# Project: Event Catalog

Author: Christy Jaeson Augustine

URL: http://localhost:8080/myproject/

Student ID: 13025

# Project Description:

Allows user to select events based on user criteria and add deals to the events purchased. Manage profile, sign up , login.

Domain Classes:

* Deal.java:Manages deals



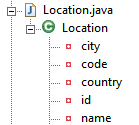
* User.java: Manages user profile



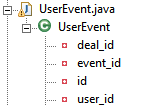
* Event.java: Manges Events



* Location.java: For criteria based event selection (criteria used: Location)



* UserEvent.java: to manage the event list each user books also to the deals purchased by user associated with the events



## PROJECT FEATURES:

> Sign Up new User

> User sign In

> Managing Profile [User can edit his profile details]

> Find Events based on criteria

* Area of Interest
* Location

> Find Deals

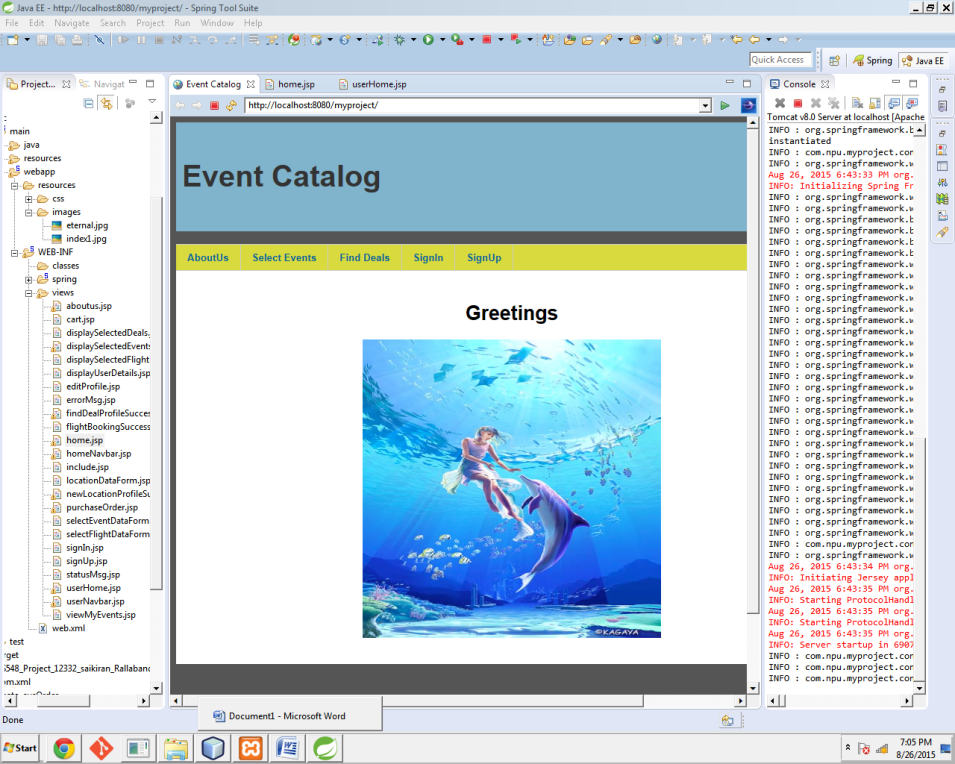
> Purchase and add deal to your events Purchase

> Viewing Purchase summary

## Features in Detail:

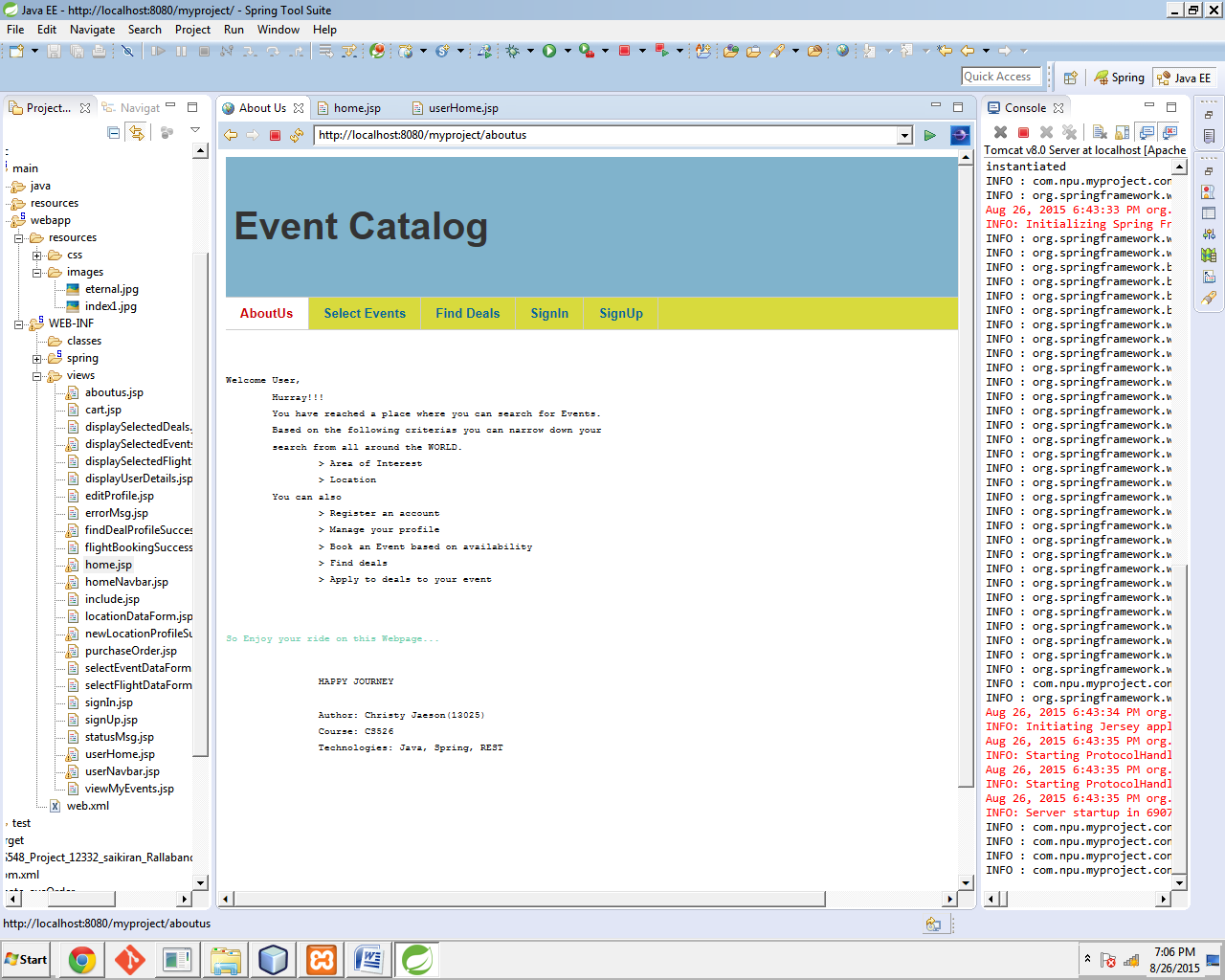
### Welcome Page:

This is the welcome page of the Event Catalog Project. Even GUEST users can find events and find deals but cannot purchase unless they signIn.

