ÉCOLE D'INGÉNIEURS EN INFORMATIQUE

Identity Access Management

Submitted By: Sonali Narayan Ghadage
Course: Data Science & Analytics

Spring 2018

Table of contents:

Sr. no.	Index	Page No.
1.	Subject Description	3-4
2.	Subject Analysis	4-6
	➢ Major Features:	
	• Create	
	• Insert	
	• Update	
	• Search	
	• Delete	
	➢ GUI (Graphical User Interface):	
	• Application feasibility	
	• Data description	
	• Expected results	
	• Scope of the application	
3.	Conception	6-9
	• Chosen algorithms	
	• Data structures	
	• Global application flow	
	• Global schema or database	
4.	Configuration Instructions	9-14
5.	Commented Screenshots	14-21
6.	Bibliography	21

Subject Description:

The Identity Management System is GUI based application which allows end user to manage and manipulate user identities. User can create, update, delete and search identities. This application is coded on POM (Project Object Model) architecture. It is an XML representation of a Maven project held in a file named pom.xml. Application runs on Tomcat server and uses MySQL database to store data.

Subject Analysis:

- Major Features:

  Create:

  This allows user to create identity using unique user_id, name and email_id.

  Insert:

  This allows user to insert identities randomly using any unique user_id which is inserted into the table in an ascending order.

  Update:

  This allows user to edit or update name and email_id of each existing unique identity.

  Delete:

  This allows user to delete existing identity.

  Search:

  This allows user to search existing identity using unique user_id.

  Graphical User Interface:

  GUI provides user interface to the user to interact with the system user friendly and efficiently.

- Application Feasibility:

  It’s a web application which has been made up of Java servlets; this gives web interface to the user. Due to web based application, cost of installation is low. This also helps to maintain and version the software at centralized point. Web applications are easier to extend than standalone PC application. This application uses structured database to store data, which is feasible to manage and restore the data. A project contains configuration files, the defect tracking system, the organization, the URL of where the project lives, the project's dependencies, and all of the other little pieces that come into play to give code life.

  It is a one-stop-shop for all things concerning the project and that is nothing but project object model (POM). The POM contains all necessary information about a project, as well as configurations of plug-ins to be used during the build process. It is, effectively, the declarative manifestation of the "who", "what", and "where", while the build lifecycle is the "when" and "how".

- Data Description:

First of all, while providing authentication it takes username and password from the table located in identities database where user’s username and password is stored. Basically, this application gives variety of options to manipulate data such as create, update, delete and search. These actions are performed on three attributes, Name, Email and Unique Id. Unique Id is unique in overall application lifecycle, so no duplication of id throughout the application lifecycle.

While creation of identity, system asks three fields which are Id, Name and Email as when we are giving Id; it becomes unique and can't be duplicate. At the time of submission, system also validates that user_id, System doesn’t allow user to create duplicate identity. There is no limitation on Name field; user can enter any alphanumeric value in this field. This is also applicable while updating the user information. Update operation works on the basis of unique id. System matches unique id to map data with database. However Search operation can search any identity based on its unique identity. To store data, system is using structured database, using MySQL.

- Expected Results:

As developed web application is to manage user identities in an organizational manner. So the expected output would be creation, deletion, updating, and searching of those unique identities stored in the database. Moreover GUI helps to manage those identities efficiently.

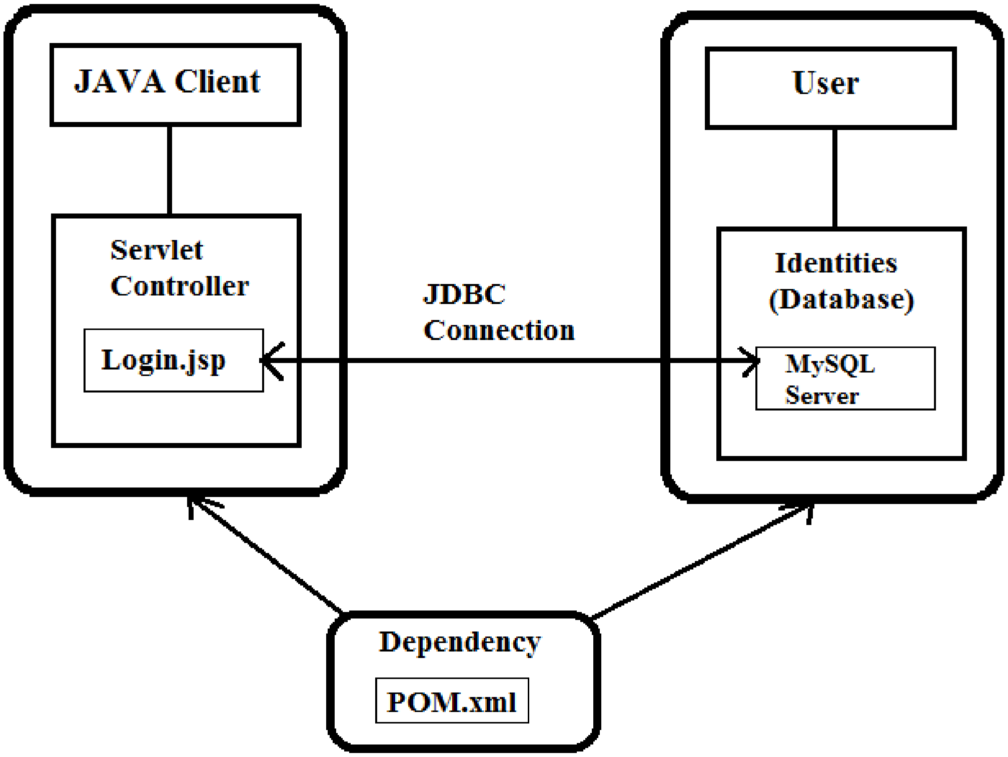
- Scope of the Application:

This application fulfills basic operation of managing data. Right now scope of the application is limited to create, update, delete and search operation. In future, system will filter user's data. Current system has only one credential access, which can be progresses in future development to many accesses. System doesn’t provide any admin and normal user for access control. In future development, users can emerge into two types such as admin and normal user.

Admin user would have full control on the system, whereas normal user would have limited control on data and system. Future system will also have feature to export data to Spreadsheet, and imported into system from spreadsheets. In future development, identities will have more attributes, which can be address, phone number, zip code, access control etc.

Conception:

The web application uses maven architecture which is based on POM. To manage and better organization of code and result business logic, UI and database have been kept separately this structure will allows easy modifications with respect to future scope.



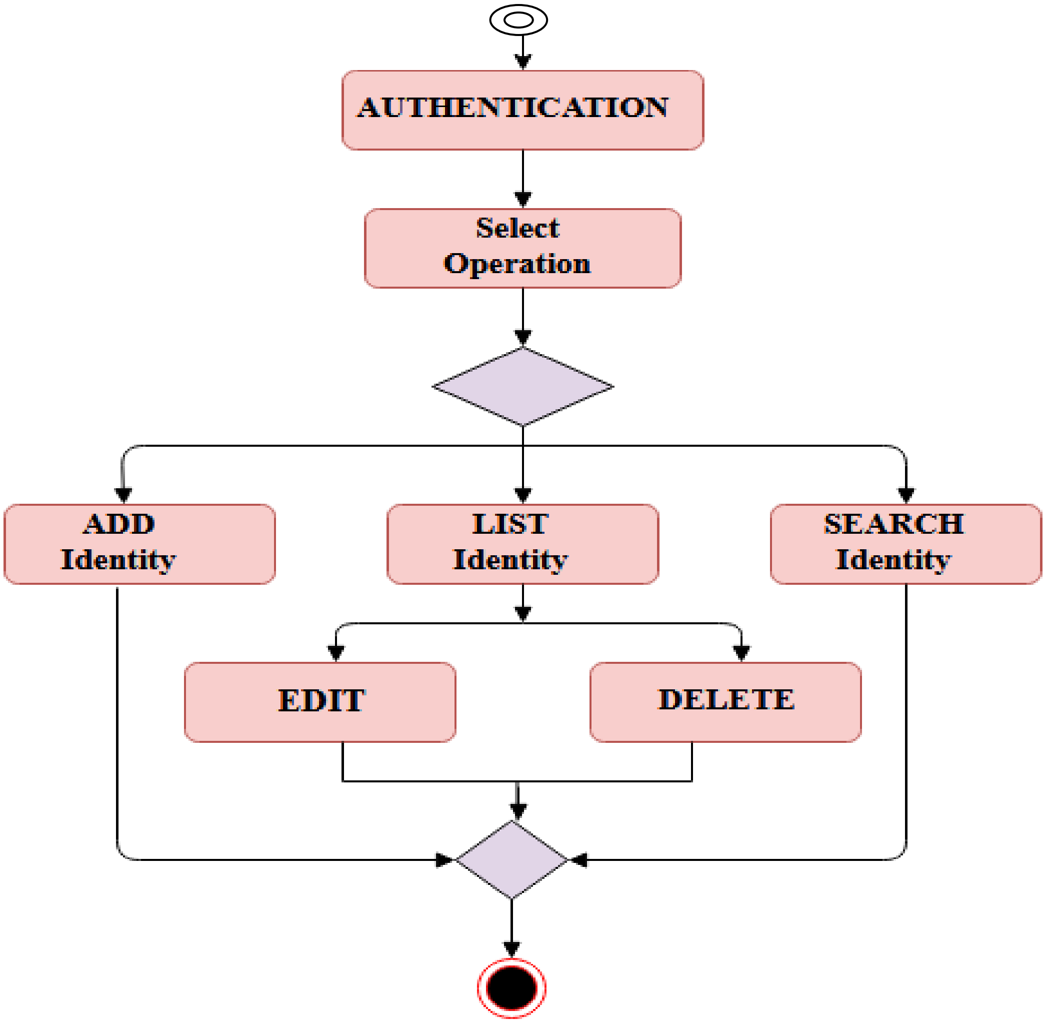
- Chosen Algorithm:

This web application is using no particular algorithm as such, but to store data MySql 5.1.44 is used. Based on the primary key user_id, update, delete and search operations has been performed. So, this mapping is handled by unique id of identity.

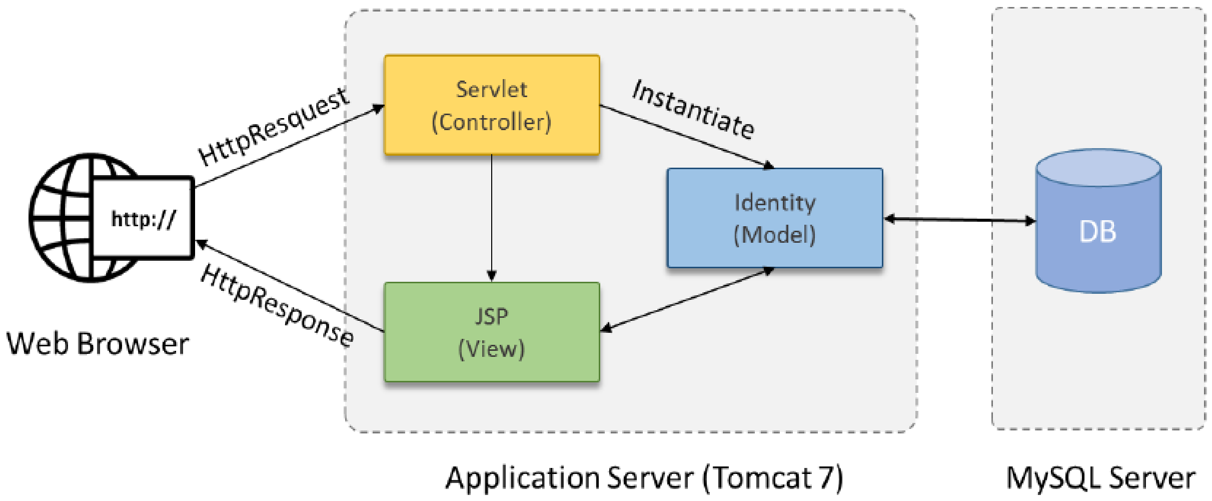
- Data Structures:

This application uses MySQL database to store data. As for development setup, system is using MySQL workbench and XAMPP Server to run MySQL service. SQL is providing structured way of data storage, therefore manipulation of data becomes easier for user. List and array data structures are also used to do operations on that data in java classes.

- Global Application Flow:



- Global Schema:



Configuration Instructions:

This application needs several configurations. Following are the requirements of the project.

a. XAMPPServer (for MySQL)
b. MySQL Workbench
c. Maven configuration
d. Apache Tomcat Server 8.0
e. Setup environment variable

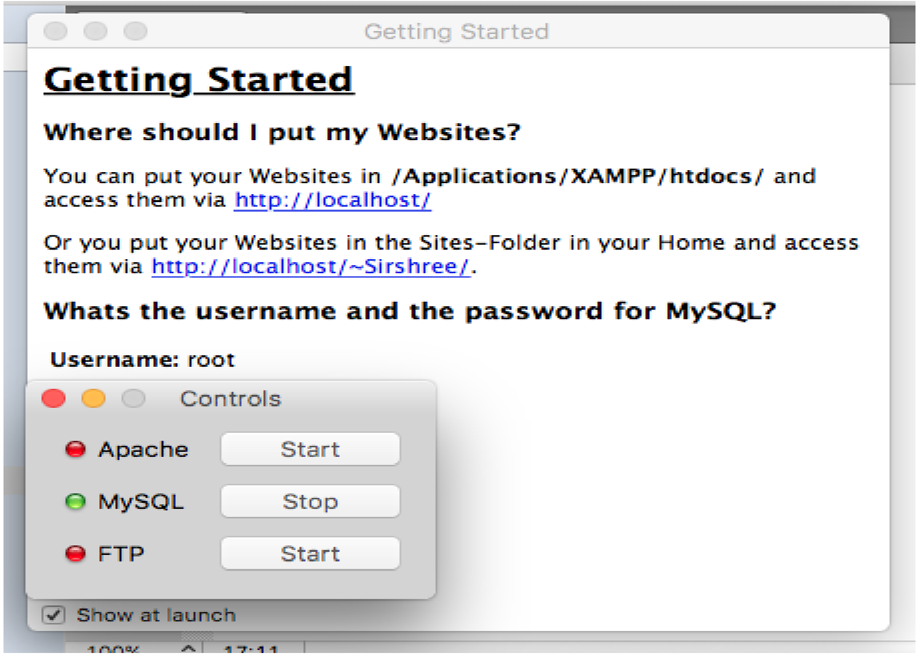
- XAMPPServer (for MySQL):

XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes.

XAMPP is also cross-platform, which means it works equally well on Linux, Mac and Windows. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server extremely easy as well. In this web application MySQL is using XAMPP server for storing database.

This is the website (**https://www.apachefriends.org/download.html)** to download XAMPP server according to your computer configurations.

It will look like as following after installation.



Getting Started

Getting Started

Where should I put my Websites?

You can put your Websites in **/Applications/XAMPP/htdocs/** and access them via http://localhost/

Or you put your Websites in the Sites-Folder in your Home and access them via http://localhost/~Sirshree/.

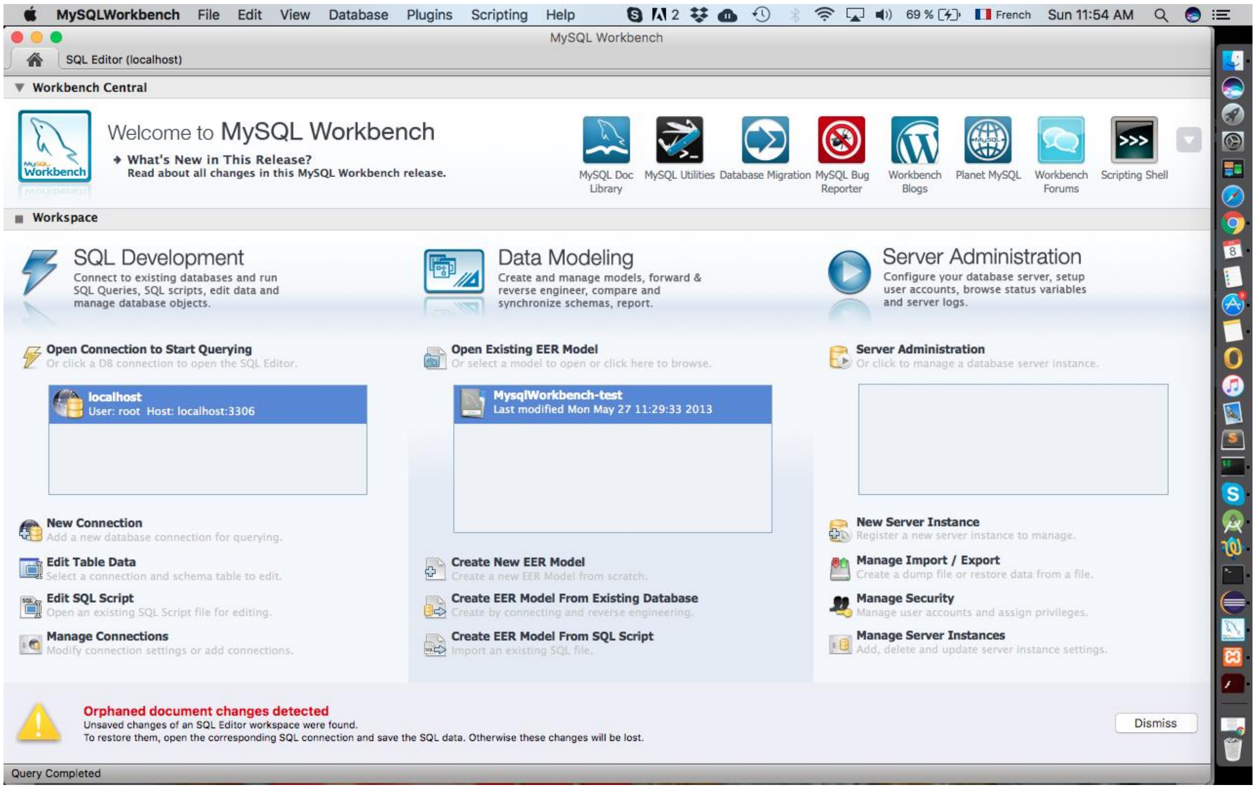
Whats the username and the password for MySQL?

Username: root

- MySQL Workbench:

XAMPP Server is controlled by MySQL Workbench. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more.

MySQL Workbench enables a DBA, developer, or data architect to visually design, model, generate, and manage databases. You can download it from this website (https://www.mysql.com/products/workbench/)



Welcome to MySQL Workbench

What's New in This Release?
Read about all changes in this MySQL Workbench release.

SQL Development
Connect to existing databases and run SQL Queries, SQL scripts, edit data and manage database objects.

Open Connection to Start Querying
Or click a DB connection to open the SQL Editor.

localhost
User: root Host: localhost:3306

New Connection
Add a new database connection for querying.

Edit Table Data
Select a connection and schema table to edit.

Edit SQL Script
Open an existing SQL Script file for editing.

Manage Connections
Modify connection settings or add connections.

Data Modeling
Create and manage models, forward & reverse engineer, compare and synchronize schemas, report.

Open Existing EER Model
Or select a model to open or click here to browse.

MysqlWorkbench-test
Last modified Mon May 27 11:29:33 2013

Create New EER Model
Create a new EER Model from scratch.

Create EER Model From Existing Database
Create by connecting and reverse engineering.

Create EER Model From SQL Script
Import an existing SQL file.

Server Administration
Configure your database server, setup user accounts, browse status variables and server logs.

Server Administration
Or click to manage a database server instance.

New Server Instance
Register a new server instance to manage.

Manage Import / Export
Create a dump file or restore data from a file.

