



**URL Shortener with JSF  
(CPSC 476: Graduate Programming Project)**

**December 7th, 2016**

**Submitted by,  
Neha Gawade**

**Prepared for**

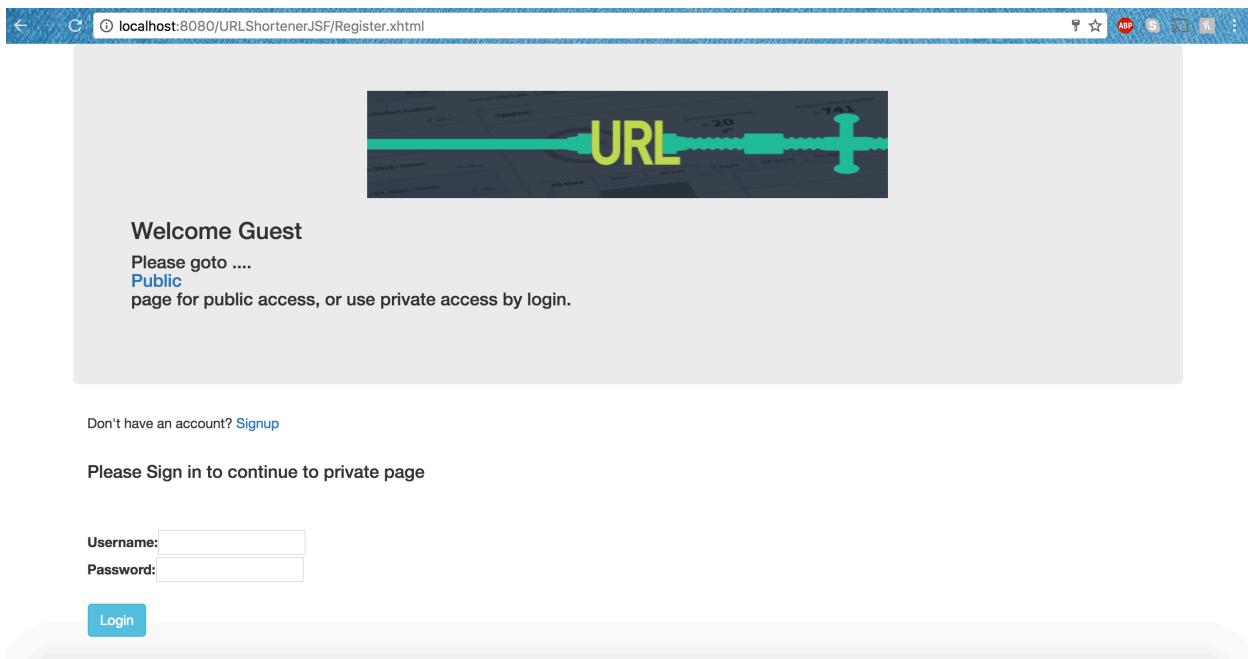
**Prof. Kenytt Avery**

**California State University Fullerton**

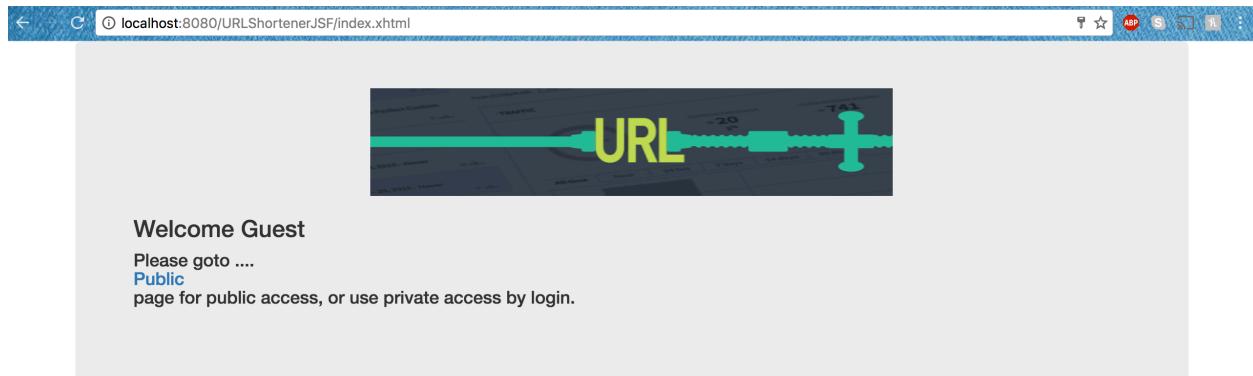
## Purpose

The purpose of this project is to build a Client server application that uses Java Server Faces () and a structured database like HSQLDB. The project's main functions apart from login and maintaining sessions for users are converting a long URL entered by user to a shortened one. This is done only when user is a registered user. The other task is to give back the long URL if user enters shortened URL in public page. No need to login for this.

## How it works



*Figure 1 Homepage*



Don't have an account? [Signup](#)

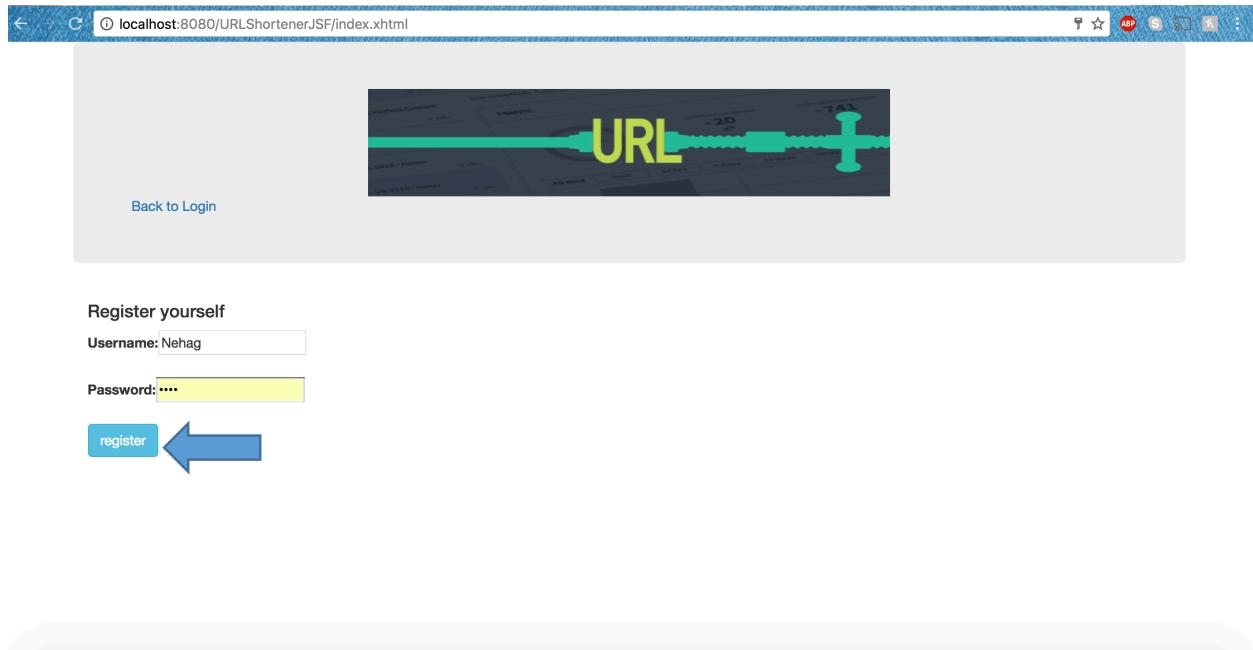
Please Sign in to continue to private page

Username: Neha

Password: Password: Validation Error: Value is required.

