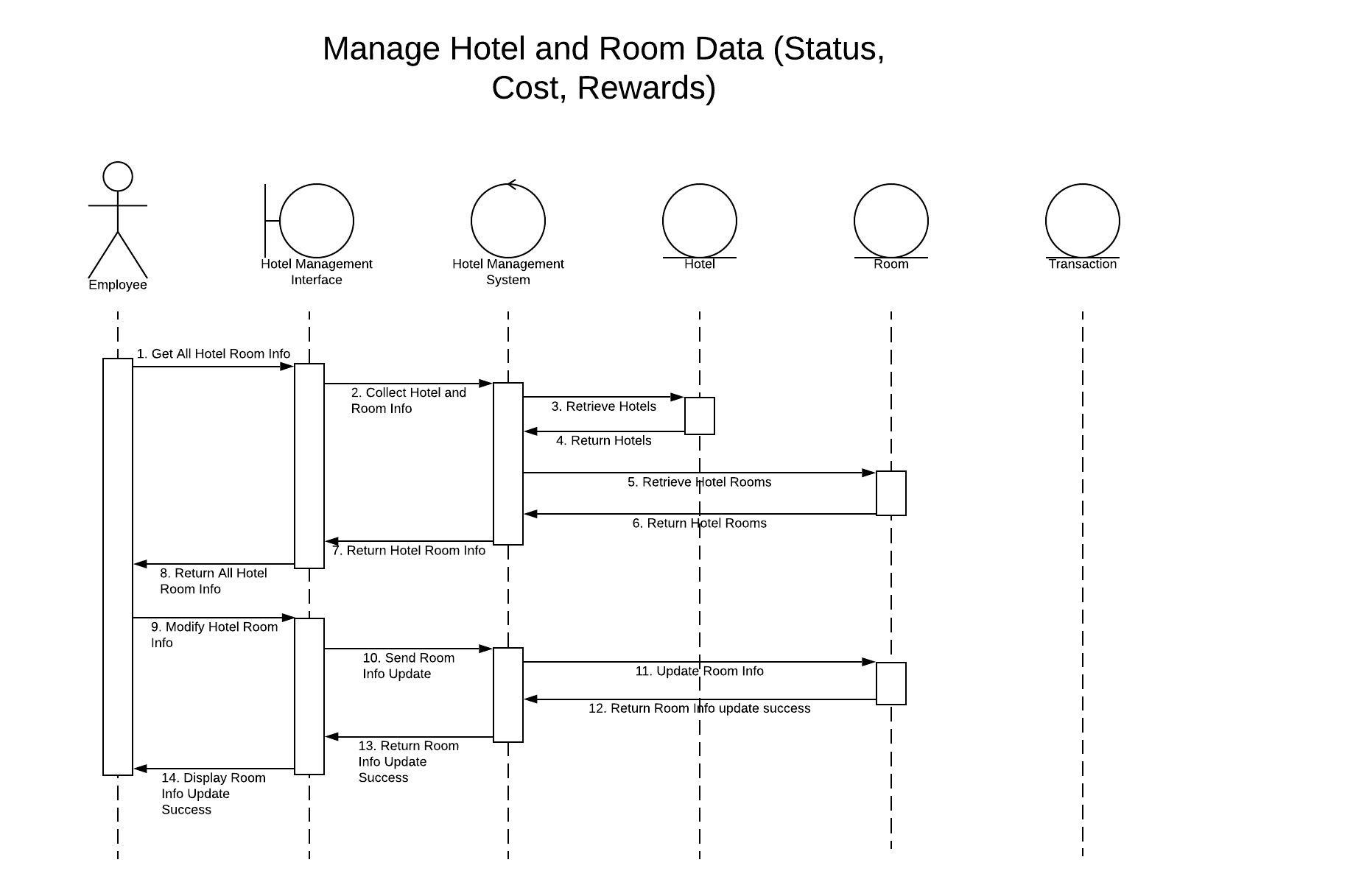
**Be sure to include “exception” reports, if any**