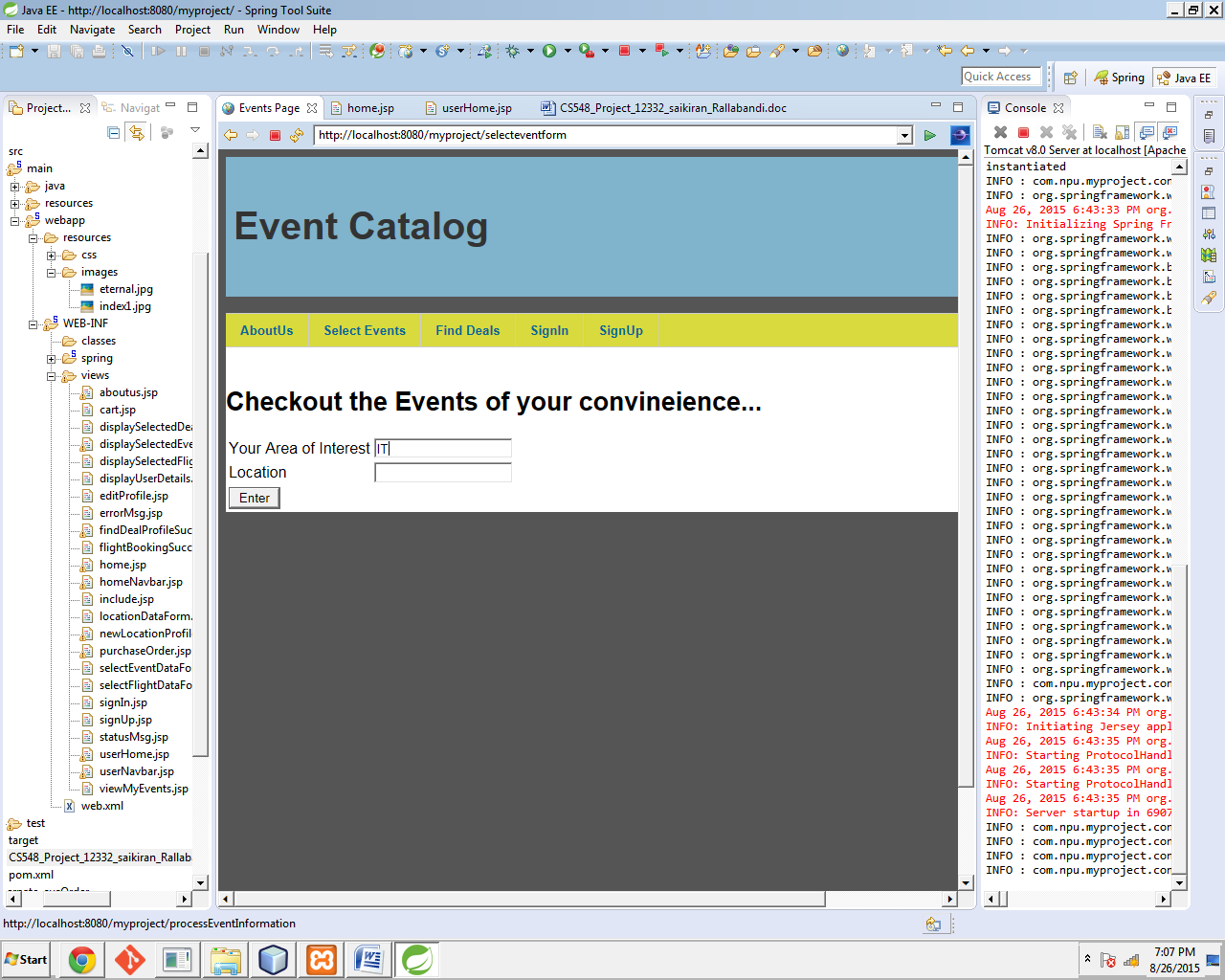


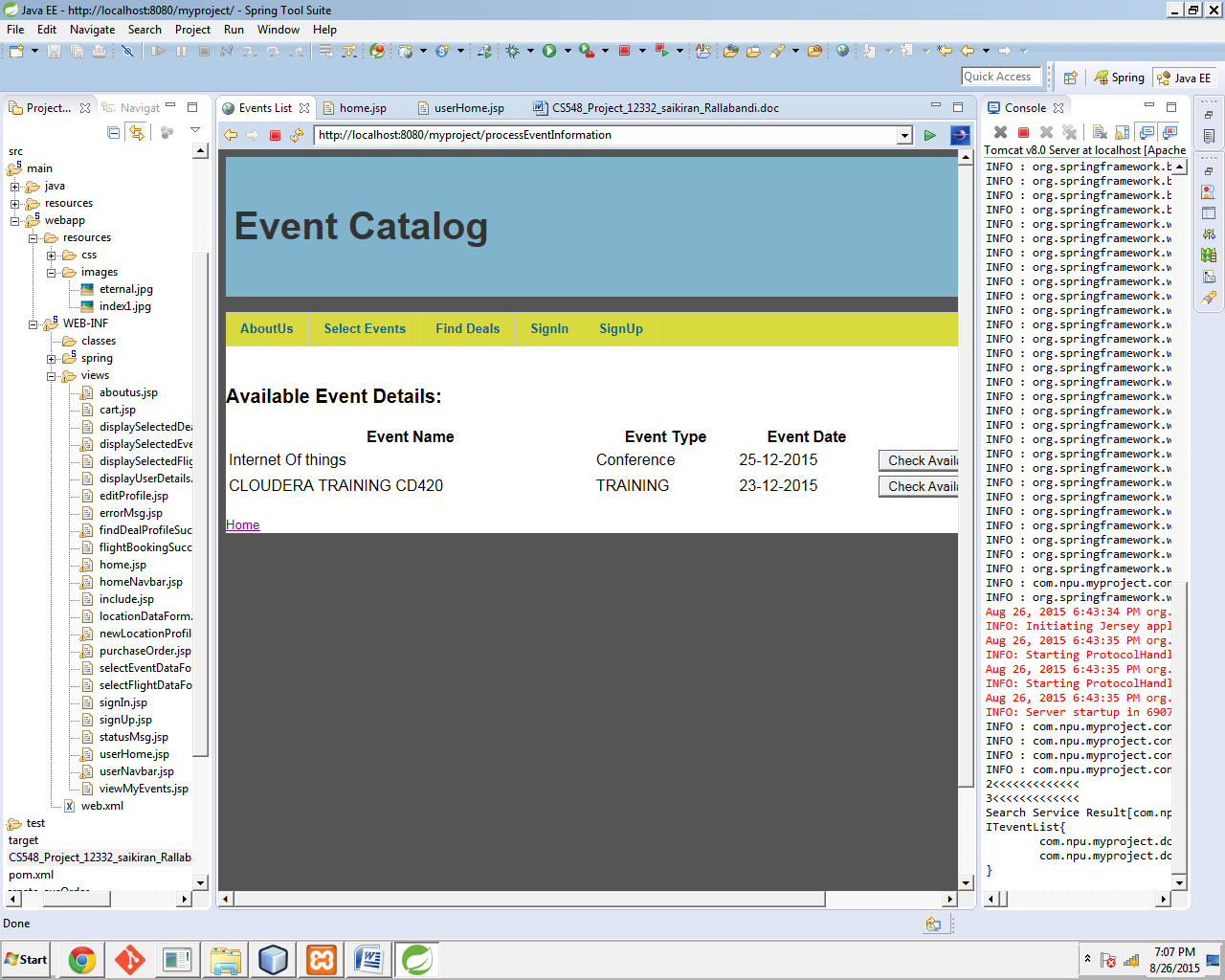
### About us Page:

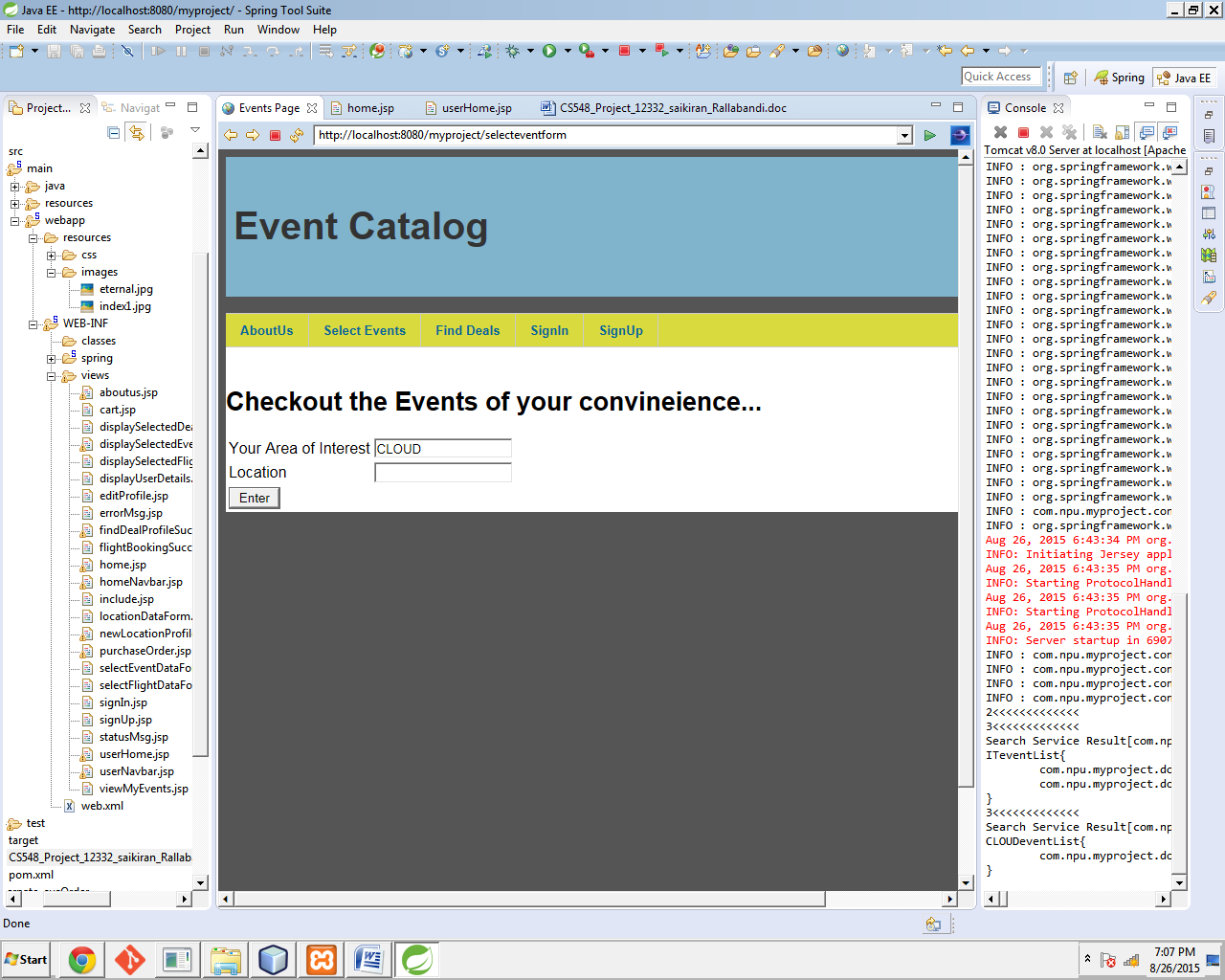
The about page provides an explanation about the feature of the website

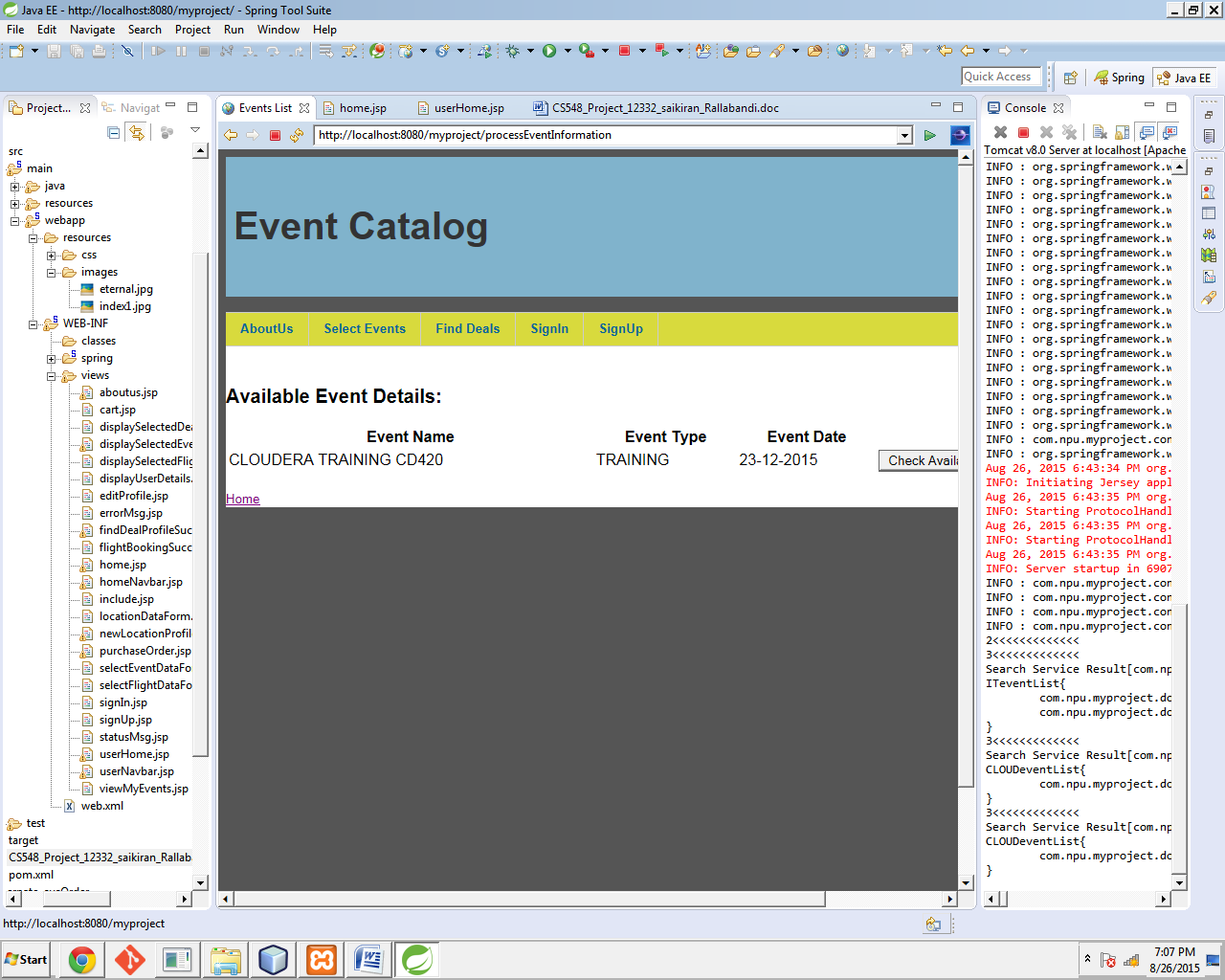


### Searching Events based on criteria:

User can provide their area of interest (e.g. IT/CLOUD/..) and user will be provided with events in the related fields, but when the GUEST user tries to book the events ,he will be prompted with SignIn page





