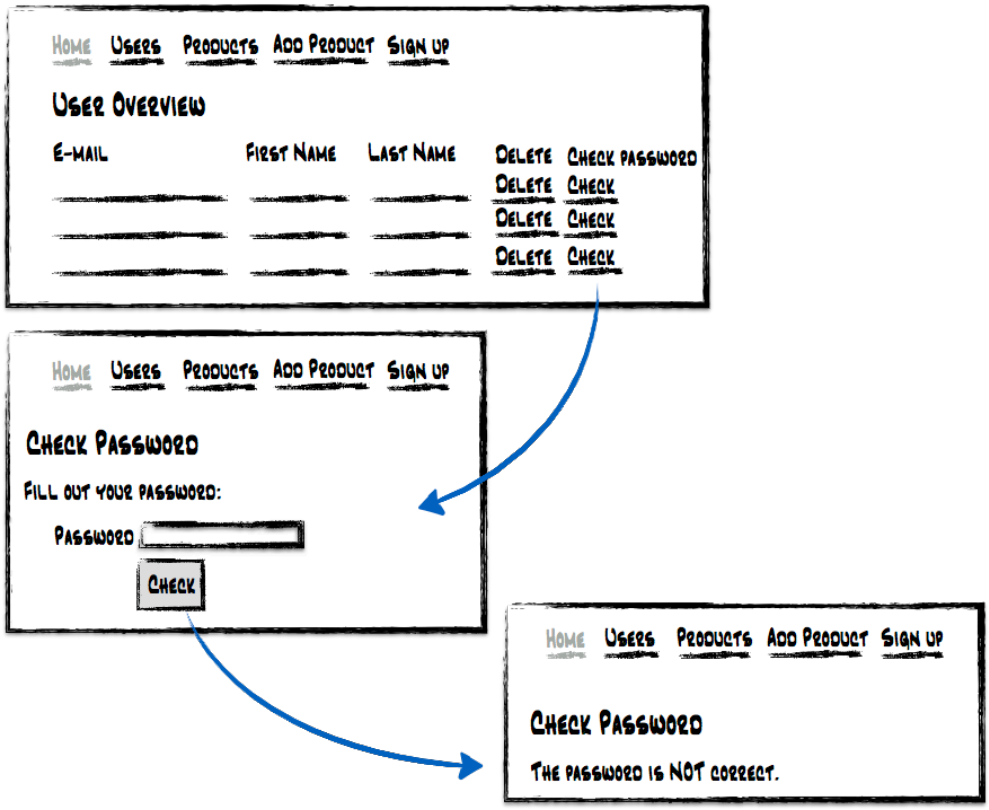
STORY W11. CHECK PASSWORD

As a user
I want to check my password
so that I am assured that it is stored correctly.

WIREFRAMES:



TECHNICAL DETAILS

Make sure the passwords are hashed when you store them in the database:

+ Person
-password : String -salt : String
+setPassword(password : String) +setPasswordHashed(password : String) -hashPassword(password : String) : String +isPasswordCorrect(password : String) : boolean +setSalt(salt : String) : void +getSalt() : String

- Change the `person` table: add 2 columns: one for the password, one for the salt. Each column has the type *character*, with a length of 128 for the password and 40 for the salt.
- Change the class `Person`: add a new setter which will first hash the given password before setting the instance variable. Use this setter in your controller-class.

The existing setter will only be used by the database class when a person object is retrieved from the database: it will replace the instance variable with the hashed password from the database. This method does not need to hash the password again.

Remark: the salt is generated and used as a byte array, but if you want to store it in the database, it is easier to use a String. To convert to a String and back to a byte array, use following methods:

- `String salt = new BigInteger(1, seed).toString(16);`
- `byte[] seed = salt.getBytes("UTF-8");`

ACCEPTANCE CRITERIA:

- When user clicks on the link “Check” in the person overview page, he/she will be asked to enter his/her password. A message is shown if the password is correct or not.