[Login](#) ←

**Figure 2** Login page gives error if any textbox left blank



**Figure 3** Register/Signup page

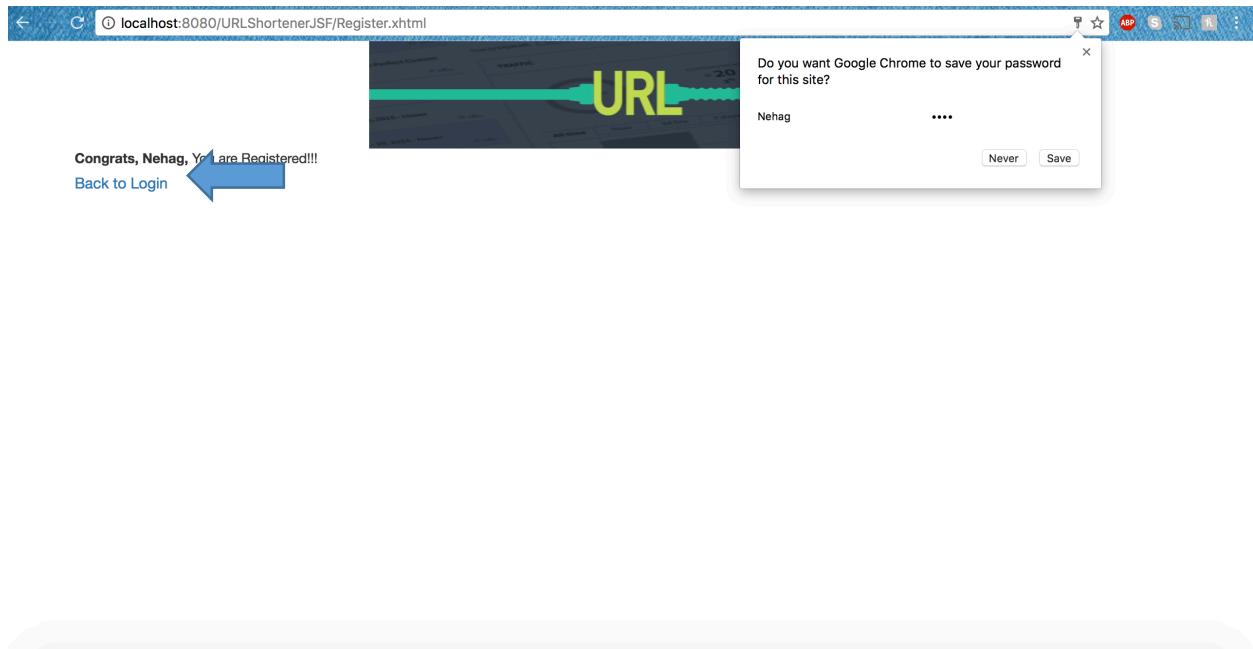


Figure 4 After Signup, success page

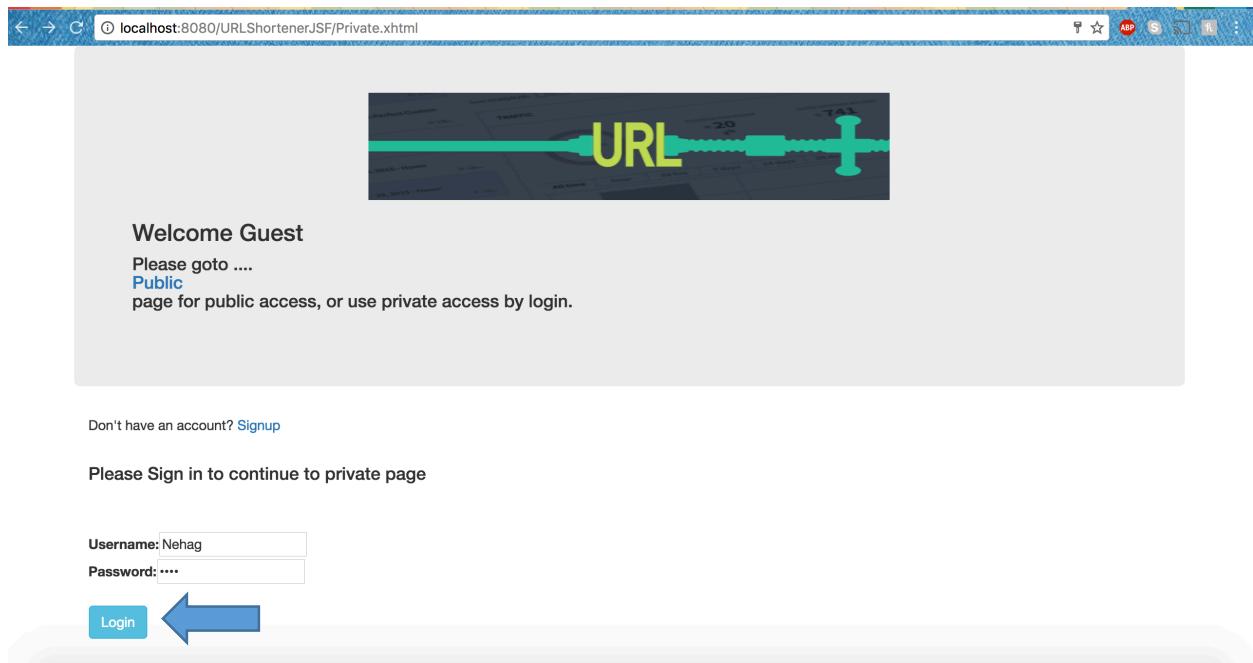
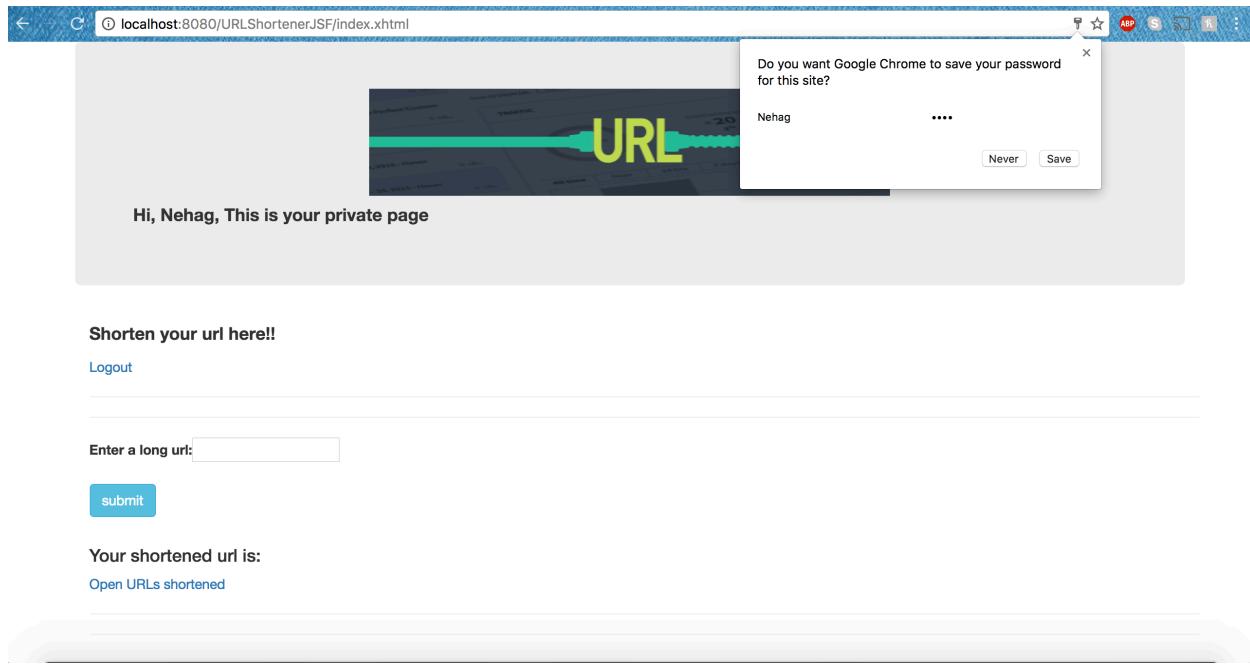
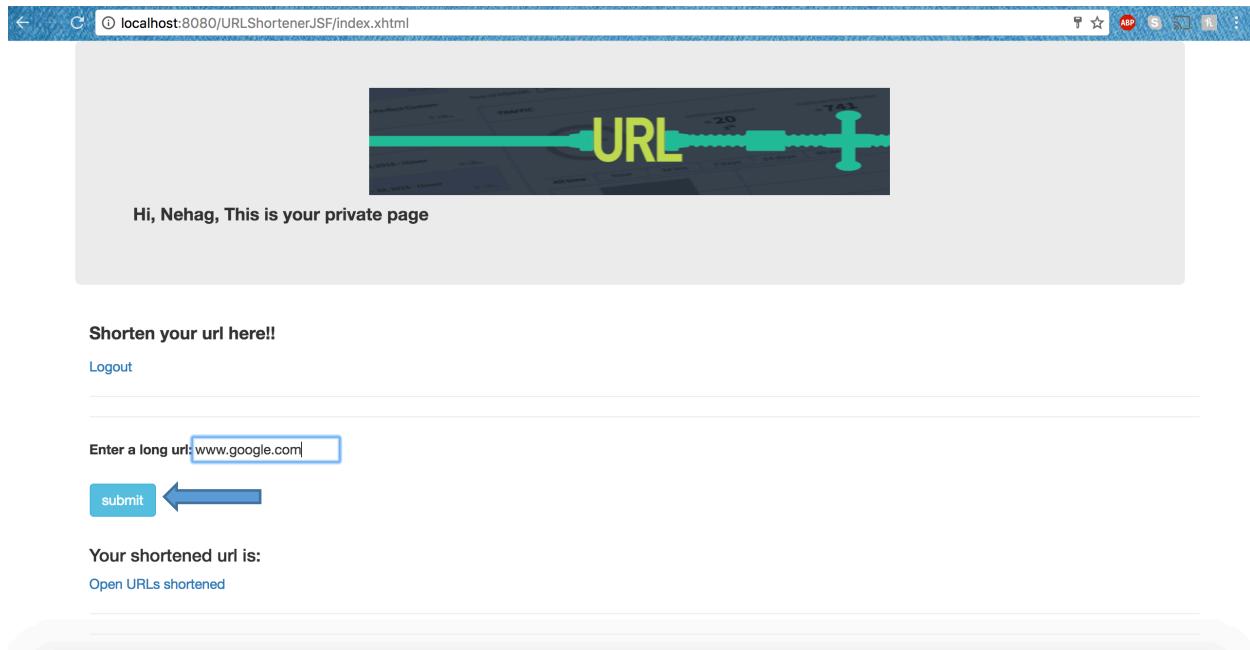