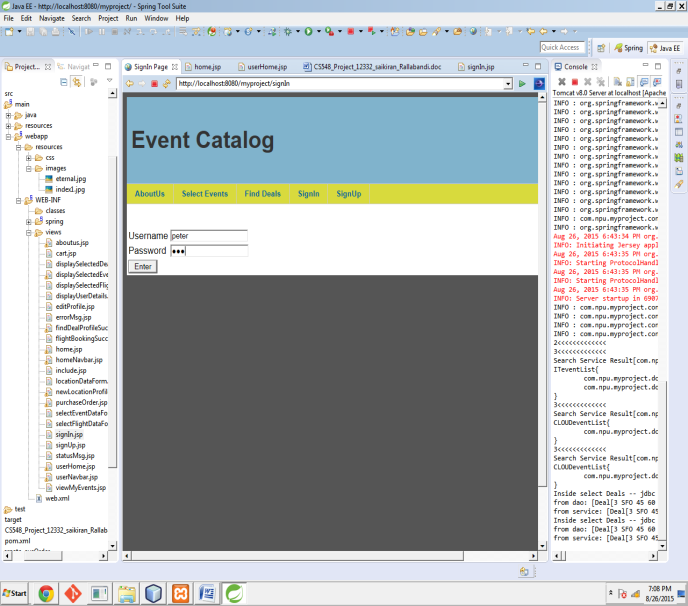
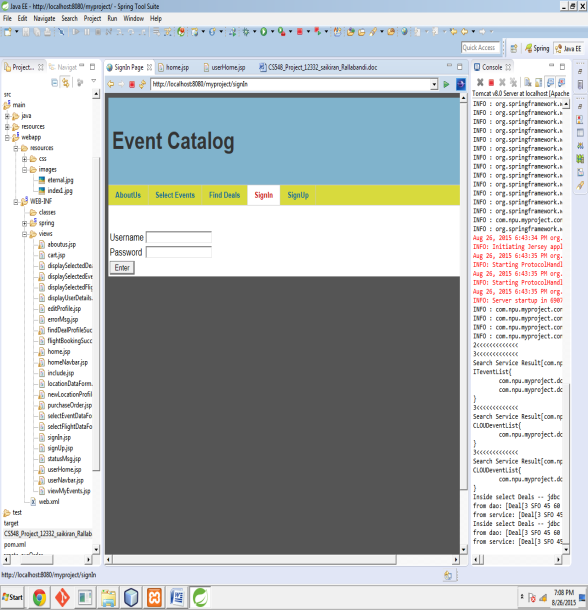


### View Deals:

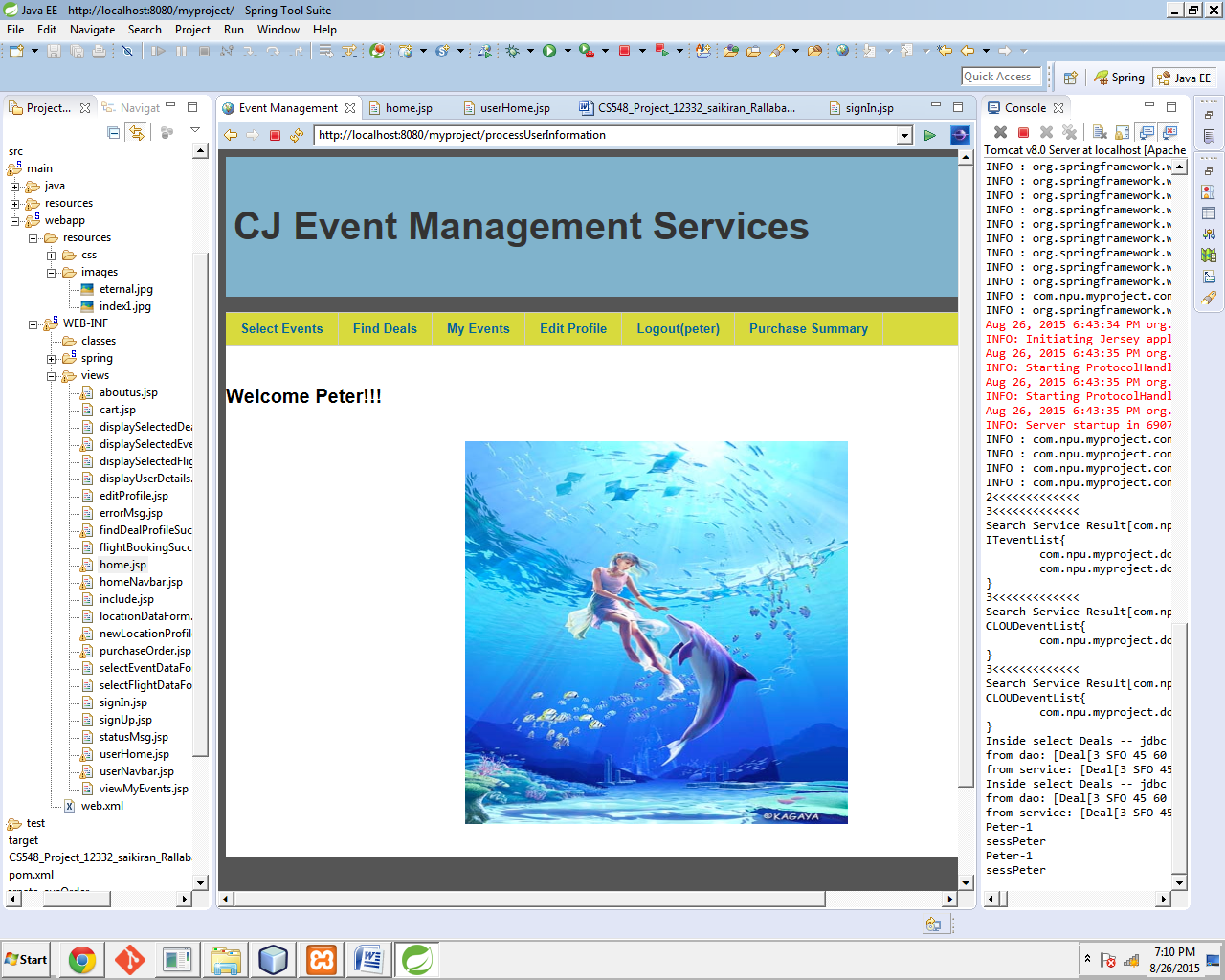
Guest users can also view available deals

