ONLINE HOTEL RESERVATION SYSTEM

PROBLEM DEFINITION:

The main aim of the entire activity is to automate the process of day to day activities of Hotel like Room activities, Admission of a New Customer, Assign a room according to customer's demand, checkout of a computer and releasing the room and finally compute the bill etc.

INTRODUCTION AND PROBLEM DESCRIPTION:

I have tried my best to make the complicated process of **HOTEL RESERVATIONSYSTEM** easy by simple modeling of UML diagrams. The main purpose of my exercise is performing each User's activity in computerized way rather than manually which is time consuming.

This project is used by two types of users.

- Online Users.
- Administrator (management of the Hotel).

Online users can see the required articles or news

Administrator can maintain daily updates in the hotel records. Administrator ismust be an authorized user. He can further change the password. There is the facility for password recovery, logout etc.

This system has been designed to computerize the following functions that are performed:

- Room Detail Functions
- Opening a New Room
- Modification to room assigned
- Check-in and check-out Detail Functions
- Admission of New customer
- Check-out of customer
- Room assigning related to customer's need.
- Statement of Customer Details
- Check-in customer-Check-out customer
- Room Details
- Total number of Customers in the Hotel
- Individual customer Report

SOFTWARE REQUIREMENT SPECIFICATIONS (SRS):

Purpose:

The software requirement specification will provide a detailed description of the requirements for the online hotel reservation system. This SRS will allow for a complete understanding of the system and the clear understanding of the system functionality will allow for the correct software to be developed for the end user. From this SRS, the system can be designed, constructed, and finally tested.

OVERVIEW:

The SRS is organized into two main sections.

- Overall Description.
- Specific Requirements.

The Overall Description

This section describes the general factors that affect the product and its requirements.

PRODUCT PERSPECTIVE

This online hotel reservation system is the stand-alone system. It Is totally self contained.

Hardware Interfaces

This system will be placed on PC's throughout the hotel.

Software Interfaces

In this system, we maintain two data bases. These databases include hotel rooms and customer's information. These can be modified by the end users. The room databases will include the room numbers and if they are vacant or occupied. The customers' information database maintains all the information about the customer such as name, number of occupants, assigned room, default room rate, phone number, whether or not the room is guaranteed, credit card number etc.

Product Functions

Reservation and Booking system

- Allows for typing in customer information
- Has a default room rate that is adjustable
- Includes a description field for the changed rate
- When a customer checks in, the room number will be changed to occupied in the database
- Ability to modify a reservation
- when a customer checks out the amount owed is displayed
- records that room is vacant
- records payment
- allows for space to write customer's feedback General Manager Services and Automated Tasks System
- Reports generated to audit hotel occupancy, future occupancy and room revenue.
- Exception reports listing to the normal cost
- Allows addition, deletion and modification of information on rooms and rates.

- Creation of users and assigning passwords.

User Characteristics

Educational level of hotel reservation system computer software-low Experience of this software-none Technical Expertise-little

Apportioning of Requirements

The audio and visual alerts will be deferred because of low importance at this time.

Assumptions and Dependencies

- The system is not required to save the generated reports.
- Credit card payments are not included.

SPECIFIC REQUIREMENT

This section contains all the software requirements at a level of details ,that when combine with the system context diagram, use cases, and usecase descriptions, is sufficient enables designers to design a system to satisfy those requirements

External Interfaces

This system will use the standard input/output devices for a personal computer. This includes the following:

- Keyboard
- Mouse
- Monitor
- Printer

User Interfaces

The user interface screens are:

- **Login** - log into the system as a CSR or Manager.
- **Reservation** – Retrieve button, update/save reservation, cancel reservation, change reservation, adjust room rate, accept payment type/credit card.
- **Check-in** – Modify room stay (e.g., new credit card), check-in customer (with or without a reservation), adjust room rate, special requests, accept payment type/credit card.
- **Checkout** – checkout customer and generate bill.
- **Hotel Payment** – accept payment for room.
- **Room service** - Create order, modify order, view order, cancel order, generate meal bill.
- **Customer Record** – Add or update customer records.
- **Administer rooms** – availability and rates.
- **Administer user** – create, modify, and delete users; change password.
- **Reports** – select, view, save, and delete reports.

Software Interfaces

The system shall interface with an oracle or access database.

Hardware Interfaces

The system shall run on Microsoft Windows based system.