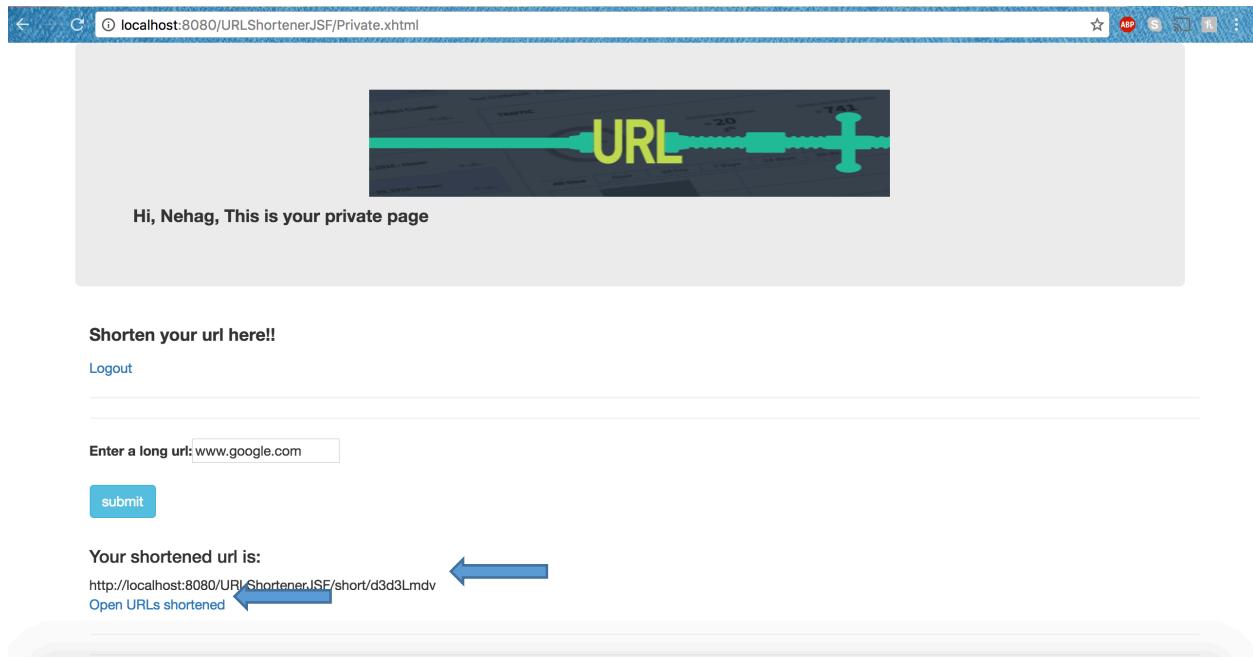
Figure 5 Now you have to login



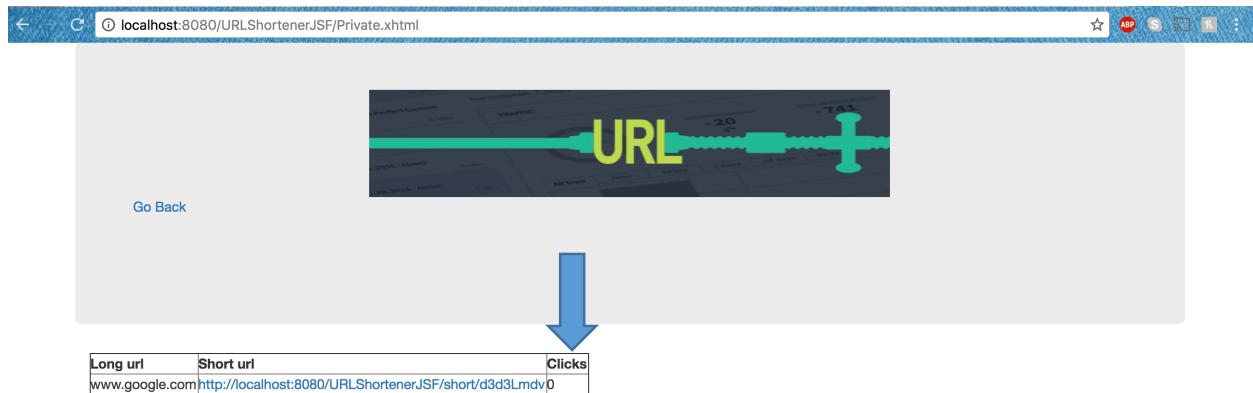
**Figure 6 Get shortened URL from long URL in private page(URL shortening is done by using java.util.Base64 class)**



**Figure 7 If you click on submit you will get shortened URL**



**Figure 8** When you click on Open URLs shortened, you will get list of URLs for that user



**Figure 9** URLs and also the clicks count for that URL. Clicks count for a particular URL is common for all users.

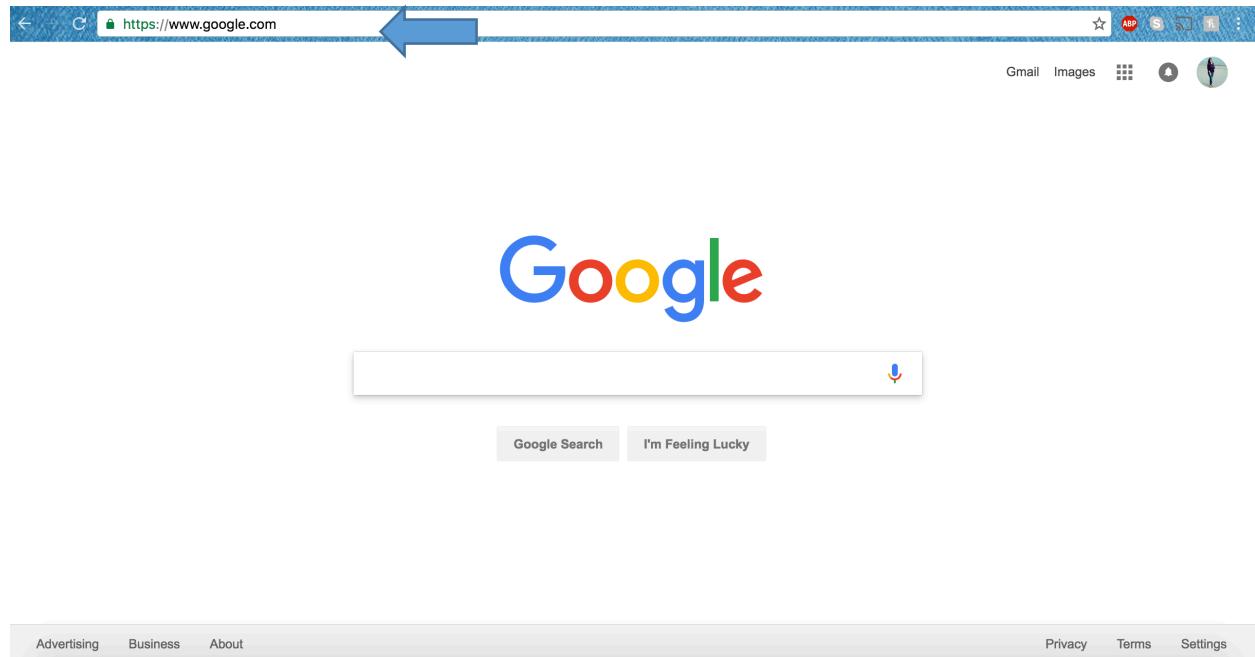


Figure 10 If you clicked on the short URL link, it will redirect you to long URL page

