Slenderware Domain Documentation

Slenderware

Web Developers - User Experience Designers - Project Management Platform Designers

Index

1. System Requirements:

NetBeans 

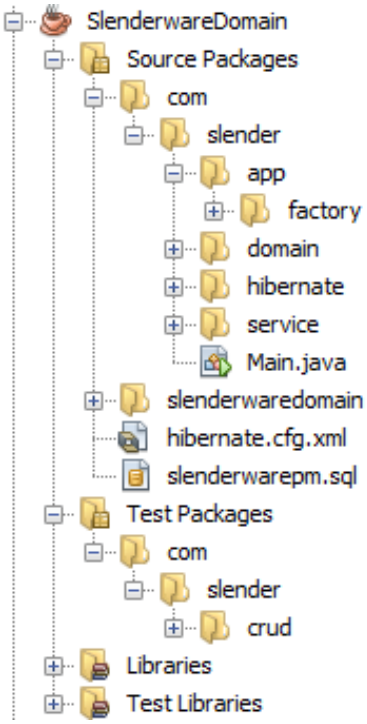
WAMP 

2. Opening the Project

All the source code can be found under this URL:
https://github.com/Slenderware/SlenderDomainServer

Once you have checked out the project from the Slenderware/SlenderDomainServer repository on NetBeans from Github

You will see the project and folder structure as the one below:



3. Folder Structure

Source Packages

Root folder

- hibernate.cfg.xml – This is the configuration file to map the domain classes with the tables in the MySQL database
- slenderwarepm.sql – This is the .sql full database export file for the slenderwarepm database

com.slender.app.factory - Contains all the factory classes that create our entities.

- AttachmentFactory.java
- CommentFactory.java
- CompanyFactory.java
- PriorityFactory.java
- ProjectFactory.java
- RoleFactory.java
- SessionFactory.java
- StatusFactory.java
- TaskFactory.java
- TaskTimeFactory.java
- UserProjectFactory.java
- UserTaskFactory.java
- UserFactory.java

com.slender.domain – Contains all the database entities

- Attachment.java
- Comment.java
- Company.java
- Priority.java
- Project.java
- Role.java
- Session.java
- Status.java
- Task.java
- TaskTime.java
- UserProject.java
- UserTask.java
- User.java

com.slender.hibernate – Contains the Hibernate files

- HibernateUtil.java
- hibernate.hbm.xml

com.slender.service.* – Contains the CRUD service interfaces and implementations

- Service.java

com.slender.service.crud

- AttachmentCrud.java
- CommentCrud.java
- CompanyCrud.java
- PriorityCrud.java
- ProjectCrud.java
- RoleCrud.java
- SessionCrud.java
- StatusCrud.java
- TaskCrud.java
- TaskTimeCrud.java
- UserProjectCrud.java
- UserTaskCrud.java
- UserCrud.java

com.slender.service.crud

- AttachmentCrudImpl.java
- CommentCrudImpl.java
- CompanyCrudImpl.java
- PriorityCrudImpl.java
- ProjectCrudImpl.java
- RoleCrudImpl.java
- SessionCrudImpl.java
- StatusCrudImpl.java
- TaskCrudImpl.java
- TaskTimeCrudImpl.java
- UserProjectCrudImpl.java
- UserTaskCrudImpl.java
- UserCrudImpl.java

Test Packages

com.slender.crud –Contains all the individual test case classes for the CRUD services and a test suite to execute all the test class's tests.

AttachmentCrudTest.java
CommentCrudTest.java
CompanyCrudTest.java
PriorityCrudTest.java
ProjectCrudTest.java
RoleCrudTest.java
SessionCrudTest.java
StatusCrudTest.java
TaskCrudTest.java
TaskTimeCrudTest.java
UserProjectCrudTest.java
UserTaskCrudTest.java
UserCrudTest.java
CrudTestSuite.xml