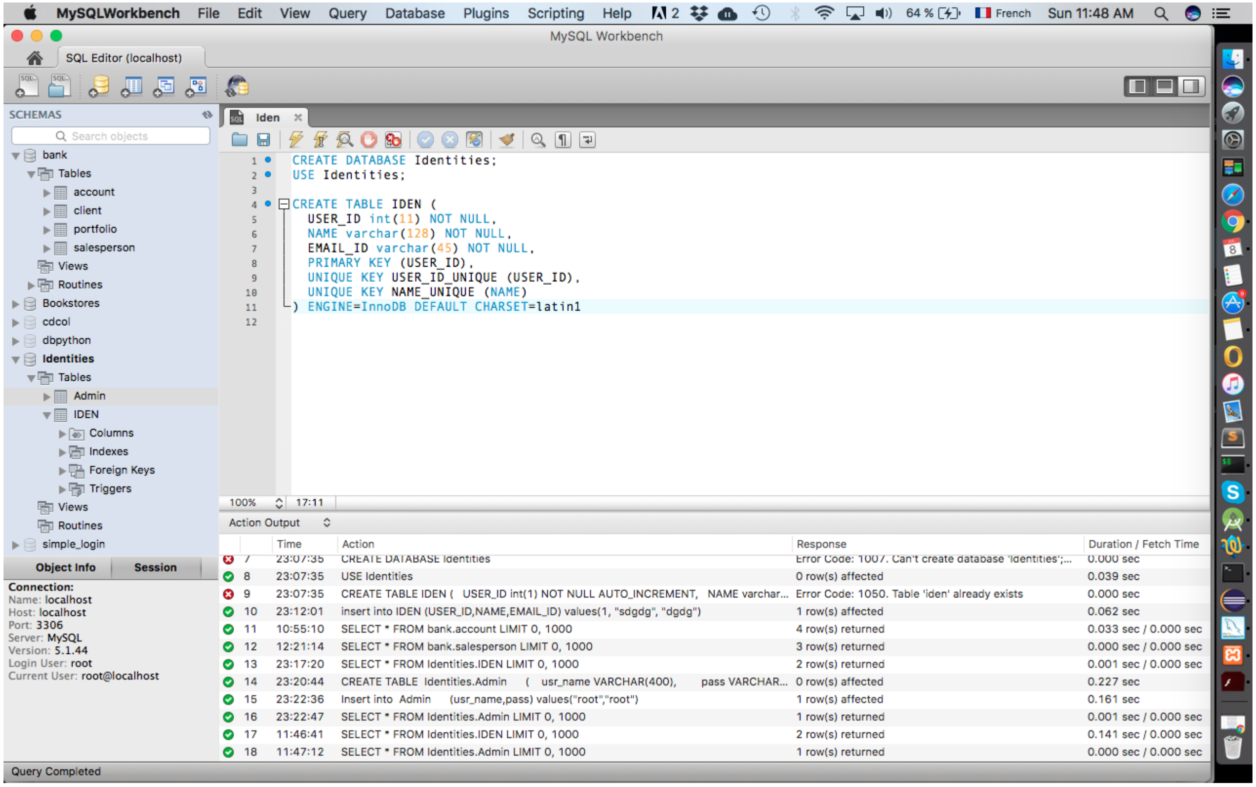
Manage Security
Manage user accounts and assign privileges.

Manage Server Instances
Add, delete and update server instance settings.

Orphaned document changes detected
Unsaved changes of an SQL Editor workspace were found.
To restore them, open the corresponding SQL connection and save the SQL data. Otherwise these changes will be lost.

Query Completed

Here XAMPP server is connected to the “localhost” having root “user”. After clicking on that following page will open where we can create a database.



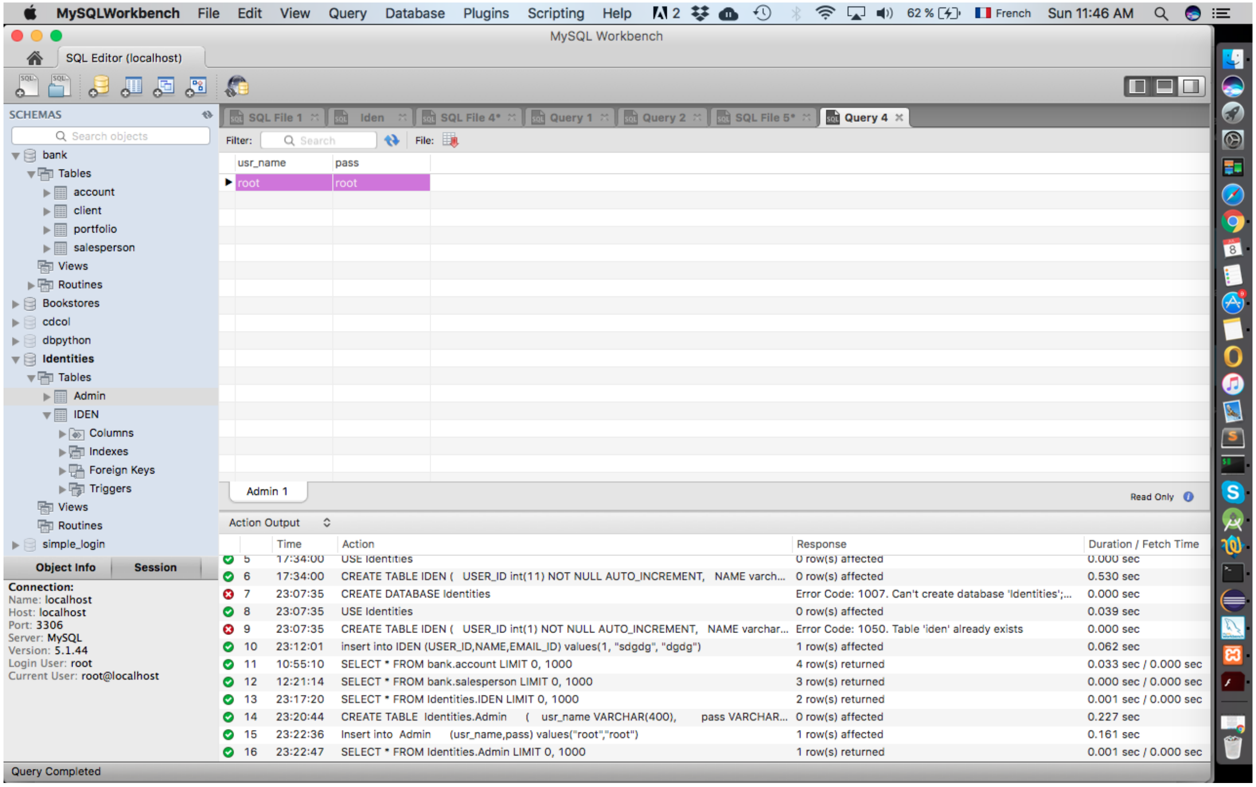
```
CREATE DATABASE Identities;
USE Identities;

CREATE TABLE IDEN (
  USER_ID int(11) NOT NULL,
  NAME varchar(128) NOT NULL,
  EMAIL_ID varchar(45) NOT NULL,
  PRIMARY KEY (USER_ID),
  UNIQUE KEY USER_ID_UNIQUE (USER_ID),
  UNIQUE KEY NAME_UNIQUE (NAME)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
```

Connection:
Name: localhost
Host: localhost
Port: 3306
Server: MySQL
Version: 5.1.44
Login User: root
Current User: root@localhost

	Time	Action	Response	Duration / Fetch Time
7	23:07:35	CREATE DATABASE Identities	Error Code: 1007. Can't create database 'Identities';...	0.000 sec
8	23:07:35	USE Identities	0 row(s) affected	0.039 sec
9	23:07:35	CREATE TABLE IDEN (USER_ID int(1) NOT NULL AUTO_INCREMENT, NAME varchar...	Error Code: 1050. Table 'iden' already exists	0.000 sec
10	23:12:01	insert into IDEN (USER_ID,NAME,EMAIL_ID) values(1, "sdgdg", "dgdg")	1 row(s) affected	0.062 sec
11	10:55:10	SELECT * FROM bank.account LIMIT 0, 1000	4 row(s) returned	0.033 sec / 0.000 sec
12	12:21:14	SELECT * FROM bank.salesperson LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
13	23:17:20	SELECT * FROM Identities.IDEN LIMIT 0, 1000	2 row(s) returned	0.001 sec / 0.000 sec
14	23:20:44	CREATE TABLE Identities.Admin (usr_name VARCHAR(400), pass VARCHAR...	0 row(s) affected	0.227 sec
15	23:22:36	Insert into Admin (usr_name,pass) values("root","root")	1 row(s) affected	0.161 sec
16	23:22:47	SELECT * FROM Identities.Admin LIMIT 0, 1000	1 row(s) returned	0.001 sec / 0.000 sec
17	11:46:41	SELECT * FROM Identities.IDEN LIMIT 0, 1000	2 row(s) returned	0.141 sec / 0.000 sec
18	11:47:12	SELECT * FROM Identities.Admin LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

On the same database Admin table is also added to store username and password “root”.