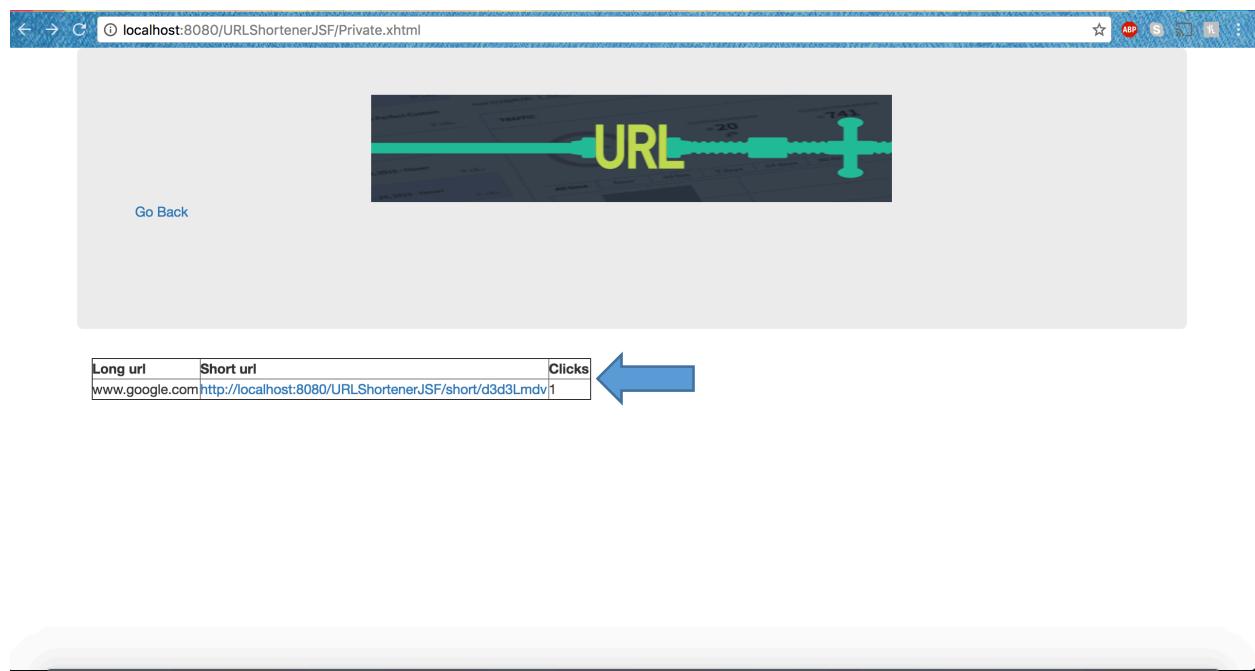


Figure 11 When you go back and refresh page, Click count increases by 1

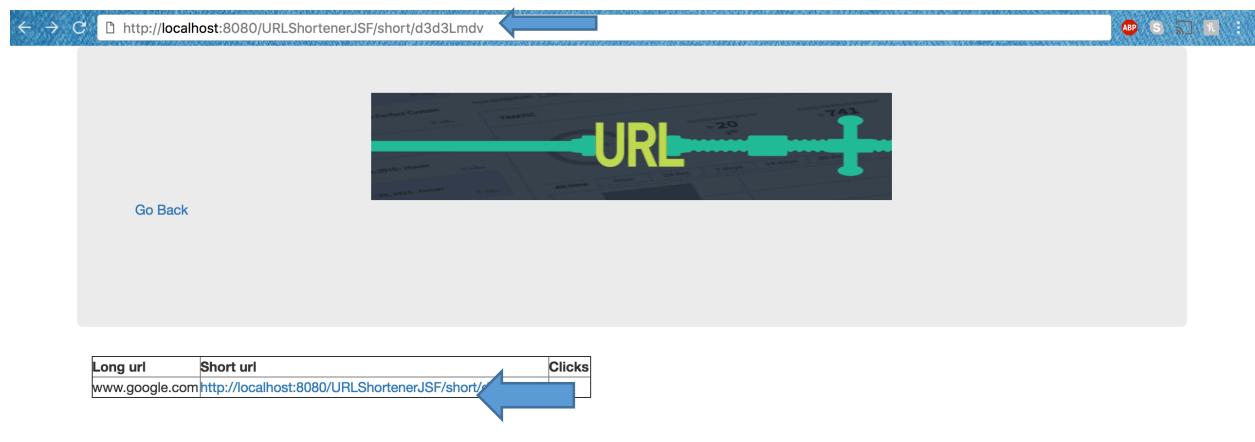


Figure 12 Copy paste shortened url in web browser address bar

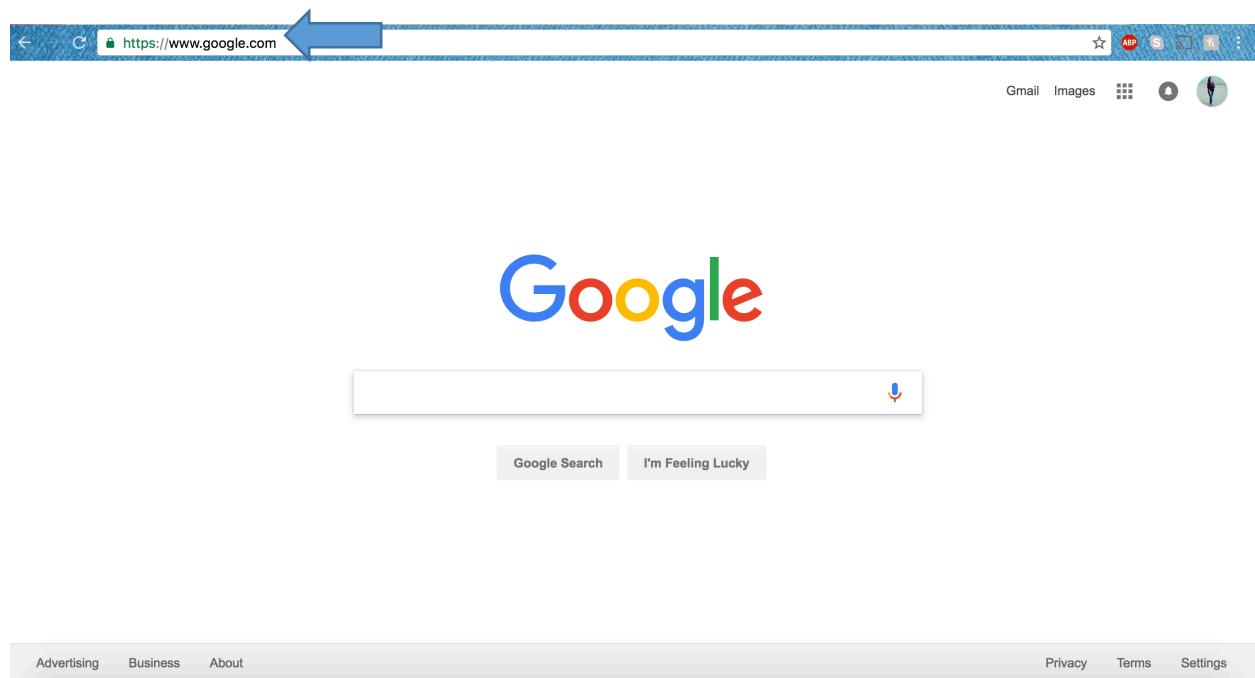


Figure 13 Redirects to long url page

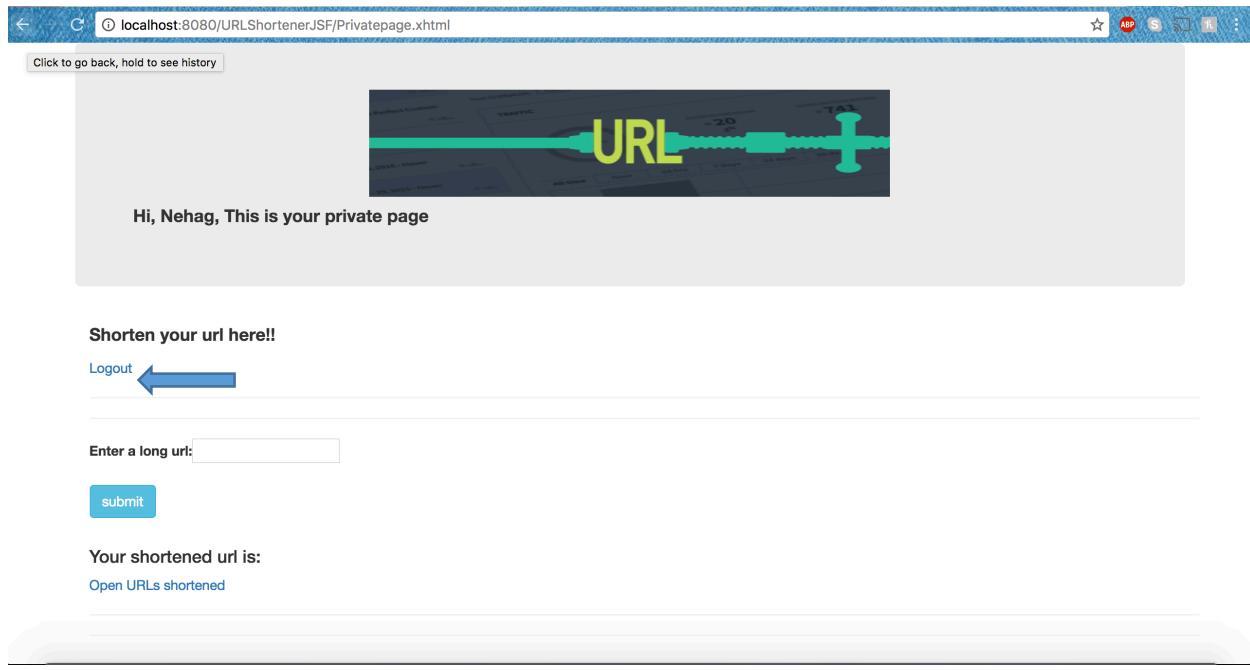


Figure 14 Go back and Logout

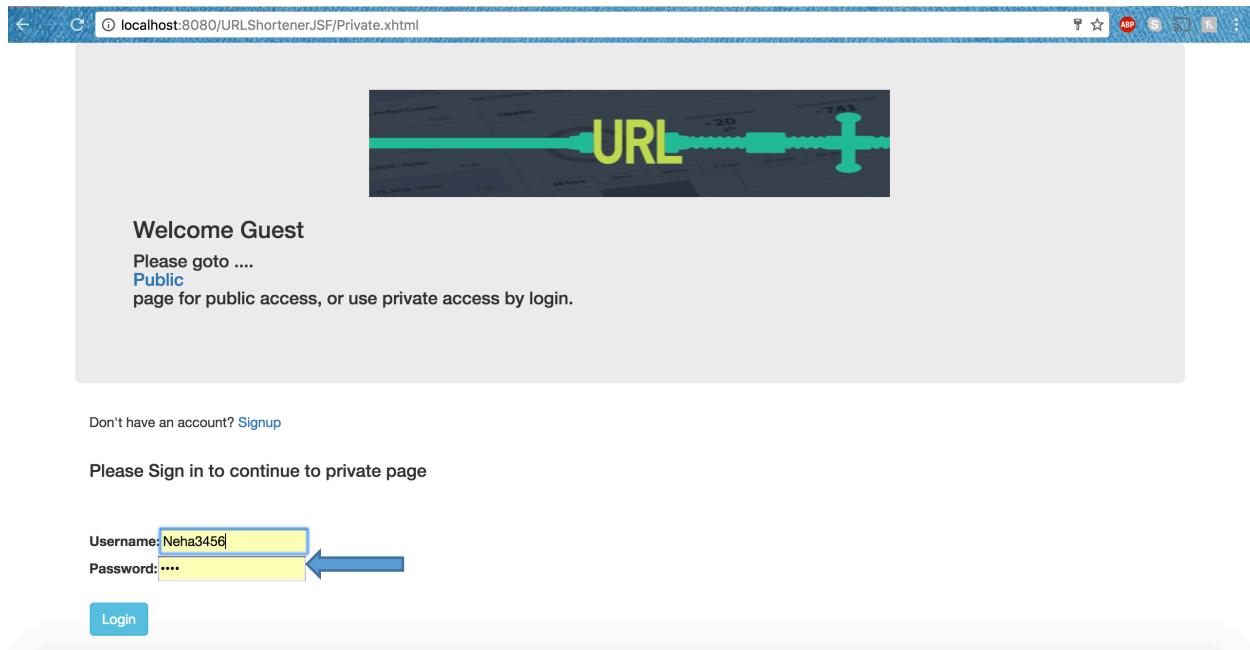


Figure 15 Login as different user, already existing

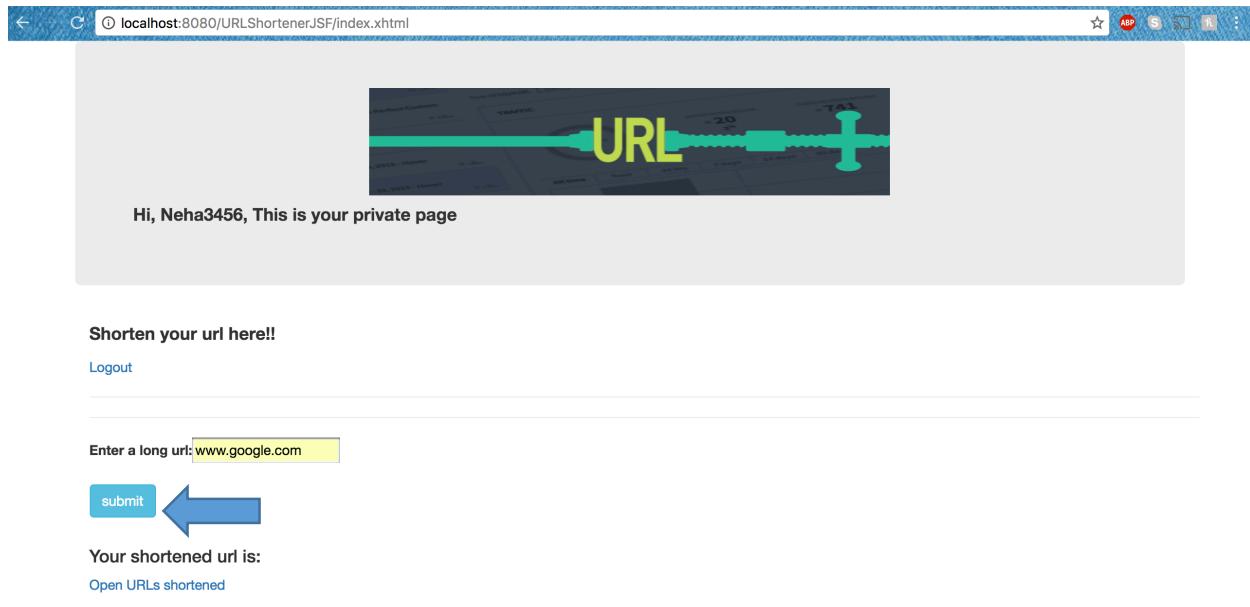


Figure 16 If you re-enter same long URL for different user and submit

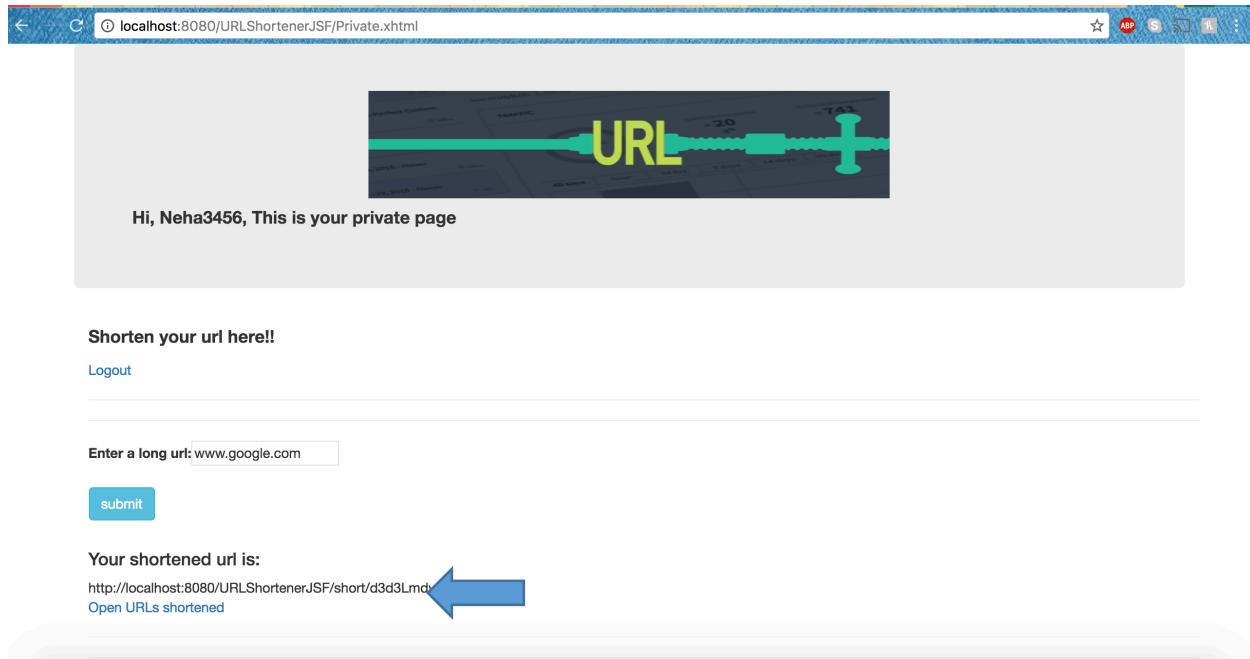


Figure 17 You get shortened url



Figure 18 You get same click count for that URL, even if current user hasn't yet clicked on the URL.