Communication Interface

The system shall be standalone product that does not require any communication interfaces.

FUNCTIONAL REQUIREMENTS

Functional requirements define the fundamental actions that system must perform. Two categories in Functional Requirements:

1. Reservation/booking.
2. Management.

Reservation/Booking

- The system shall record reservations.
- The system shall record customer details.
- The system shall record the room number.
- The system shall display the default room rate.
- The system shall display whether or not the room is guaranteed.
- The system will generate unique confirmation for each reservation.
- The system will record expected check in time and date and also expected check out time and date.
- The system shall display the amount owed by the customer and record the payment.
- The system shall record the customer feedback.

Management

- The system shall display the hotel occupancy for a specified period of time.
- The system shall display the room revenue for a period of time.
- The system shall display an exception report where default room has been overridden.
- The system shall allow for the addition, deletion and also modification of information, regarding rooms, rates, and user profiles.
- The system shall allow managers to assign user passwords.

Non-functional Requirements

This section defines the needs in terms of performance, logical database requirements, design constraints, reliability, availability, security, maintainability, and portability.

CHANGE MANAGEMENT PROCESS

Changes to this document may be made after approval from the project manager and the client approval officer

USE CASE DIAGRAM

A Use Case specifies the behavior of a system or a part of a system and is a description of a set of sequences of actions. Use Case Diagram :Use case diagram shows the relationship among use cases within a system or semantic entity and their actors. The relationships area associations between the actors and the use cases, generalization between the actors.

Actors in Online Hotel Reservation System:

- Client.
- Travel Agent.
- Hotel Receptionist.
- Hotel Administrator .Use cases with brief description:

Inquire Information:

The customer needs to interact with the hotel interface through the personal system. It involves set of actions.

Reserve/Update Reservation:

To reserve the room, the customer needs to interact with the website. If the customer already reserved room, want some updates regarding the increase in occupancy, he must able to update the data by using his login and password which can be provided by the administrator.

Pay for Hotel:

This involves set of actions. If the customer ready to book or reserve the room, he needs to enter his complete details and occupancy level. After furnish the details in the form, he demanded to pay the advance. That can be through credit card/debit card. He must enter the correct card number.

Cancel Reservation:

The system software should support all the requests made by the each authorized customer. If the customer not satisfied with the facilities and occupancy of each room after few hours he checked in, he may want to cancel the reservation for remaining hours. System software must be in a position that the payment for the staying hours should captured and cancel the reservation.

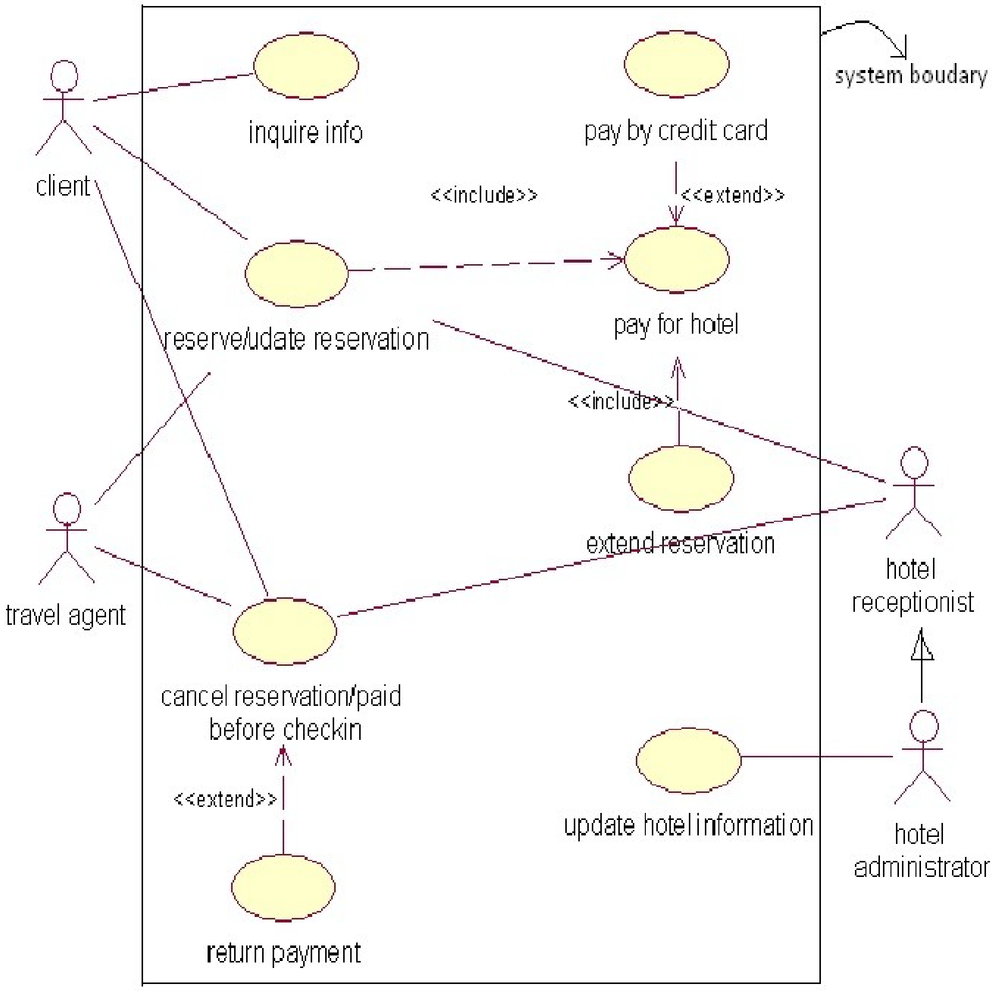
Update Hotel Information:

