**Description**

1. **What happened?** Our team was tasked with solving a programming problem where we had to insert cows into an array of animals based on specific rules. The assignment included implementing methods to find specific animal pairs, insert cows between these pairs, and then solve the overall problem.
2. **When and where?** The assignment took place over one week as part of our SSVV course, with meetings held both in class and online.
3. **Who else was involved?** The team consisted of five members: Guia Alex, Groza Vlad, Hideg Paul, Harangus Dan, and Gligor Ovidiu.
4. **What did you do?** I contributed by solving the statement problem and helping others woth their tasks.
5. **What did other people do?**

* Guia Alex: Implemented the InsertCow method and conducted white-box testing.
* Groza Vlad: Worked on the integration of the methods and conducted integration testing.
* Hideg Paul: Implemented InsertCowbetween a pair of mouse-cat or cat-dog feature, including test case design for it.
* Harangus Dan: Conducted code reviews and helped with debugging issues across all methods.
* Gligor Ovidiu: Worked on the white-box testing and black-box testing

1. **What was the outcome?** The final solution worked correctly, and the testing strategies were effectively documented and implemented.

**Feelings**

1. **During the situation:** I felt a mix of excitement and pressure, particularly about ensuring my method worked correctly and integrating smoothly with others' work. There were moments of frustration when initial tests failed.
2. **Others' feelings:** Hideg Paul seemed confident and focused, particularly when explaining the white-box testing strategies. Dan and Ovi were intrigued and detail-oriented, feeling the pressure of ensuring error-free code. Groza Vlad felt unsured by the integration process but gained confidence as the work progressed.
3. **Current feelings:** Reflecting on the situation, I feel proud of our collaborative effort and the successful outcome. The experience was challenging but rewarding.

**Evaluation**

**What went well?**

1. **Team Collaboration:** Effective division of tasks and mutual support.
2. **Quality of Implementation:** Each method was well-implemented and tested thoroughly.

**What didn’t go well?**

1. **Initial Coordination:** Some initial misalignment in understanding the scope of tasks.
2. **Integration Challenges:** Some issues during the integration phase due to differences in tasks management.

**Why did things go well?**

1. **Effective Communication:** Regular meetings and updates helped address issues promptly.
2. **Commitment:** Each member was committed to their tasks and the overall project success.

**Why didn’t things go well?**

1. **Initial Miscommunication:** Early misalignment on task scope caused some delays.
2. **Integration Issues:** Differing coding styles led to challenges during the integration phase.

**Conclusions**

**Learnings:**

1. **Importance of Initial Alignment:** Clear initial communication can prevent misunderstandings.
2. **Value of Regular Updates**: Frequent updates and meetings can help address issues promptly.
3. **Benefit of Thorough Testing:** Comprehensive testing strategies ensure high-quality code.

**Future Approach:**

1. **Early Clarification:** Ensure all team members understand the project scope and tasks from the start.
2. **More Frequent Check-ins:** Increase the frequency of check-ins to monitor progress and address issues quickly.

**Action Plan**

**Skills to Develop:**

1. **Effective Communication**: Enhance skills in clear and concise communication.
2. **Consistent Coding Practices:** Improve skills in adhering to common coding standards.

**Development Plan:**

1. **Workshops:** Attend workshops on team management and communication.
2. **Practical Experience:** Engage in more team projects to practice and refine these skills.
3. **Feedback Mechanisms:** Implement regular feedback sessions within teams to continuously improve.