TP01: Introduction to the problems of multi-layered applications

0 – Getting started:

Follow the instructions under “Prog setup” in Section 1 in Cyberlearn.
Download the file “TP01-TESTU-POJO-DAO-E-donnee.zip” from Section 1 in cyberlearn and unzip in the folder used for the Eclipse workspace: /rootJavaEE/Labs/
Add it in the Project Explorer in Eclipse using New > Project… > Java Project (the Import… wizard may create problems). Then uncheck the box “Use default location” and use the button “Browse” to locate your unzipped project (e.g., Z:\rootJavaEE\Labs\TP01-TESTU-POJO-DAO-E-donnee) and click “Finish”.

I – Preliminary remarks:

The application is a simplified banking application. It allows to list all clients of a bank and to manage the accounts (transfers).

The implementation respects a multilayer architecture:

- **Service layer** (ch.hevs.bankservice)
- **Persistence layer** including:
    - Business Objects (ch.hevs.businessobject)
    - Data access (ch.hevs.dao)
- “Consuming” layer that uses the services offered (ch.hevs.test). In a real application, this layer would be replaced with interface layers to constitute the **Presentation layer**.

The database used in this exercise is a hsqldb database. It is quite simple to install and admin. In rootJavaEE\database\, you find all scripts useful for the exercise:

- startDB.bat: starts the DB
- data.bat: creates the DB schema and inserts test data. You can also edit the file data.sql that contains information on the schema and on the data to insert
- runManager.bat: allows launching the graphical interface for managing the DB. You can also use another tool.

As long as the database is running (see above), you can run the test scripts in ch.hevs.test:

- TestClient.java: adds one client in Table CLIENT
- TestTransfer.java: to perform a transfer of money between two accounts.

II - Questions and reflections:

The goal of this exercise is NOT to develop an application, but to see where the problems in multilayered applications are, and where Java EE may give solutions.

Topic 1: Persistence

- How is an account stored? Describe it.

Topic 2: Transactions

- What do you think about the implementation of the method transfer in the Bank class?

Topic 3: Habilitation

- Habilitation means that a resource is only accessed by allowed persons. What would you propose to limit the access to the service layer

- What would you propose the access to your service layer only to habilitated persons?

Your report addressing the questions above must be **uploaded on Cyberlearn** (e.g., word, pdf or equivalent) under “Devoir – TP01”.