According to the customer feedback, the administrator is update the hotel information such as offers, discounts on rooms and meals provided by the hotel. When the customers' checkout from the hotel, the database need to Update and show the availability to the new customers.

Return Payment:

This use case extends cancel reservation. Whenever the customer wants to cancel the reservation, system must cancel the reservation and return the payment which remains. Each and every transaction made by the customer or administrator must be updated with the customer database and room database. This is the brief explanation about the Use cases in the hotel reservation system which helps in modeling the dynamic aspects of the system. Generalization relationship exists between hotel receptionist and hotel administrator. The relationship between them like child and parent relation. Here Administrator assumes as a parent whereas Receptionist acts as a child.

USECASE DIAGRAM:



CLASS DIAGRAM

Class diagrams are the most common diagrams found in modeling object oriented systems. A class diagram shows a set of classes, interfaces, and collaborations and their relationships. This gives the static design view of the system.

Class:

A class is rendered as a rectangle with three components separated with horizontal lines. The top compartment is used for displaying the name of the class; the next two compartments are used for listing attributes and operation of the class.

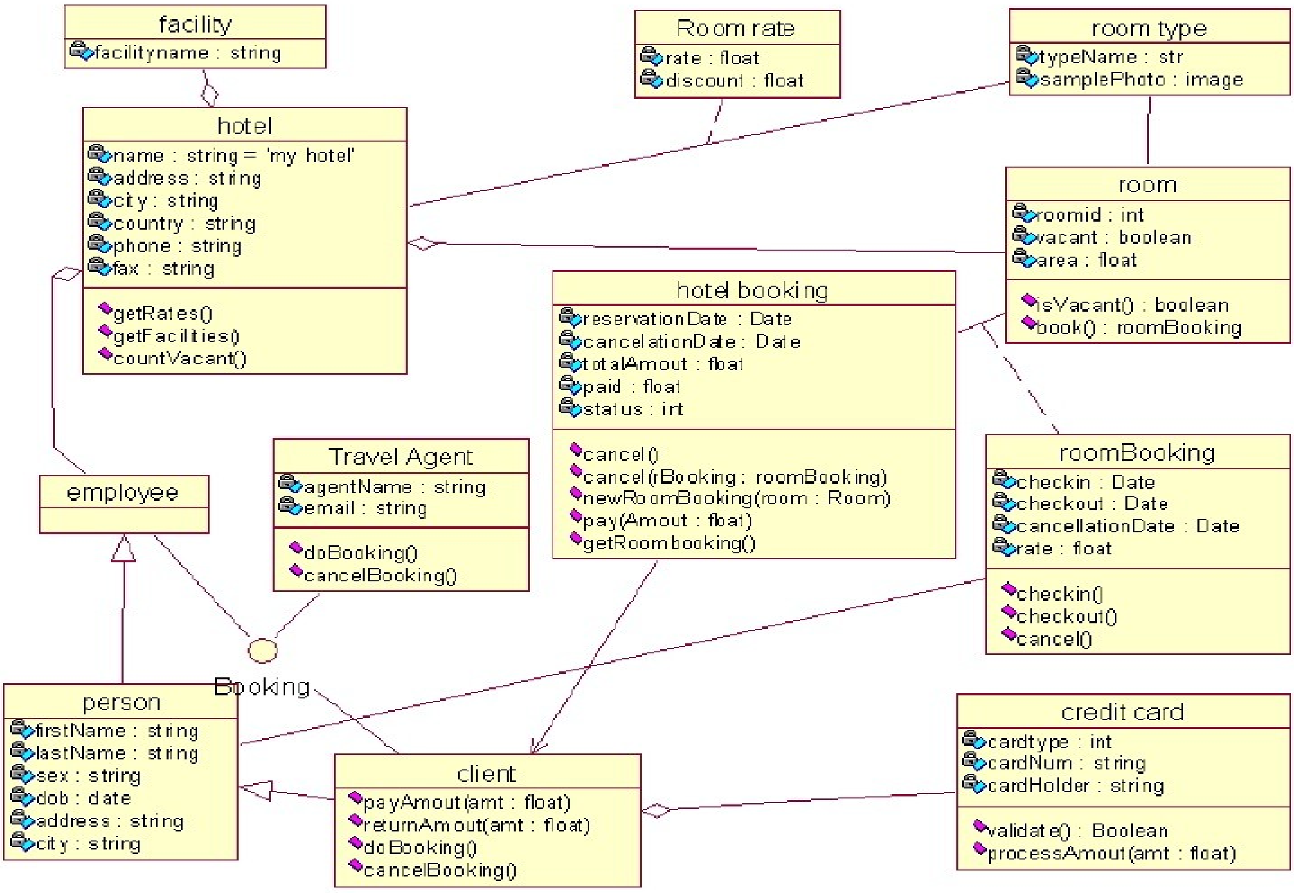
Classes declared for Reservation system:

- Facility.
- Room Rate:

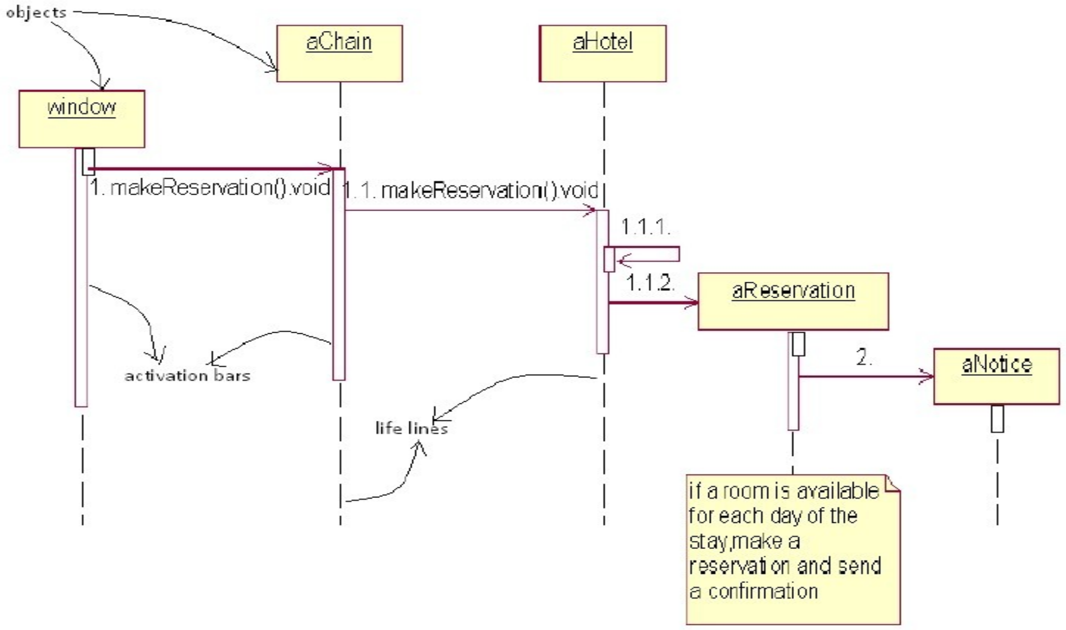
- Hotel.
- Room Type.
- Employee.
- Travel Agent.
- Hotel Booking.
- Room.
- Room Booking.
- Person.
- Client.
- Credit Card.