usr_name	pass
root	root

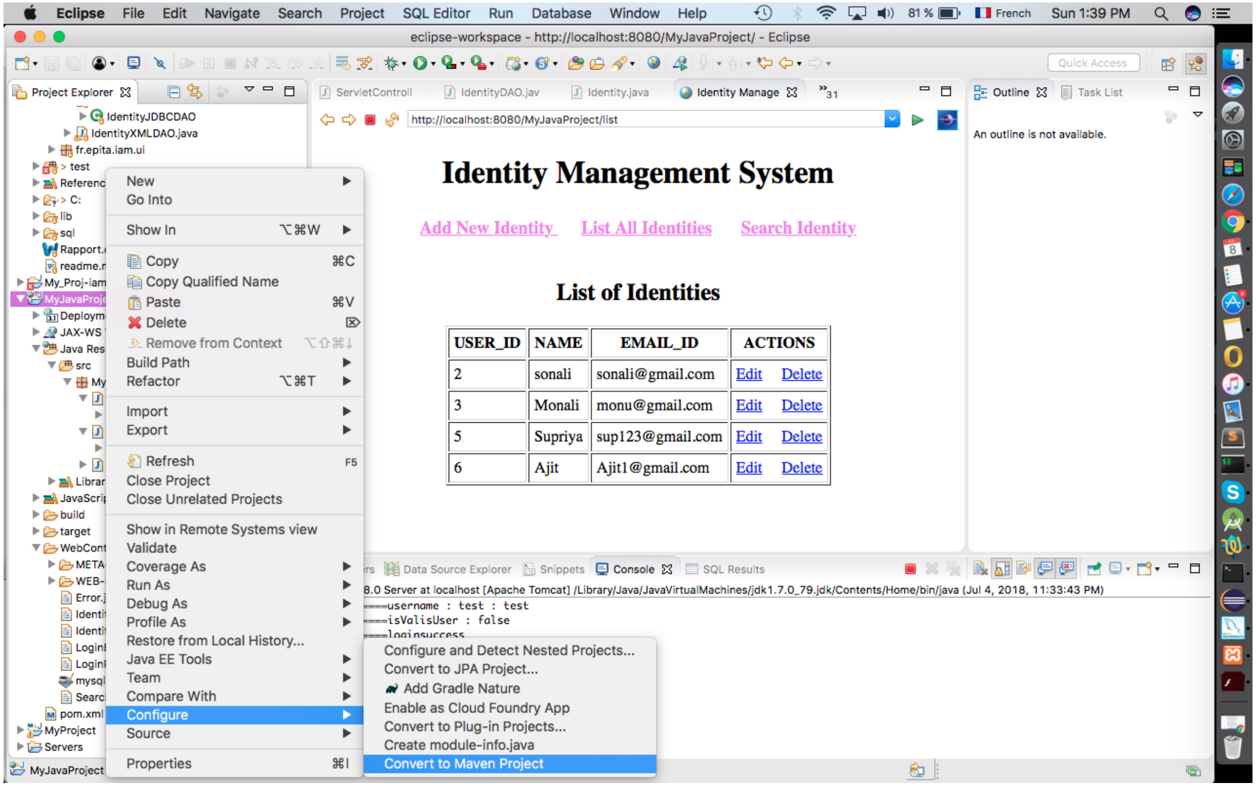
Connection:
Name: localhost
Host: localhost
Port: 3306
Server: MySQL
Version: 5.1.44
Login User: root
Current User: root@localhost

	Time	Action	Response	Duration / Fetch Time
5	17:34:00	USE Identities	0 row(s) affected	0.000 sec
6	17:34:00	CREATE TABLE IDEN (USER_ID int(11) NOT NULL AUTO_INCREMENT, NAME varch...	0 row(s) affected	0.530 sec
7	23:07:35	CREATE DATABASE Identities	Error Code: 1007. Can't create database 'Identities';...	0.000 sec
8	23:07:35	USE Identities	0 row(s) affected	0.039 sec
9	23:07:35	CREATE TABLE IDEN (USER_ID int(1) NOT NULL AUTO_INCREMENT, NAME varchar...	Error Code: 1050. Table 'iden' already exists	0.000 sec
10	23:12:01	insert into IDEN (USER_ID,NAME,EMAIL_ID) values(1, "sdgdg", "dgdg")	1 row(s) affected	0.062 sec
11	10:55:10	SELECT * FROM bank.account LIMIT 0, 1000	4 row(s) returned	0.033 sec / 0.000 sec
12	12:21:14	SELECT * FROM bank.salesperson LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
13	23:17:20	SELECT * FROM Identities.IDEN LIMIT 0, 1000	2 row(s) returned	0.001 sec / 0.000 sec
14	23:20:44	CREATE TABLE Identities.Admin (usr_name VARCHAR(400), pass VARCHAR...	0 row(s) affected	0.227 sec
15	23:22:36	Insert into Admin (usr_name,pass) values("root","root")	1 row(s) affected	0.161 sec
16	23:22:47	SELECT * FROM Identities.Admin LIMIT 0, 1000	1 row(s) returned	0.001 sec / 0.000 sec

- Maven configuration:

  Then convert this project to a Maven project by right click on the project, select Configure Convert to Maven Project, as shown below.

  The needed information create Maven POM file, such as group ID, artifact ID, etc. Then it will add the dependencies. You will find it in WEB CONTENT->pom.xml file.



Identity Management System

Add New Identity List All Identities Search Identity

List of Identities

USER_ID	NAME	EMAIL_ID	ACTIONS	
2	sonali	sonali@gmail.com	Edit	Delete
3	Monali	monu@gmail.com	Edit	Delete
5	Supriya	sup123@gmail.com	Edit	Delete
6	Ajit	Ajit1@gmail.com	Edit	Delete

```
8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2018, 11:33:43 PM)
====username : test : test
====isValisUser : false
====loginsuccess
```

- Apache Tomcat Server 8.0:

  Apache tomcat server is used to host java application to make servlet and JSP runable.

  Go to http://tomcat.apache.org/index.html and download Tomcat version 8.

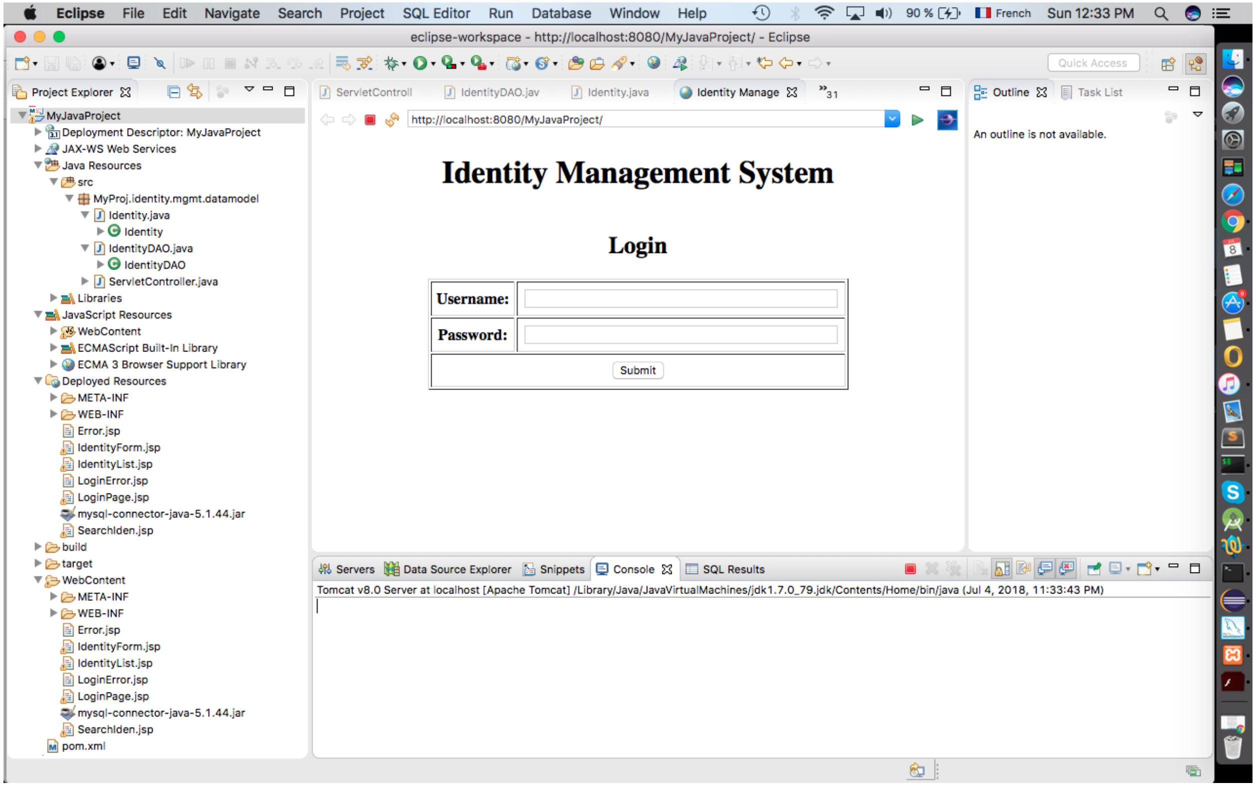
  Install the executable file.

  Open project in eclipse and right click on project. You will see menu, go to run as and select run on server.

  There will be run on server window, Select Tomcat v8.0 Server. If you want to change server name, then you can do it by inputting new name in Server Name textbox. Now, click on Next.

  There you will find new dialog, having two columns called Available and Configured. Select the project from Available and click on Add button. This will add project to Configured column.