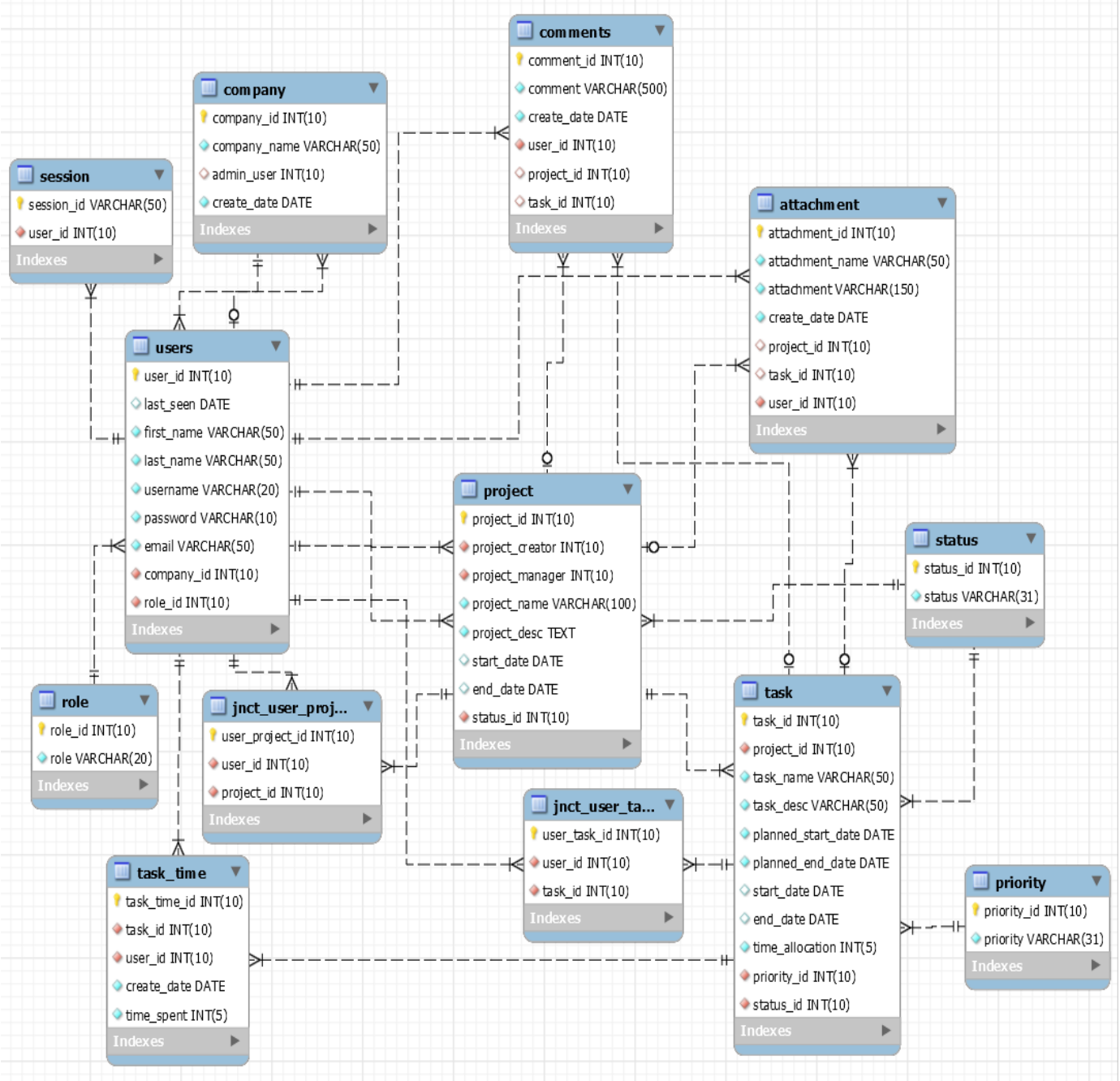
4. Libraries required

Hibernate JPA
Hibernate
TestNG 6.5.1 –testng-6.5.1.jar
Javax.faces-2.1.7.jar
Mysql-connector-java-5.1.18.bin.jar

