



**POLITECNICO
DI MILANO**

TravelDream

GUIDA ALL' INSTALLAZIONE

Bresciani Giacomo
Cazzaniga Gian Piero
Colombo Giacomo

Matr. 804979
Matr. 804359
Matr. 820029

Corso di Laurea Specialistica in Ingegneria Informatica
Docente: Luca Mottola

SOMMARIO

1. INTRODUZIONE	3
1.1. COMPONENTI NECESSARI	3
1.2. INSTALLAZIONE GLASSFISH E MY-SQL CONNECTOR	3
2. CREAZIONE DEL DATABASE	4
3. CONFIGURAZIONE GLASSFISH	6
3.1. CONFIGURAZIONE JDBC CONNECTION POOL	6
3.2. CONFIGURAZIONE JDBC RESOURCES	8
3.3. CONFIGURAZIONE JDBC REALM	9
4. INTEGRAZIONE DELLA PIATTAFORMA IN GLASSFISH	10
5. NOTE FINALI	10

1.INTRODUZIONE

TravelDream è sviluppata come applicazione per Glassfish 4.0 e include un database MySQL per la gestione dei dati. Nella guida illustrata in questo testo si ipotizza di utilizzare il server e il database su rete locale.

1.1. COMPONENTI NECESSARI

Per poter iniziare con l'installazione dell'applicazione e utilizzare il sistema bisogna prima procurarsi e installare alcuni componenti:

- Glassfish 4.0

scaricabile all'indirizzo <http://glassfish.java.net/>

Il server Oracle GlassFish, offre il supporto per la costruzione e lo sviluppo dell'applicazione

- MySql Server 5.1

scaricabile all'indirizzo <http://mysql.com/>

Database Oracle per l'archivio dei dati dell'applicazione

- Driver JDBC, per potersi connettere al database mysql
- Browser web

1.2. INSTALLAZIONE GLASSFISH E MY-SQL CONNECTOR

Come primo passo è necessario avviare l'installazione di Glassfish 4.0.

Per poter configurare il collegamento con l'applicazione viene fornito il driver "MySql-Connector-java-5.1.27", che andrà copiato nella cartella di installazione di Glassfish, nel percorso: glassfish4/glassfish/lib.

2.CREAZIONE DEL DATABASE

Per la creazione del database si deve avviare MySql Server. Dopo aver installato correttamente tutti i file del pacchetto MySql sul nostro pc, apriamo il prompt dei comandi e digitiamo la stringa:

```
mysql -u NomeUtente -p
```

per nome utente si intende lo username da voi dato durante la creazione dell'utente, al termine dell'installazione di MySql.

Analogamente l'avvio può essere effettuato dalla MySql Command Line Client, compresa nel pacchetto di installazione, sempre inserendo la password.