**4.3 Sequence Diagrams**

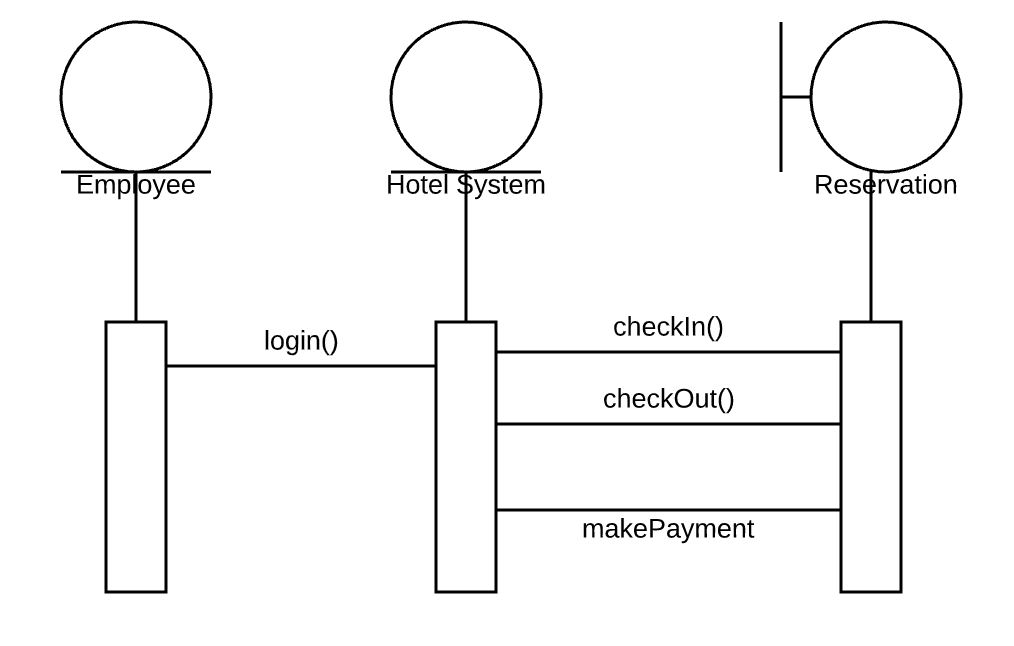
**Used to model the class interactions needed for the use cases.**

Hotel Administration:

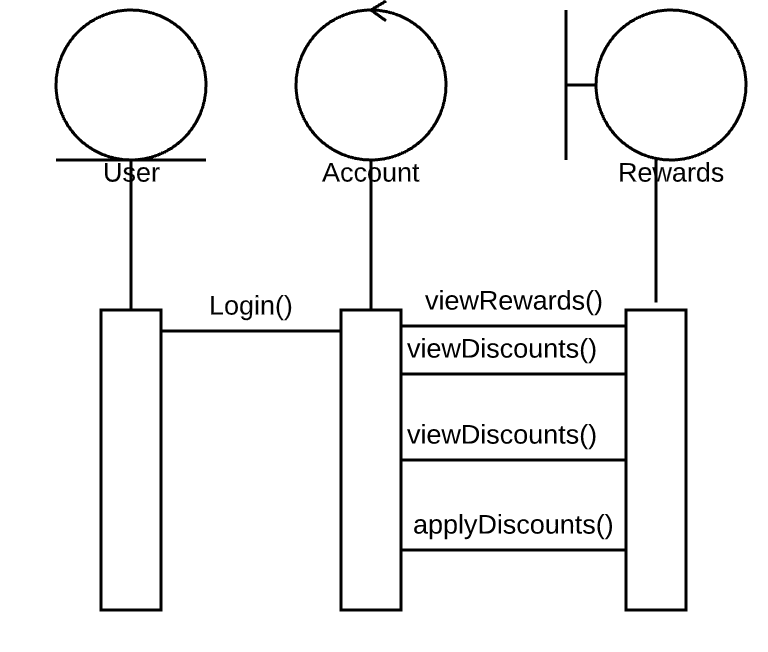




Check In/Check Out:



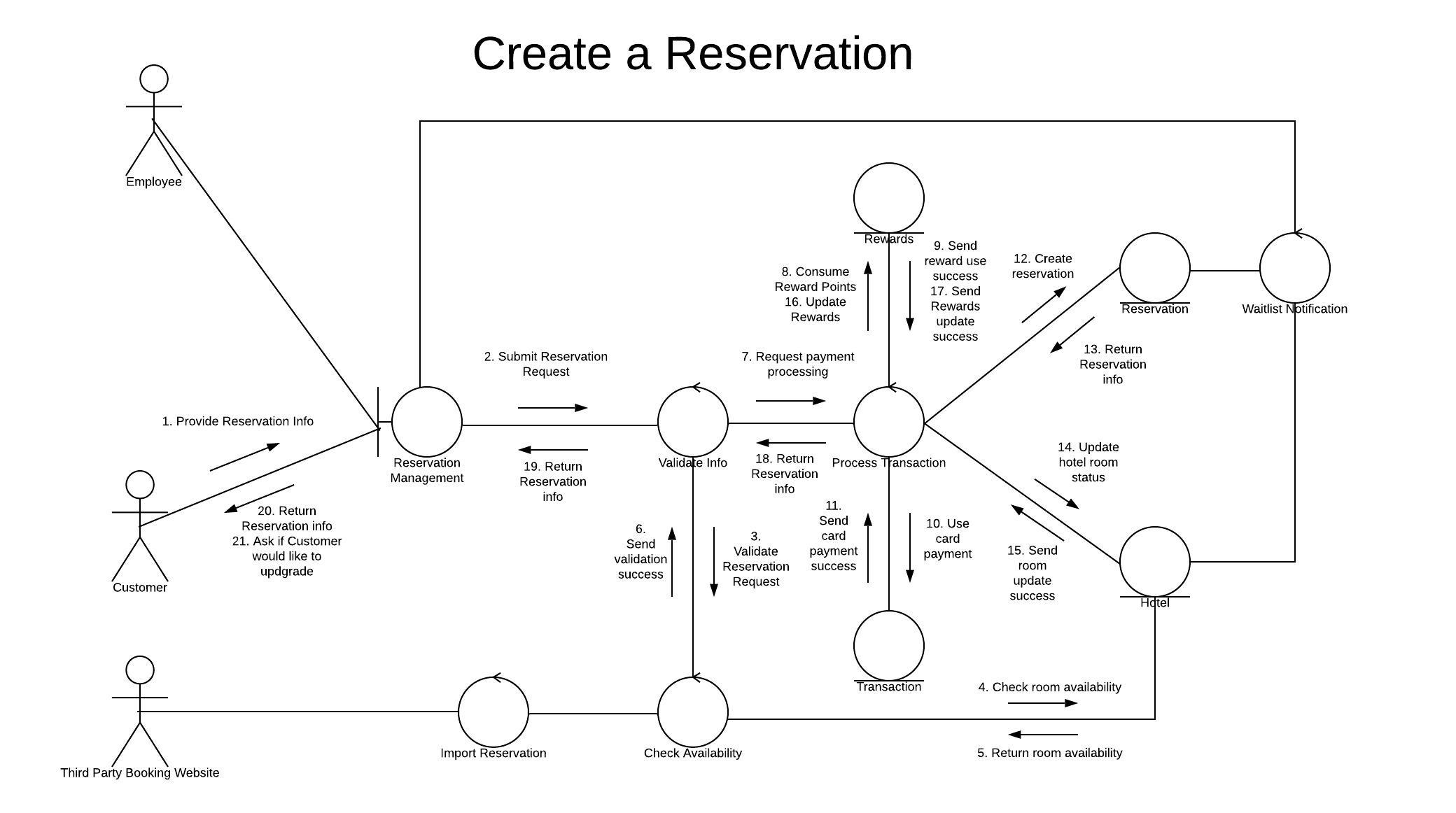
Rewards:

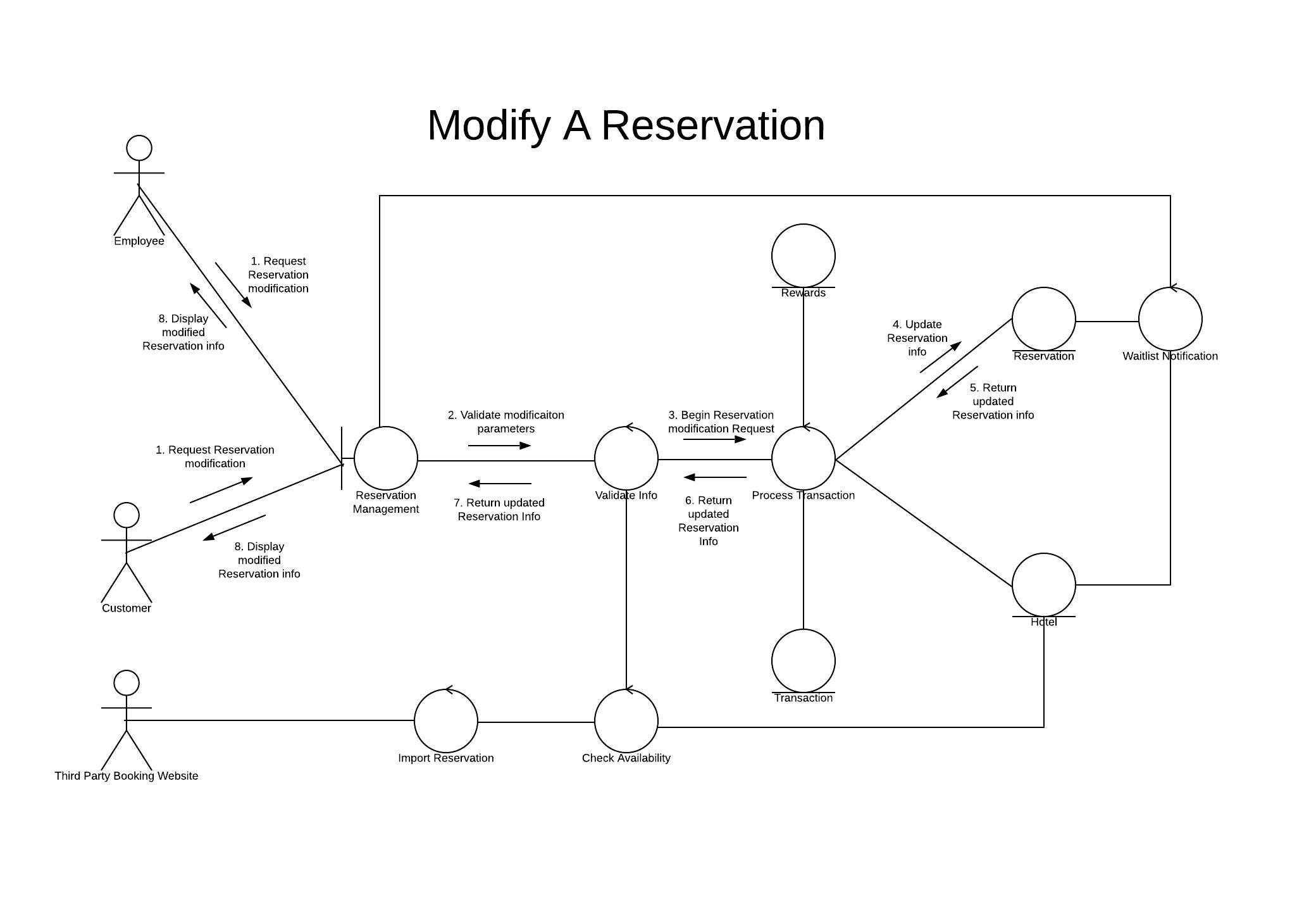


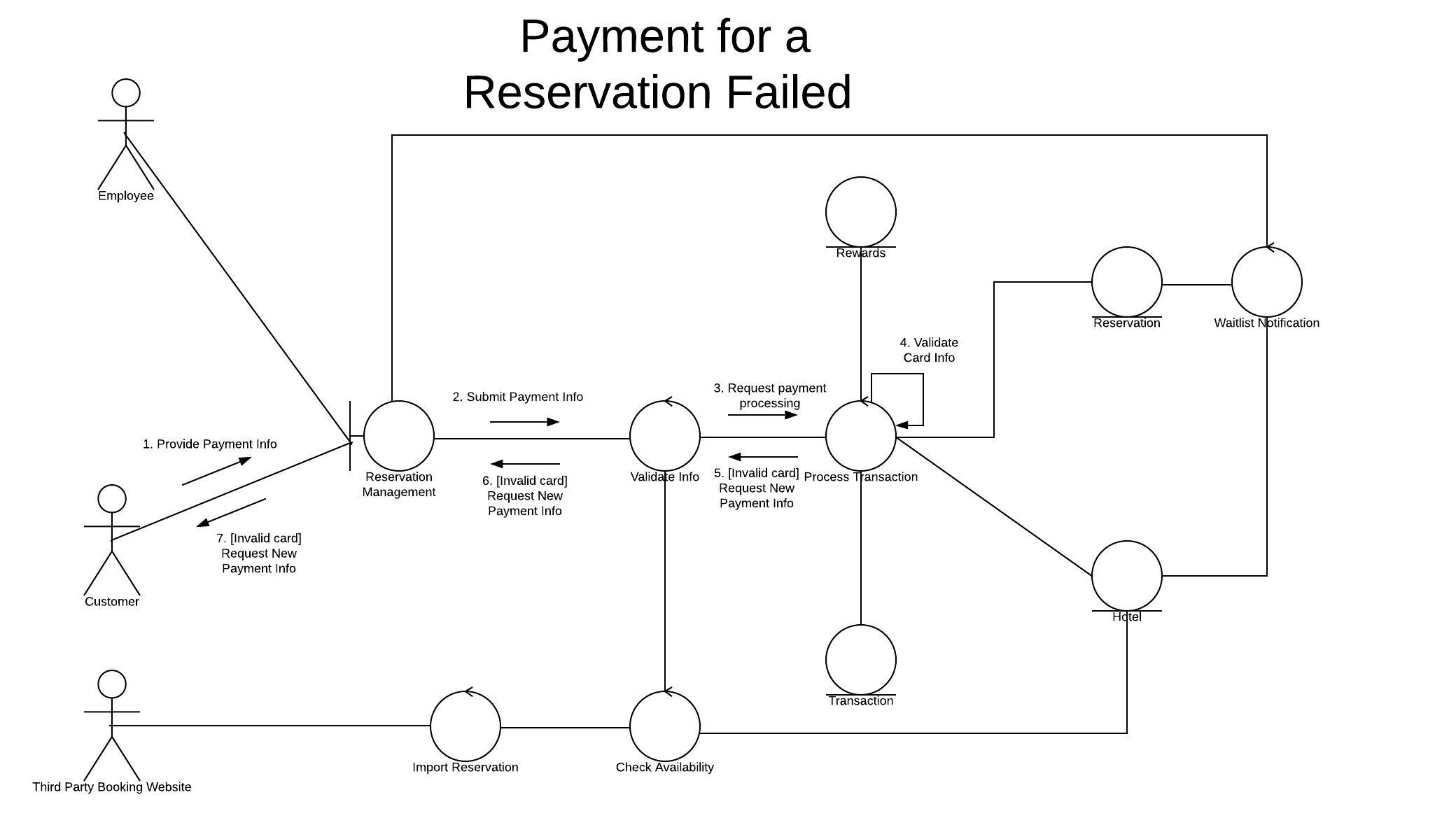