### SignIn:

Here is an example of how to sign in as Existing User

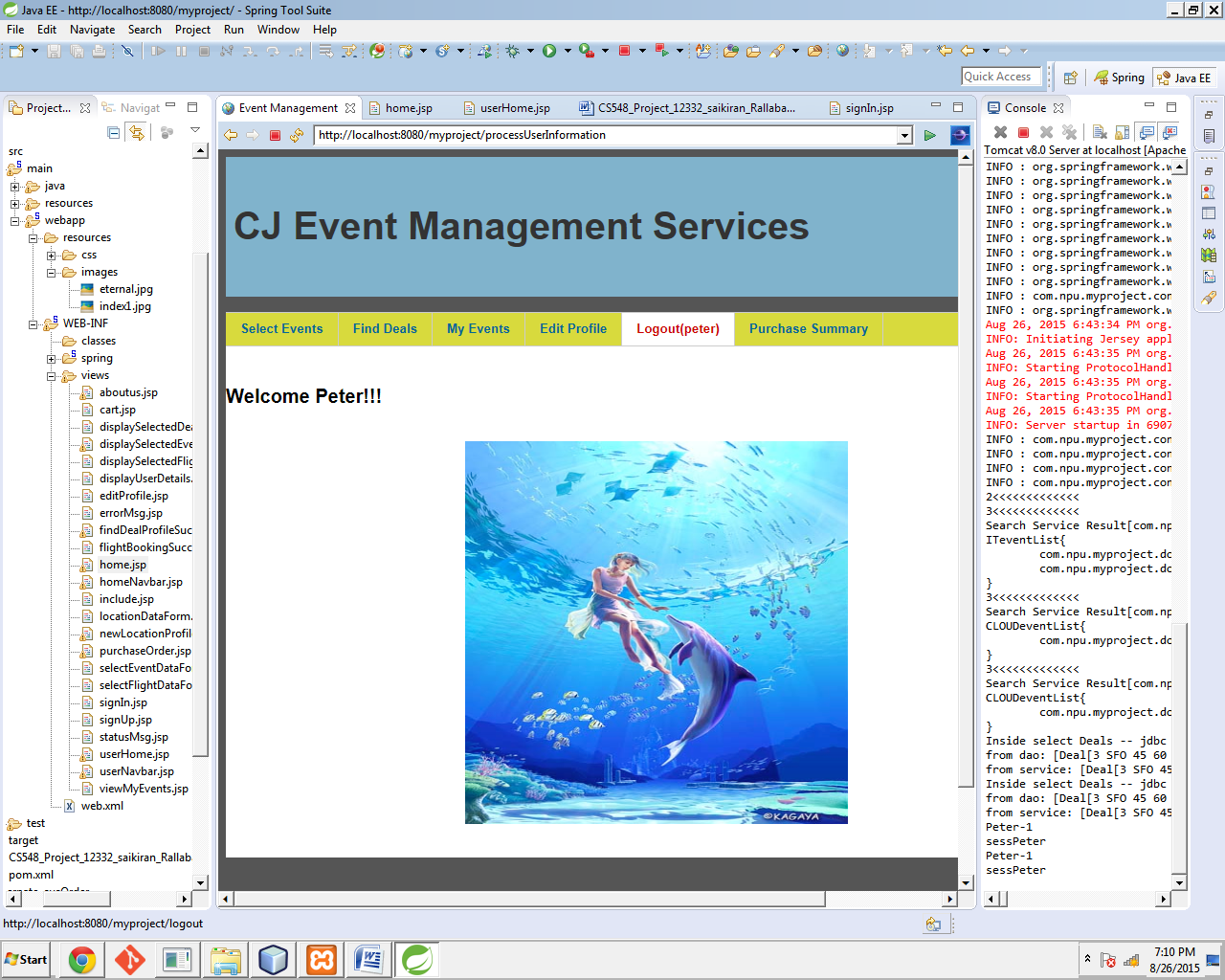


Successfull Login



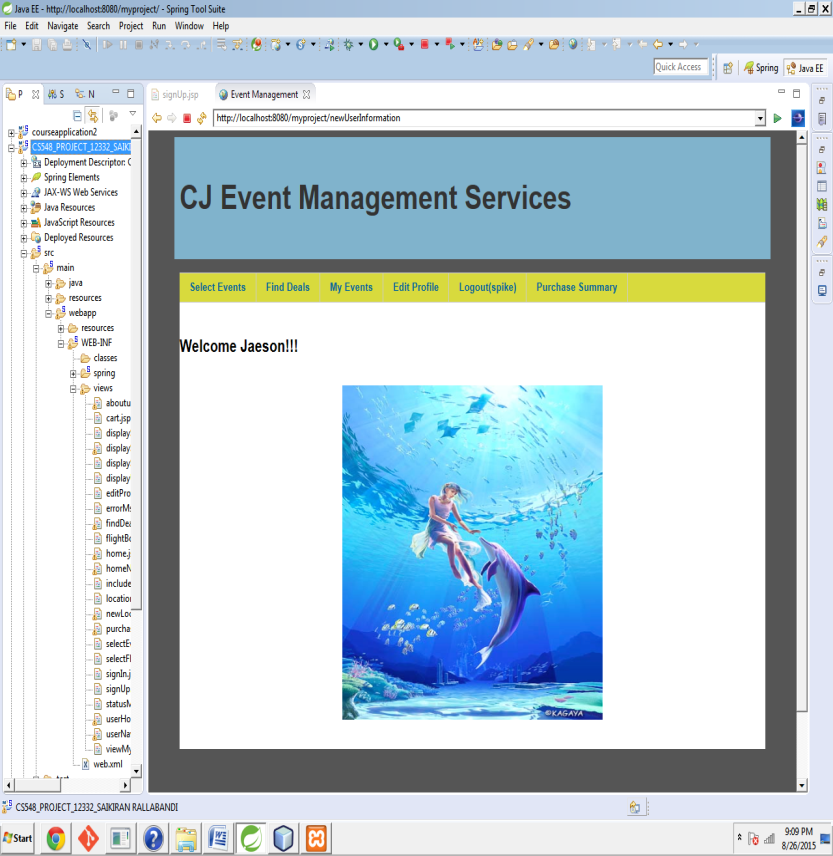
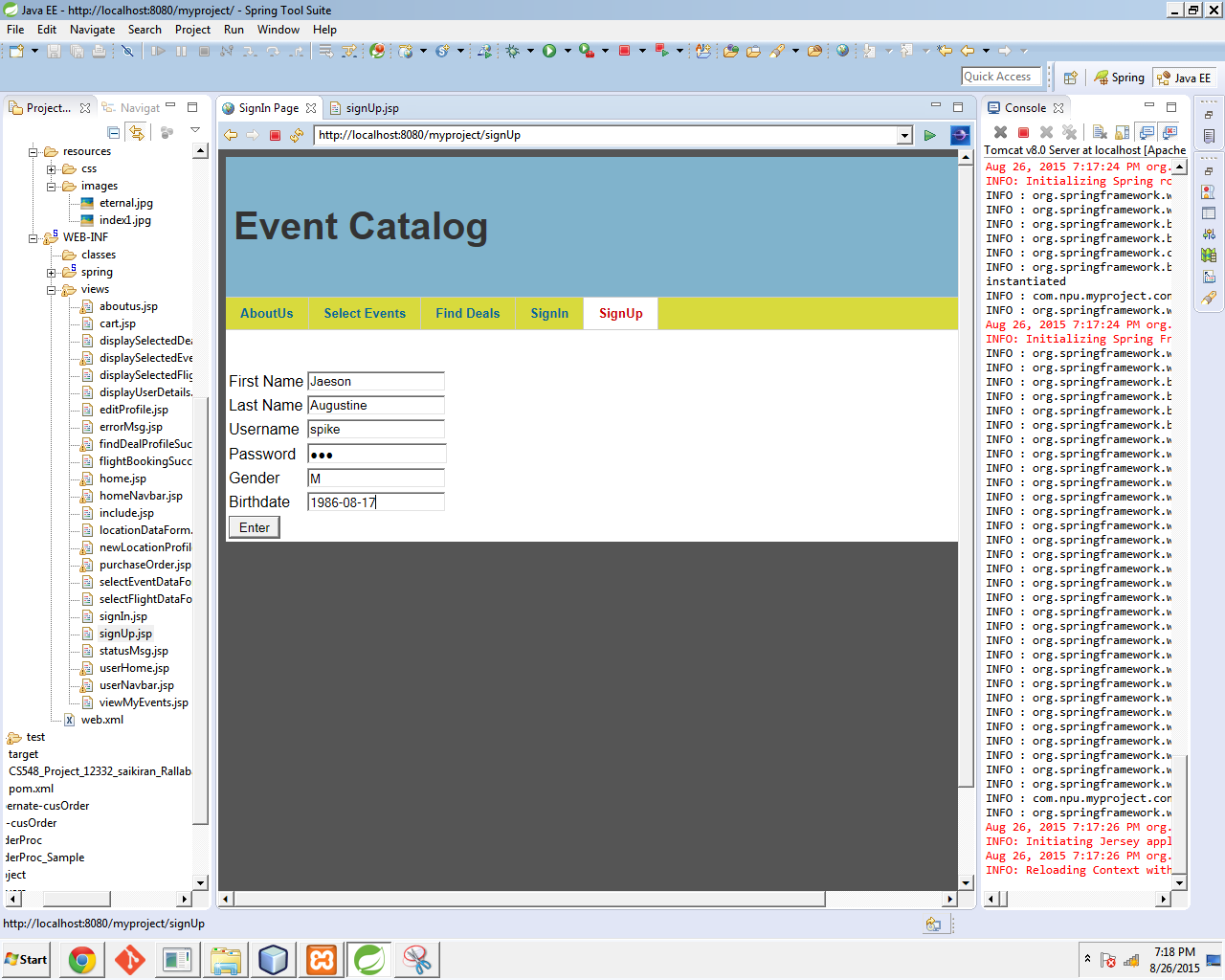
### Logout

