

TravelDream Project

Implementation

Riccardo B. Desantis - matr. 765106

1 Purpose

This document describes the differences between the implementation and the proposed solution, and eventually the unresolved bugs found.

2 Differences

I've simplified the tables, eliminating the `id` column when possible and using the `name` of the element instead.

I've followed strictly all the other details in the RASD and DD documents, or at least no other important detail has been changed. Surely, some details about the design of the webpage is changed, but all the things changed are trivial.

3 Instructions

To make the system work, we'll need:

1. to configure the connection to the database, and the realm for the authentication;
2. to create the table the first time, and this is doable modifying the file:

`ejbModule/META-INF/persistence.xml`

in the EJB component by selecting to *"drop and create the tables"*, and generate all the tables from the JPA Entities via the menu;

3. to initialize the data in the database there's a SQL file that you should run: it's

`ejbModule/META-INF/init_entries.sql`

4. there are two users already registered, and they are:
 - `effetti@gmail.com`: password password, role employee;
 - `lino@gmail.com`: password password, role customer.

4 Configuration

My actual configuration for Glassfish 4.0 is:

- the connection pool is called `TravelDream_connection`
- the jdbc resource is `jdbc/traveldream_connection`
- the authentication realm has these details:
 - Realm Name: `authJdbcRealm`
 - JAAS Context: `jdbcRealm`
 - JNDI: `jdbc/traveldream_connection`
 - User Table: `Users`
 - User Name Column: `EMAIL`
 - Password Column: `PASSWORD`
 - Group Table: `Nomrole`
 - Group Name Column: `NAME`
 - Password Encryption Algorithm: `MD5`
 - Assign Groups: `employee,customer`
 - Digest Algorithm: `SHA-512`

5 Tests

I've tried doing a lot of things without problems:

- login to the system (with correct data and not) and logout;
- sign up;
- list packages (with filtering) and show details;
- creation of a custom package for a customer;
- listing, modification and elimination of a custom package for the customer that created it;
- list components for the employee;
- adding of a new component for an employee, where if we put a different country and city in the form they'll be handled automatically;
- removing a component or a package for an employee;
- etc.

6 Bugs

Sadly solving some bugs took away a lot of time, and I didn't make it in time. Anyway, the only other things I've found that can't be solved easily are:

- **the system can't distinguish between the two roles:** that means that both roles are the same for it, so I implemented a direct check that doesn't show unallowed elements at all;
- weird behavior happens when the database is empty.