Interfaces for Reservation System:

Booking is an interface which can be declared to provide the operations such as doBooking(), cancelBooking() to the classes Employee, Travel Agent and Client where the implementation of those methods can be done. Here the relationship between the class and interface is the realization where classes realize the interface such that the interface give the contract and the classes need to carry out that work. This is the brief description about the class diagram and the classes, interfaces declared for the online hotel reservation system.

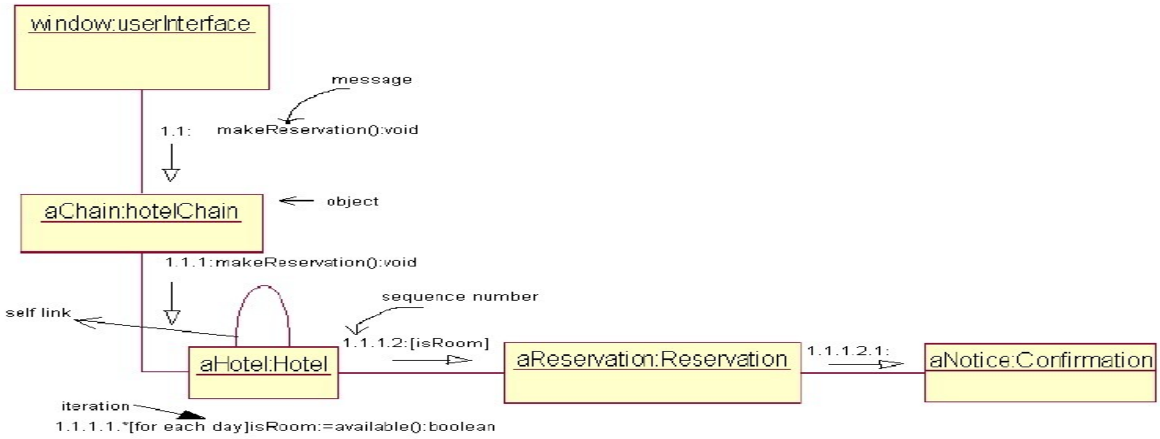


SEQUENCE DIAGRAM FOR ONLINE HOTEL RESERVATION SYSTEM


objects
aChain
aHotel
window
1. makeReservation().void
1.1. makeReservation().void
1.1.1.
1.1.2.
aReservation
2.
aNotice
activation bars
life lines
if a room is available for each day of the stay,make a reservation and send a confirmation

COLLABORATION DIAGRAM

Collaboration diagrams are also interaction diagrams. They convey the same information as sequence diagrams, but they focus on object roles instead of the times that messages are sent. In the sequence diagram, object roles are the vertices and messages are the connecting links. The object-role rectangles are labled with either class or object names (or both). Class names are preceded by colons (:).Each message in collaboration diagram has a sequence number. The top level message is numbered 1.Messages at the same level have the same decimal prefix but suffixes of 1, 2, etc.In the diagram which is at next page, it is clear that the message of make Reaservation() or doBooking() passing from one object to another followed by the sequence number from one to another object as 1.1,1.1.1, 1.1.1.2, etc.



STATE CHART DIAGRAM

State chart Diagram:

State chart diagrams are for modeling the dynamic aspects of the system .A state chart diagram shows a state machine. State chart diagram is useful to model the life time of an object. And it shows the flow of control from one state to another state.

State Machine:

A state machine is a behavior that specifies the sequences of states an object goes through during its lifetime in response to events.

State:

A state is a condition or situation in the life of an object during which it satisfies some condition, performs some activity, or waits for some event.

Event:

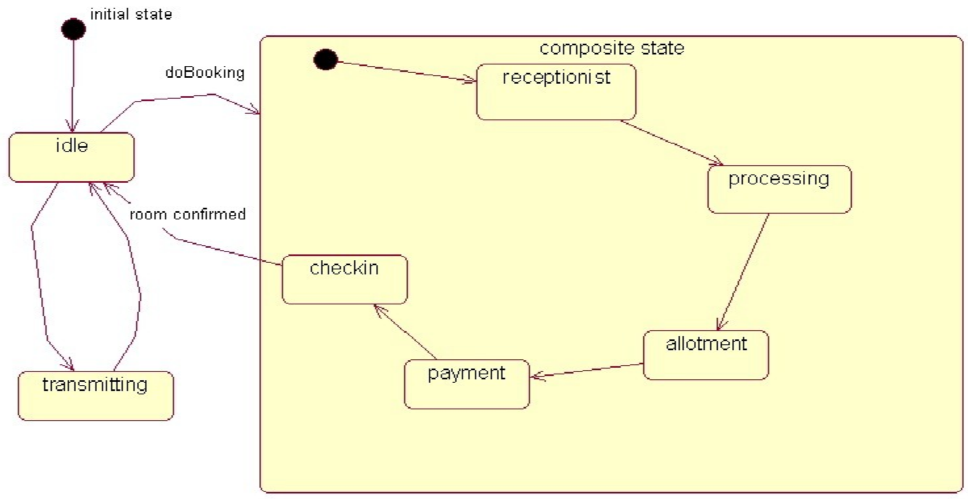
An event is the specification of a significant occurance that has a location in time and space. State chart diagram commonly contains:

