

Software Engineering: Mini-project - Group sw605f12

Our project

Our project is part of the “Autism software on the android platform”-multi-project. An android-based platform is to be developed which helps autists in their daily life.

In short, our part of the project aims to split the usage of android tablets up in two basic modes: “Guardian mode” and “Autist mode”. Guardian mode aims to help the guardians, by allowing them to customize and personalize the look and feel of each app which the guardian uses in their daily work with autists. As autists sometimes have trouble handling all the options which the android platform by default provides, the “autist mode” aims to lock down the device such that all not-required buttons, notifications etc. are removed such that the autist cannot “mess up” the configuration of the phone, by entering the settings and switching random features on/off.

We will be creating an application, which will be responsible of launching all other applications. We call it *the launcher*.

Multi-project work interface

As the multi-project is composed of several independent groups working together, we have all agreed on an interface that we must adhere to. This interface sets requirements to the work done in the groups and the way the groups communicate, in order to create a uniform standard that allows for effective communication and work across the groups.

There is one weekly meeting for the entire multi-project, where every group shows up and gives a short presentation on their progress. This helps keep each group aware of how far they are compared to the other groups, and makes it easy for groups to communicate with each other about the progress of their work.

Choice and modifications

It will be clear to the reader that we have chosen to use the XP development method. We acknowledge right away, that we are set out to fail, as we know that we will be committing the worst of all errors, listed at the top of the “How to Fail with Extreme Programming”-list, namely “no on-site customer”. However, we will try to compensate for this by exchanging email addresses, phone numbers, Skype contact information, and make a plan for how we can stay in touch as much as possible.

Rationale

Pair programming

The study regulation notes that this semester project should focus on building ‘High quality software’. The use of pair programming increases the quality of the software, on behalf of it may slow down the quantity, but as the focus this semester is high quality software, we find it suitable to use pair programming. The pair programming-concept is new to all the members

of the group, which means the use of it would increase the overall learning of this project and increase our experience with pair programming. As pair programming enforces you to work together in pair, it helps creating a common understanding of the code within the group as a whole, as more people is involved in each line of code.

Refactoring & Simple design

All code we write should be accessible, both for ourselves but also for other sub groups of the multi-project, and easy to document in the report. Refactoring helps make the code simpler and more readable, making it easier to share and easier to discuss, an advantage when working across project groups but also when writing the report. Therefore we choose to apply refactoring.

Test driven development

Having good understanding of what the functionality of the product should accomplish is important, and having clear use cases and tests outlined before the code is written helps clarify it. As we do not have an on-site customer, the clarification provided by test driven development can be essential in staying true to the goals set by our customer.

Whole team together

The concept of everyone working in the same direction decreases the amount of time wasted on unwanted code and tasks. Communication is essential for us to be working in the same direction. The feeling of working as a group also lowers individual ownership of member created parts of the project.

Implementation plan

We will be implementing the XP approach by setting our iteration length to the same of the multi projects scrum-iteration length, such that we can deliver working code on time. We will both use the TDD which is part of XP, and perform additional integration tests as part of the multi project.