JEE
Mini Project
Hotel Bookings Management System (HBMS)

Document Control

Reference No:			
Security Classification:	**Internal**		
Issue Date:			
Author(s):	**Name**		**Title**
	Shrilata Tavargeri		Manager – T & D
Reviewer(s):	**Name**		**Title**
	Training Team		Executive, Managers – T&D
Issuer(s):	**Name**		**Title**
	Dinesh Anantwar		Sr. Manager
Issuer Signature(s):			
Distribution:	**Name**		**Title**
	Team		All Designations in Training
Document History:	**Date**	**Revision**	**Change**
	Nov 24, 08	0.01D	Initial draft
	Nov 29, 08	0.02D	Review comments incorporated
	July 03,13	0.03D	Changes in Miniproject document content to standardize for 60 days training.
	June 29,15	0.04D	Changes in Miniproject document content as per the upgraded courses
	June 27,16	0.05D	Changes in Miniproject document as per upgraded courses

Table of Contents

1 INTRODUCTION

This document outlines a mini project for the JEE LOT. The project is to develop Hotel Bookings Management System (HBMS). This document contains the work flow of the system and gives guidelines on how to build the functionality gradually in each of the course modules of the JEE LOT.

1.1 SETUP CHECKLIST FOR MINI PROJECT

Minimum System Requirements

- Intel Pentium 90 or higher (P166 recommended)
- Microsoft Windows 95, 98, or NT 4.0, 2k, XP, Windows 7
- Memory: 32MB of RAM (64MB or more recommended)
- Internet Explorer 6.0 or higher
- Oracle 9i client and access to oracle 9i server
- JDK 8
- Eclipse Luna
- JUnit 4.0

1.2 INSTRUCTIONS

- The code modules in the mini project should follow all the coding standards.
- Create a directory by your name in drive **<drive>**. In this directory, create a subdirectory **MiniProject**. Store your Project here.
- You can refer to your course material.
- You may also look up the help provided in the java
- The total time required to complete this mini project is 50 hrs.
- Since this project work will span over couple of months, you will need to take care of maintaining the code

2 PROBLEM STATEMENT

2.1 OBJECTIVE

Development of Hotel Bookings Management System (HBMS)

2.2 ABSTRACT OF THE PROJECT

This project is aimed at developing a Hotel Bookings Management System (HBMS). This system can be used to search for Hotel rooms and reserve them. This is an integrated system that contains both the user component, Hotel-Employee component and the Admin component. There are features like report generators etc in this system.

2.3 FUNCTIONAL COMPONENTS OF THE PROJECT

Following is a list of functionalities of the system. Wherever, the description of functionality is not adequate; you can make appropriate assumptions and proceed.

There are three categories of people who would access the system viz. customer, hotel-employee & Admin. Each one of them would have some exclusive privileges (for e.g. customer and hotel-employee can just search for hotel rooms and reserve them, but only the admin has the right to add new and modify hotels.)

1. Customer and Hotel-Employee should be able to

    - Register into the system.
    - Login to the system using his/her credentials.
    - Search for hotel rooms.
    - Book hotel rooms.
    - View Booking Status

2. The Admin should be able to

    - Login to the system using his/her credentials
    - Perform Hotel Management (add/delete/modify Hotel info like description, any special offers etc)
    - Perform Room Management (add/delete/modify Room info like revised tariff)
    - Generate various reports like:

- View List of Hotels
- View Bookings of specific hotel
- View guest list of specific hotel
- View bookings for specified date

2.4 TECHNOLOGY USED:

- *Front End :–*
  1. Java Classes coding
- *Business Logic Components and Services :-*
  1. Java Beans
- *Databases:-*
  1. Oracle 9i

3 IMPLEMENTATION IN JEE LOT

3.1 SUMMARY OF THE FUNCTIONALITY TO BE BUILT:

The participants need to develop the **HBMS** by building the functionality incrementally in each of the course modules of JEE LOT.

Sr. No	Course	No. of Saturdays	Functionality to be built
1	Programming Foundation with Pseudo code		Analyze the given case study
2	Web Basics (HTML 5,CSS 3, JavaScript, XML)		
3	Oracle Basics	1	Creating relevant database tables
4	OOP & UML	1	Creating relevant Use case and class diagrams
5	Programming Foundation with Pseudo code + Web Basics +Oracle Basics Test		
6	Core Java 8 & Development Tools (Junit, Log4j)	2	Developing Business components (java classes). Coding for test classes & testing the functionality using JUnit
7	Core Java with JAXB + Dev Tools + OOP/UML Test		
8	Servlets	2	Developing the web application using the prototypes. Implementing coding in java classes (business components) for displaying reports and Integrating java classes with business components to complete the entire functionality.
9	JSP		
10	Developer Workbench (PMD, MAVEN)		
11	Servlets + JSP + Dev Workbench Test		
12	Basic Spring 4.0	1	Prepare document for presentation.
13	Basic Spring Test		
14	Mini Project presentation		

3.2 GUIDELINES ON THE FUNCTIONALITY TO BE BUILT:

The functionality and components to be built in each of the course modules of JEE LOT is as follows: : Here screen refers to the Console Screen.