- Simple states and composite states.
- Transitions, including events and actions.

Diagram Description:

From the diagram which is for reservation of room in a hotel, it is clear that there is a transition from the idle state to composite state. Composite state is a state in which again the object undergoes different transitions throughout its life time. Whenever the customer interact with the window interface and make a request for reservation, then the object states changes from idle to composite state. In composite state, the object moves from one state to another until the customer reserved room by paying some amount in advance. Once the room reserved, then the reservation letter may send to the customer mail id by moving its state to transmitting.

Once the room reserved by the customer, then the object comes to the idle state again by coming out of the composite state. This is the brief description about the state chart diagram for the hotel reservation system.



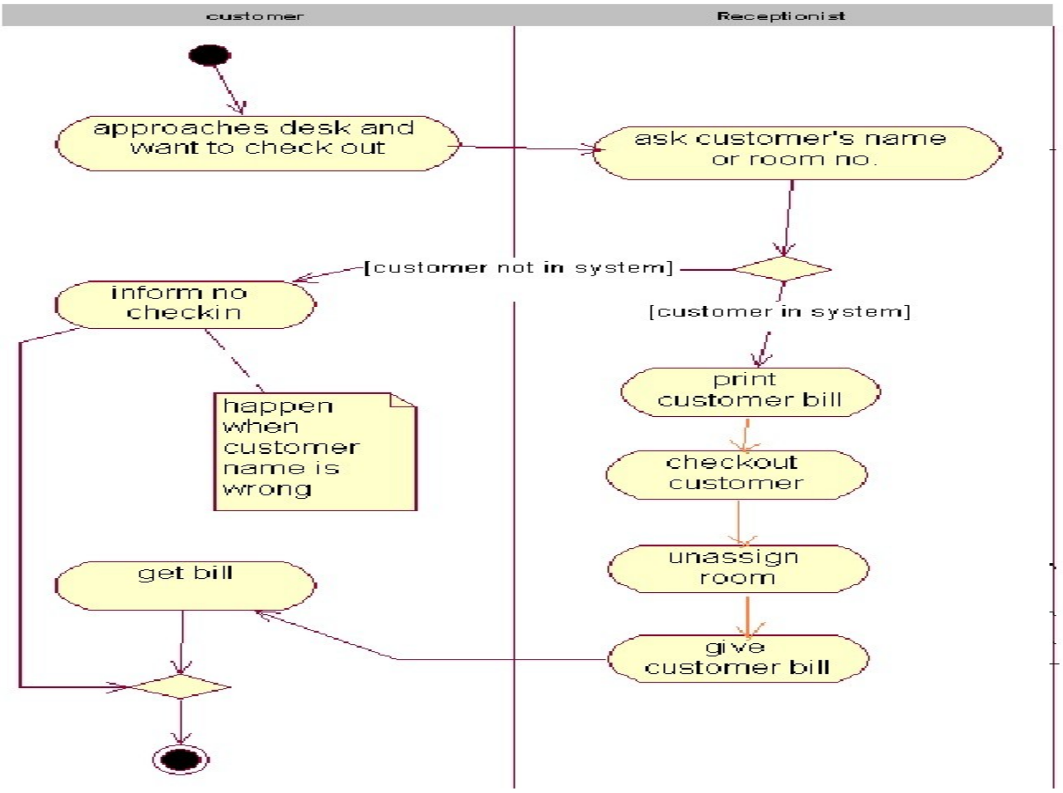
ACTIVITY DIAGRAM

Activity Diagram:

An activity diagram is essentially a fancy flowchart. Activity diagrams and state chart diagrams are related. While a state chart diagram focuses on an attention on an object undergoing a process, an activity diagram focuses on the flow of activities involved in a single process. The activity diagram show how the activities depend on one another.