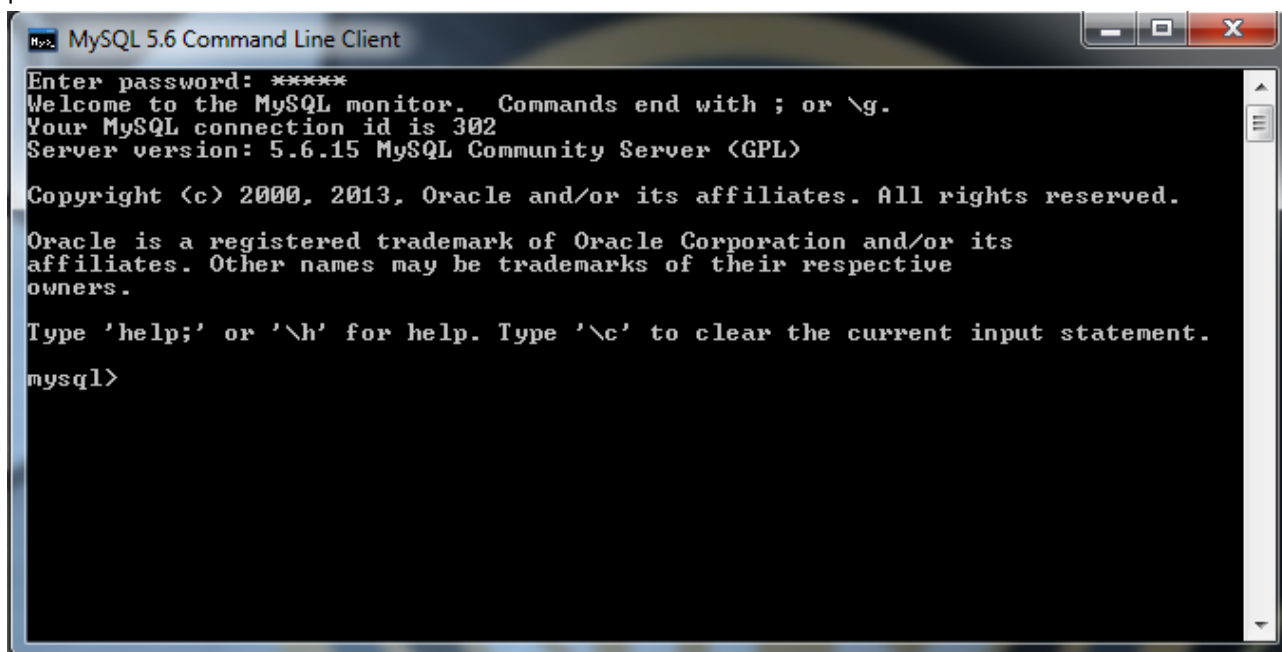


Figura 1 – Schermata MySql

- Per creare il database basta digitare il comando:
create database TravelDreamDB;
- Per creare un utente in qualsiasi momento basta digitare il comando:
create user 'utente'@'localhost' identified by 'password';
- Per abilitare all'utente all'uso del database creato:
grant all on TravelDreamDB.* to 'NomeUtente'@'localhost'
identified by 'password';

- Per controllare che il database sia stato creato correttamente bisogna lanciare il comando
`show databases;`

e verificare che TravelDreamDB sia presente nella tabella.

L'ultimo passo è quello di riempire il database importando lo schema sql fornito nel pacchetto, creando in questo modo le tabelle necessarie per il funzionamento del sistema.

- Per importare lo schema è necessario digitare nel prompt il comando:
`source C:\insert_default.sql`

seguito come mostrato dal percorso completo in cui abbiamo messo il file con estensione .sql fornito.

3.CONFIGURAZIONE GLASSFISH

Per l'utilizzo dell'applicazione configuriamo Glassfish attraverso la sua console di amministrazione.

L'interfaccia grafica del server, visibile in Figura 2, è raggiungibile all'indirizzo: <http://localhost:4848/common/index.jsf>

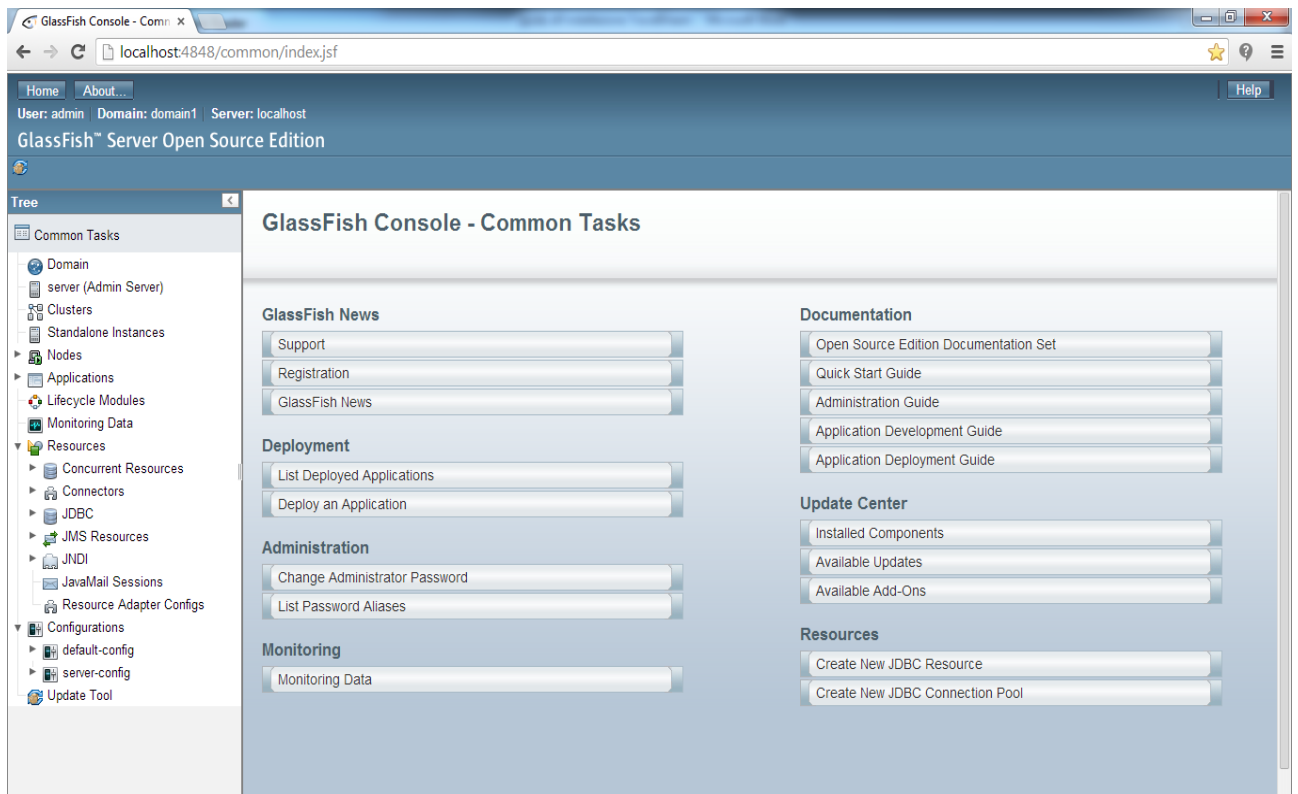


Figura 2 – Glassfish console

3.1. CONFIGURAZIONE JDBC CONNECTION POOL

Creiamo prima una JDBC Connection Pool, andando nella colonna Common Tasks su *Resources/JDBC/JDBC Connection Pool* e cliccando su "New". Compiliamo i campi come indicato in figura:

General
Advanced
Additional Properties

Edit JDBC Connection Pool

Modify an existing JDBC connection pool. A JDBC connection pool is a group of reusable connections for a particular database.