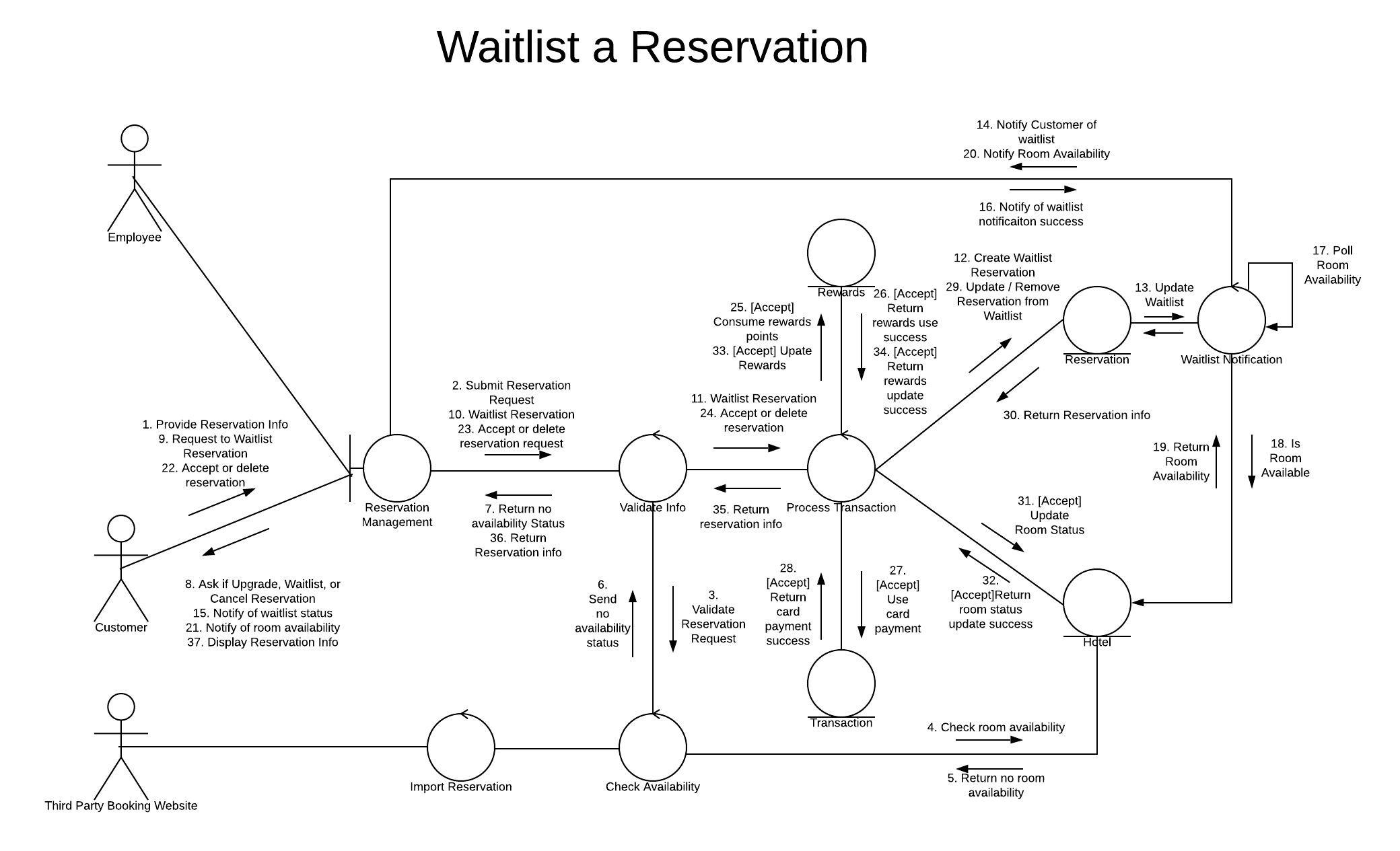