5. Creating the Database

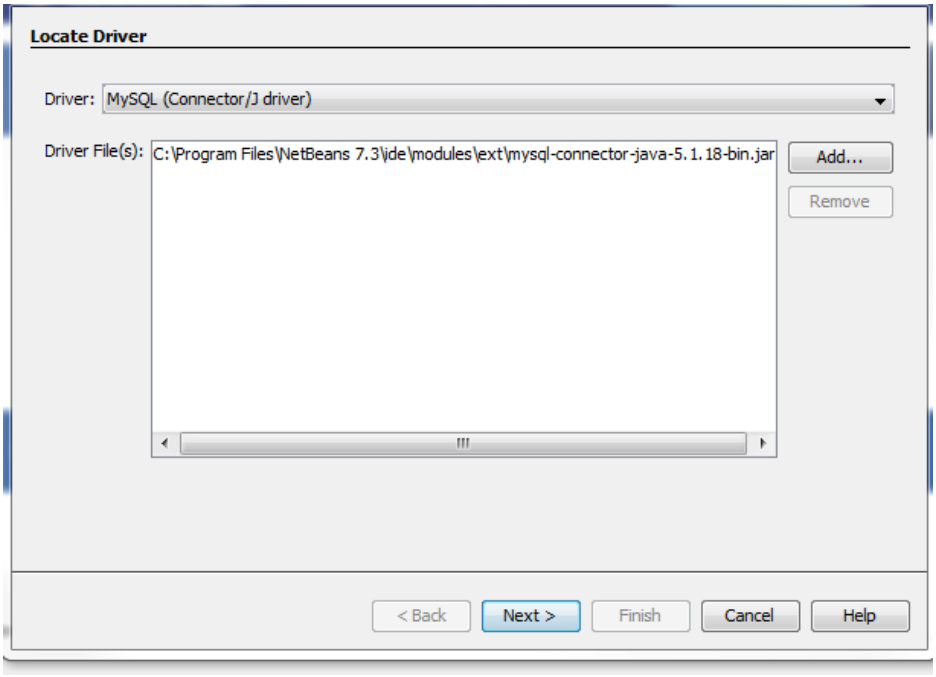
1. Start WAMP Server
2. Open PHPMyAdmin
3. Create a new database and call it gdqmxhyb_dev_slen
4. Click on the *Import* tab
5. Click on *Choose File* and locate the slenderwarepm.sql file found under your projects Source Packages or under this URL: https://github.com/Slenderware/SlenderDomainServer/blob/master/src/slenderwarepm.sql
6. Then click *Go*
7. You should then see the gdqmxhyb_dev_slen database connection on your left next to your other database connections
8. Lastly create a user by clicking on the *Users* tab and then click *Add User*
9. Create a user with the username: 'gdqmxhyb_slen', password: 'poesw4#32EWa32', localhost as the host, and tick all under Global Privileges.
10. Then click Go and you have completed creating your Database and user.

6. SlenderwarePM Database ERD

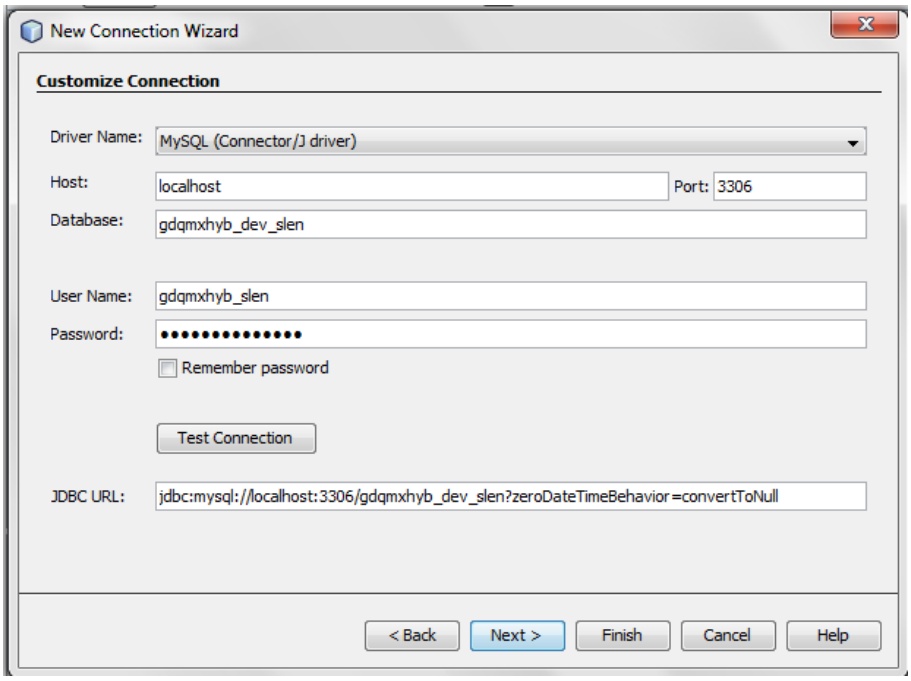


7. Connecting NetBeans to the SlenderPM MySQL Database

1. Start WAMP Server if you have not already
2. Click on the *Services* tab in NetBeans
3. Right Click on *Databases* and select *New Connection*
4. Select MySQL (Connector/J Driver) and click *Next*