Now, click on finish. Project has been published to server and ready to use. Put (http://localhost:8080/MyJavaProject/) login URL into your browser. You will see following screen.



```
MyJavaProject
  Deployment Descriptor: MyJavaProject
  JAX-WS Web Services
  Java Resources
    src
      MyProj.identity.mgmt.datamodel
        Identity.java
          Identity
        IdentityDAO.java
          IdentityDAO
        ServletController.java
    Libraries
  JavaScript Resources
    WebContent
    ECMAScript Built-In Library
    ECMA 3 Browser Support Library
  Deployed Resources
    META-INF
    WEB-INF
    Error.jsp
    IdentityForm.jsp
    IdentityList.jsp
    LoginError.jsp
    LoginPage.jsp
    mysql-connector-java-5.1.44.jar
    SearchIden.jsp
  build
  target
  WebContent
    META-INF
    WEB-INF
    Error.jsp
    IdentityForm.jsp
    IdentityList.jsp
    LoginError.jsp
    LoginPage.jsp
    mysql-connector-java-5.1.44.jar
    SearchIden.jsp
  pom.xml
```

http://localhost:8080/MyJavaProject/

Identity Management System

Login

Username:	
Password:	
Submit	

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2018, 11:33:43 PM)
```

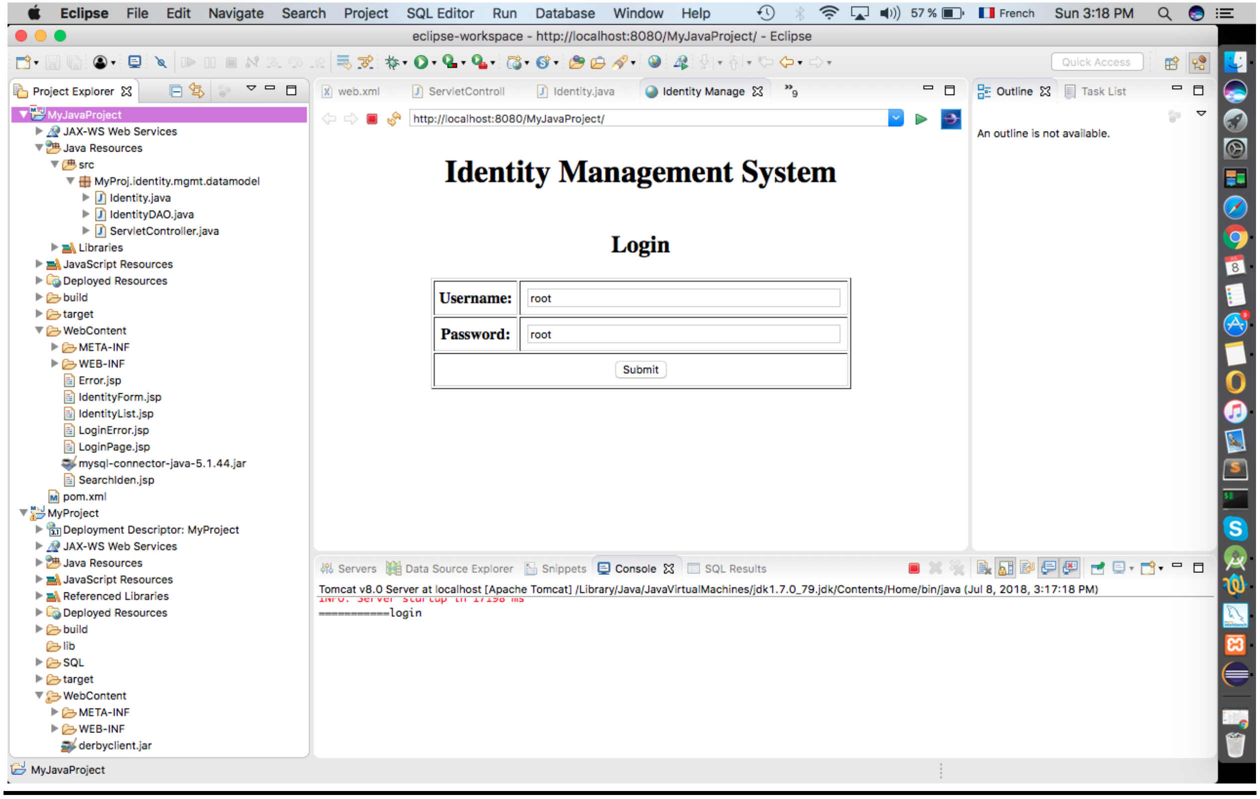
- Setup Environment Variable:

  You will find this file in project at /MyJavaProject/WebContent/WEB-INF/web.xml

  In web.xml you will find the environment variable of the whole project named as “ServletController”.

Commented Screenshots:

Following screenshots will show overall execution of this web project.



http://localhost:8080/MyJavaProject/

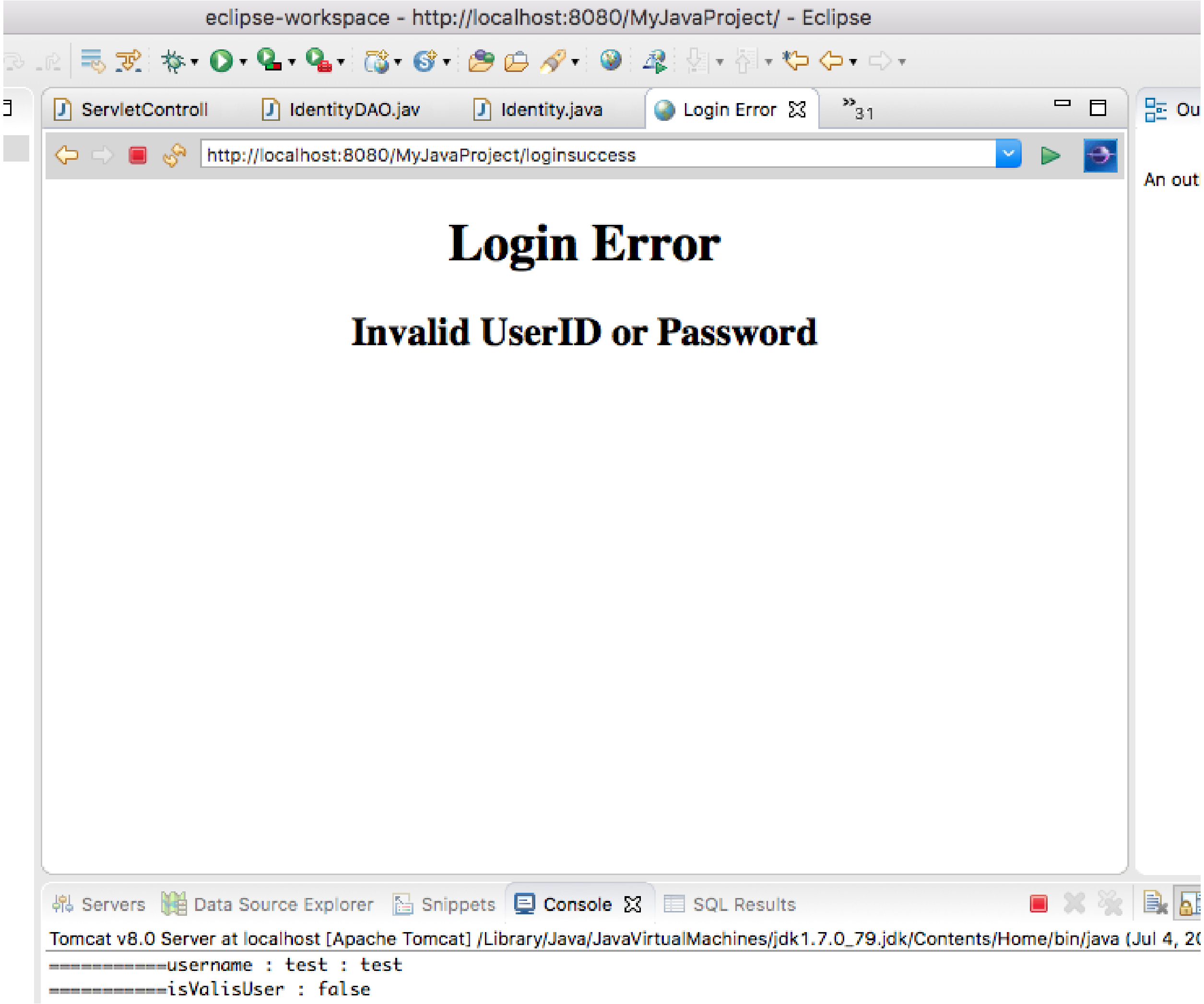
Identity Management System

Login

Username:	root
Password:	root
Submit	

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 8, 2018, 3:17:18 PM)
INFO: Server startup in 17198 ms
===========login
```

This is login page of web application. If the username or password is invalid then it will display following message.

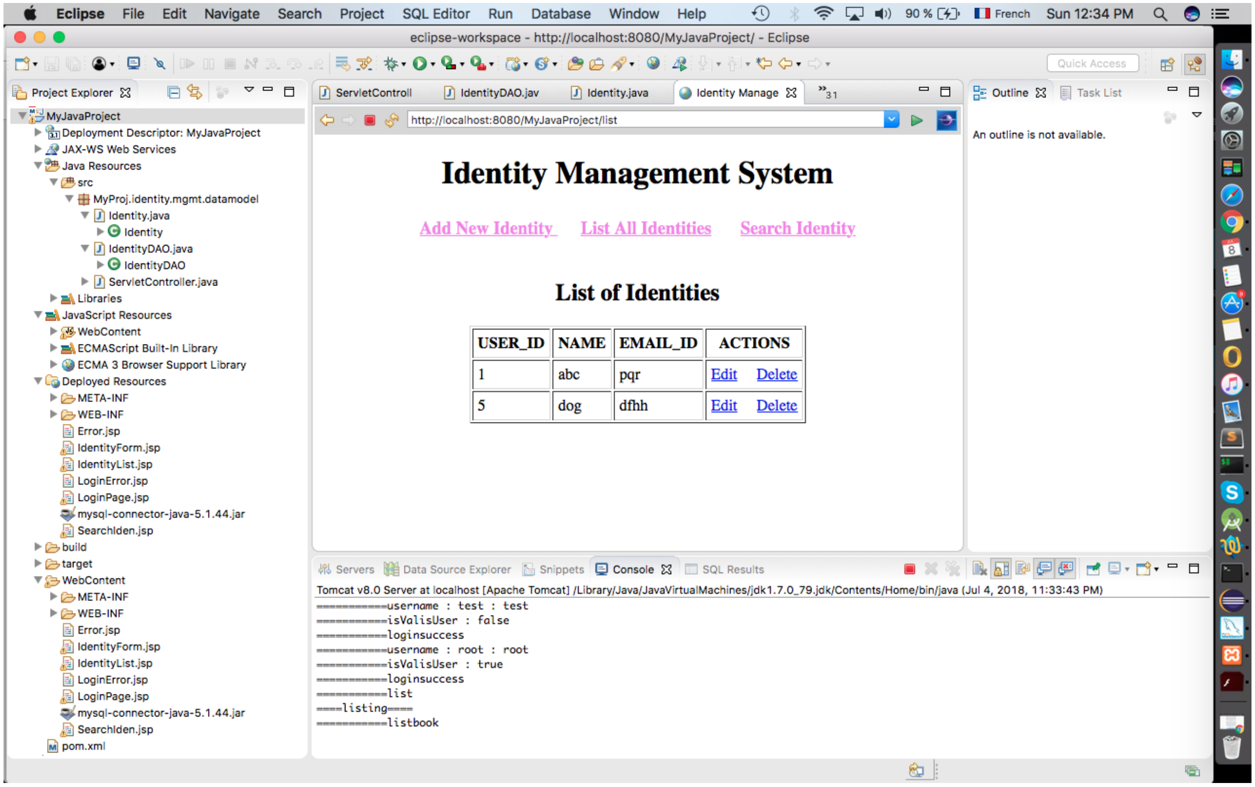


Login Error

Invalid UserID or Password

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 20
===========username : test : test
===========isValisUser : false
```

After successful login you will see the options like “Add new Identity”, “List of Identities”, “Search Identity” and you will see two hyperlinks named as “Edit” and “Delete”.



Identity Management System

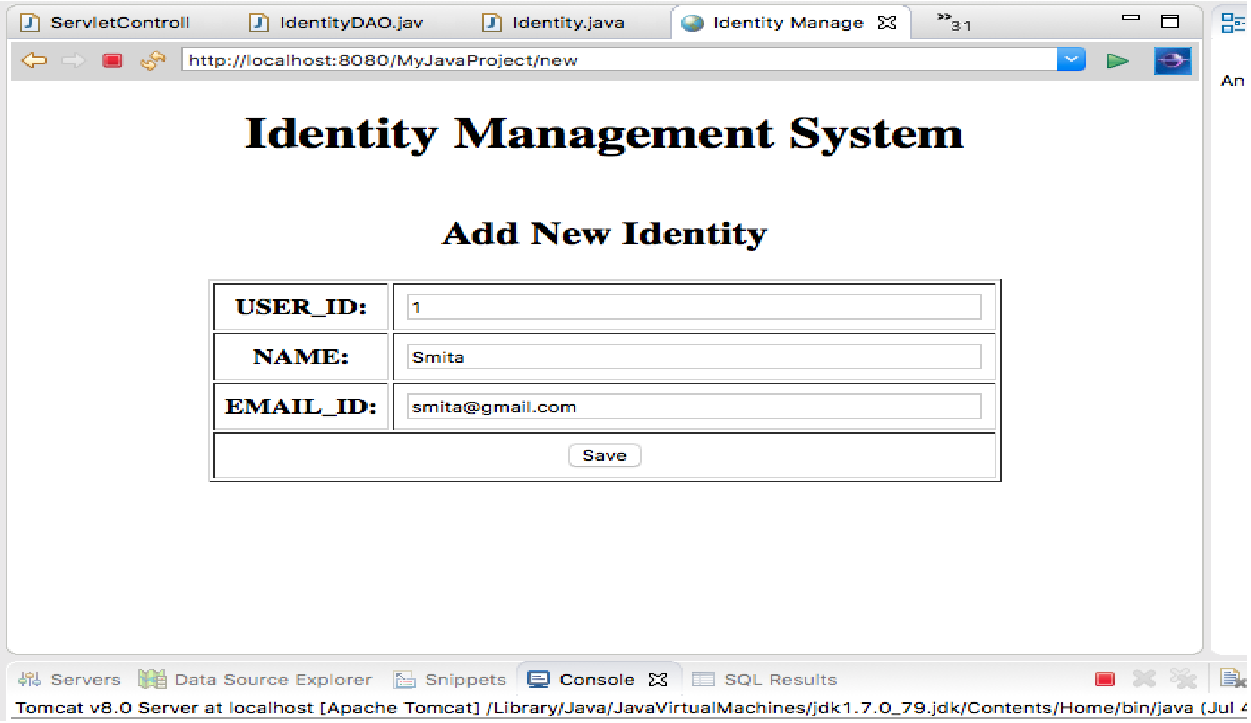
Add New Identity   List All Identities   Search Identity

List of Identities

USER_ID	NAME	EMAIL_ID	ACTIONS
1	abc	pqr	Edit Delete
5	dog	dfhh	Edit Delete

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2018, 11:33:43 PM)
===========username : test : test
===========isValisUser : false
===========loginsuccess
===========username : root : root
===========isValisUser : true
===========loginsuccess
===========list
====listing====
===========listbook
```

Now, after clicking on "Add New Identity" you will see following page to add new identity.

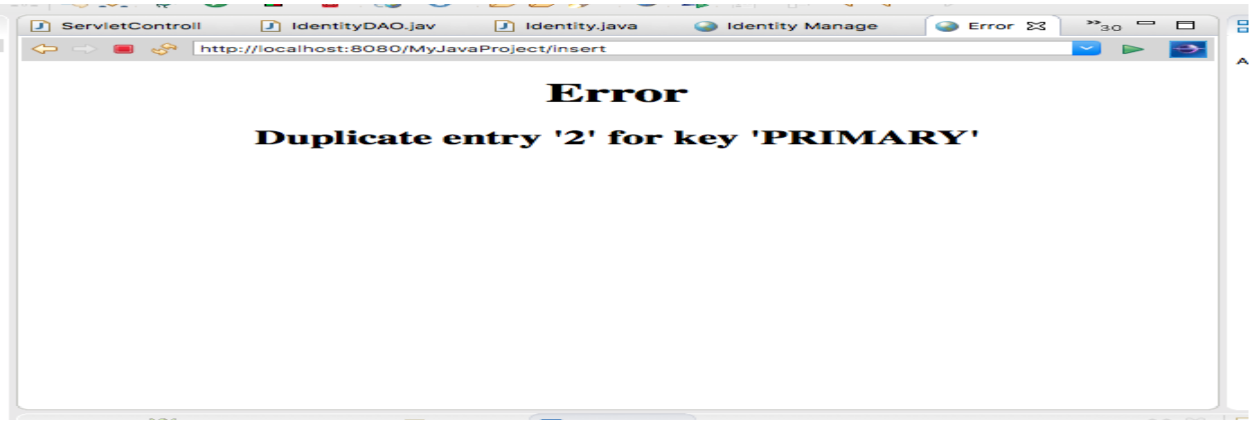

ServletControll
IdentityDAO.jav
Identity.java
Identity Manage
http://localhost:8080/MyJavaProject/new
An
Servers
Data Source Explorer
Snippets
Console
SQL Results
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul

Identity Management System

Add New Identity

USER_ID:	1
NAME:	Smita
EMAIL_ID:	smita@gmail.com
Save	

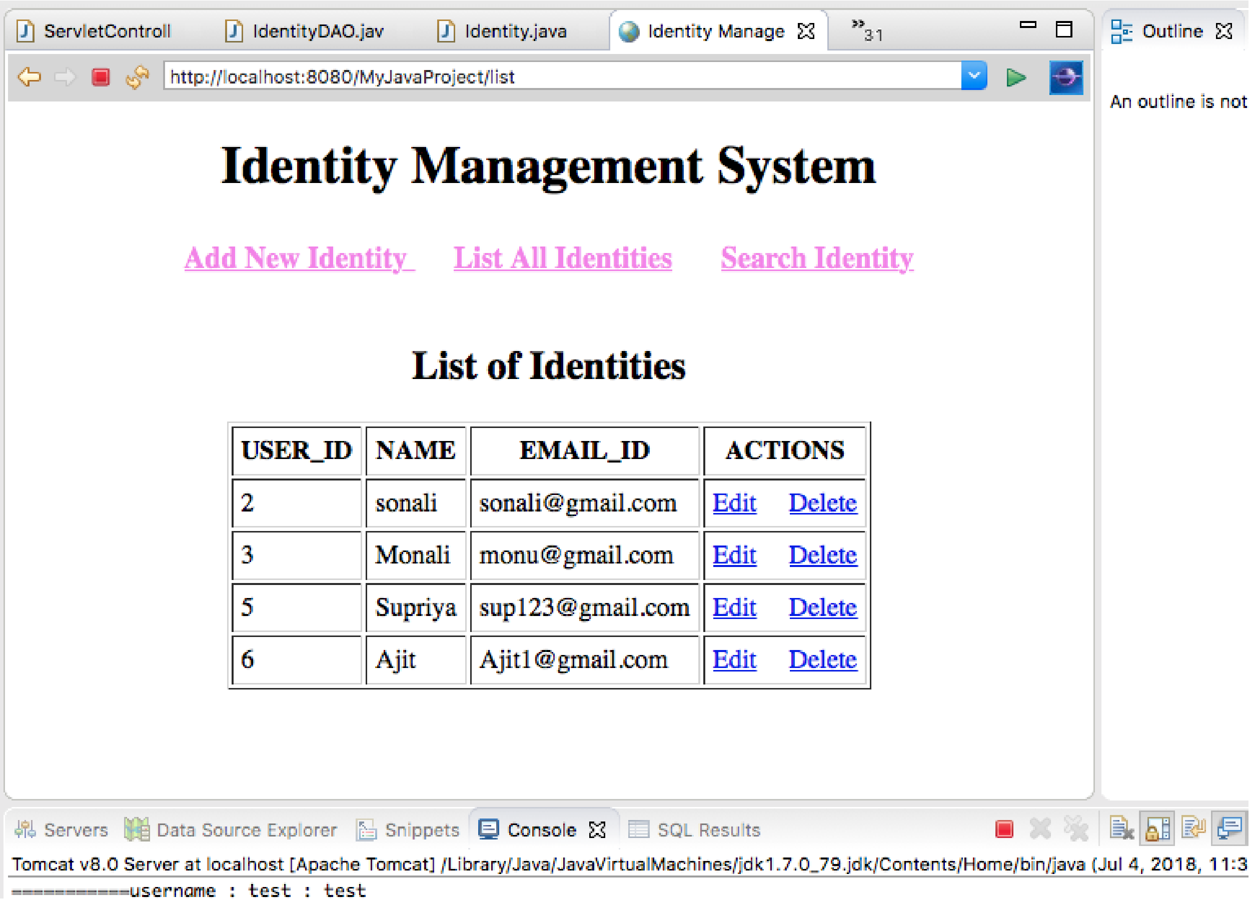
When you are adding the same identity of same user_id then it will display the following error.



