SSVV TakeHome-B Student name…Chirea Liviu…….

Assignment Group………………932…………………..

**Teams for this exam (all the students that contributed partially): Campean Tudor-Alexandru, Buiga Andreea-Diana, Ceontea Vlad-Calin**

Courses Reflective Journal – as an individual task

**Description**: What happened? When and where? Who else was involved? What did you do? What did other people do? What was the outcome?

**Feelings**: What were you feeling during the situation? What do you think other people were feeling about the situation? How do you feel about the situation now?

**Evaluation**: What went well? What didn’t go well? What positive or negative things did you (or other people) contribute to the situation?

**Analysis**: Why did (or didn’t) things go well? What theories or research can help you better understand the situation?

**Conclusions**: What did you learn from this situation? If this situation happened again, what would you do differently?

**Action Plan**: What skills do you need to develop to handle a situation like this better? How will you develop the skills you need?

* Laboratories Reflective Journal - – both as an individual task and as a group task

**Description**: What happened? When and where? Who else was involved? What did you do? What did other people do? What was the outcome?

individual task

I had to build, test and deploy my Maven project using the Jenkins automation server. This was sometime between the second and third laboratory. My teammate had some technical problems with Docker, which worked on my machine (I already prepared the environment from home, as I heard that we needed to from another colleague that week), so I had to manage to make it work myself. I scoured several forums and documentations. My colleague didn’t know how to help. Right before we presented our assignment, I figured out that the problem was that the build of the project files were compiled using an outdated Java version, and changing the version of the project did not result in rebuilding those files, unless they were manually deleted first.

a group task:

Me and my colleague, Campean Tudor-Alexandru, were working on laboratory 3 in class. We split the task into writing the test cases (me) and documenting them (him). I wrote and debugged the test cases while he analyzed the example documentation and came up with a plan. We successfully finished it.

**Feelings**: What were you feeling during the situation? What do you think other people were feeling about the situation? How do you feel about the situation now?

individual task

I felt stressed, because at the time I thought that the laboratory work would only get harder and harder, and me and my colleague would fall behind if we didn’t figure it out right then and there. This fortunately was not the case. I also felt frustrated, because I find these kinds of software problems to be very misplaced, especially such a minor details as the files not recompiling after a compiler version change. I imagine that my colleague was also stressed, maybe even more than myself, as he doesn’t do well in these kinds of situations; he panics. In hindsight, there was nothing to be afraid of. Software systems that use Java usually take a couple of tries to get the versioning right.

a group task:

I was feeling confident and at ease, because JUnit has simple mechanisms for writing simple tests. He was more confused, as it was the first time we had to write documentation. I do not have different feelings today.

**Evaluation**: What went well? What didn’t go well? What positive or negative things did you (or other people) contribute to the situation?

individual task

I think my search for a solution went well. I think that my frustration still affected it negatively, and my colleague’s panic didn’t help either. But I would also consider that, depending on the situation, I sometimes feel better and more motivated when under pressure. So, maybe if we didn’t have a time limit, or if my teammate was not as anxious, I wouldn’t have managed to solve the problem.

a group task:

Everything went well. The tutorials helped a lot, as did random programming forums I found while searching for solutions. My teammate also came up with great ideas.

**Analysis**: Why did (or didn’t) things go well? What theories or research can help you better understand the situation?

individual task

This is probably a negative outlook to have, but I think this experience is essential to software development as a whole. Most systems do not work as seamlessly as we would want them to, and finding solutions to these problems is a crucial skill to have. I consider that more robust tutorials regarding the setup could have helped, but only as far as no more exceptional bugs came up. Not everything can be solved by tutorials.

a group task:

The task went well because of our skill as programmers, the good tutorials and the simple requirements.

**Conclusions**: What did you learn from this situation? If this situation happened again, what would you do differently?

individual task

I learned that the problem might be hidden deeper than my solutions can reach. I now have the perspective that the recompiling of files is the step that actually transforms our configurations into usable programs, and I will pay closer attention to make sure that a project’s build files reach their destination in a correct condition.

a group task:

I would do nothing differently. I learned how to write simple test cases and document White Box / Black Box testing.

**Action Plan**: What skills do you need to develop to handle a situation like this better? How will you develop the skills you need?

individual task

I need to develop my understanding of how software environments are set up, and how the Java ecosystem functions. The aspect of CI/CD, of building, testing and deploying software that is autonomously accomplished by tools such as Jenkins is also relevant. The best way to develop these skills is to continue to build varied applications and use different tools; learning through experience.

a group task:

Knowing how to use Microsoft Excel better, or how to quickly solve problems with Maven dependencies.

