**Software Development Tools**

**LABORATORY WORK 2**

**EXERCISE 01 – Thinking About Risk**

IMPORTANT: PERFORM ALL TASKS ACCORDING TO YOUR OWN TOPIC.

PURPOSE

In this exercise, you will begin to think about difficulties that might arise while the Automated Teller Machine (ATM) software is being developed.

In the task, you will, as a group, discuss possible risks associated with the project.

So, your task at this point is to think carefully about what features of the ATM software the project description might be missing. The best way for you to think about this is to think about your own experience with using ATMs.

ACTIVITIES

Perform each of the following activities. If you have questions, issues, or doubts, please ask for help and do not just guess.

1. Each person in the group should reread the requirements and review the use case diagram already created.
2. Start an Excel spreadsheet to record the identified risk.
3. As a group, go through the software requirements, looking for quality requirements or functions that might be difficult to implement. Write these up as risks. Be sure to write the risk in such a way that you can objectively tell when the risk has been mitigated. For instance, “The requirements state that the software should work on machines that use insert card readers as well as swipe card readers, but we only currently have access to the API (Application Programmatic Interface) for the insert card reader”. This risk will be mitigated when the team has been able to locate the API for the swipe interface.
4. As a group, go through the scenarios one at a time:
   1. Consider what assumptions are embedded in the first scenario.
   2. Discuss whether each assumption would be likely to cause problems if left to be decided later. This is the most difficult step, because it requires experience in software design and development.
   3. If you decide ignoring an assumption is risky, add it to your risk list.
   4. Repeat for the next scenario.
5. Repeat steps for #4, but this time consider each of the actors in your use case diagram.
6. Repeat steps for #4, but this time consider the hardware and software with which your ATM software will interact.
7. Add the following column heading: A: Risk, B: Importance, C: Difficulty, D: Risk Level, and E: Techniques for Mitigation Strategy, F: Addressed.
8. Rate each risk on your list from 0: no concern to 5: extremely important for perceived importance to stakeholder.
9. Rate each risk on your list from 0: a technical non-issue to 5: very difficult to address later from a technical standpoint.
10. Set Column D to compute the average of the cells in Columns B and C. For example, cell D2 would include the formula =AVERAGE(B2:C2).
11. Sort your spreadsheet by selecting Columns A-D then on the Data Ribbon select Sort, Custom Sort, Descending and sort on Column D.
12. Now your highest risks are at the top of the list. Discuss whether each of the top five risks is related to management or software design decisions. Enter either Management or Architecture as appropriate in Column E of the spreadsheet.
13. Discuss the three types of views (the module view type, the runtime view type, and the allocation view type.)
14. Each team member should select a view type and add one question that could be answered by consulting a view of that kind (write it).
15. Beginning with your highest priority risk, decide whether each risk is a management risk or an architectural risk. Enter “Management” or “Architectural” in Column E.
16. For the purposes of this exercise, reformulate each architectural risk as a question about the architecture.
17. For each question, identify which view type can be used to answer the question and write in the next column.
18. Add two new columns headed Mitigates and Risky.
19. Refer to the prioritized list of risks that were identified as being architectural. Starting with the highest priority risks on the list, discuss how the design of the system impacts the risk for the top five risks you wrote questions earlier. Once the team has come to consensus on the impact, discuss design choices that could:
    1. Help mitigate the risk.
    2. Leave the system at risk.

Record a description of responses to a and b above in the new columns of the spreadsheet.

1. Submit your group’s Lab2 document (Risk Assessment spreadsheet) to the Dl.