Setup Instructions

On Ubuntu:

Deploying the application using war file:

What is a war file?
In software engineering, a **WAR file** (or Web application ARchive) is a JAR **file** used to distribute a collection of JavaServer Pages, Java Servlets, Java classes, XML **files**, tag libraries, static web pages (HTML and related **files**) and other resources that together constitute a web application.

If you do not want to modify the code and just want to deploy the Clickr application using apache tomcat service then first install apache tomcat on your system.
Here is a good link:
https://www.digitalocean.com/community/tutorials/how-to-install-apache-tomcat-7-on-ubuntu-14-04-via-apt-get

The ClickrServer war file can be downloaded from the following link:
https://github.com/bhargavchippada/ClickrServer/releases
(Download the latest version or an older version)
The Clickr admin page username is clickrdemo and password is clickr@iitb.

The essential steps are as follows (Install through apt-get. This is the simplest method):

1. Update system
   ```
   sudo apt-get update
   ```

2. Install Tomcat
   ```
   sudo apt-get install tomcat7
   ```

   Tomcat is not completely setup yet but you can access the default splash page by going to your domain or IP address followed by :8080 in a web browser:
   http://your_ip_address:8080

3. Installing additional packages
   `sudo apt-get install tomcat7-docs tomcat7-admin tomcat7-examples`

   This is recommended for people new to tomcat

4. Install java development kit to develop java apps to run on server(optional)
   `sudo apt-get install default-jdk`

   In addition to JDK, the Tomcat documentation suggests also installing Apache Ant, which is used to build Java applications, and a source control system, such as git.

   `sudo apt-get install ant git`

5. Configure tomcat web management interface
   In order to use the manager webapp installed in Step 3, we must add a login to our Tomcat server. We will do this by editing the tomcat-users.xml file:
   `sudo nano /etc/tomcat7/tomcat-users.xml`

   You will want to add a user who can access the manager-gui and admin-gui (the management interface that we installed in Step Three). You can do so by defining a user similar to the example below. Be sure to change the password and username if you wish, the user tag should be enclosed between tomcat-users tag as follows:

```
<tomcat-users>
   <user username="admin" password="admin" roles="manager-gui,admin-gui"/>
</tomcat-users>
```

   Save and quit the tomcat-users.xml file. To put our changes into effect, restart the Tomcat service:

   `sudo service tomcat7 restart`

   Further, to start tomcat server run: `sudo service tomcat7 start`
   to stop tomcat server run: `sudo service tomcat7 stop`

6. Access the web interface
   Now that we've configured an admin user, let's access the web management interface in a web browser:

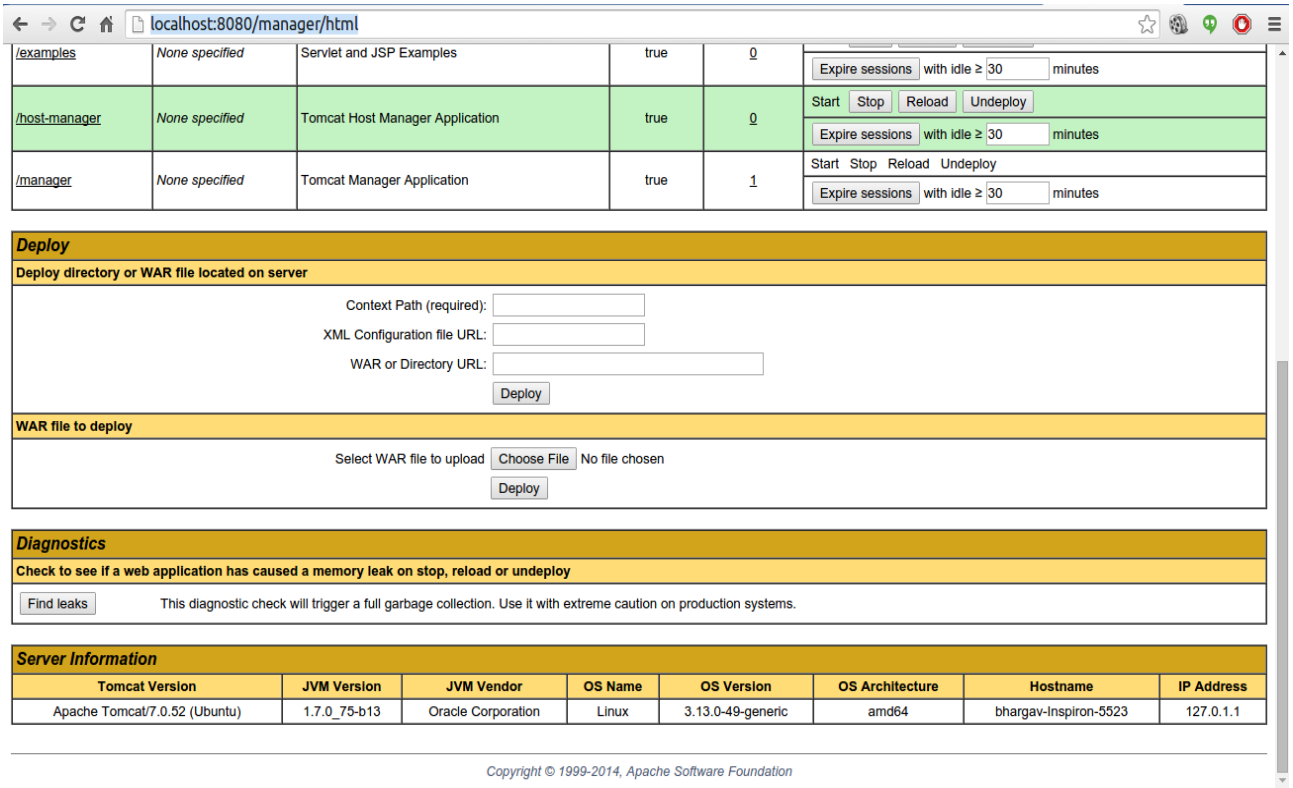
   http://your_ip_address:8080

7. You will see something like the following message:

   # It Works!

   As you can see there are some links, we are concerned with manager webapp link, click on it and you will be taken to http://your_ip_address:8080/manager/html

   There is a section which says ‘war file to deploy’



You can select a war file using choose file button and click on deploy. Scroll up and you can see a section named ‘Applications’ and you should see ClickrServer listed. You can start, stop, reload and undeploy the Clickr application.

You can access the Clickr web interface by opening the following url in your web browser:
http://your_ip_address:8080/ClickrServer

The ClickrServer war file can be downloaded from the following link:
https://github.com/bhargavchippada/ClickrServer/releases
(Download the latest version or an older version)
The Clickr admin page username is clickrdemo and password is clickr@iitb.

Note: If you don't want to install using apt-get then you have to download the binary distribution from Apache Tomcat site. Refer to Apache Tomcat Documentation for instructions.

Setup for developing the application:

Following is the directory structure of the project:-

- ❖ src
  - ➢ datahandler (contains classes for storing/retrieving the data)
  - ➢ support (helper classes)
  - ➢ webconnectionjdbc (servlets for exchanging data with Clickr mobile application)
  - ➢ webinterfacehandler (servlets for exchanging data with Admin Clickr application)
- ❖ WebContent
  - ➢ bower_components (polymer dependencies)
  - ➢ bower.json file to manage polymer dependencies
  - ➢ media (images used in html)
  - ➢ elements (polymer webcomponent elements used for making the web interface)
  - ➢ WEB-INF
    - ■ web.xml (contains servlet mappings)
    - ■ lib
      - gson (gson library)
      - doc folder
        - ◆ gson doc jar
  - ➢ html pages with polymer dependencies
- ❖ examplefiles (example input files for Clickr application)

As you can see there are two external dependencies:

- polymer v0.5 (Updating this is ok but thorough testing has to be done to check that nothing is broken!
- gson v2.3.1 (You can update this jar and its doc jar too if a newer version is released in future)

The essential steps are as follows (Eclipse IDE):

1. Download eclipse Luna for Java EE developers (Easiest way).
   If you have some other eclipse release then you have to install web development tools software from inside eclipse marketplace.
2. Download the latest version of war file from the following link:
   https://github.com/bhargavchippada/ClickrServer/releases
3. Download apache tomcat tar.gz core binary distribution and extract it in your desired location. (you can also download this using eclipse in next step)
4. Open eclipse, click File->Import->Web->War file and select the downloaded war file.
   Select the Target runtime as Apache tomcat v7.0 (or v6.0 or v8.0), if there is no target runtime available then click on 'new', select apache tomcat version that you have downloaded in Step 3, click next, select the tomcat installation directory of Step 3. If you have not installed tomcat in Step 3 then click and download & install button and select your installation folder.
   Click next or finish. Now target runtime is shown. click next and then click finish.
5. Let's start the server now, Go to File->New->Server
   Select tomcat version, click next, select ClickrServer project and add it to server. Click finish.
6. To host the Clickr application, right click the ClickrServer project and select 'Run As', select 'Run on Server', select available server and click finish.
7. The application is now hosted on the server, type the following url in your browser to access the application: `http://your_ip_address:8080/ClickrServer`
8. You can now start developing the ClickrServer.