POLITECNICO DI MILANO
SOFTWARE ENGINEERING 2

GuessBid

Installation Manual

Mirjam Škarica

Milan, June 2015

Contents

Chapter 1

Introduction

This document is intended to provide the necessary information to install the GuessBid web application on a GlassFish server. In addition to this manual, in the Installation folder you can find the following files:

- The archive GuessBid.war containing the web application
- The createSrcipt.sql containing database structure

Chapter 2

Requirements

There are the requirements to run GuessBid web application:

Operating System the application can be executed on any operating system that is able to run Glassfish 4.1, including Windows, MacOSX and Linux (for further details see here[1]).

Application Server Glassfish 4.1 (Open Source project by Oracle)

DBMS MySQL Community Server 5.7

JDK 8 Java SE Development Kit 8

Browser any modern browser with HTML5, CSS3 and JavaScript support (e.g. Chrome, Firefox)

2.1 Database configuration

This section provides the instructions to correctly setup the database for the application. If you have not configured a password for your MySQL server follow these instructions[2].

The application uses a database named guessbid_db.

Here are the instructions from command line, alternatively you could do the same from your IDE or using any mysql GUI client (e.g., MySQL Workbench).

Position yourself in the installation folder:

```
shell> cd <path>/Installation
```

Once you there, to create the DB structure needed, execute the following command:

```
shell> mysql < createScript.sql;
```

[1] https://blogs.oracle.com/theaquarium/entry/glassfish_supported_platforms

[2] http://dev.mysql.com/doc/refman/5.0/en/resetting-permissions.html

2.2 Server configuration

This section provides the instructions to correctly configure Glassfish server to run the application. Here is the overview of this section:

1. Install MySQL connector/J, a driver that provides for client applications developed in the Java programming language with to connect to MySQL database through Java Database Connectivity (JDBC) API.

2. Create a Glassfish Connection Pool, a group of reusable connections for a particular database

3. Create a JDBC Resource which provides the applications with a means of connecting to a database (Multiple JDBC resources can specify a single connection pool).

2.2.1 Install MySQL connector

This section explains[3] how to use MySQL Connector/J with GlassFish ™ Server Open Source Edition 3.0.1[4].

1. Download MySQL Connector/J from http://dev.mysql.com/downloads/file.php?id=456315

2. Copy the MySQL Connector/J jar file to <path>/glassfish4/glassfish/domains/domain1/lib directory. Alternatively, if that doesn't work, copy it in the <path>/glassfish4/glassfish/lib directory

3. Restart the GlassFish Application Server.

2.2.2 Create a Connection Pool

Assuming the Glassfish server is running, follow the steps below to create a connection pool.

1. In the GlassFish Administration Console, using the navigation tree, navigate to Resources > JDBC > Connection Pools.

2. In the JDBC Connection Pools frame click *New* where you will enter a two step wizard.

3. Under General Settings fill out the fields as follows:

   Name: MySQLGuessBidConnectionPool

   Resource Type: javax.sql.DataSource

   Database Vendor: MySQL

[3] http://dev.mysql.com/doc/connector-j/en/connector-j-usagenotes-glassfish-config.html

[4] For more information, see "Integrating the JDBC Driver" in GlassFish Server Open Source Edition Administration Guide, available at GlassFish Server Documentation.

4. Click *Next* to go to the next page of the wizard.

5. Scroll down to *Additional Properties* and set these properties:

   URL: jdbc:mysql://localhost:3306/guessbid_db

   User: root (or any user granted to access guessbid_db)

   Password: the root(user) password

   ServerName: localhost

   DatabaseName: guessbid_db

6. To test your connection pool click the Ping button at the top of the frame. A message will be displayed confirming correct operation or otherwise. If an error message is received recheck the previous steps, and ensure that MySQL Connector/J has been correctly copied into the previously specified location.

2.2.3 Creating a JDBC Resource

Your Java application will usually reference a data source object to establish a connection with the database. This needs to be created first using the following procedure.

1. Using the navigation tree in the GlassFish Administration Console, navigate to R*esources > JDBC > JDBC Resources*. A list of resources will be displayed in the JDBC Resources frame.

2. Click *New* and fill out these fields:

   JNDI Name: jdbc/guessbid_db

   Pool Name: MySQLGuessBidConnectionPool

3. Click OK to create the new JDBC resource.

2.2.4 Creating a JDBCRealm:

From the Glassfish control panel navigate to Configurations > Server-config >Security > Realms

1. Click *New* and use the following configuration:

   Realm Name: jdbcRealmRegistration (must be the same referenced in the web.xml)

   Class Name: JDBCRealm

   JAAS Context: jdbcRealm

   JNDI: jdbc/guessbid_db

   User Table: user

User Name Column: user_id

Password Column: password

Group Table: user

Group Table User Name Column: user_id

Group Name Column: groupname

Password Encryption Algorithm: MD5

DigestAlgorithm: SHA-256

2. Server restart may be required

Chapter 3

Application deployment

Assuming your MySQL server is running, access the GlassFish control panel an open *Applications*

1. Click on *Deploy*
2. Under *Location* select the file GuesBid.war archive from the *Installation* folder.
3. Make sure that the *Context Root* is set to GuesBid
4. Click OK to deploy the application.
5. If everything has been correctly set up, you will see GuessBid application in the list of your applications.
6. Restart the the server
7. Connect to http://localhost:8080/GuessBid