

Fejlesztői dokumentáció

Szoftverfejlesztés J2EE platformon tárgy házi feladat

Autókölcsönző webalkalmazás

Kiss Balázs DCZUT 2016.december

Tartalomjegyzék

Alkalmazás rövid elírása.....	3
Futtatási környezet.....	3
Alkalmazás telepítése.....	3
Adatbázis.....	4
Alkalmazás szerkezet.....	5
Felhasználói felületek, és működésük.....	6
Csatolmányok:.....	7

Alkalmazás rövid elírása

A webalkalmazás segítségével az adminisztrátor felvihet autókat és felhasználókat az adatbázisba. A felhasználók belépés után tudnak saját maguk számára autókat kölcsönözni. Az adminisztrátori felület csak admin számára érhető el.

Futtatási környezet

Az alkalmazás eléréséhez a következőkre van szükségünk:

- Javascriptet támogató böngésző, pl. aktuális Google Chrome 55.0

Az alkalmazás fejlesztéséhez szükséges eszközök:

- JDK 1.8 java verzió, tetszőleges IDE
- Glassfish 4.0 a használt webalkalmazás szerver.
- MySQL 5.6 az adatbázis verzió.
- SOAP UI 5.0 a Webservice teszteléséhez.
- Maven 3.3.1 az automatikus dependencia feoldásokhoz.
- Egyéb hasznos java api-k, mint pl. Lombok, Primefaces, stb.

A projekt elérhetősége: <https://github.com/KissBalazs/j2eehazi>

Alkalmazás telepítése

1. Glassfish telepítése.

Töltsük le a Glassfish-t a <https://glassfish.java.net/> oldalról. Fontos, hogy ne a Web Profile-t, hanem a teljes verziót töltsük le. (ennek ellenőrzése a futó szerver esetén `./asadmin list-containers` parancs kiadása után futnia kell a webservice konténernek.)

A projekt setupFiles mappájában található a mySql konnector driver. Bontsuk ki, és a mappában található `.jar`-t másoljuk be a `/glassfish4/glassfish/domains/domain1/lib` mappába.

Indítsuk el a szerveret. Glassfish4/bin mappában `./asadmin start-domain` (ha debugolni is szeretnénk, akkor `-debug` kapcsolóval)

Ezek után hozzuk létre az connection pool-t:

```
./asadmin create-jdbc-connection-pool --restype  
javax.sql.DataSource --datasourceclassname  
com.mysql.jdbc.jdbc2.optional.MysqlDataSource --property  
user=jee:password=jee:DatabaseName=jeehazi:ServerName=localhost:po  
rt=3306 PM-Pool
```

Majd hozzuk létre a resource-ot:

```
./asadmin create-jdbc-resource --connectionpoolid PM-Pool
```

jdbc/jeeDS

2. Glassfish Role-ok konfigurálása

Glassfish adminisztrátor konzolban <http://localhost:4848/common/index.jsf> a

Configurations/server-config/Security fülön pipáljuk be a “Security Manager”, “Audit Logging”, és “Default Principal To Role Mapping” értékeket.

Ez után a Security/Realms/File realm részen az “Assign Groups”-hoz írjuk be a default role-t: “Users”

Hozzunk létre a Manage Users gombbal két usert: egyik neve legyen mondjuk “simpleuser”, role-ja legyen üresen hagyva. Másik user legyen “adminuser”, role-ja legyen “Admin”

A “Users” és “Admin” role-ok fontosak, ezek vannak felmappelve az alkalmazásban is, de felhasználót tetszőlegesen hozhatunk létre.

3. Alkalmazás deploy-olása

Az alkalmazást kedvünk szerint deploy-olhatjuk beépülő IDEA maven plugin segítségével (a repositoryban található .idea mappában le van mentve a profil), vagy kézzel is lefordíthatjuk, majd az elkészült .WAR fájlt kell deploy-olni a webalkalmazás “Applications” menüjében.

