

Process and Plan for Organizing the Programming Project

Process

We've settled on using an agile development method, with Slack as a tool. In slack we can keep track of how the project is moving forward, and keep all important information about meetings and so on. We also have a google drive, to help us share documents that are related to the project.

We're going to use Scrum as our management style, and have decided on having meetings 2 times a week. One on monday (arranged group) and one on thursday (group initiative). We're also planning to divide the meetings into smaller groups, so that people that work with the same things, can discuss those, without everyone else being passive watchers. Here we're going to go through the progress since last meeting (follow up), and how far we've gotten with the project. Together we review some of the progress, and divides new tasks amongst us. We also keep working on tasks that weren't finished. Mostly, team members can decide what they want to do, as long as everything gets done. That makes for a more motivated team, and we can fully utilize our skills. If we can't divide, team leader has authority to hand out tasks. We will also update to-do document between meetings, so we know what's done and not.

In addition to Scrum, we'll use a TTD-oriented development style (Test Driven Development), especially towards the back-end. We use this to create code that is more easily managed, and to be able to find bugs faster and with better precision. We think it will be good for this project, because we are multiple people working at it, and changes may affect other parts of the code, and hence it'll be easier to fix. This is not, however, saying we will always have perfect TDD, but we'll work towards it, when possible.

For now, we're working on planning and design, and dividing those tasks amongst us. We've also assigned some important tasks, like team leader (Jonas Triki) and git-manager (Jonas Mossin Wagle), (see more in Team Plan). Later it will be more programming and practical tasks, like testing and re-work of the documentation we've already made. We're going to keep using the meetings to arrange those things. We also have a goal to keep our project easily expand- and changeable.

For the more "human" aspect of the process, we've established rules, like no deleting of git repo, and guidelines for how to act if something is bothering you. We need to be able to talk to each other, and communicate in an effective manner. That means talking about issues, and finding (creative) solutions.

Specific plans

We plan to organize in subgroups based on previous experience and also the desire of each member.

- Additional group meeting at Thursday at 14.15 every week, adjusting if needed.
- Communication between group meetings and through Slack.
- Delegation of work will occur at work meetings or through Slack. The work tasks are determined by the obligatory assignment or contemporary demands. If reallocation of human resources is needed an agile approach will be employed.

- We will organize our Git in two branches, one dev(development) and one master (stable).
- Project documentation and illustrations will be placed in the folder “docs”.
- Code under development will be placed in the development branch and with group consensus be pushed to the production branch when finished, potentially in the end of a sprint.
- If unforeseen and problematic situations occur they will be dealt with as they arise or at group meetings/through Slack.
- Allocation of tasks based on the expertise of each team member.
- Well defined tasks to each team members to avoid duplication of same tasks.
- Breakdown of bigger tasks in smaller units to have better visibility and management.
- Document unresolved bugs in a dedicated Slack channel to ensure that we do not forget about them, and resolve them as quickly as possible. We will also strive to update this channel as soon as we find new bugs or fix known bugs.