**J2EE Project Proposal**

**By: Cup ‘O Java (Katie Sullivan, Yuliia Shymbryk, Ujjawal Tivari)**

**Definition of Project**

Our project is a hotel booking application, modeled after the site waringhousereservations.com. We chose to only model the booking functionality of the site, not the entire site.

In this site, users can select to stay at a large country inn with 4 different sites (hotels). They can select one of 5 different room types to stay in, and can select the duration of their stay. In order to sign up, the user needs to register themselves, and can save a pin number in order to access their registration information again. If they choose not to enter a pin, their booking will be saved, but they will have to re-enter information to make a new booking. They will still be able to modify or remove an existing booking. They can search for available rooms, add a booking, modify a booking, and remove a booking.

After making a booking, the user will be charged online, and will enter card information. We decided not to process payment, but our site will validate card information.

The administrator can access all bookings for all clients, and can add new bookings (such as when someone calls in to book), and can remove or modify all bookings.

**Parts of Application**

**User:**

1. User sees a page with option to create, change, or cancel a reservation.
2. To create a reservation, the user can search for availability by entering reservation details including: check in date, checkout date, rooms, number of adults, number of children and type of room. At this point, users may also enter a promotional or corporate code to receive discounts
3. After pressing enter, user is taken to a page displaying available rooms for the duration of their stay that match the room type requested. They can select a room, and begin making a reservation.
4. They user is taken to a reservation page, where either the user can sign in to access their account details, or they can enter their details, including credit information. All details will be validated. Once registered, user receives a confirmation page with a reservation number.
5. To change a reservation, the user needs to enter their reservation number, and credit card information. Once they press continue, the information will be validated in the database. If the reservation with matching details exists, user will see their reservation details populated in the table in the initial reservation page, and can modify it and save it as per above.
6. To cancel a reservation, user needs to enter reservation number, and credit card information. Once they press continue, the reservation will be removed.

**Admin:**

1. Admin will go to a separate page to login, and will be validated.
2. Once admin is logged in, they can search for a booking by reservation number, email address, or phone number and full name. After searching, it will return one or more reservations for that user.
3. Admin will then have the same ability to modify or delete bookings as the user.
4. Admin will also be able to create a reservation like the user does above.

**Application Implementation:**

We will be implementing our application using Spring MVC framework.

Our model classes will represent our database tables, and include classes for:

* Guest
* Booking
* Hotel
* Room

Our controller classes will implement the functionality of the site, including:

* Initial – returns the index page
* Search
* SearchResults
* AddBooking
* ModifyBooking
* DeleteBooking
* AdminSearch
* AdminLogin

Our views will match the controllers and include:

* Welcome/search page
* Search results page
* Add page
* Modify/delete page
* Admin login page
* Admin search page

Additional Configuration Classes:

* Database Handler (JDBC Template)

Responsibilities: