CS2102 Database Systems
AY 2013/2014 Semester I

Project: Online Booking System

DESCRIPTION

The manager of your company, ThinkCan Pte Ltd, a local software house, has asked your team to design and implement the prototype of an online booking system. The prototype should be realistic in order to convince a major customer to commission your company to develop the full system, but should also illustrate the use of relational database technology in order to serve as an in-house showcase application for engineers in your company.

For example, in an online hotel booking system, the user can search the catalog of hotels and rooms based on location, price range, features (e.g. swimming pool, fitness club, etc) and category of rooms (e.g. standard, superior, single and double bed, etc.). They can book rooms in a hotel for a certain period only if there is no conflicting booking. They can cancel and modify their booking before the checking date.

The system should also have an interface for administrators to create, modify and delete bookings.

It is left to your creativity to design a realistic model for the description of the elements and ancillary information in the system. You should populate the database with sufficiently enough data to both make the demonstration realistic and to illustrate the use of interesting SQL and DBMS features.

You can refer to http://www.cheaptickets.com.sg/ and http://www.hotels.com/ for examples and data.

OBJECTIVES

The objective of this project is to familiarize you with database technologies, to give you an opportunity to use the school's available software and infrastructure, and to apply the concepts and techniques learned in class for the design and programming of a database application. The project is to be carried out in teams of 3 students. There is no constraints on the composition of groups, team members can be in different tutorial groups.

The evaluation of the project will consider the scope of concepts and techniques used and their relevance. For instance, you should try and appropriately use the simple and advanced SQL constructs that you have learned: simple queries, aggregate queries, integrity constraints, views, etc. Feel free to extend the application requirements and add features in order to demonstrate interesting use of the technology learned. The amount of data in the database should be sufficient for a complete and realistic demonstration of the system.

TECHNOLOGY

Figure 1 is an overview of the architecture of your application. The architecture consists of a Web server, a server page language and, of course, a relational database management system. The DBMS and possibly the Web Server can be located on the Cloud.

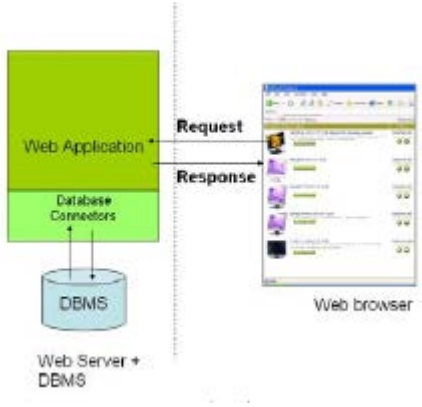

Figure 1: General Architecture

DELIVERABLES

The project deliverables are:

1. Two brief reports (2%+3%).
   a. The first report is due at the end of Week 5 (e-Learning week) and should contain the ER diagram for the application and the relational schema (in SQL DDL code).
   b. The second report is due at the end of Week 12 and should contain implementation details of the application (indication of the web server, server page language and database management system used, whether you use the zone or else); sample and representative SQL code of the functions it helps to implement; 2 or 3 representative screen dumps of the Web interface.

   All reports must include team members' names and matric numbers on the front page.

2. A demonstration of your system (5%). The system should include, but is not limited to, the following features:
   a. User browsing and searching of items. Items can be searched using multiple attributes.
   b. Functions to rank items if there is more than one item satisfying the search condition.
   c. User can create, delete, and modify his/her own booking, while administrators can create, delete and modify all the bookings. When creating a booking, the date must be valid (e.g., the departure date should be later than the arrival date, the departure date cannot be later than the actual date). When modifying a booking, make sure there is no conflict; otherwise, the modification cannot be processed.