You can Logout your Profile and your session will be cleared.

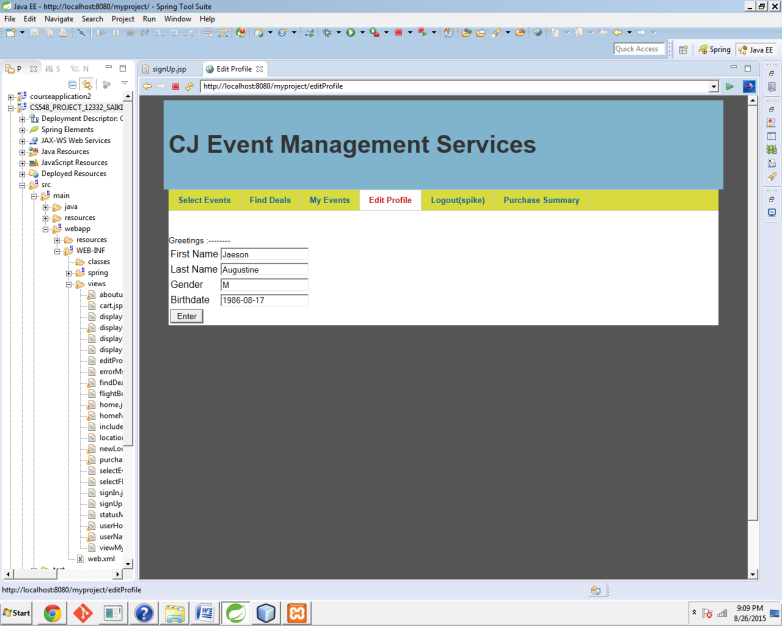


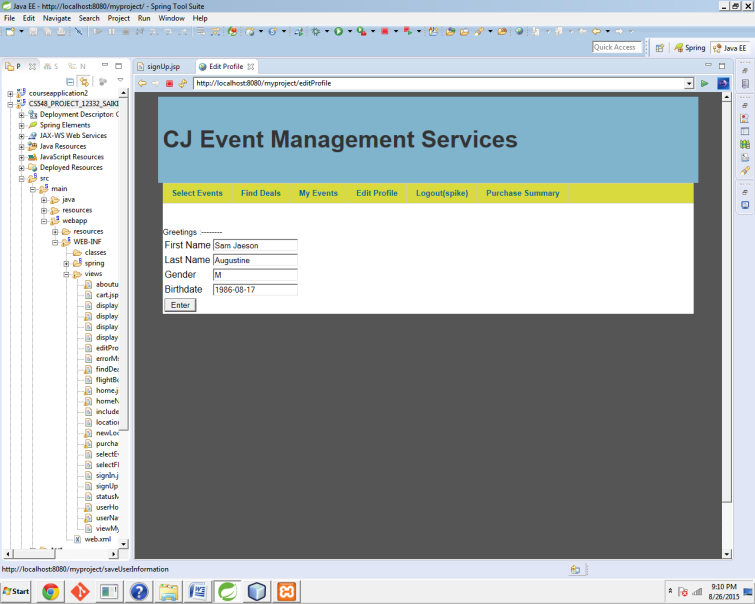
### Sign UP

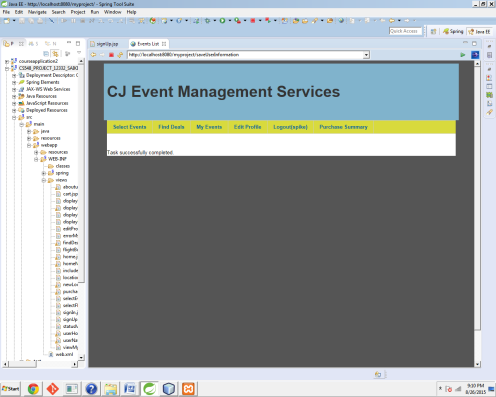
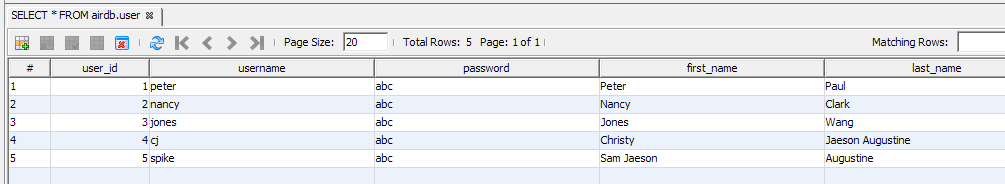
user can create his own new profile.



