URL Shortner

(CPSC 476: Programming Project- 1)

October 26th, 2016

Submitted by,

Husamaldeen Naji

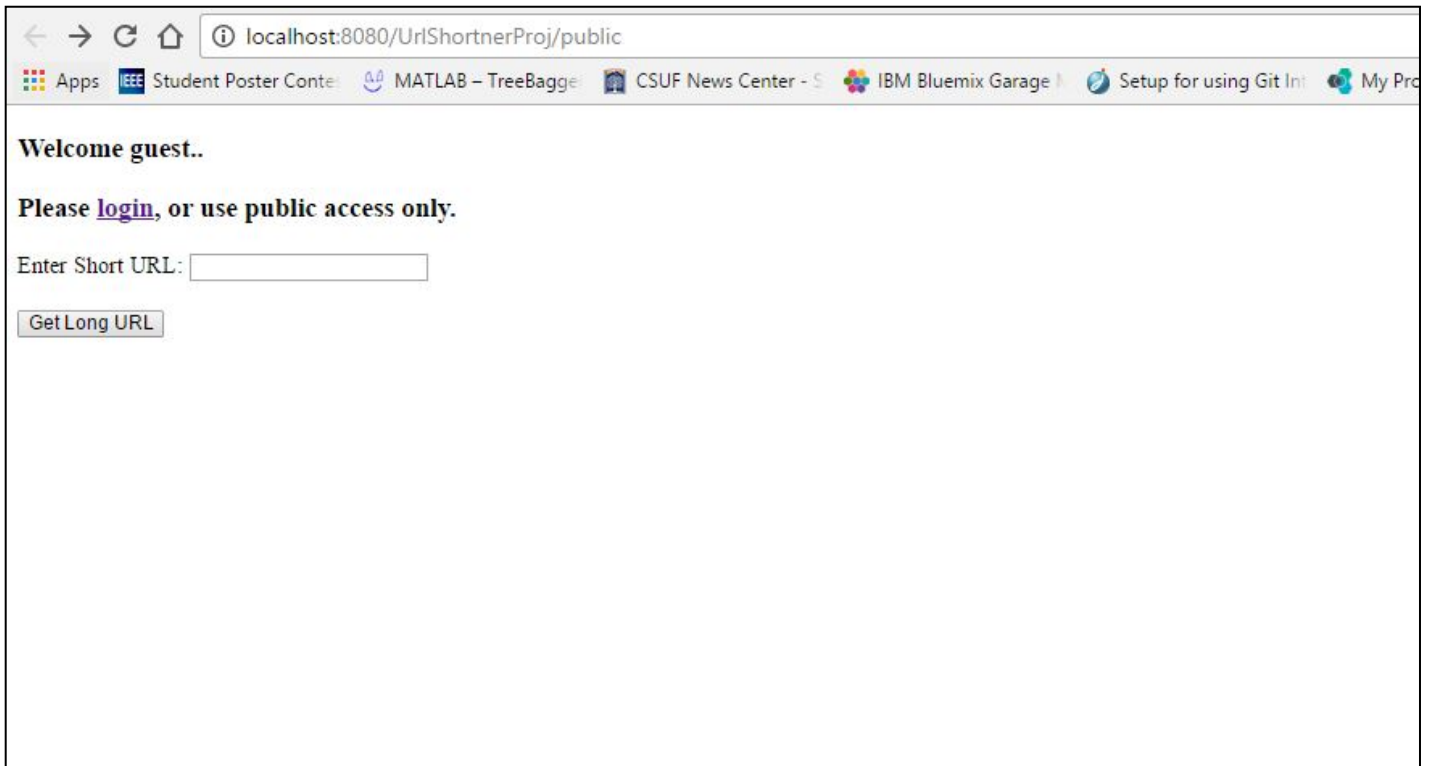**Neha gawade**

Vinay ramappa

Karthik Karunanithi

Purpose

The purpose of this project is to build a Client server application that reflects a jsp servlet communication without using a structured database. The project talks about shortening a URL when a user inputs a long URL and the reverse of it when the user enters a shortened URL.

How it works

The public page looks as shown in the screen shot below, when the user visits the localhost website.

The user can submit the long URL here and get the short URL as output as shown below

Once we click on login it navigates to the login page where users can either sign up or login

Test accounts are username: Nicholas password: password

Next Once we login and sign up we end up in the private page where user can enter a long URL and get the shorter URL. Clicking on the link navigates to the original long URL and also shows the number of clicks each link has received .

We have included a Base 64 library (`java.util.Base64)` that supports the Short URL generation

← → C △ ⓘ localhost:8080/UrlShortnerProj/private

::: Apps IEEE Student Poster Conte: 😊 MATLAB – TreeBagge: 🏛 CSUF News Center - S 🌸 IBM Bluemix Garage !

Hello Nicholas, This is you private page!

Logout

Shorten your url here!!

Enter a long url:

Submit

Your shortened url for www.yahoo.com is :

http://localhost:8080/short/3d3Lnl

Short URL's	Clicks
http://localhost:8080/short/3d3Lnl	0
http://localhost:8080/short/3d3Lmd	1

Setup Instructions:

1. We have implemented the project using the Eclipse IDE. So importing the ZIP file and also adding the JSTL jar file 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/UrlShortnerproj/public
3. To check the incremented click value after redirection, coming back to the page doesn't increment the click but retyping the URL again in the browser works.

Architecture of the project:

1. Login and Signup: The login and singup is built on login.jsp and loginservlet.java file
2. Once the user signup or login, he/she is redirected to a private.jsp page
3. The private page where the Short url to long url conversion happens contains the view Private.jsp and the controller PrivateServlet.jsp
4. The public page where the long url to short url conversion happens contains the view public.jsp and guest.jsp which is controlled by the publicservlet.java servlet file
5. The user details are stored in a Hash Map data structure.
6. The urls and the clicks for a particular user are stored in an array of objects defined by class Pojo.java.
7. The project is packaged as a maven build with all the dependencies mentioned in pom.xml

References:

http://docs.oracle.com/javaee/6/api/javax/servlet/package-summary.html

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

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

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