Motel Management

OVERVIEW

Motel management software. Create a representation of a functioning motel and allow for conceptualization of the process of running a hotel.

GOALS

Who uses it?

- Any member of the Hotel Management

Why does he use it? What is he trying to accomplish when he uses it?

- To manage all aspects of the hotel such as employees, bookings, general info, etc.
- It is used to help him accomplish his managerial tasks through automation of smaller tasks

What functions does your software provide to the user that help him achieve his goal(s)?

- Shift management
- Manage bookings/availability
- Manage room service
- View and change Checkin/checkout times
- View guest information
- Search for a given sized room to view availability/occupancy information
- General Accounting (revenue, salaries, overall profits)

How does he use it? What **steps** does he go through in order to achieve his goal(s)? What are the **workflows** he progresses through when using it?

- Shift management
  - Create a shift for any department
  - Assign a department member to that shift
  - Add shift hours to employees total hours worked
- Manage bookings/availability

- - ○ Declare how many total rooms exist
    - ○ Distinguish which rooms are currently available and which are currently occupied
    - ○ Know when rooms will be available and unavailable in the coming dates
  - Manage room service
    - ○ When room service is requested add it to a queue
    - ○ Manage/Distinguish different types of request, such as food, cleaning, other stuff
    - ○ Make sure the request goes to the correct department
  - View guest information
    - ○ Search by guest
    - ○ Show guest info (Phone #, Total $ spent at the hotel)
  - Search for a given sized room to view availability/occupancy information
    - ○ Book available rooms
    - ○ See if room is "special"(balcony room, suite)
  - General Accounting (revenue, salaries, overall profits)
    - ○ Total money invested
    - ○ Monthly expenses
      - ■ Employee salaries
      - ■ Bills

What data will your system deal with to meet the user's needs? Define this in terms of **data only** - (classes, instance variables, enums) - **no logic yet**

- Employee class
- Guest class
- Room class
- Hotel class
  - ○ Instance of employee
  - ○ Instance of guest

What data structures should you use to store and access your data? Decide based on how the user will use the system; pick the data structures that **work best for what the user wants to accomplish**

Stage 2: Data Model

What data will your system deal with to meet the user's needs? Define this in terms of **data only** - (classes, instance variables, enums) - **no logic yet**

- Hotel class
  - ○ HashMap, room number to room object

- ○ Array of employees
- ○ Int currentBalance
- ○ Int totalMoneyInvested
- ○ Int bills
- ○ Int monthlyExpenses
- ○ Queue of Service staff on duty
  - ■ Queue of employees, when a shift is over the next employee moves to the top and is now 'on duty'
- ○ Queue of Cleaning staff on duty
- ○ Queue of kitchen staff on duty
- ○ RoomsReqRoomService Queue of Rooms
- ○ RoomsReqMaintenece Queue of Rooms
- ○ RoomsThatNeedCleaning Queue of rooms
- ○ HashMap Directory
- ● Employee class
  - ○ Boolean onDuty
  - ○ String name
  - ○ String staffPosition
  - ○ Double hourly rate
  - ○ In weeklyHours
  - ○ Int potentialHours (to calculate overtime needs or to manage budget)
- ● Guest class
  - ○ Instance of Room class
  - ○ String Name
  - ○ Int Age
  - ○ Int ID
  - ○ Int phoneNumber
  - ○ Int moneySpent
  - ○ Int daysBooked
  - ○ Int daysLeftInRoom
  - ○ Int checkInTime
  - ○ String encodedCreditCard
- ● Room class
  - ○ Boolean isBooked
  - ○ Boolean NeedsService
  - ○ Boolean NeedsCleaning
  - ○ Int daysBooked
  - ○ Final int roomNum
  - ○ Int roomPrice

- Boolean isNonResidential
- Final Boolean isSpecialResiRoom
- String RoomTitle
- String checkoutDay
- Instance of guest
- Int extraExpenses