Project Retrospective

Project 7: NFT Development

Dear future students participating in the AMOS project,

In the past few months, we participated in the AMOS project ourselves and in this letter, we will try to summarise the things that worked well, and the things we would do a little bit differently in future projects.

First of all, we would like to emphasise that having weekly meetings, especially in the early phase of the project, were very helpful for getting a good understanding of the project scope and for getting feedback for newly implemented features quickly.

Another thing that was crucial to the success of our project, was good communication. In order to achieve this, we set up a Slack workspace with multiple channels for different topics and integrated it with GitHub. Even though communication, especially online and in bigger teams, can be a challenging task, using this approach, we were able to stay up to date fairly easily.

What we also found very important were thorough pull request reviews, compared to only checking if a feature does what it is supposed to do. This helped improve the overall quality and to keep other developers up to date with the code base. Besides, developers also were able to learn from others (code styles, technical knowledge, ways to solve a problem) and therefore were able to improve their own skills, which is proved at the end of the project. For future projects, we would keep this approach and continue to have at least two people review pull requests.

Even though being a product owner is a very time consuming task, it is very important and should be taken seriously. If a product owner does a good job at planning a sprint, the work of the developers is made quite a bit easier and less stressful. The team can also support their product owner(s) by giving issue suggestions that could refer to bugs encountered during testing, new features that were communicated by the industry partner, or other features that could be helpful for achieving the project goal.

Keep in mind that even though during this project all of you have to work closely together, all team members have different studying/working/sleeping schedules. It is helpful to establish guidelines from the beginning, e.g. giving feedback on a pull request within 48 hours to avoid frustration.

In retrospect, we would choose a better strategy for each phase of the project. Initiation phases should have different requirements (i.e code quality) due to the nature of uncertainty in design choices, in favour of MVP and development speed. This “technical debt” can be paid in the later phases. However, choosing the proper compromising level is tricky and usually only realised when the project’s deadline approaches.

We made some important learnings. Github workflows are neat, but pretty hard to develop and debug. Fork/create a private repository and test the action there to avoid polluting the main repository. Burn-out is real, and recovering from exhaustion takes time. Try to determine your and your team-mate’s limit. Testing is hard, especially decentralised. (Dijkstra once said that testing can prove the presence of bugs, but not their absence). Try to think about testing from early on. Create Stubs to mockup yet non-existing services.

Best of luck,
NFT-The-World-Team