Adatbázis

Kapcsolódás, létrehozás

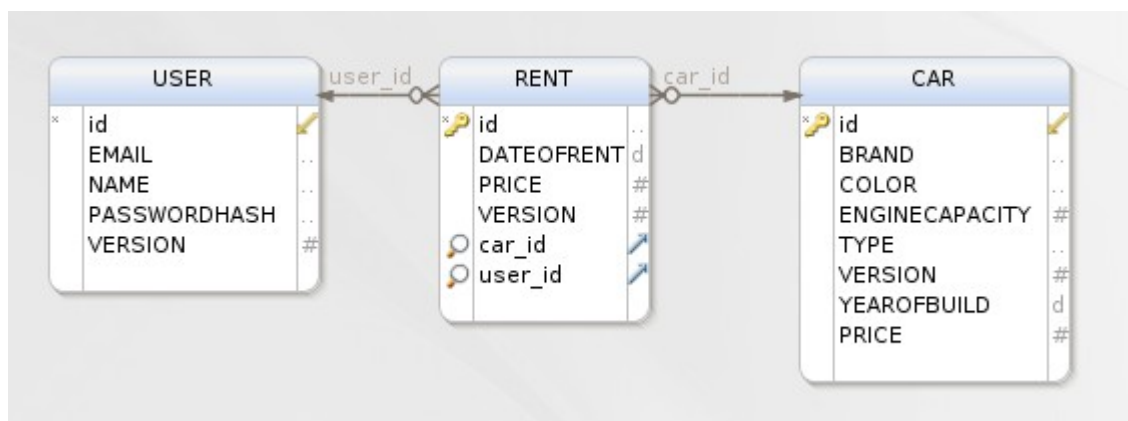
Az adatbázist a következő paraméterekkel hozzuk létre (a datasource definíciói alapján):

- Database név: “jeehazi”
- User:”jee”
- Password: “jee”
- url: jdbc:mysql://localhost:3306/jeehazi

Adatszerkezet

Adatbázis táblák létrehozására nincs szükség, a persistence.xml megfelelő konfigurálásával az entitások alapján létrejön a struktúra. A generált DDL megtalálható a Documents/db_ddl.sql fájlban is.

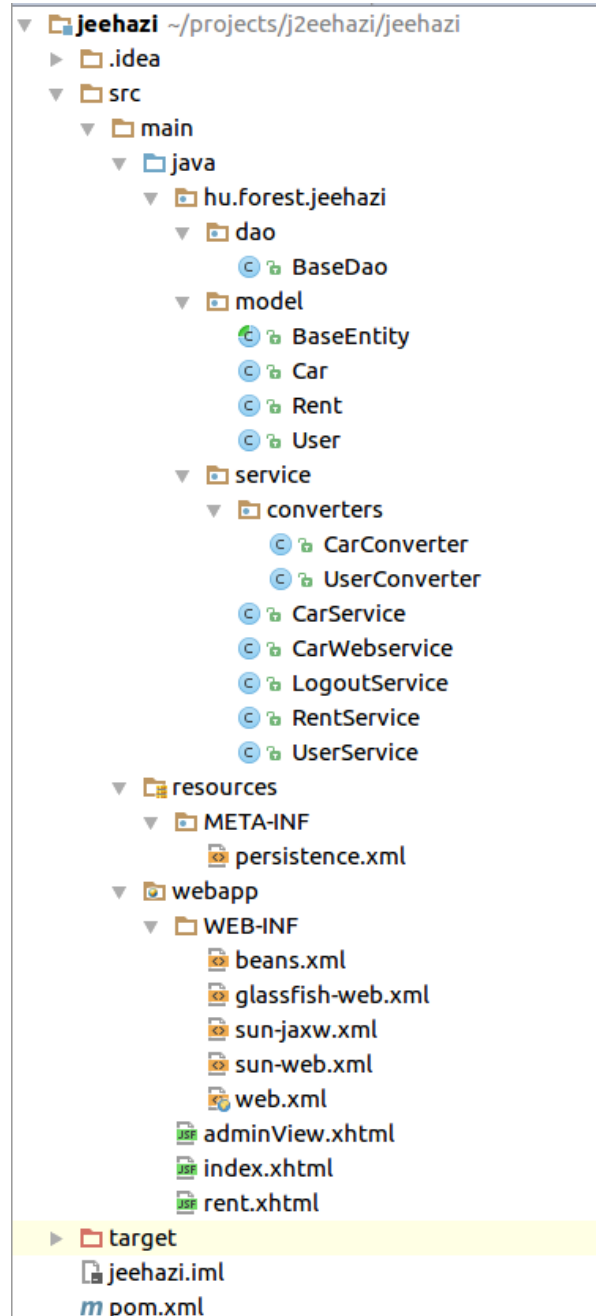
Összesen három adatbázis entitást használunk, ezért négy tábla is lesz: CAR, USER és a kapcsolótábla köztük a RENT.



