**Grading Policy**

The [course goals] state that, after completion of this course the student should possess the following understanding, skills, abilities, and judgement:

1. **Knowledge and understanding**, the student should:
   1. Understand the principles of software development processes and projects
   2. Understand the principles of working with requirements and test cases
   3. Understand the complexity of working in software development teams
   4. Understand the difference between small scale programming and taking part in software engineering projects
2. **Skills and abilities**, the student should:
   1. Be able to plan and execute a small software development project in a team;
   2. Be able to fill at least two of the following software engineering roles: project manager, architect, analyst, designer, tester.
   3. Be able to apply skills from programming courses and other relevant courses in a project-like environment.
3. **Judgement and approach**, the student should:
   1. Be familiar with various software development process models, tools and techniques
   2. Be able to reflect on the choice of software engineering methods in the project.

During the course you will take part in a project to produce an Android application. During this project, you will learn various roles (such as scrum master, developer, etc.), produce a number of artifacts (such as user stories, test cases, etc.), and deliver a finished application. We will use the*finished application*, the *development process* and the *produced artifacts* to evaluate how well you fulfill each of the critera listed above. We will judge these three according to the following, weighted themes.

**1. Product, Vision and Requirements (30% of final grade)**

The finished product will be judged based on how "good" it is as well as in terms of the inital vision and the produced stories. How "good" it is will be based general qualities, such as perfomance, usability, and stability as well as qualities take make sense for the domain. Any non-functional requirements that are expressed as stories will be taken into account when determining "goodness".

We will also consider how well the finished application captures the vision, how well the stories capture the vision, and how these are realized in the application.

This step could be considered as a final acceptance test by a customer.

**2. Design Decisions (10% of final grade)**

Various design elements, such as user interface, classes and packages, as well as external dependencies (libraries and services) will be considered. Each of these should be well motivated by user stories and application needs. If you choose to use any external dependencies (other than the Android SDK, such as graphics engines or web services), these should be motivated.

**3. Development and Code Quality (15% of final grade)**

The produced source code should be of high quality. This includes, for example, that you take object-oriented design patterns (GRASP) into account, that you comment your code, that you refactor your code, etc. We will, for example, check the size of classes and the amount of comments. We expect you to continuously integrate your code, so your github repository should be updated and used in a good way.

**4. Documentation and Testing (15% of final grade)**

Major design decision, such as external dependencies, should be documented. These can be documented via user stories, source code, or other artefacts. We also expect you to produce tests for the user stories, as documentation for both your application and your source code.

**5. Development Process (30% of final grade)**

You will each (indivually) submit a postmortem report that discusses what you learned as part of the development process. This report should include a description of how your team worked, what was good/bad, and what you would change the next time you work on a similar project.

[More information](https://github.com/morganericsson/DAT255/wiki/Post-Mortem-Report) about the Post-mortem report (format, pages, etc.) is available in the course wiki (on github).

**Final Grade**

At the end of each project, each group will recieve a total score (0-100), based on the five themes. Each theme score will be briefly motivated. The total score will determine your final grade.