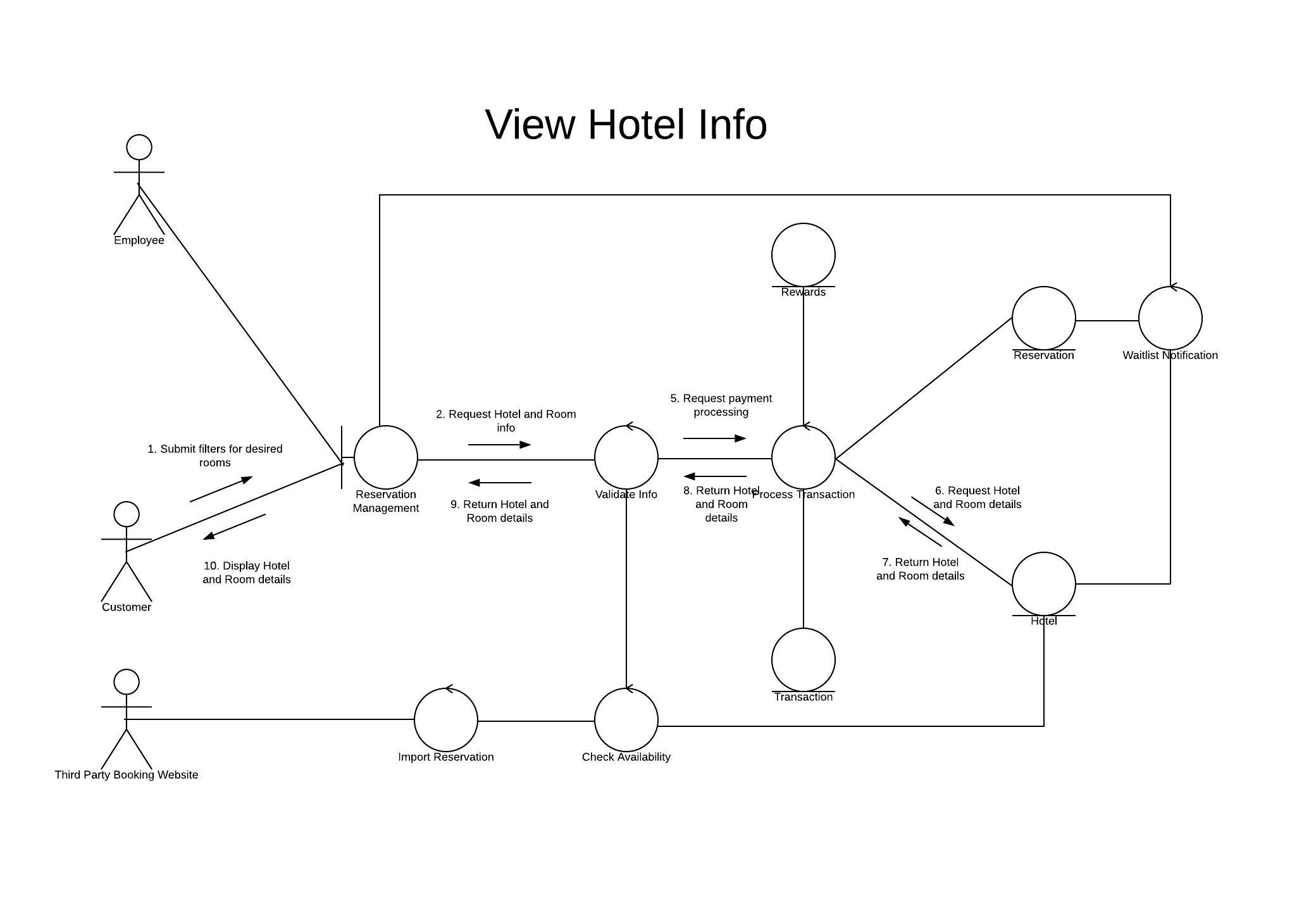
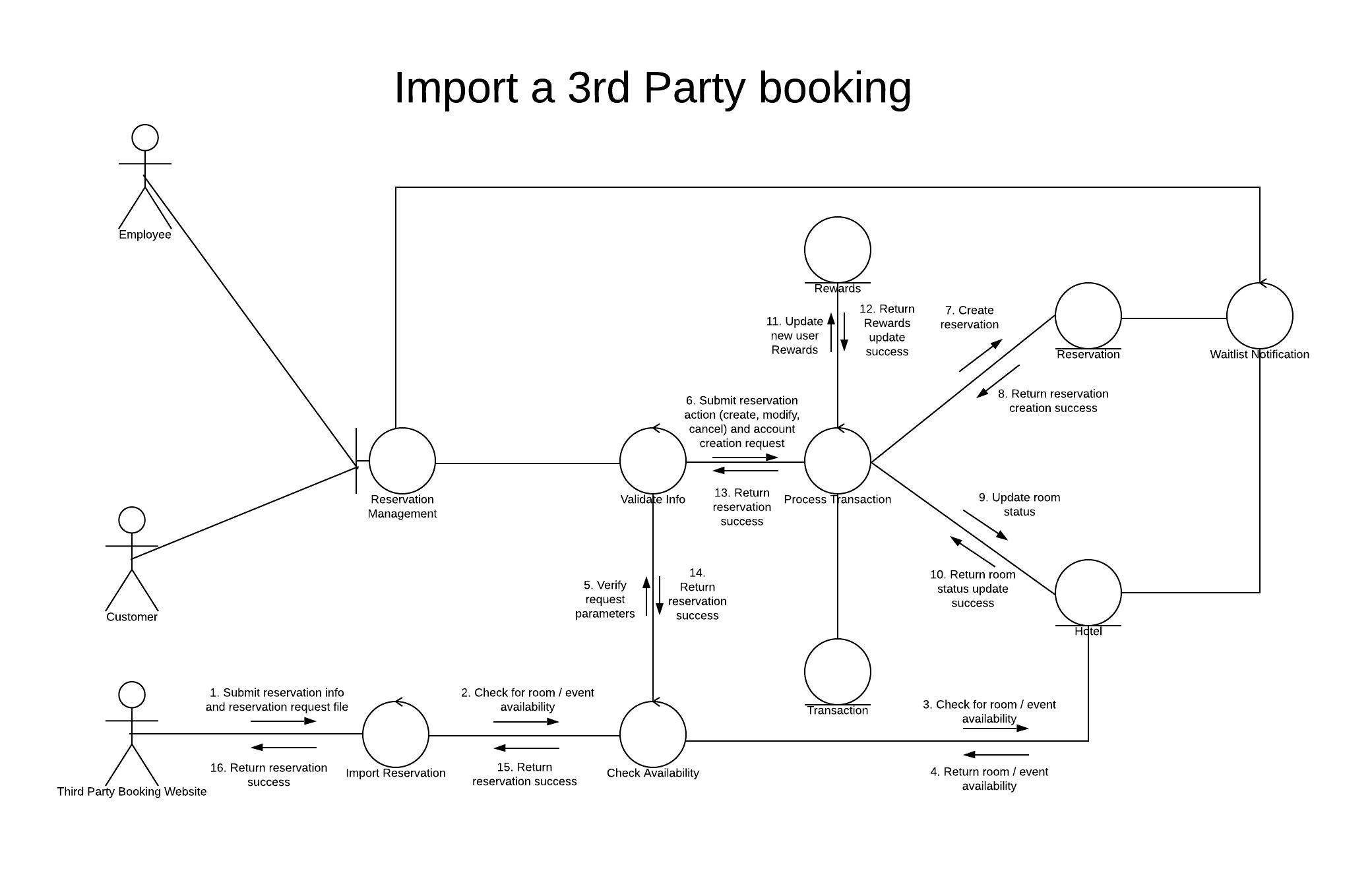