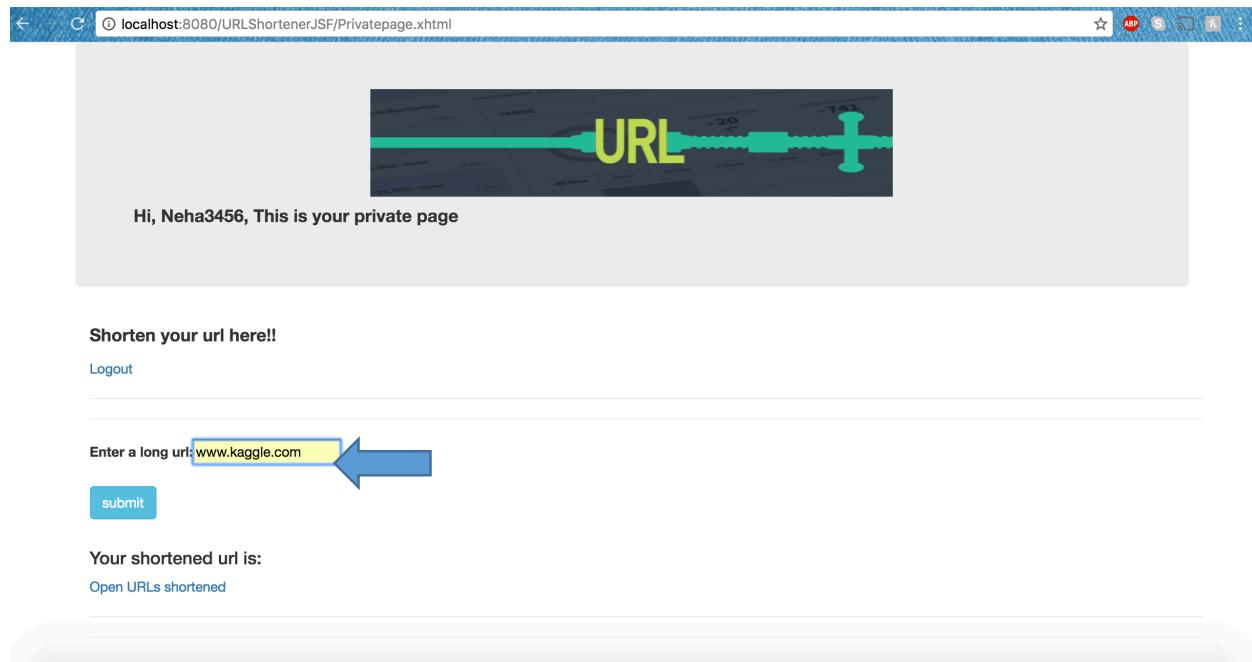
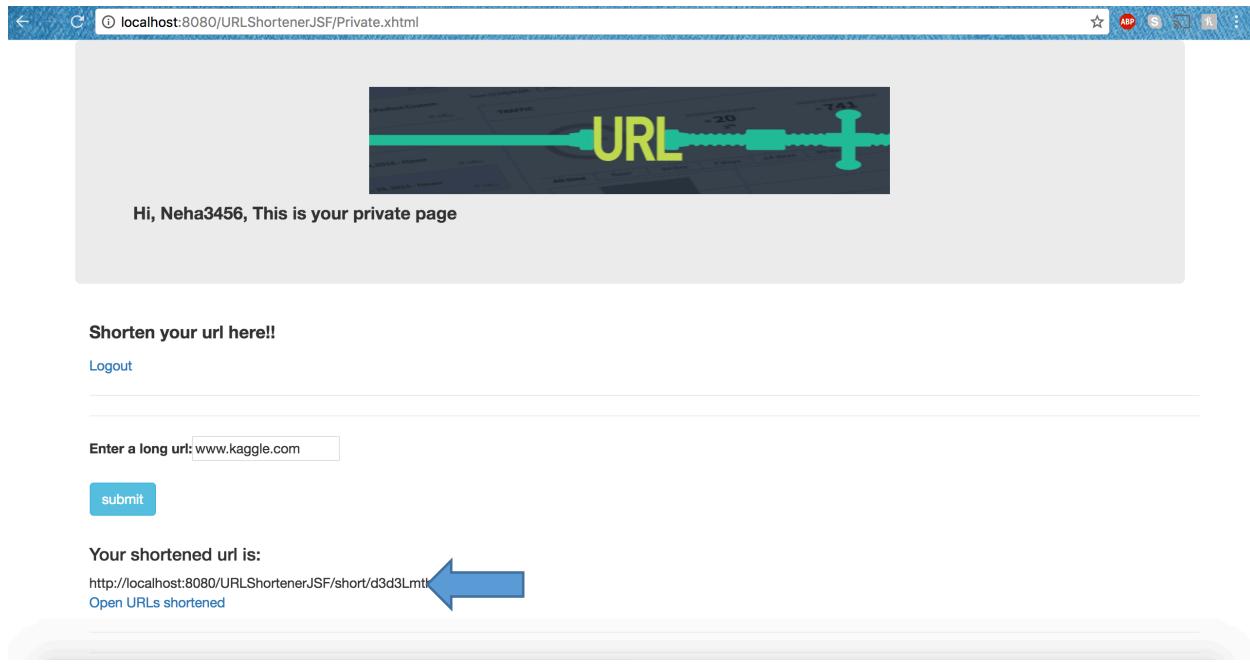
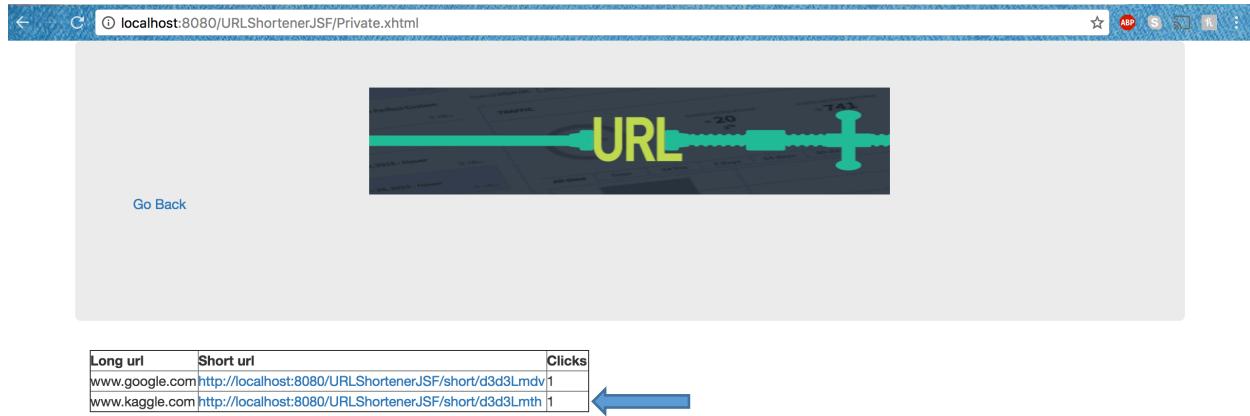


Figure 19 Add Other URL



**Figure 20 Shorten it**



**Figure 21 you get click for that URL**

HSQl Database Manager

File View Command Recent Options Tools Schemas Help

Clear S... Execute SQL

SELECT \* FROM GradProj.url

GRADPROJ.URL

- GRADPROJ
- USERNAME
- ENTRY\_ID
- LONGURL
- SHORTURL
- CLICKS
- Indices

GRADPROJ.USERDETAILS

PROJECT2.URL

PROJECT2.USERDETAILS

Properties

Ready / 7 rows retrieved

USERNAME	ENTRY_ID	LONGURL	SHORTURL	CLICKS
neha123	9	www.twitter.com	http://localhost:8080/URLShortenerJSF/short/d3d3LnR3	3
neha123	10	www.kaggle.com	http://localhost:8080/URLShortenerJSF/short/d3d3Lmth	1
neha123	11	www.facebook.com	http://localhost:8080/URLShortenerJSF/short/d3d3LmZh	1
Neha	12	www.kaggle.com	http://localhost:8080/URLShortenerJSF/short/d3d3Lmth	0
Nehag	13	www.google.com	http://localhost:8080/URLShortenerJSF/short/d3d3Lmdv	1
Neha3456	14	www.google.com	http://localhost:8080/URLShortenerJSF/short/d3d3Lmdv	0
Neha3456	15	www.kaggle.com	http://localhost:8080/URLShortenerJSF/short/d3d3Lmth	0

Figure 22 Database for URLs

HSQl Database Manager

File View Command Recent Options Tools Schemas Help

Clear S... Execute SQL

SELECT \* FROM GradProj.userdetails

GRADPROJ.URL

- GRADPROJ
- USERNAME
- ENTRY\_ID
- LONGURL
- SHORTURL
- CLICKS
- Indices

GRADPROJ.USERDETAILS

PROJECT2.URL

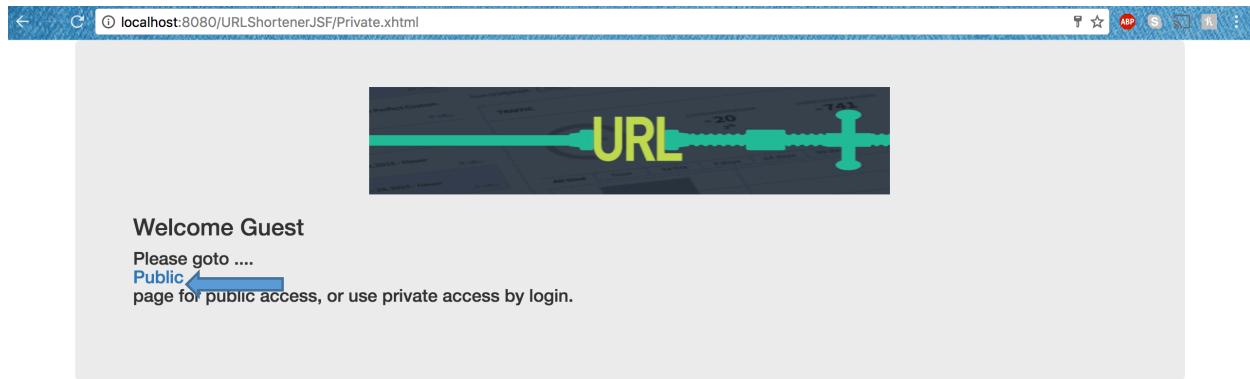
PROJECT2.USERDETAILS

Properties

Ready / 6 rows retrieved

USERID	USERNAME	PASSWORD
1	Neha	neha
2	Neha3456	neha
6	Nehag	neha
3	k7	neha
4	neha123	neha
5	neha123245	neha

Figure 23 Userdetails Database



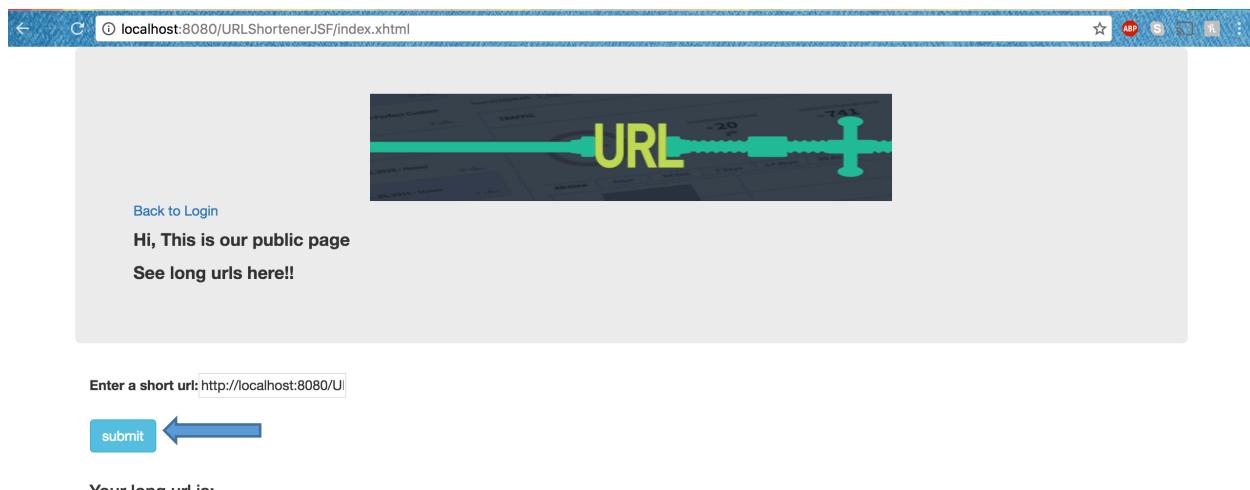
Don't have an account? [Signup](#)

Please Sign in to continue to private page

Username:   
Password:

[Login](#)

