**3. For each of the 12 assignment reports you have at hand, analyze the proposed architectural solution with respect to:**

**a. Risks**

Because there is no interface between application-layer and data-layer modularity could become an issue.

**b. Non-risks**

Security should not be too much of a problem since the system is client-server.

**c. Sensitivity points**

‘Cart Management’ is very important for availability-reasons.

**d. Tradeoff points**

Performance has been traded in favour of security by having the system running on one server.

**4. Review each of the 12 proposed solutions with respect to the following abstract design principles:**

**a. Information hiding**

Very bad, from looking at the sequence diagrams. Information is sent all over the place. No interfaces between layers.

**b. Minimize coupling**

There is absolutely no coupling at all.

**c. Coherence**

Good.

**d. Divide and conquer**

Good.

**e. Separation of concerns**

Good.

**f. Keep it simple**

It’s actually too simple.

**g. No circular dependencies**

There’s no dependencies depicted at all.

**h. Layering**

The layering is good, every subsystem is in the correct layer.

**i. Modularity**

Looks good, but the lack of dependencies makes modularity hard to judge.

**5. Give an overall grade for each architectural solution at the scale of 1 (poor) to 10 (excellent). Justify your score in terms of your answers to the previous questions**.

The system has been split up nicely into coherent subsystems divided into layers. However, there are no associations depicted at all between the subsystems. This is also evident in the sequence diagrams: ‘Register Screen’ (Presentational Layer) communicates directly to ‘User Table’ (Data Layer). This defeats the purpose of layering.

Final score is 4/10.