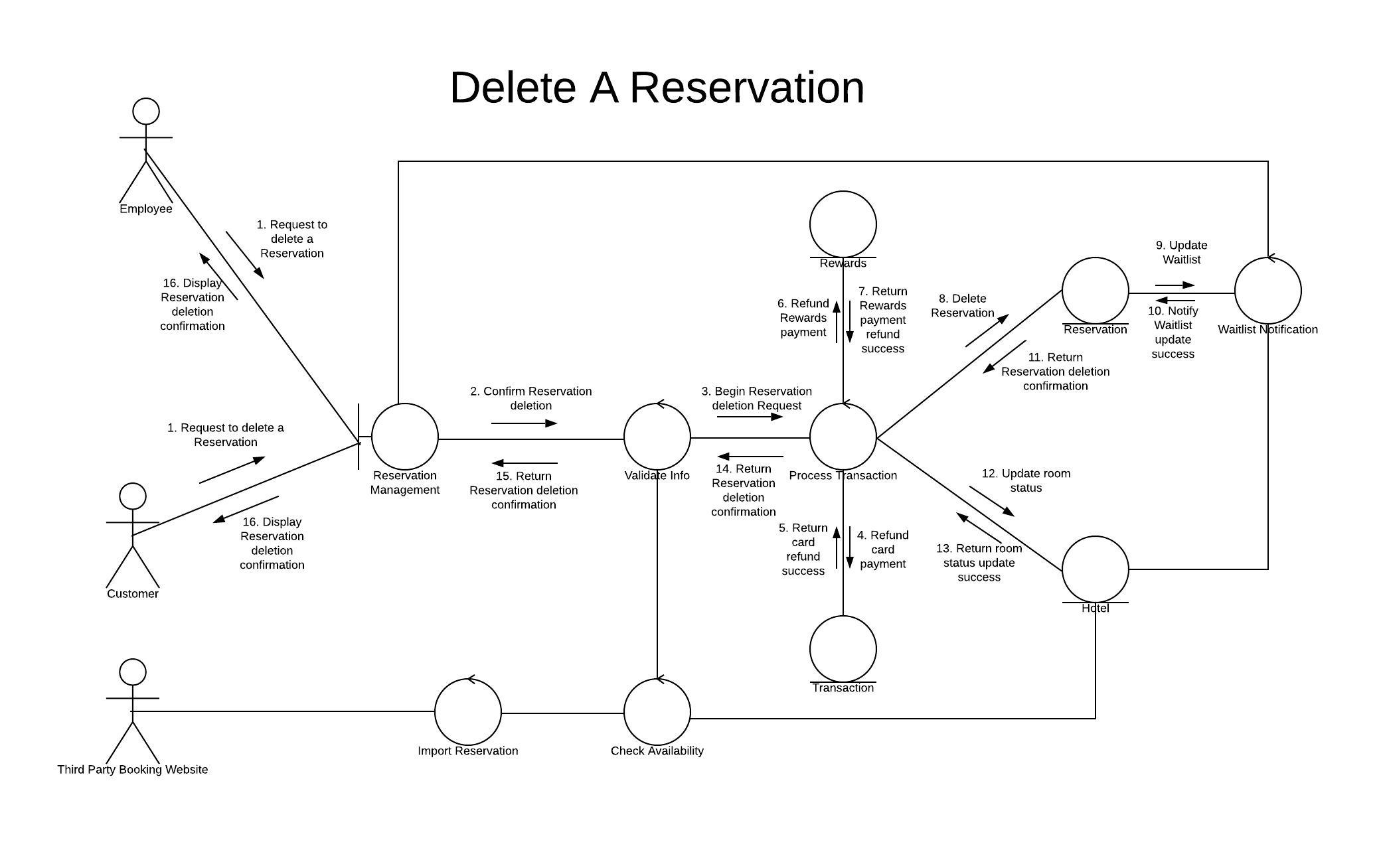
**4.4 Communication Diagrams**

