LU04b - Benutzer

Benutzer

Anforderung 5: Es soll erfasst werden können, wer welches Buch ausgeliehen hat.

Dazu werden Benutzer benötigt. Erstellen Sie eine entsprechende Klasse `User` mit folgenden Attributen und verwenden Sie dabei die Hibernate-Annotationen analog der Klasse `Book`:

- id
- firstname
- lastname
- dateOfBirth
- email
- passwordHash
- passwordSalt

Als nächstes benötigen wir eine Möglichkeit Benutzer anzulegen und abzuspeichern. Erstellen Sie dafür in ihrer Applikation den Befehl `createUser …` gefolgt von den Werten (ausser id, Salt und Hash). Der Commit `bd45e4c` enthält die nötigen Anpassungen an `persistence.xml` und die Tests.

Anstatt dem Salt und dem Passwort-Hash wird das Passwort im Klartext angegeben. Der Salt kann beim Erstellen eines neuen Benutzers folgendermassen zufällig generiert werden. Mit dem Salt und dem übergebenen Klartextpasswort wird der Hash berechnet, welcher im Gegensatz zum Klartextpasswort, in der Datenbank gespeichert wird.

```
    public static byte[] generateSalt() {
        byte[] salt = new byte[16];
        new SecureRandom().nextBytes(salt);
        return salt;
    }

    public static String hashPassword(String password, byte[] salt) throws
NoSuchAlgorithmException {
        MessageDigest md = MessageDigest.getInstance("SHA-256");
        md.update(salt);
        byte[] hashed = md.digest(password.getBytes());
        return Base64.getEncoder().encodeToString(hashed);
    }
```

Überprüfen Sie, ob folgende Eingabe einen Benutzer anlegen und in die Datenbank speichern kann.

```
createUser Max Mustermann 1990-05-21 max.mustermann@example.com geheim123
```

Überprüfen Sie Ihren Code ebenfalls mit dem JUnit-Test.

From:
https://wiki.bzz.ch/ - **BZZ - Modulwiki**

Permanent link:
https://wiki.bzz.ch/de/modul/ffit/3-jahr/java/learningunits/lu04/b?rev=1757376693

Last update: **2025/09/09 02:11**