Alkalmazás szerkezet

Az alkalmazás a következőképp épül fel:

- Dao package:
 - BaseDao: alap adatbázis perzisztencia osztály.
- Model package:
 - BaseEntity: alap entitás osztály, absztrakt, csak összegyűjti a főbb attribútumokat, és a többi entitás belőle származik le.
 - Car: autók entitás
 - User: felhasználók entitás
 - Rent: kölcsönzések entitás
- Service package:
 - converters package: az autó- és felhasználó névfeloldása entitássá a legördülő menükhöz.
 - CarService, UserService, RentService: az adott entitások létrehozásához, mentéséhez, törléshez, listázásához használt osztályok.
 - CarWebservice: autókat listázó webservice.
 - LogoutService: BaseAuthentication-höz kijelentkezés megoldása.
- Resources: a webalkalmazás belső működéséhez szükséges fájlok.
 - Persistence.xml: a JPA konfigurálásához szükséges beállítások.
- Webapp: a frontendhez tartozó statikus fájlok
 - *.xhtml: XHTML weboldalak, melyekből a statikus tartalom generálódik JSF segítségével.
 - WEB-INF: különböző szolgáltatásokhoz tartozó konfigurációs xml-ek
 - beans.xml – nem kötelező üresen, de a CDI működését inicializálja
 - glassfish-web.xml: role-ok automatikus mappelése glassfishben
 - sun-jaxw.xml: ahhoz, hogy a webservice elérhető legyen basic authentication



segítségével

- sun-web.xml: általános role mappelés
- web.xml: javaEE és kapcsolódó beállítások

Az alkalmazás kódbázisának külön dokumentálását itt nem teszem meg, a feladhoz mérten a kódban helyeztem el a szükséges helyeken megértést segítő kommenteket.

Felhasználói felületek, és működésük

1. Menü


Az alkalmazás megnyitása után a menü fogad minket. Az adminisztrátor számára egyel több menüpont érhető el.


JEE házifeladat

Autókölcsönző webalkalmazás

Kiss Balázs - DCZUKT

Menü - simpleuser

 Kölcsönzés


 Kijelentkezés


JEE házifeladat


Autókölcsönző webalkalmazás

Kiss Balázs - DCZUKT

Menü - adminuser

 Kölcsönzés

 Admin felület

 Kijelentkezés

2. Admin felület

Vissza a menübe

User felvétele

Név: Forest Jelszó: Email: Mentés

Autó felvétele

Gyártó: Típus: Iszin: Gyártás éve: Motor térfogat: Ár: Mentés

Új kölcsönzés

Felhasználó: Válassz! Auto: Válassz! Ár szorzó: Mentés

Adatbázis:

ID	userName	email	Törtés
1666dfa5-7330-44d5-85a2-9a1c89726a4e	Forest	a@a	
4317349b-3c2d-4ba8-8e9e-b5a3378e207c	adminuser		
8b006201-fa98-4e70-ae05-7dc364f00a11	Peti	nincs	
8bde4b24-6309-4068-9448-3e5467c2504f	Kovács Pista	nincs	
aa7f7074-5e01-45d3-b494-4311c019788b	simpleuser		

ID	brand	type	engineCapacity	price	yearOfBuild	Törtés
61a81088-bbcd-4875-9ea4-24140fa9930	Ford	Escort	2200	5	Wed Dec 07 00:00:00 CET 2016	
83e24fb-75a8-4f0f-9e98-4c7d364a02a3	Skoda	Niva	500	7	Wed Sep 14 00:00:00 CEST 2016	
a568821-7e2b-46a2-ad33-fac9998bbad	Renault	Thalia	1000	10	Tue Sep 06 00:00:00 CEST 2016	

