dBModeler- Installation and Configuration

Version-1

|  |  |
| --- | --- |
| **Prepared By:** | Dhivya Shanmugaraj |
| **Technical Lead/Manager:** | Dhivya Shanmugaraj |
| **Dated:** | 21st January 2019 |
| **Review BY:** |  |
| **Review Date:** |  |

**dBModeler Installation and configuration Setup:**

1. **Tools to install:**
   1. Java JDK 5 or higher

* Ensure that JAVA\_HOME Environment variable is set to java installation directory (e.g. "Drive:\...\java\jdk1.x")
* Ensure that PATH Environment variable is extended with "\bin"-folder of java installation directory (e.g. "Drive:\....\java\jdk1.x\bin")
  1. Firefox Browser
  2. Tomcat 6 (unpack to arbitrary folder)
  3. Eclipse IDE for Java EE Developers
  4. Python version 2.5.2
* Just follow the install wizard. No special setup is needed here.
  1. PostgreSQL 8.3.x
* Use included wizard for installation
* Within the install wizard select "install postgreSQL as service"
* ATTENTION: The Procedural language dialog has to show PL/python (otherwise python installation failed). (Re-)Install PL/python!
* Ensure that PATH Environment variable is extended with "\bin"-folder of postgresql installation dir (e.g. "C:\program files\PostgreSQL\8.3\bin")
* IMPORTANT: remember the password for windows user postgres and database. You will need them later

1. **Source import:**

2.1) Import existing source into eclipse workspace, Select File->Import->Select Existing project into workspace->Next->Browse for the source and check copy projects into workspace->Finish.

1. **Database setup:**

3.1) Run command line as user postgres: type "run as /user:postgres cmd" in run-prompt in Windows. On the Keyboard press the Windows-key and R at the same time to get the run-prompt. To logon use windows user (postgres) password from postgreSQL installation.

3.2) change to schema file directory which is located in "dB Modeler-workspace/poem-jvm/data/database/db\_schema.sql"

3.3) Run the following commands to import schema:

* createuser -U postgres --echo --pwprompt --encrypted poem
* createdb -U postgres --echo --encoding utf8 --owner poem poem
* psql poem < db\_schema.sql postgres

1. **Change Build Properties:**

4.1) open the file build.properties in root dir

4.2) edit "deploymentdir" and set it to your apache tomcat wepapps folder (e.g. Drive:/apache-tomcat-6.0.32/webapps)

4.3) If you created a password for the poem user (see 3, bullet point 3.1), edit line "<property name="connection.password">poem</property>" in /poem-jvm/etc/hibernate.cfg.xml

1. **dB Modeler Build:**

5.1) Right-Click on "build.xml" in root dir and select "Run As -> External Tool Configuration"

5.2) Set buildfile to "build.xml", e.g. ${workspace\_loc:/oryx/build.xml}

5.3) Set base directory to oryx root directory, e.g. ${workspace\_loc:/oryx}

5.4) choose tab "Targets" and check the following targets: "build-all" and "deploy-all"

5.5) Start build by clicking on run-button.

5.6) If the build was successful, the two files "oryx.war" and "backend.war" should have been copied to your Apache Tomcat "\webapps" folder

1. **Server Start:**

6.1) If Tomcat was already running, it automatically re-deploys the war files.

1. **Start your browser and open** "<http://localhost:server_port/backend/poem/repository>".

7.1) Opening the backend enables users to create new process models or to browse/manage the process models that are already stored in dB Modeler. A double-click on a process model then opens the model in the frontend, where models can be modified.