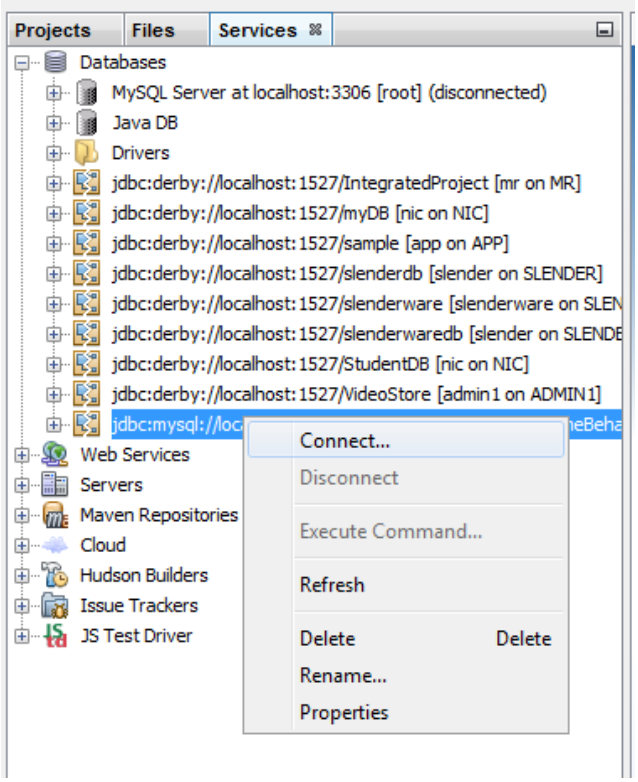


5. Fill in 'gdqmxhyb_dev_slen' as the Database name, 'gdqmxhyb_slen' as the User Name, and the password 'poesw4#32EWa32'



6. Select Next at Choose Database Shema
7. Type in a name for your connection and click Finish.
8. You will then see your SlenderwarePM database connection on your left under your databases

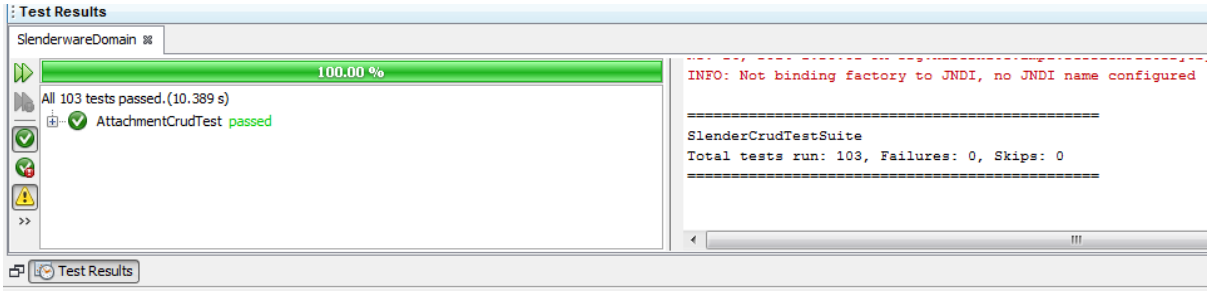
9. Right click on your connection and click *Connect*



10. You have now successfully connected to your Slenderware PM MySQL database on Netbeans.

8. Running Test Cases and Completion

1. To Test and make sure that everything is working and you have successfully completed the installation, you will need to run the test cases.
2. Locate and run the CrudTestSuite under com.slender.crud under the Test Packages in the Project or run an individual entity's test cases.
3. If all 103 Tests are pass successfully then you have completed the installation process successfully.



9. CRUD Service Functions

findById
Return type: Entity Object
Parameters: int
Description: This function returns the entity from the database with the matching ID you passed as a parameter.

findAll
Return type: List<Entity Objects>
Parameters: none
Description: This function returns a list of all the entities in the database table.

persist
Return type: Entity Object
Parameters: Entity Object
Description: This function persists the entity passed as a parameter into the database and returns the entity that was just persisted with its auto generated ID.

merge
Return type: void
Parameters: Entity Object
Description: This function passes an entity as a parameter and updates that entity in the database.

remove
Return type: void
Parameters: Entity Object
Description: This function removes the entity passed as a parameter from the database.

removeById
Return type: void
Parameters: int
Description: This function removes the row from the database table that matches the ID passed as a parameter.

count
Return type: int
Parameters: none
Description: This function returns the number of rows/entities there are in the database table.

getByPropertyName

Return type: Entity Object

Parameters: String name, String/int value

Description: This overridden function allows you to search for an entity by a property/column name in the database table. It returns the entity retrieved from the database table that matches its property/column name with the value passed as an input parameter.

getEntitiesByProperName

Return type: List<Entity Objects>

Parameters: String name, String/int value

Description: This overridden function allows you to search for entities by a property/column name in the database table. It returns a list of entity retrieved from the database table that matches its property/column name with the value passed as an input parameter.