ID	carBrandID	carType	userType	price	date	Törtés
0f4530b-c021-43bb-8f5b-a877e3026454	Ford	Escort	adminuser	100	Tue Dec 06 00:00:00 CET 2016	
6da8d305-c347-41b9-9f14-ca7a0c0215a62	Renault	Thalia	simpleuser	210	Tue Dec 06 00:00:00 CET 2016	
700c90f-4000-4e7c-9f5d-e3c1503457ba	Renault	Thalia	adminuser	20	Tue Dec 06 00:00:00 CET 2016	
745e2a01-3a94-4468-9803-e3209a5d3a45	Ford	Escort	Forest	10000	Mon Dec 05 00:00:00 CET 2016	
8b148dc-d941-463b-a7f3-e3af5428577	Ford	Escort	Forest	10	Tue Dec 06 00:00:00 CET 2016	

Az admin felületen a következő funkciókat érhetjük el:

- Új felhasználó hozzáadása
- Új autó hozzáadása
- Új kölcsönzés összeállítása
- Adatbázis táblák megtekintése, bejegyzések törlése.

3. Kölcsönzés felület

Vissza a menübe

Autók					
Azonosító	Gyártó	Típus	Hengerirtartalom cm ³	Alap ár	Gyártás éve
> 61a81088-bbcd-4875-9ea4-24140fa99930	Ford	Escort	2200	5	7
> 83e24fb-75a8-40f8-9e98-fc7b364a02a3	Skoda	Niva	500	7	14
> a568621-7e2b-46a2-add3-face9960bab	Renault	Thalia	1000	10	6

Új kölcsönzés

Felhasználó: adminuser Autó: Ár szorozó:

Kölcsönzéseim:

Kölcsönzések					
Kölcsönzés azonosító	Kölcsönzött autó gyártója	Kölcsönzött autó típusa	Felhasználónév	price	date
0fa4530b-c021-43bb-8f5b-a877e3026454	Ford	Escort	adminuser	100	Tue Dec 06 00:00:00 CET 2016
700c90f-4000-4e7c-9f5d-e3c1503457ba	Renault	Thalia	adminuser	20	Tue Dec 06 00:00:00 CET 2016

A kölcsönzés felületen a következőket tehetjük meg:

- Autók listájának megtekintése, szűrés rájuk különböző paraméterek szerint
- Új kölcsönzés leadása
- Saját kölcsönzések megtekintése

4. Webservice felület

Az alkalmazás megvalósít egy webservice-t, ami kilistázza a rendszerben létező autókat.

Elérése deploy után a <http://localhost:8080/jeehazi-0.1-SNAPSHOT/CarWebserviceService?wsdl> felületen érhető el.

A

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ser="http://service.jeehazi.forest.hu/">
  <soapenv:Header/>
  <soapenv:Body>
    <ser:listCars/>
  </soapenv:Body>
</soapenv:Envelope>
```

kérést leadva hasonló eredményt kell hogy kapjunk az adatbázisunk állapotától függően:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:listCarsResponse
xmlns:ns2="http://service.jeehazi.forest.hu/">
      <return>Car(brand=Ford, type=Escort, color=piros,
price=5, engineCapacity=2200, yearOfBuild=Wed Dec 07 00:00:00 CET
2016)</return>
      <return>Car(brand=Skoda, type=Niva, color=fekete,
```

```
price=7, engineCapacity=500, yearOfBuild=Wed Sep 14 00:00:00 CEST
2016)</return>
    <return>Car(brand=Renault, type=Thalia, color=Kék,
price=10, engineCapacity=1000, yearOfBuild=Tue Sep 06 00:00:00
CEST 2016)</return>
    </ns2:listCarsResponse>
</S:Body>
</S:Envelope>
```

A teszteléshez SOAP UI-t használtam.

Csatolmányok:

A Projekthez a következő csatolmányok tartoznak a repositoryban:

- documents/Fejlesztői dokumentáció: ez a dokumentum.
- Documents//Kiss Balázs DCZUKT Házi feladat specifikáció Szoftverfejlesztés J2EE platformon.docx: a kezdeti specifikáció a feladathoz.
- fejlesztoi_jegyzet.txt: saját jegyzetek fejlesztés során.
- SetupFiles/mysql-connector-java-5.1.40.zip: mysql konnektor glassfish-hez.