



# GuessBid

---

## Installation Manual

Milica Shulevska

**POLITECNICO DI MILANO | SOFTWARE ENGINEERING 2**

23.06.2015

## CONTENTS

---

1	Introduction .....	2
1.1	Purpose .....	2
1.2	Tools.....	2
1.3	Attachments.....	2
2	Configuration.....	2
2.1	Setting up the database .....	2
2.1.1	Create a new connection .....	2
2.1.2	Data Import/Restore.....	3
2.1.3	Import from file.....	4
2.1.4	Target Schema .....	4
2.2	Setting up the server.....	4
2.2.1	MySQL Connector .....	5
2.2.2	Creating a connection pool .....	5
2.2.3	Creating a JDBC resource.....	6
2.2.4	Creating a JDBC realm .....	6
3	Deployment .....	7

# 1 INTRODUCTION

---

## 1.1 PURPOSE

This document has the aim to provide the installation steps for a successful deployment of the web application GuessBid which is made in the course Software Engineering 2 at Politecnico di Milano. By following the steps, one would be able to successfully deploy the application.

## 1.2 TOOLS

You will need the following tools to successfully deploy the GuessBid web application.

- **NetBeans IDE (8.0.2)**
- **GlassFish 4.1 Server**
- **MySQL Community Server 5.7**
- **MySQL Workbench 6.2 CE**

## 1.3 ATTACHMENTS

These are the files needed for the installation:

- `guessbid.war` – the package of the GuessBid application
- `guessbid-db.sql` – the script to create the GuessBid database

# 2 CONFIGURATION

---

This section provides the details of every step in the configuration process.

## 2.1 SETTING UP THE DATABASE

There are many ways of setting up the database needed for the web application. In this manual, the approach with MySQL Workbench is covered as being one of the simplest.

It is assumed the MySQL server is started.

### 2.1.1 Create a new connection

If you don't have any connection in Workbench, create a new connection as it is shown on the photo.

**Setup New Connection**

Connection Name:  Type a name for the connection

Connection Method:  Method to use to connect to the RDBMS

Parameters **SSL** Advanced

Hostname:  Port:  Name or IP address of the server host - and TCP/IP port.

Username:  Name of the user to connect with.

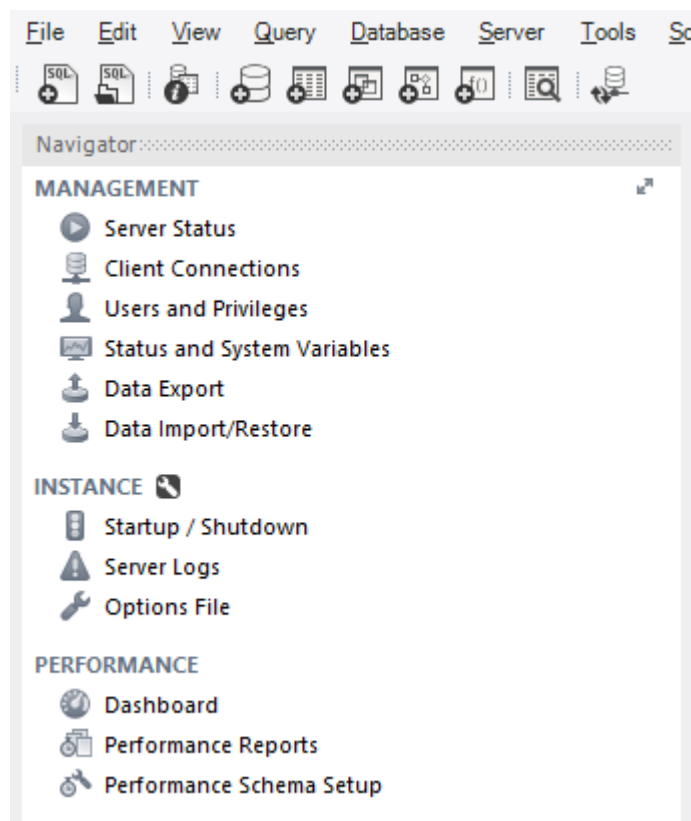
Password:   The user's password. Will be requested later if it's not set.

Default Schema:  The schema to use as default schema. Leave blank to select it later.

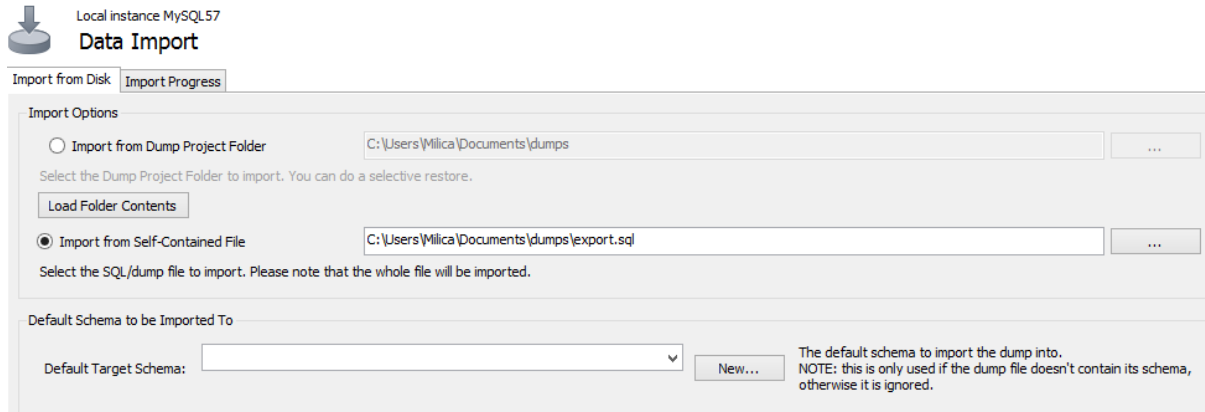
### 2.1.2 Data Import/Restore

When you connect, on the left side you should see a Data Import/Restore button.

Click on it.

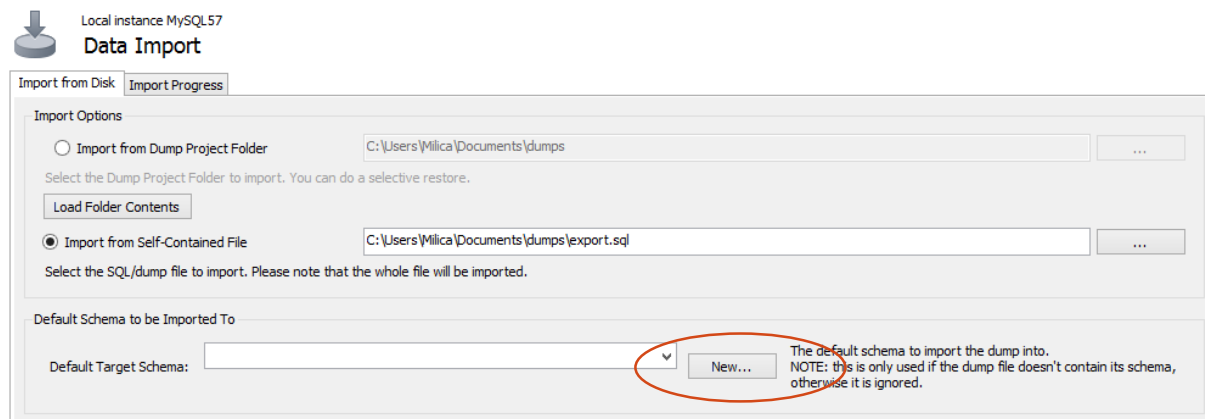


### 2.1.3 Import from file



Choose the option for importing from Self-contained file and import the provided SQL: guessbid-db.sql.

### 2.1.4 Target Schema

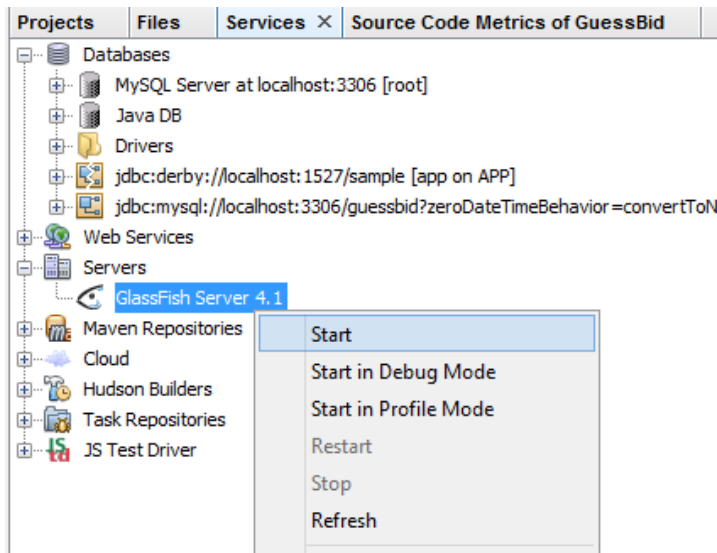


Click on New and put the name guessbid.

Click on Import and you should have the database set.

## 2.2 SETTING UP THE SERVER

In this section, the setting up of the server is describe. First, the Glassfish server should be started. This could be done through NetBeans IDE, select Services -> Server -> GlassFish. Right click and start.



### 2.2.1 MySQL Connector

Download from <http://dev.mysql.com/downloads/connector/j/> the platform independent connector. Extract the jar file to into your GlassFish installation directory, under the path *glassfish/lib/*.

### 2.2.2 Creating a connection pool

- Open the GlassFish console and navigate to Resources -> JDBC -> Connection Pools. Click on New.
- Input as the following and click Next

#### New JDBC Connection Pool (Step 1 of 2)

Identify the general settings for the connection pool.

##### General Settings

Pool Name:

Resource Type:  Must be specified if the datasource class implements more than 1 of the interface.

Database Driver Vendor:  Select or enter a database driver vendor

Introspect: ☒ **Enabled** If enabled, data source or driver implementation class names will enable introspection.

Add these additional properties:

Password: root

User: root

PortNumber: 3306

ServerName: localhost

URL: jdbc:mysql://localhost:3306/guessbid?zeroDateTimeBehavior=convertToNull.

DatabaseName: guessbid

driverClass: com.mysql.jdbc.Driver

### 2.2.3 Creating a JDBC resource

- Open the GlassFish console and navigate to Resources - > JDBC - > JDBC Resources. Click on New.
- Input as the following

#### New JDBC Resource

Specify a unique JNDI name that identifies the JDBC resource you want to create.

JNDI Name: \*

Pool Name:  ▼  
Use the [JDBC Connection Pools](#) page to create new pools

Description:

Status: ☒ Enabled

### 2.2.4 Creating a JDBC realm

- Open the GlassFish admin console and navigate to Configurations - > Server- config - > Security - > Realms. Click on New
- Input as the following

JAAS Context: jdbcRealm

JNDI: jdbc/guessbid

User Table: guessbid.user

User Name Column: user\_email

Password Column: user\_password

Group Table: guessbid.user

Group Table User Name Column: user\_email

Group Name Column: user\_group

Password Encryption Algorithm: AES

Scroll down to find Charset, input UTF-8 and click Save.

### 3 DEPLOYMENT

---

Assuming that the previous steps are successfully done, we should be able to deploy the GuessBid web app. Restart the GlassFish server and open the admin console again.

From the left panel locate applications and click on Deploy. Select the included .war file and the application should be deployed.