Error

Duplicate entry '2' for key 'PRIMARY'

You can as many identities as you want. After adding multiple identities you will see the "List of Identities".



Identity Management System

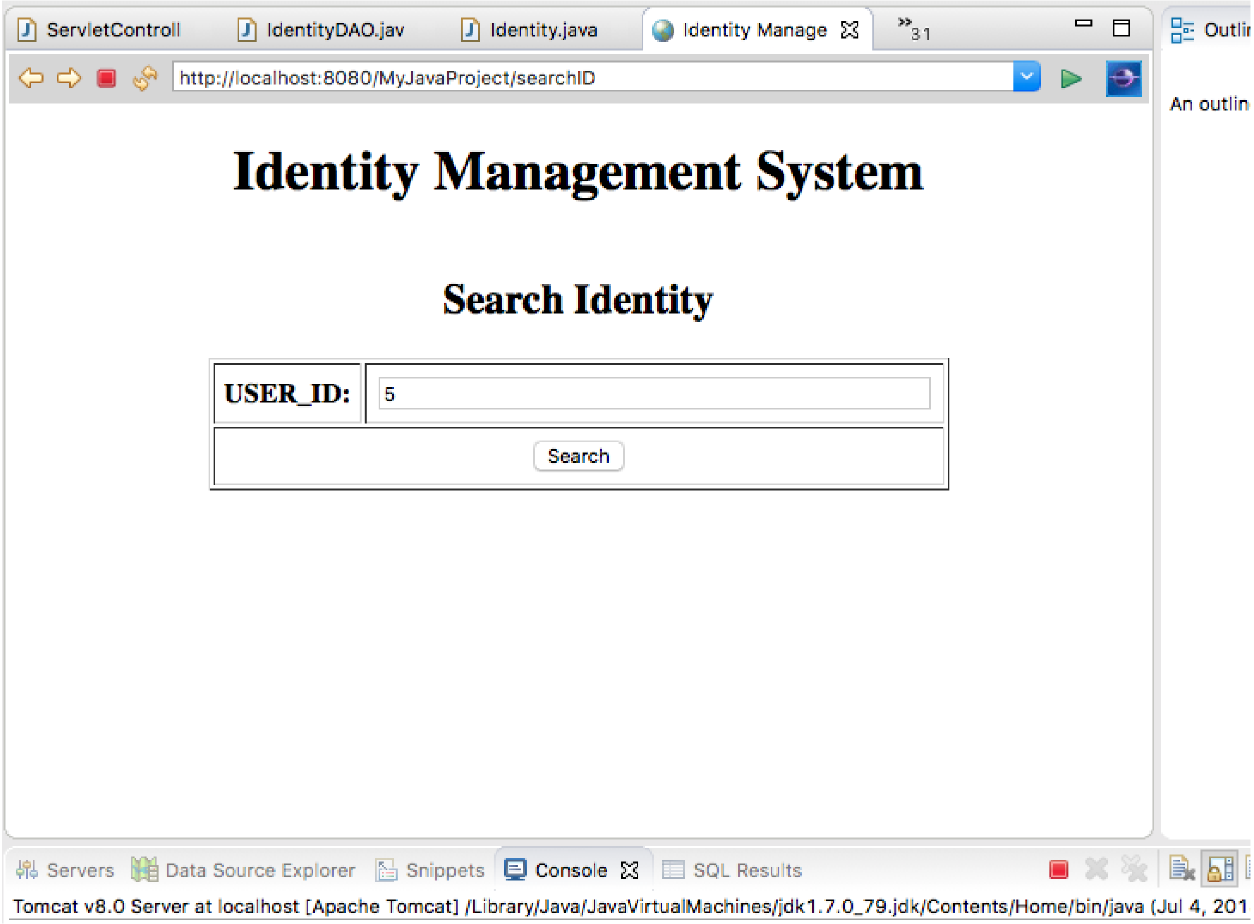
Add New Identity   List All Identities   Search Identity

List of Identities

USER_ID	NAME	EMAIL_ID	ACTIONS
2	sonali	sonali@gmail.com	Edit Delete
3	Monali	monu@gmail.com	Edit Delete
5	Supriya	sup123@gmail.com	Edit Delete
6	Ajit	Ajit1@gmail.com	Edit Delete

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2018, 11:3
===========username : test : test
```

Now, by clicking on “Search Identity”, you can search any identity by giving user_id to the application as shown in the following image.



http://localhost:8080/MyJavaProject/searchID

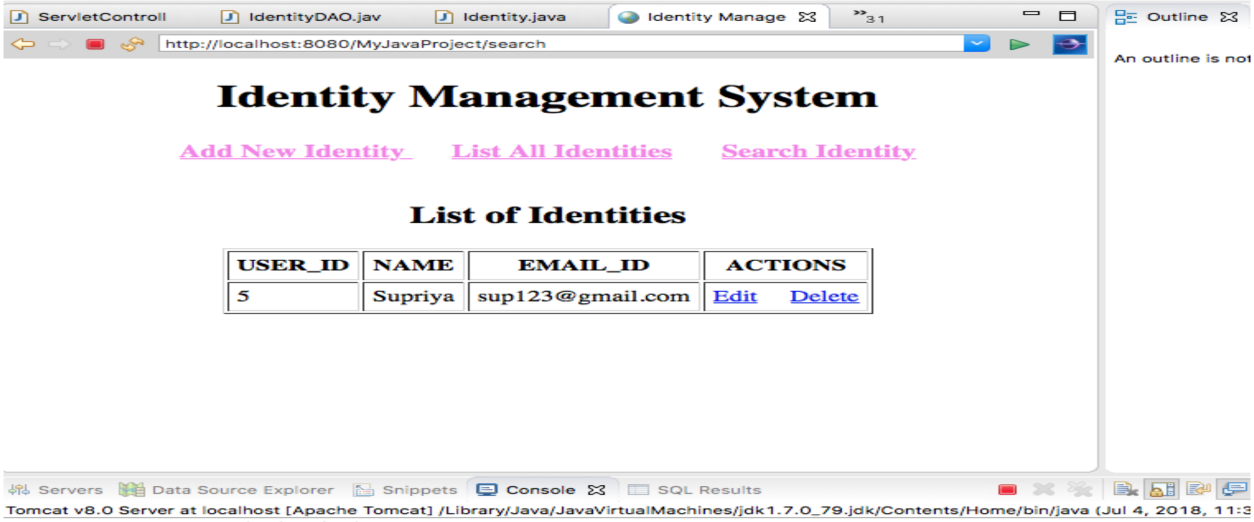
Identity Management System

Search Identity

USER_ID: 5

Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 201

As given in the search identity the 5th identity will display on the screen.



http://localhost:8080/MyJavaProject/search

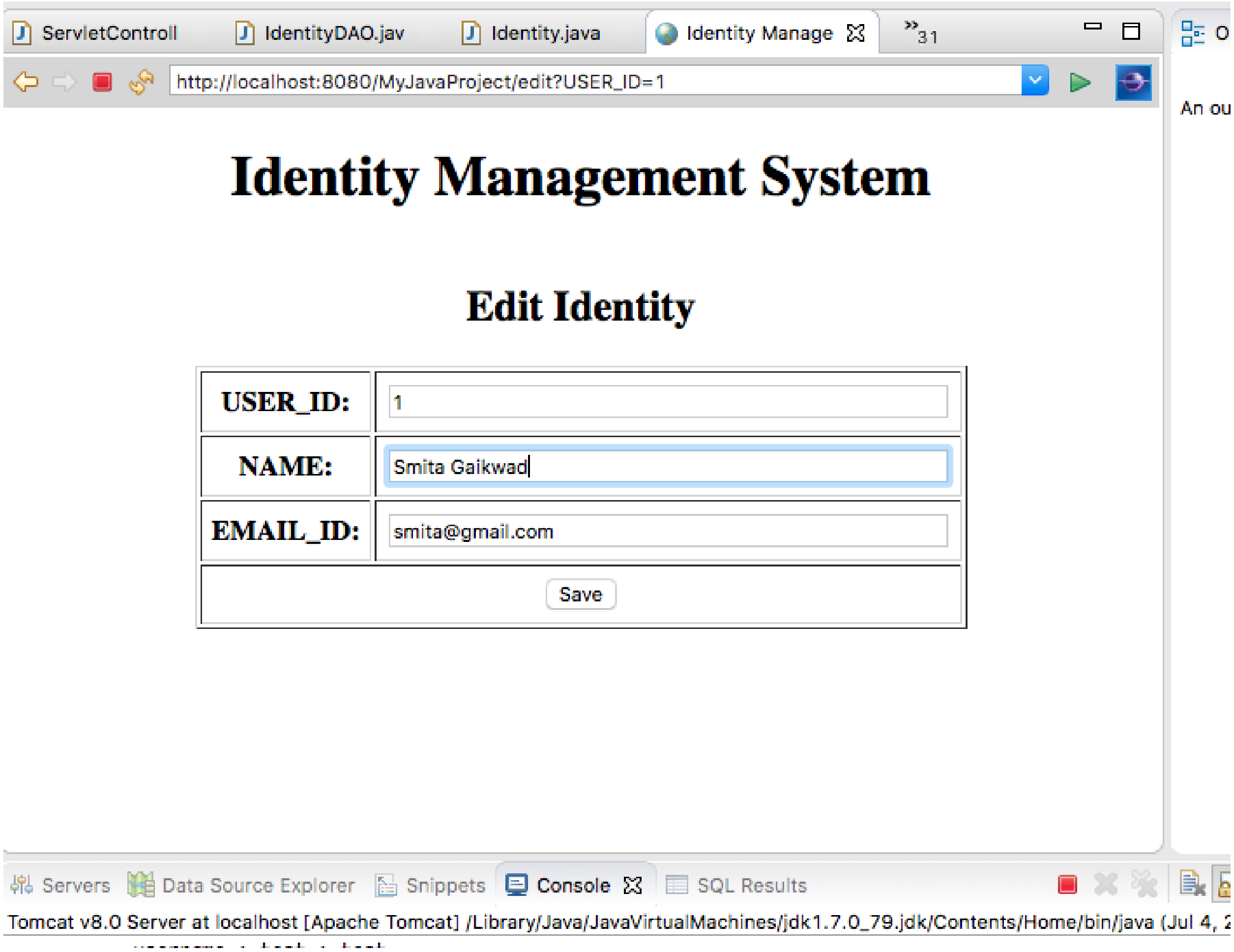
Identity Management System

List of Identities

USER_ID	NAME	EMAIL_ID	ACTIONS
5	Supriya	sup123@gmail.com	Edit Delete

Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2018, 11:3

After search operation by clicking on “Edit” hyperlink, you can edit particular identity. In the following image I am editing the first identity.



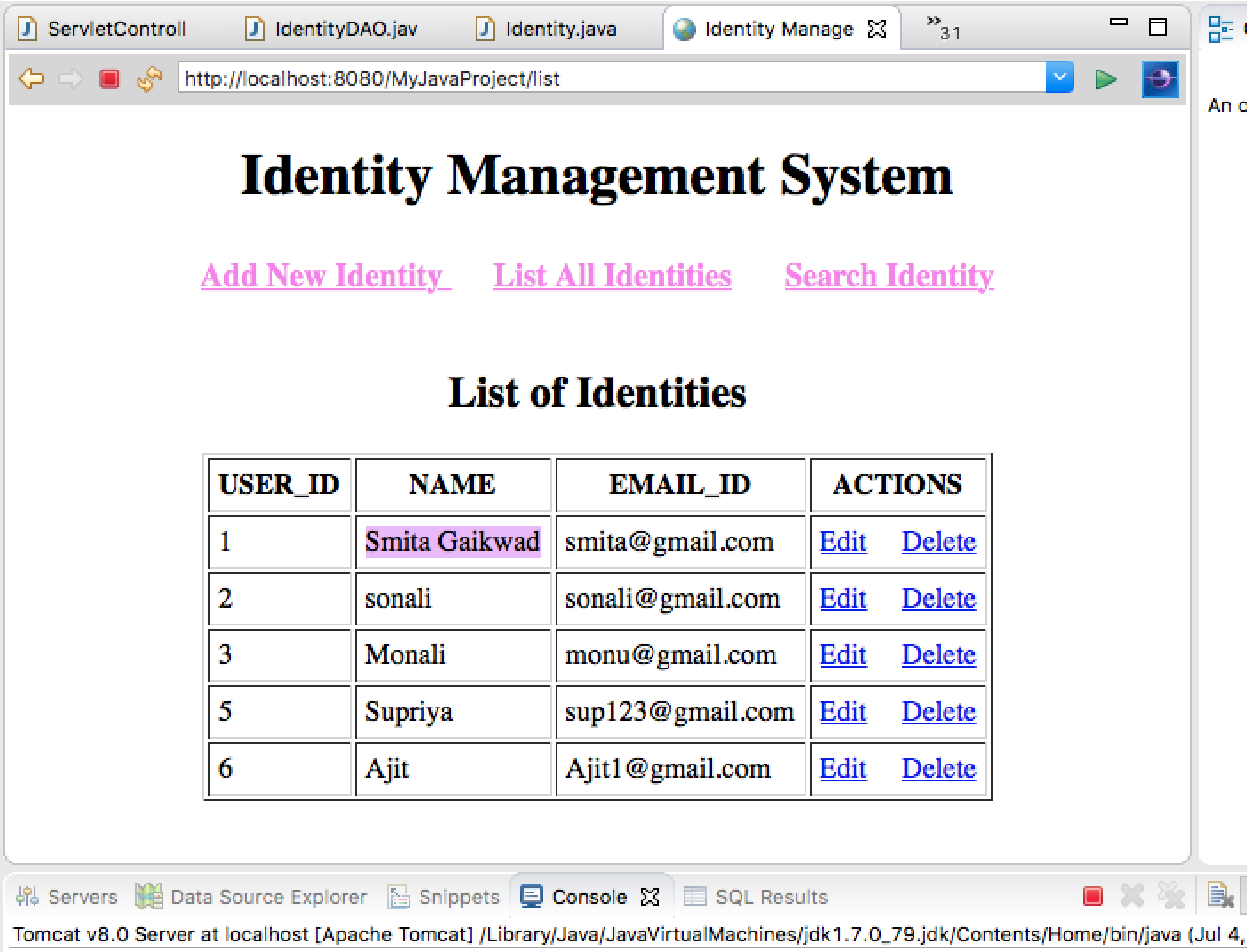
Identity Management System

Edit Identity

USER_ID:	1
NAME:	Smita Gaikwad
EMAIL_ID:	smita@gmail.com

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2
```