Diagram description:

The activities are rounded rectangles. The activity diagram for the customer who wants to check out of the room from the hotel with swim lanes can be shown in the diagram. Each swim lane talks about each object i.e. which object is responsible for which activity. A single transition comes out of each activity, connecting it to the next activity .Here we are dealing with customer and receptionist/administrator. From the diagram, the activity started by the customer object which starts as request made by the customer wants to check out of the room at desk top interface. Then it given a form such that asking about the customer details which are already loaded in the database. If the customer enters the information which matches with the database information, then the customer bill should be printed. And then the sequence of activities is done step by step such that checkout customer, unassigned room, finally gives the customer bill to the customer. Then the customer gets the bill and simply pays by the credit card and simply leaves the room. If the details entered by the customer at the starting stage are not matched with the information from the database, the customer must be informed that he didn’t check into the hotel room.



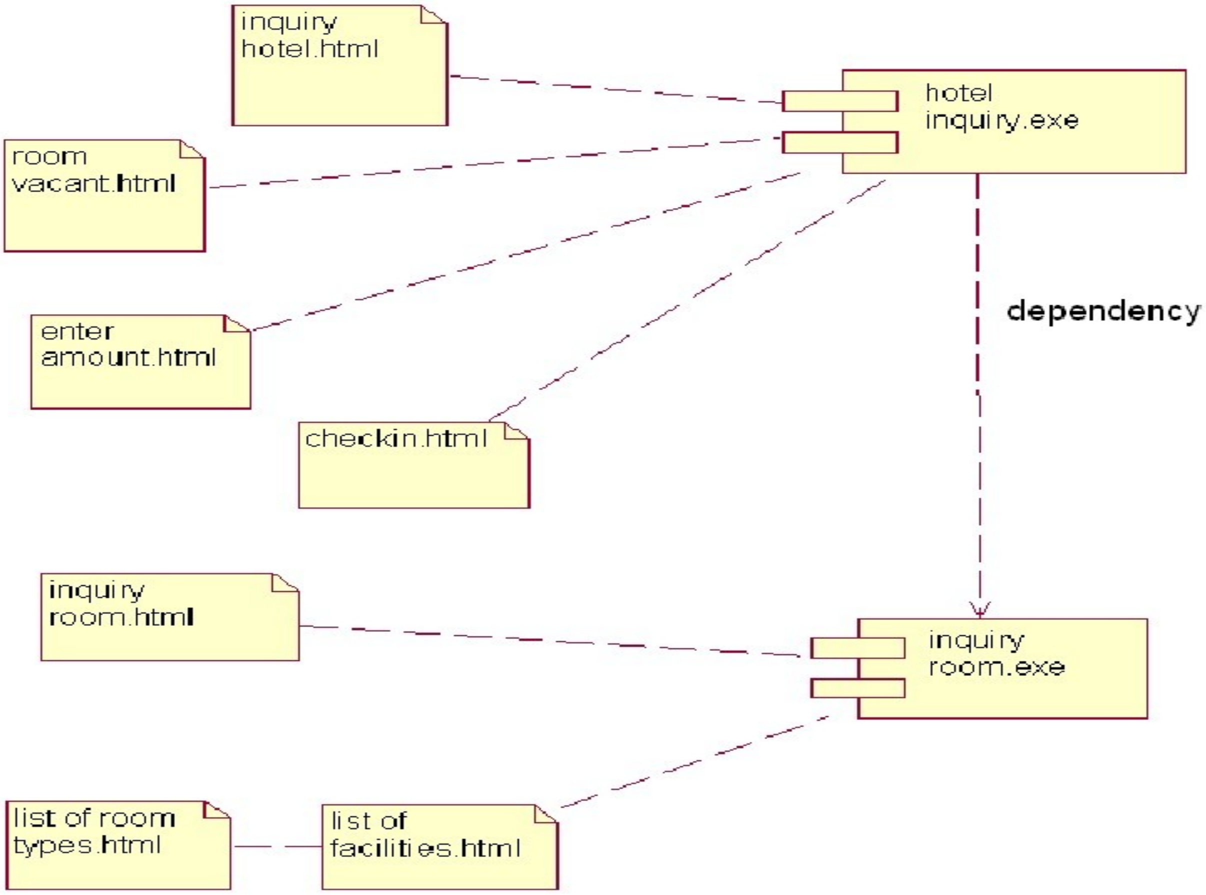
COMPONENT DIAGRAM

Component Diagram:

It helps to model the physical aspects of object-oriented systems. A component diagram shows the organization and dependencies among a set of components. We use the component diagrams to model the static implementation view of a system. This involves modeling the physical things that reside on anode such as executables, libraries, tables, files, and documents. A component diagram commonly contains:

•Components.
•Interfaces.
•Relationships.