**Figure 24 Go to Public page from Homepage**

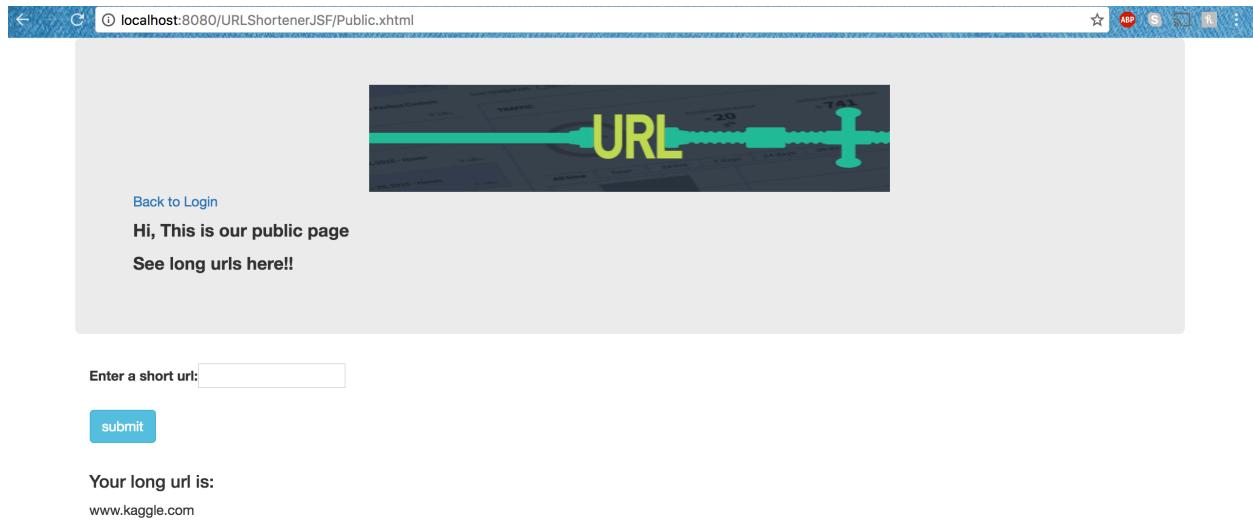


Enter a short url:  http://localhost:8080/U

[submit](#) ←

Your long url is:

**Figure 25 Submit short URL to get back long URL**



**Figure 26 Get back Long URL from database**

## **Setup Instructions:**

### **Database**

1. Download and install [Apache Ant](#).
2. Download [build.xml](#) and place it in a new, empty directory
3. Go to the directory in command prompt and type **ant start** followed by **ant manage** to open the HSQL Database Manager
4. I have attached a sql file containing the queries for creating all the databases. You can execute them by following steps:

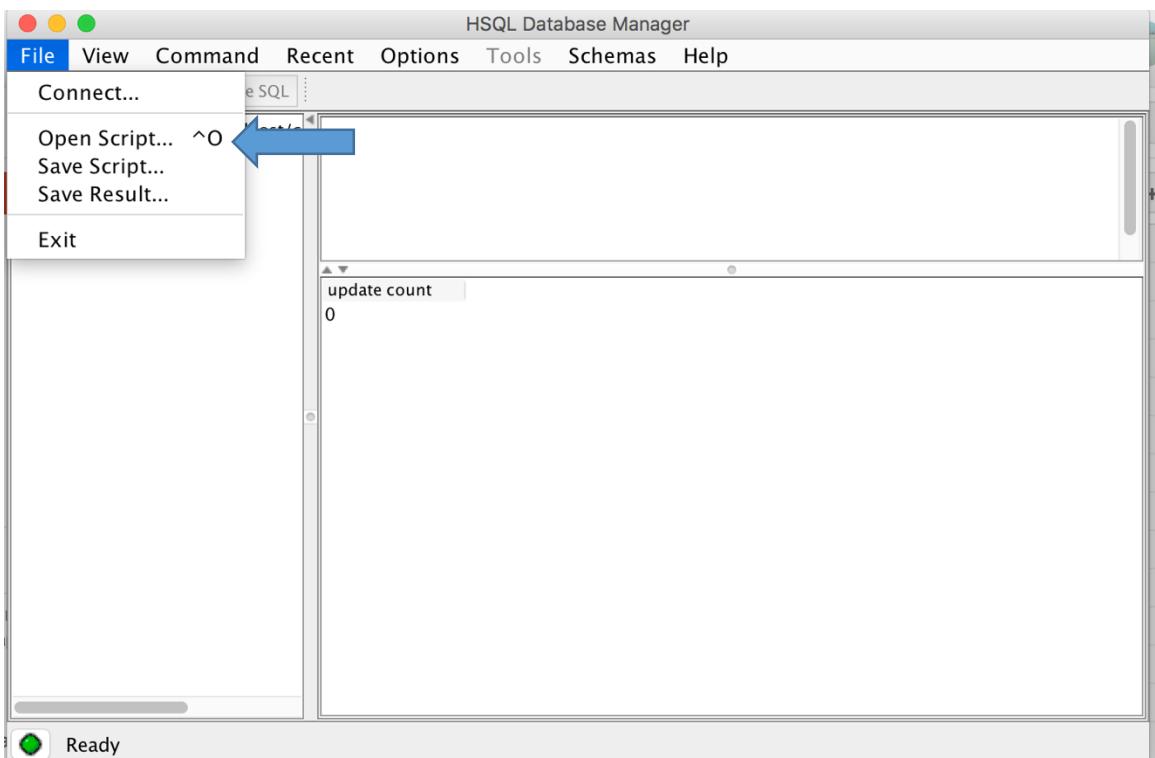


Figure 27 In HSQL Database Manager go to file -> Open Script

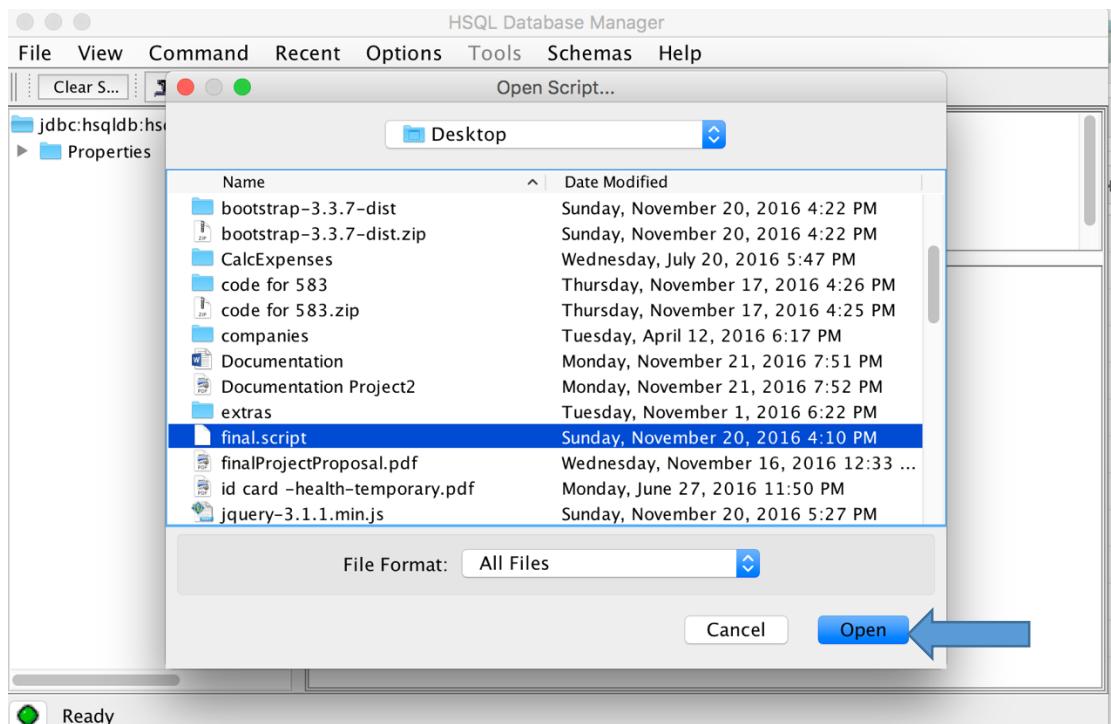
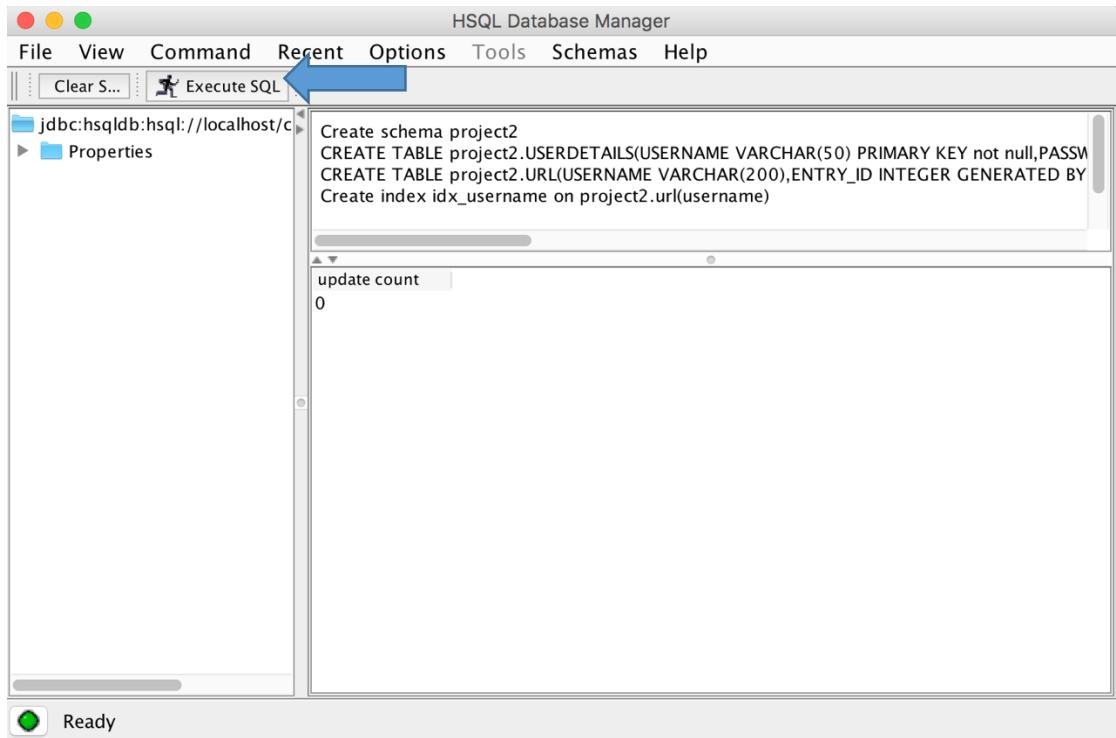
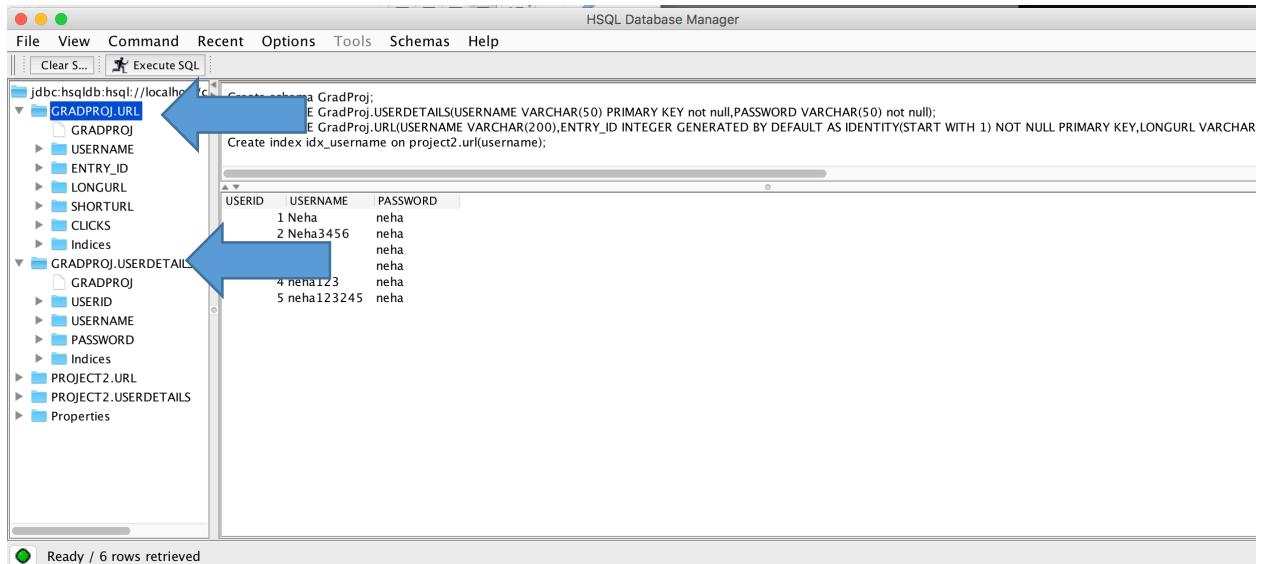


