**Process**

We work with a protected master. This means that we need a development branche, and once a week merge this one into the master. The master should always be stable and working. The assessors will only look at the master.

We need to set up a Continuous integration. We need to use analysis tools. Hopefully the meaning of these things will become clear in the lectures.

We have to do the setup ourselves. Hopefully we will learn about Gradle in the lecture. Otherwise we will have to read about it ourselves.

**Application**

We develop a tool that can gives multiple choice questions to students and grade them.

The grade is calculated as:

The exam consists of 10 randomly chosen questions. These always have four options, and multiple answers can be selected. The exam lasts 20 minutes. The exam ends when the time passes or student click on a button to end it. Students immediately receive their grade. Students can change their answers during the exam.

**Authentication**Authentication testing: Users should log in with their Ned-ID and a password. We need to store the passwords securely. This could be done by hashing. The advice is to use Spring Security C4 (I think I heard C4).

**Roles**Students:

* Should be able to see their own grades and their own answers.
* They should only be allowed to participate in the assessments that they are enrolled in.
* They have three attempts per assessment. Highers grade counts.

Teachers:

* Can add, edit and delete questions.
* Access and edit everything.
* Can see a list of students that take the exam.
* Should see grades of the students.
* Can see the average grade for the course.
* See the number of students that took the assesment, and their names.

**Misc**

The API should be clean so it’s easily combined with different systems. It’s should be extendable, so be able to deal with high load.

There is no GUI. We only need to provide the functionality.

**Process**

Meet same time very week. 15:00 o clock.

Next Wednesday:

Have a formal requirements list. We get feedback.

We should write reviews on merge requests. We will get assessed on that.