Component diagrams mainly used to model source code, executable releases, physical databases and adaptable systems. In this online hotel reservation system we maintain mainly two databases i.e. customer database and room database. Mainly the information about the customer and rooms are stored in databases in tabular forms (relational).These tables are components. In the diagram, two components which are executable files areinquireroom.exe and inquire hotel.exe Which are having the documents regarding the available rooms and vacant rooms. And also the customer information. There is a dependency relationship between the two components in whichinquireroom.exe depends on inquirehotel.exe



DEPLOYMENT DIAGRAM

Deployment Diagram:

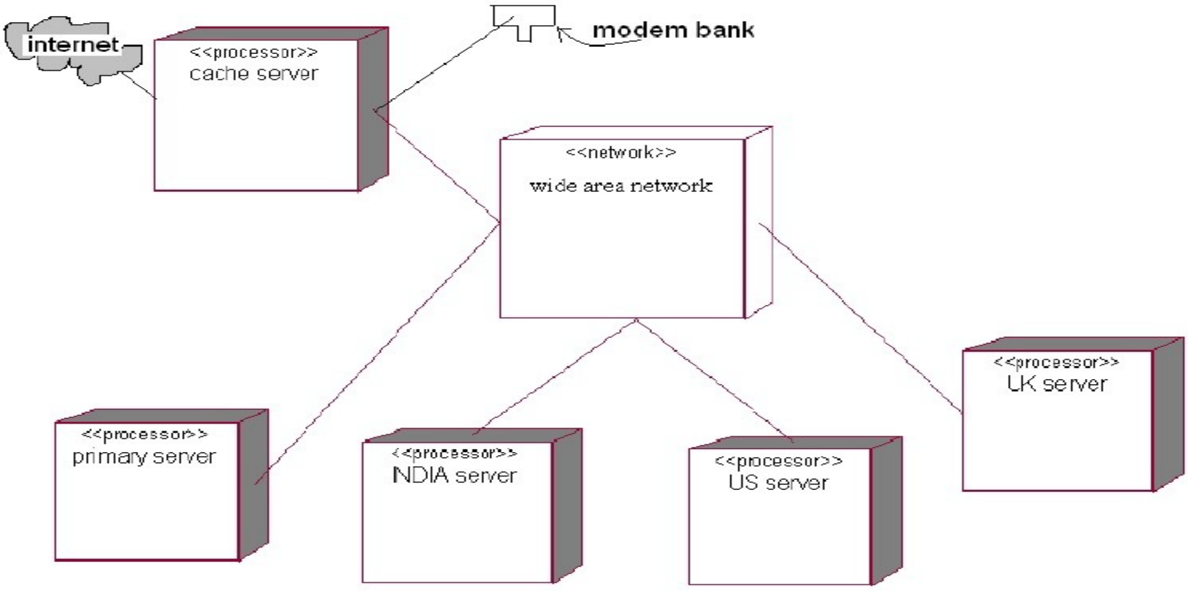
It also helps in modeling the physical aspects of an object-oriented system. A deployment diagram shows the configuration of run time processing nodes and the components that live on them. For most part, this involves the modeling the topology of the hardware on which the systems executes.

Deployment diagram commonly contains:

- Nodes.
- Dependency and association relationships.

Diagram Description:

Deployment diagram for the online hotel reservation system mainly depicts the processors which are widely distributed around the world and the components which are configured them .These all processors are connected through the device in the distributed system Cache server is used to store the transactions which are done by the customer at the time shortage of memory until the memory storage should extended. The primary server is placed at the head branch of hotel chain and remaining servers or processors are located geographically dispersed and connected through the internet .In our project we assumes as we have a chain of hotels which are at different places such as UK, US etc.



SOFTWARE ENGINEERING

CASE STUDIES : ONLINE HOTEL RESERVATION SYSTEM

SUBMITTED BY

C.INDHUMATHI

12MI004