Figure 28 In open script select the final.script file attached along with project source code in the email. Then click on open.



**Figure 29** Make sure the queries don't have semicolon at the end, then click on Execute SQL



**Figure 30** The schema and databases are ready to use

5. Sample insert statement for project2.url database is:

```
insert into GradProj.url(username,longurl,shorturl,clicks) values
('red','www.google.com','www.kkk.com',1);
```

Here, in this database 'entry\_id' is generated uniquely on its own and we don't need to insert it. It is the primary key of the table.

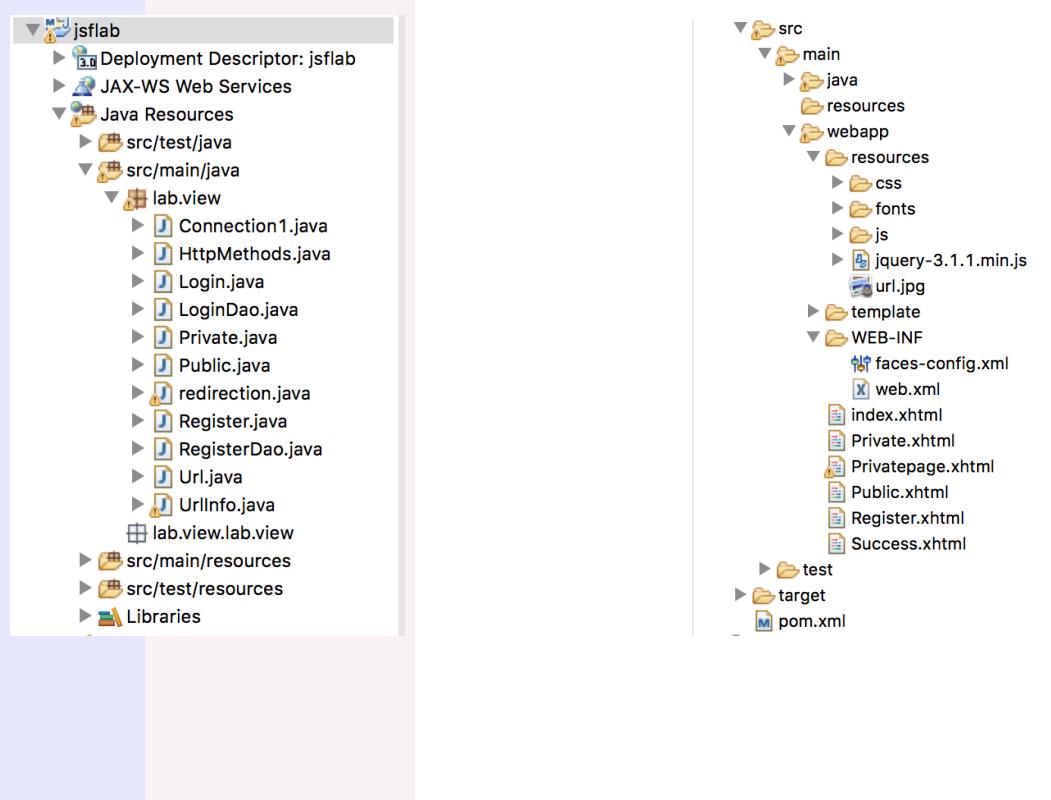
6. In GradProj.userdetails table ‘username’ is the primary key. It is also the foreign key in GradProj.url database.

## Web Application

1. We have implemented the project using the Eclipse IDE. So importing the ZIP file and also adding the JSTL jar file (if missing) to WEB-INF/lib folder and JAVAX servlet jar to the eclipse build path should help the application running
  - 1.1 In eclipse import the zip file as maven java EE file
  - 1.2 Right click on the project root folder and choose Run on server to get the application running on the browser.
2. The other way to do it is by placing the war file in the tomcat webapps directory and launching the webserver. On launching the server navigate to  
<http://localhost:8080/URLShortenerJSF>
3. To check the incremented click value after redirection, coming back to the page doesn't increment the click but retyping the URL or refreshing the page again in the browser works.
4. The project is also compatible with maven build, so we can navigate to the project directory and do **mvn clean install**. Once that is done use **mvn package** command to package the build application.
5. The war file is built in the target folder
6. Start the apache tomcat server once and use the below command to deploy and start the application

Command = mvn tomcat8:deploy

## Architecture of the project:



1. Login and Signup: The login is built on index.xhtml and Login.java (Login managed bean) & LoginDao.java (Database interactions) files. The signup is built on Register.xhtml and Register.java (Register managed bean) and RegisterDao.java (Database interactions)
2. Once the user is done with signup or login, he/she is redirected to a Private.xhtml page
3. The private page where the Short url to long url conversion happens contains the view private.xhtml and the managed bean Private.java and Database interactions are present in Url.java.
4. The public page where the long url to short url conversion happens contains the view Public.xhtml which is controlled by the Public.java file and Database interactions are present in Url.java.
5. The user details are stored in a GradProj.userdetails database.
6. The url's and the clicks for a particular user are stored in GradProj.url database.
7. There are separate connection and Http objects for creating connection with Database and session and their respective method definitions are present in Connection1.java and HttpMethod.java respectively.
8. Redirection of short url's when entered in address bar of browser is taken care of in redirection.java web filter file.
9. The project is packaged as a maven build with all the dependencies mentioned in pom.xml
10. JSF configurations for my web application is configured in faces-config.xml.

## References:

<https://javaserverfaces.java.net/nonav/docs/2.2/>

<http://tomcat.apache.org/tomcat-7.0-doc/>

<http://proquest.safaribooksonline.com/book/programming/java/9781118909317>

<https://maven.apache.org/what-is-maven.html>

<http://getbootstrap.com/>

<http://hsqldb.org/doc/2.0/guide/index.html>