1. **Core Java (Duration: 10 hours)**

    a. Develop the following screens:
        i. Home page screen: Home page for the HBMS which provides a options for the login page.
        ii. Login Screen: Allows the valid user or amount to logon to the system and display the Main option screen.
        iii. View Hotel screen: For all the users, this screen shows a list of hotels and rooms.
        iv. Book Rooms: For the customers and the hotel-employees, this screen will allow booking of available hotel rooms.
        v. Add Hotels: This screen allows the admin to add new hotels or rooms.
        vi. Update Hotels: This screen allows the admin to update existing hotels or rooms.
        vii. Delete Hotels: This screen allows the admin to delete existing hotels or rooms.
        viii. View Reports: This screen offers administrator to view various reports by clicking on an appropriate link.

    b. In this course you need to develop the user interface using java classes coding and document the flow of your application. The screenshots should include the fields as per the functionality mentioned above. Also, include client-side validations using regular expression.

2. **Course: Oracle (Duration: 5 hours)**

    a. Create the following database tables:
        i. Users: This contains the list of valid users with details
        ii. Hotel: This will contain the list of hotels.
        iii. RoomDetails: This will contain the details of all the rooms available in all the hotels.
        iv. BookingDetails: This will contain the list of booked rooms and their details.

    b. The structure of the above listed tables is as follows:

i. Users: user_id(varchar(4)), password (varchar(7)), role(varchar(10), user_name(varchar (20)), mobile_no(varchar(10)), phone(varchar(10)), address (varchar(25)), email (varchar(15))
ii. Hotel: hotel_id(varchar(4)), city (varchar(10)), hotel_name(varchar (20)), address(varchar(25)), description varchar(50)), avg_rate_per-night (number(m,n)), phone_no1(varchar(10)), phone_no2(varchar(10)), rating(varchar(4)), email (varchar(15)), fax (varchar(15))

iii. Note : Description here could be a brief teaser about the hotel like – "Centrally located in the main city centre at Shivajinagar Railway Station, this budget accommodation is designed with spacious rooms…". It could also give some special offers etc.

iv. RoomDetails: hotel_id(varchar(4)), room_id (varchar(4)), room_no(varchar(3)), room_type(varchar(20)), per_night_rate (number(6,2)), availability (Boolean), photo (blob))

v. Note : Room_type could be Standard non A/C room, Standard A/C room, Executive A/C room, Deluxe A/C room etc.

vi. BookingDetails: booking_id(varchar(4)), room_id(varchar(4)), user_id(varchar(4)), booked_from (date), booked_to(date), no_of_adults, no_of_children, amount(number(6,2))

Note: You may add/normalize/denormalize the tables if your application so demands it.

3. **Course: OOP & UML (Duration: 5 hours)**
   a. Develop relevant Use case and Class diagrams for the **HBMS** application.

4. **Course: Core Java + Developer Tools (Duration: 14 hours)**

   a. Develop business components (java classes) for the following functionality:
      i. User verification (on Login): This component will verify if the user who is trying to access the system is a valid user. This verification is as against the valid users listed in the users table.

ii. Add/Delete/Modify: This component will allow the admin to add new hotels, update and delete existing hotels.
iii. Book Rooms: this component will allow the customers to book rooms.

b. Develop test classes for testing the following functionality
   i. Login
   ii. Booking Rooms
   iii. Modify hotel details
c. Test the application using JUnit.
d. Configure Logger to log the status of an application

5. Course: Java Classes+ Developer Workbench (Duration: 14 hours)

a. Convert all the java classes (business components) created in Java module to Java beans
b. Integrate all screens(java classes designed for this) with business components (java beans) to complete the entire functionality
c. Configure the DataSource and modify the data access classes to use DataSource object configured.

6. Documentation (Duration: 2 hours)

a. Project Documentation: Document your project details (Duration: 1 hour 30 mins).
b. Project submission: Submit your project with all the artifacts including the test cases & documentation (Duration: 30 mins).

3.3 EVALUATION AND ASSESSMENT PARAMETERS:

This miniproject will be done in groups of five. Each group will identify a Team Lead who will decide which team member will code for which functionality. This project shall be evaluated at the end of spring module.

Evaluation Criteria (out of 100):

Look and Feel,	**10**
Client-side validation	**15**
Code Documentation and using coding standards	**15**
• Functionality	**40**

Dataset	**10**
Appropriate test cases using JUnit 4.0; logging application using	**10**

Log4j	