### Edit Profile





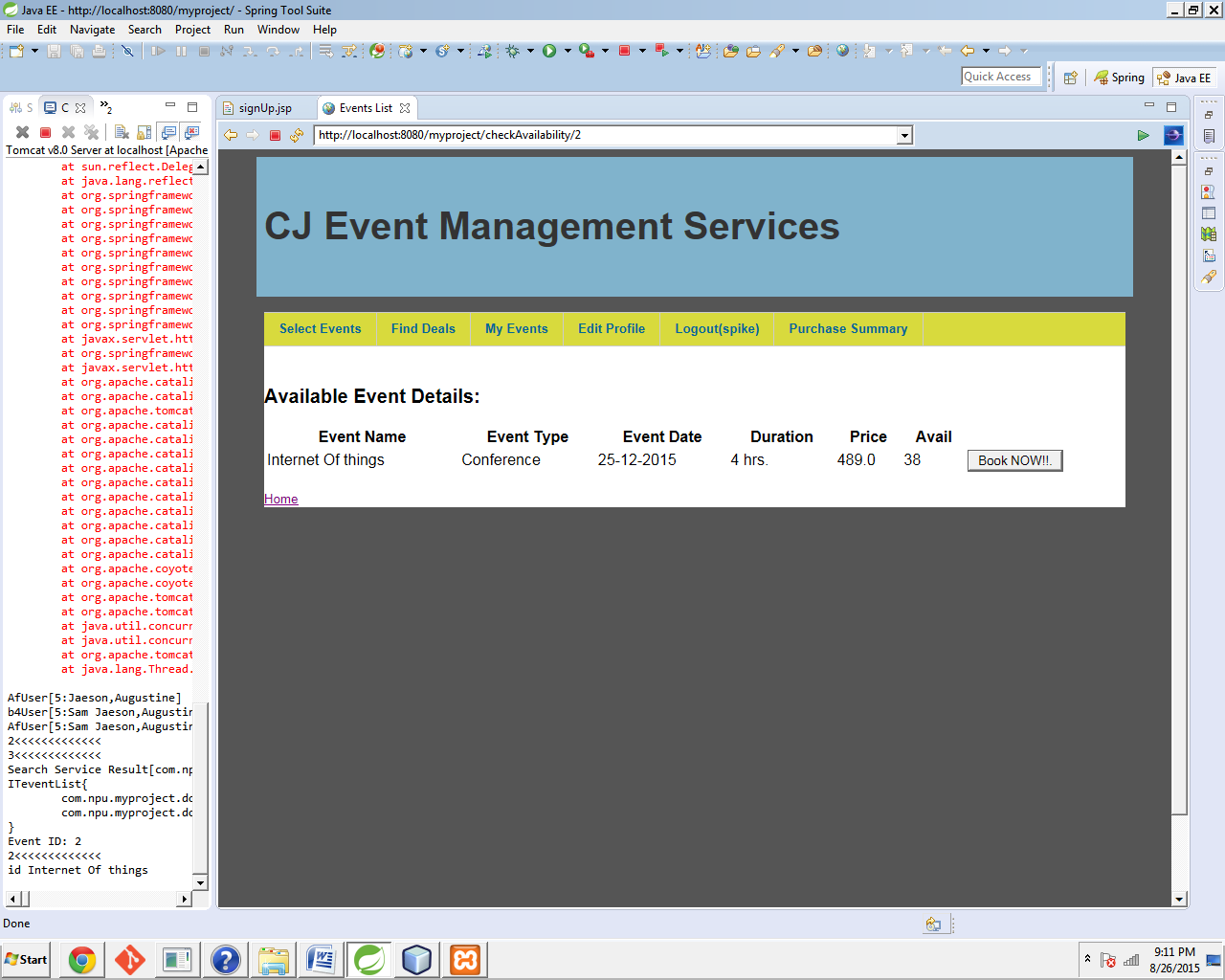
### Event Selection

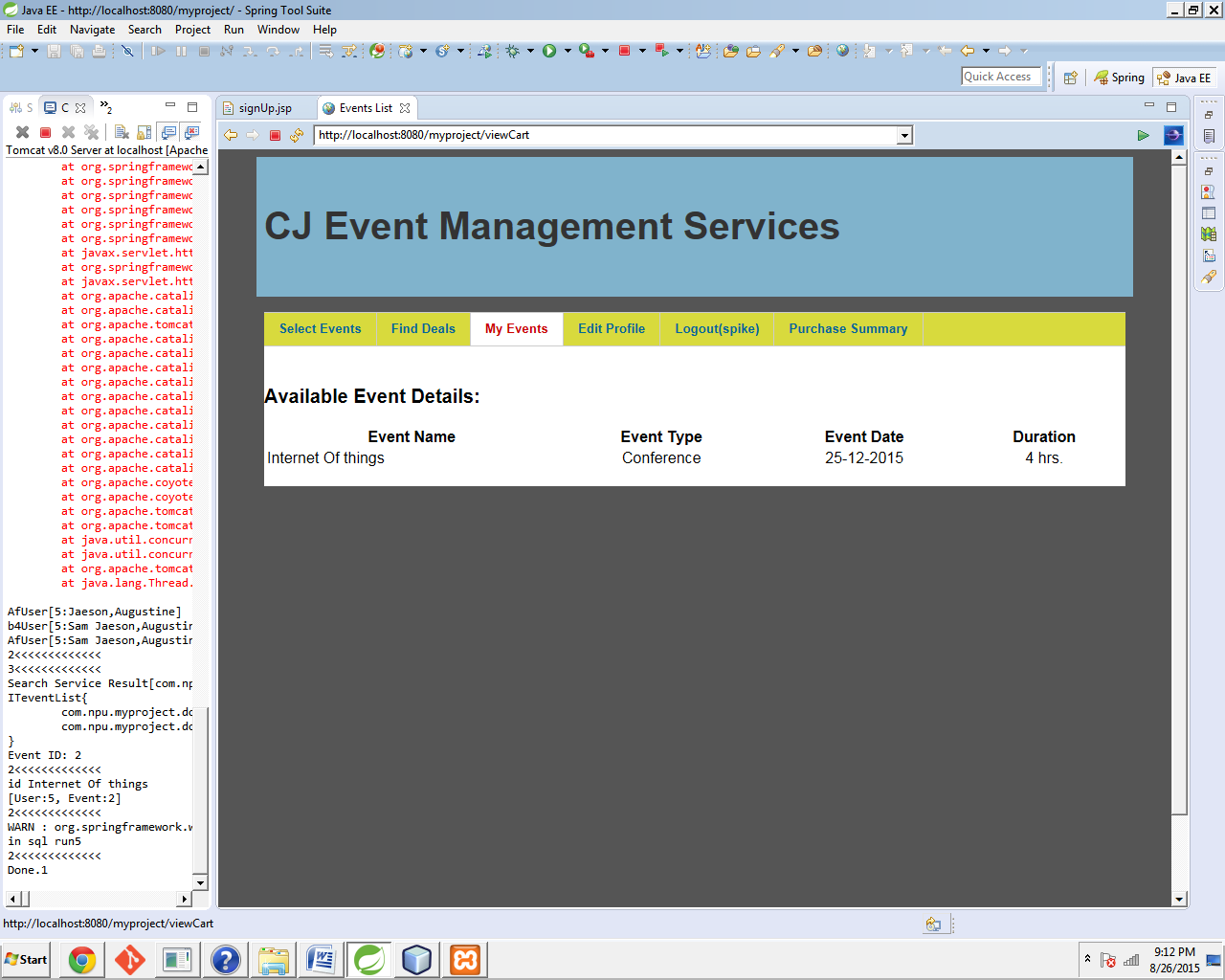
User will be shown the list of events as per his criteria and then the user can check availability.

If the available seats are more than zero then the user will be allowed to book and the availability will be reduced by one. This is a **compound transaction statement** used in the project.