Load Defaults
Flush
Ping

General Settings

Pool Name:

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

Datasource Classname:
Vendor-specific classname that implements the DataSource and/or XADataSource APIs

Driver Classname:
Vendor-specific classname that implements the java.sql.Driver interface.

Ping: ☒ Enabled
When enabled, the pool is pinged during creation or reconfiguration to identify and warn of any erroneous values for its attributes

Deployment Order:
Specifies the loading order of the resource at server startup. Lower numbers are loaded first.

Description:

Pool Settings

Initial and Minimum Pool Size: Connections
Minimum and initial number of connections maintained in the pool

Maximum Pool Size: Connections
Maximum number of connections that can be created to satisfy client requests

Pool Resize Quantity: Connections
Number of connections to be removed when pool idle timeout expires

Idle Timeout: Seconds
Maximum time that connection can remain idle in the pool

Max Wait Time: Milliseconds

Figura 3 – JDBC Connection Pool

Successivamente cliccare nelle Tab in alto su Additional Properties e settare tutti i campi come in Figura:

Edit JDBC Connection Pool Properties

Modify properties of an existing JDBC connection pool.

Pool Name: traveldream_connection

Additional Properties (5)				
<div> Add Property Delete Properties </div>				
Select	Name	Value	Description	
<input type="checkbox"/>	User	<input type="text" value="Username"/>	<input type="text"/>	
<input type="checkbox"/>	ServerName	<input type="text" value="localhost"/>	<input type="text"/>	
<input type="checkbox"/>	Port	<input type="text" value="3306"/>	<input type="text"/>	
<input type="checkbox"/>	DatabaseName	<input type="text" value="TravelDreamDB"/>	<input type="text"/>	
<input type="checkbox"/>	Password	<input type="text" value="password"/>	<input type="text"/>	

Figura 4 – Proprietà del JDBC Connection Pool

3.2. CONFIGURAZIONE JDBC RESOURCES

Creiamo il JDBC Resources che si trova sempre nella cartella *Resources* dei Common Tasks, dandogli un nome e collegandolo con il JDBC Connection Pool appena creato, quindi utilizzando lo stesso nome per la pool.

Edit JDBC Resource

Edit an existing JDBC data source.

[Load Defaults](#)

JNDI Name:

Pool Name:

Use the JDBC Connection Pools page to create new pools

Deployment Order:

Specifies the loading order of the resource at server startup. Lower numbers are loaded first.

Description:

Status: ☒ Enabled

Additional Properties (0)

[Add Property](#)

[Delete Properties](#)

Select	Name	Value	Description
No items found.			

Figura 5 – JDBC Resources

3.3. CONFIGURAZIONE JDBC REALM

Per poter utilizzare le funzioni di sicurezza dell'applicazione creiamo il nostro JdbcRealm andando su "*Configurations/server-config/Security/Realms*" nella colonna dei Common Tasks, e clicchiamo su "*New*" per crearne uno nuovo.

Diamo un nome al nostro Realm (nel nostro caso è travelDreamJdbcRealm) e impostiamo tutti i campi come mostrato in figura. I campi non visibili nella figura sono da lasciare vuoti.

Edit Realm
Edit an existing security (authentication) realm.

Configuration Name: server-config

Realm Name: travelDreamJdbcRealm

Class Name: com.sun.enterprise.security.ee.auth.realm.jdbc.JDBCRealm

Properties specific to this Class

JAAS Context: *	jdbcRealm Identifier for the login module to use for this realm
JNDI: *	jdbc/traveldream_connection JNDI name of the JDBC resource used by this realm
User Table: *	users Name of the database table that contains the list of authorized users for this realm
User Name Column: *	username Name of the column in the user table that contains the list of user names
Password Column: *	password Name of the column in the user table that contains the user passwords
Group Table: *	users_groups Name of the database table that contains the list of groups for this realm
Group Table User Name Column:	username Name of the column in the user group table that contains the list of groups for this realm
Group Name Column: *	groupname Name of the column in the group table that contains the list of group names
Password Encryption Algorithm: *	MD5 This denotes the algorithm for encrypting the passwords in the database. It is a security risk to leave this field empty.

Figura 6 – JDBC REALM

4. INTEGRAZIONE DELLA PIATTAFORMA IN GLASSFISH

Per poter far girare l'applicazione fornita sul server Glassfish si potrà utilizzare sempre la facile interfaccia grafica della console di amministrazione di Glassfish.

Nella schermata di home cliccheremo su *Deploy an Application* e andremo a caricare sul server il pacchetto applicativo fornito. In questo modo la piattaforma sarà pronta per essere utilizzata.

5. NOTE FINALI

Per un corretto utilizzo della piattaforma fornita si consiglia di leggere il manuale d'uso fornito.