Those modifications will reflect in the list.



Identity Management System

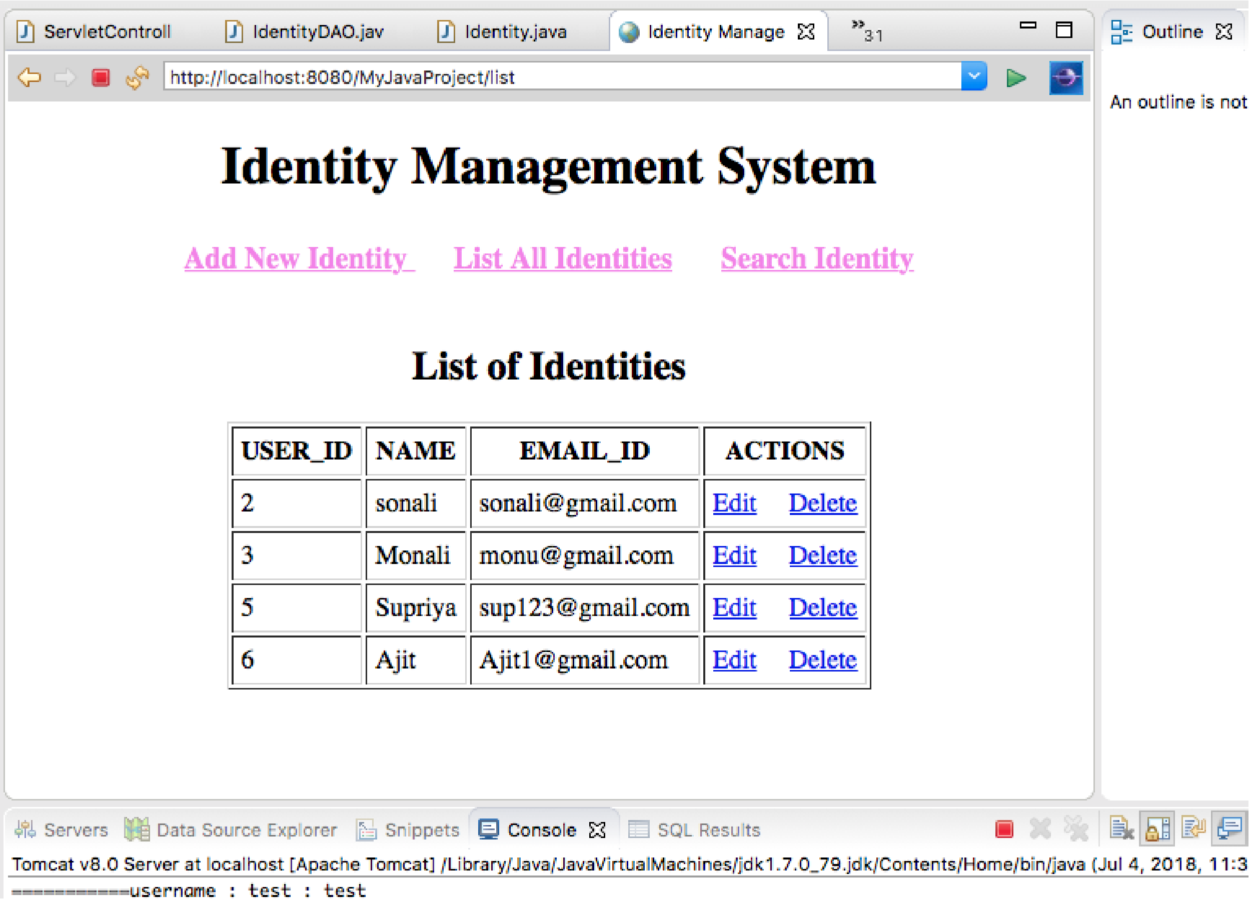
Add New Identity   List All Identities   Search Identity

List of Identities

USER_ID	NAME	EMAIL_ID	ACTIONS
1	Smita Gaikwad	smita@gmail.com	Edit Delete
2	sonali	sonali@gmail.com	Edit Delete
3	Monali	monu@gmail.com	Edit Delete
5	Supriya	sup123@gmail.com	Edit Delete
6	Ajit	Ajit1@gmail.com	Edit Delete

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4,
```

Now, after clicking on “Delete” hyperlink, the identity will get delete. Here in the following image I am deleting first identity, and then the result will be, as shown in the following image.



Identity Management System

Add New Identity   List All Identities   Search Identity

List of Identities

USER_ID	NAME	EMAIL_ID	ACTIONS
2	sonali	sonali@gmail.com	Edit Delete
3	Monali	monu@gmail.com	Edit Delete
5	Supriya	sup123@gmail.com	Edit Delete
6	Ajit	Ajit1@gmail.com	Edit Delete

```
Tomcat v8.0 Server at localhost [Apache Tomcat] /Library/Java/JavaVirtualMachines/jdk1.7.0_79.jdk/Contents/Home/bin/java (Jul 4, 2018, 11:3
===========username : test : test
```

Bibliography:

- www.google.com
- https://www.mysql.com/products/workbench/
- **https://www.apachefriends.org/download.html**
- https://stackoverflow.com/
- http://tomcat.apache.org/