* Seminars Reflective Journal – both as an individual task and as a group task

**Description**: What happened? When and where? Who else was involved? What did you do? What did other people do? What was the outcome?

individual task

For the PortfolioTestCaseDesign, me and my teammates decided to each search and use a tool we found, and then compare our experiences to decide which we should write about. We did this over a weekend, sometime in week 6 or 7. I found a couple of papers with interesting tools, but most of them were dead ends. My colleagues did the same, more or less. In the end, we chose my tool because it was the easiest to setup and use and gave us decent results.

a group task:

We had to find an investigative tool for bugs, during the first seminars. Me and my colleagues split up so we each searched for one. My find was chosen at the end.

**Feelings**: What were you feeling during the situation? What do you think other people were feeling about the situation? How do you feel about the situation now?

individual task

I was interested by the idea of mutation testing, which was one of the main techniques that we found while searching for papers. I’m not sure what the other members of the team felt, but I imagine they also must have found the idea interesting. In retrospective, we had quite an intriguing subject, and even it was quite easy and enjoyable.

a group task:

I was very stressed and confused. My colleagues felt the same. I feel the same now.

**Evaluation**: What went well? What didn’t go well? What positive or negative things did you (or other people) contribute to the situation?

individual task

I found a lot of tools, but almost all of them were outdated. I got very discouraged and that must have slowed down my research quite a bit.

a group task:

We couldn’t find a good tool that met our criteria. We kind of rambled a bit and that didn’t help either.

**Analysis**: Why did (or didn’t) things go well? What theories or research can help you better understand the situation?

individual task

There just weren’t enough up-to-date papers with relevant software. I think a better understanding of how this kind of academia works would have helped know where to look.

a group task:

It was the first time we had to research this kind of subject. We didn’t know how to explore the different sites and didn’t understand the terminology. There was also a distinct lack of literature in useful testing investigation tools.

**Conclusions**: What did you learn from this situation? If this situation happened again, what would you do differently?

individual task

I learned that it’s a very important skill to know which journals to search in.

a group task:

I learned a lot about researching in academic papers. I would now know how to find what I needed faster.

* Exams Reflective Journal – both as an individual task and as a group task

**Description**: What happened? When and where? Who else was involved? What did you do? What did other people do? What was the outcome?

individual task

My interpretation of the assignment's requirements had to be debated. During the first week of our test period, something took place. We were at a meeting of the teams. There was only the two of us participating. We engaged in a contentious discussion on whether strategy is preferable: to address the issue on our own, in the best way we can; or, given that this topic is about testing for bugs, to try a more direct approach, which would result in a software that performs worse. Finally, we went with the first course of action.

a group task:

We had to debate our understanding of the assignment requirements. This happened in the first week of our exam session. We were in a Teams meeting. No one else was involved, except the two of us. We participated in a heated discourse involving what course of action is best: to, of our own accord, solve the problem as best as possible; or, since the purpose of this subject is testing for bugs, try a more straightforward approach that would lead to a worse performing program. In the end, we chose the former strategy.

**Feelings**: What were you feeling during the situation? What do you think other people were feeling about the situation? How do you feel about the situation now?

individual task

I was concerned about the effects of selecting a poor methodology, which may or may not result in useless test results or an outright inability to thoroughly test the program. Looking back, the issue still looks concerning.

a group task:

We were troubled by the implications of choosing a suboptimal methodology, which may or may not lead to useless test data or outright inability to test the software comprehensively. In retrospect, the situation still seems worrisome.

**Evaluation**: What went well? What didn’t go well? What positive or negative things did you (or other people) contribute to the situation?

individual task

I researched different aspects and, because the subject was hard to settle, I ended up falling in a cycle of doubting my solution.

a group task:

We had a productive debate, but the subject matter was prone to personal disagreements regarding the rendering of the code (internal logic, structure, error handling). We both could have easily given more constructive thoughts.

**Analysis**: Why did (or didn’t) things go well? What theories or research can help you better understand the situation?

individual task

The discourse was significantly influenced by our propensity to rush and do tasks as quickly as possible. Better strategies could need a deeper comprehension of the exam's criteria or a quicker accommodation of our differing viewpoints.

a group task:

Our tendency to hurry and finish things as soon as possible was a big influence on the conversation. Some better tactics would call for a stronger understanding of the exam requirements or a faster compromise between our beliefs.

**Conclusions**: What did you learn from this situation? If this situation happened again, what would you do differently?

individual task

This incident once again showed me how crucial it is to understand the specifications of a test. Except for trying a little bit more to comprehend, I cannot think of anything I would alter.

a group task:

This happening proved, again, the importance of comprehending what an exam’s requirements are. We cannot think of something we would change, except for trying a bit harder to understand.

**Action Plan**: What skills do you need to develop to handle a situation like this better? How will you develop the skills you need?

individual task

Reading comprehension and a deeper understanding of software engineering ideas like application structure and program requirements are important to me. I believe these skills are acquired through practice, such as writing and testing software and researching various viewpoints on computer science as a whole.

a group task:

Relevant skills include reading comprehension and a better grasp of software engineering concepts like application structure and program requirements. These proficiencies develop through experience, i.e. creating and testing software and studying different perspectives on computer science as a whole.