**Used to model the message passing structure of the system functions.**

**Reservations:**

****

****

****

**5.0 Behavioral Model and Description**

**A description of the behavior of the software is presented.**

**5.1 Description for software behavior**

**A detailed description of major events and states is presented in this section.**

**5.1.1 Events**

**A listing of events (control, items) that will cause behavioral change within the system is presented.**

**5.1.2 States**

**A listing of states (modes of behavior) that will result as a consequence of events is presented.**

**5.2 State Transition Diagrams**

**Depict the manner in which the software reacts to external events.**

**6.0 Restrictions, Limitations, and Constraints**

Not Applicable

**7.0 Validation Criteria**

**The approach to software validation is described.**

**7.1 Classes of tests/Test Strategy**

**The types of tests to be conducted are specified, including as much detail as is possible at this stage. Emphasis here is on black- box testing.**

**7.2 Expected software response**

**The expected results from testing are specified.**

**7.3 Performance bounds**

**Special performance requirements are specified.**

**8.0 Appendices**

**Presents information that supplements the Requirements Specification**

**8.1 System traceability matrix**

Please see the Traceability Matrix spreadsheet at project root

**A matrix that traces stated software requirements back to the system specification.**

**8.2 Product Strategies**

**If the specification is developed for a product, a description of relevant product strategy is presented here.**

**8.3 Analysis metrics to be used**

**Summary Reports**

**8.4 Supplementary information (as required)**

Not Applicable