The user can later see his events by clicking on the "My Events" tab

User's are of interest: IT

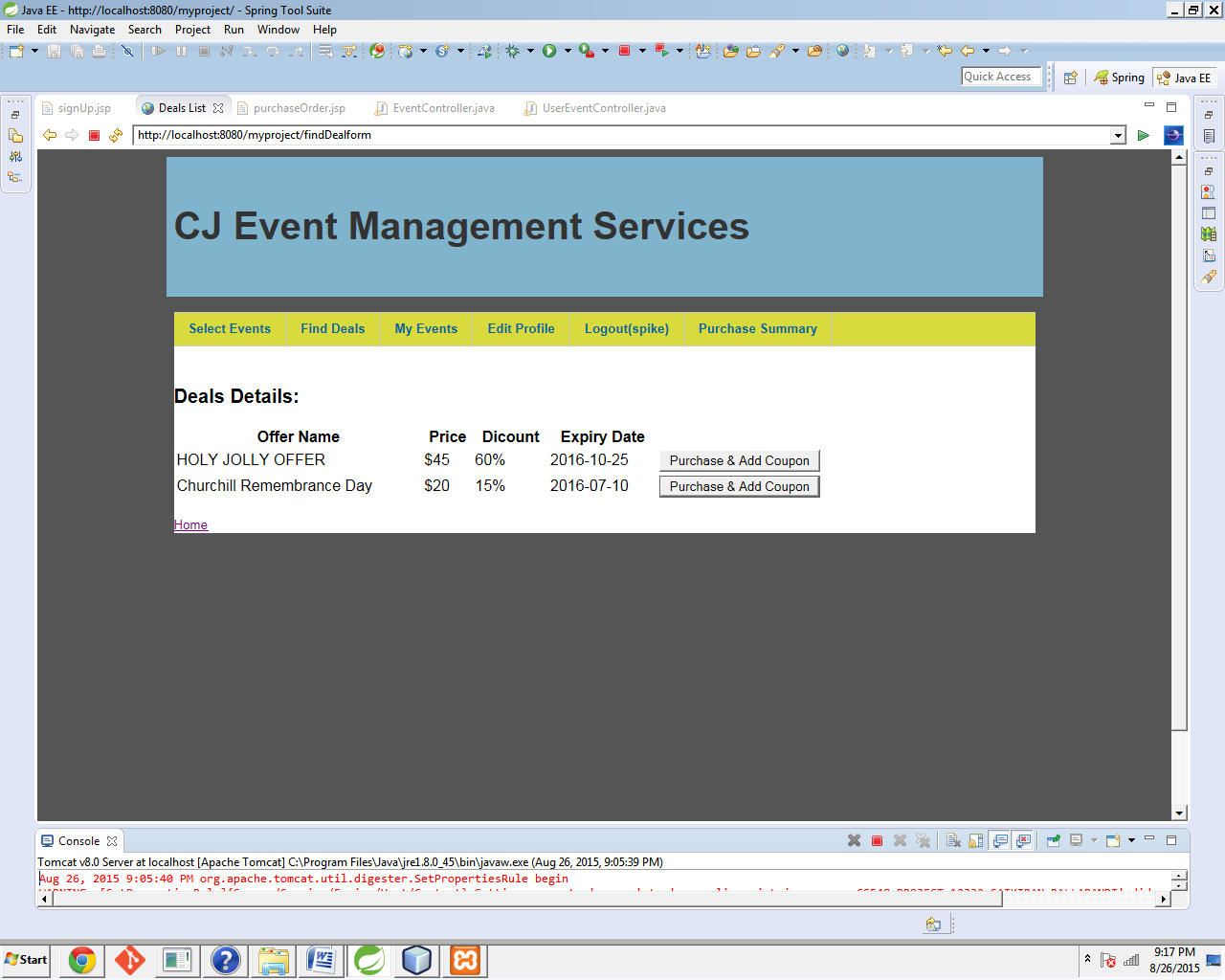




### Deals Page

To view the page click on "Find Deals " tab.

Adding Offers to Event Purchases



After clicking on "Purchase and Add Coupon" user can check his purchase summary to view his savings

### Purchase Summary

SQL statements required for Execution:

**DROP** **SCHEMA** IF **EXISTS** `airdb`;

**CREATE** **SCHEMA** airdb;

use airdb;

**CREATE** **TABLE** airdb.location (

ID **INT**(3) **NOT** **NULL** AUTO\_INCREMENT,

CODE **VARCHAR**(5) **NOT** **NULL**,

NAME **VARCHAR**(64) **NOT** **NULL**,

CITY **VARCHAR**(64) **NOT** **NULL**,

COUNTRY **VARCHAR**(10) **NOT** **NULL**,

**PRIMARY** **KEY** (ID)

);

**CREATE** **table** airdb.deals (

id **int**(3),

location\_code **varchar**(5),

price **int**(11),

discount **int**(11),

event **varchar**(256),

**date** **varchar**(256),

place **varchar**(256),

deal\_id **int**(11) **primary** **key** auto\_increment);

Create **table** airdb.events (

id **int**(3) **primary** **key** auto\_increment,

eventName **varchar**(255),

eventType **varchar**(255),

eventArea **varchar**(255),

eventDate **varchar**(10),

duration **int**(2),

location\_id **int**(11),

eventPrice **float**,

eventCapacity **int**(11));

Create **table** airdb.**user** (

user\_id **int**(11) **primary** **key** auto\_increment,

username **varchar**(32),

password **varchar**(40),

first\_name **varchar**(32),

last\_name **varchar**(32),

gender **varchar**(1),

birthdate **varchar**(10),

picture **varchar**(32),

join\_date datetime **default** current\_timestamp);

**drop** **table** airdb.user\_event

**create** **table** airdb.user\_event (

id **int** **not** **null** auto\_increment **PRIMARY** **KEY**,

user\_id **int**,

event\_id **int**,

deal\_id **int**);

**insert** **into** airdb.location **values** (2,'JFK','John F Kennedy Intl','New York', 'USA');

**insert** **into** airdb.location **values** (3,'LHR','London Heathrow','London', 'UK');

**insert** **into** airdb.events(eventName,eventType,eventArea,eventDate,duration,location\_id,eventPrice,eventCapacity)

Values ("Hadoop Basics","SEMINAR","CLOUD,HADOOP,BIGDATA","23-09-2015",6,3,120,5);

**insert** **into** airdb.events(eventName,eventType,eventArea,eventDate,duration,location\_id,eventPrice,eventCapacity)

Values ("INTERNET OF THINGS","CONFERENCE","IT","25-12-2015",4,2,489,40);

**insert** **into** airdb.events(eventName,eventType,eventArea,eventDate,duration,location\_id,eventPrice,eventCapacity)

Values ("CLOUDERA CD420","TRAINING","CLOUD,HADOOP,IT","15-01-2016",2,3,239,20);

**insert** **into** airdb.deals (ID,location\_code,Price,Discount,Event,Dates,Place)

**values**(2,"LHR",20,15,"Churchill Remembrance Day","2016-07-10","IT");

**insert** **into** airdb.deals (ID,location\_code,Price,Discount,Event,Dates,Place)

**values**(3,"SFO",25,35,"CHRISTMAS DEAL","2016-01-10","CLOUD");

**INSERT** **INTO** airdb.**USER** **VALUES** (1, 'peter', 'abc', 'Peter', 'Paul', 'M', '1980-01-01', 'Peter-Paul.jpg', now());

**INSERT** **INTO** airdb.**USER** **VALUES** (2, 'nancy', 'abc', 'Nancy', 'Clark', 'F', '1980-02-02', 'Nancy-Clark.jpg', now());

**INSERT** **INTO** airdb.**USER** **VALUES** (3, 'jones', 'abc', 'Jones', 'Wang', 'M', '1980-03-03', 'Jones-Wang.jpg', now());

**insert** **into** airdb.USER\_event **values** (1,1,2,3);

**insert** **into** airdb.USER\_event **values** (2,4,2,2);