

# CIT 594 – Homework 6 – Project

---

*Due – May 5, 2017 at 12.00pm*

## Programming Project (80 points)

Please take into account the feedback given to you and use that to improve your projects.

In addition to your code, please submit a user manual with screenshots, which describes how to run and use your program (the different functionality, the various options, etc.).

In addition, also submit a 1 page summary.txt file – this should contain the name and description of your project (4-5 sentences), the actual work breakdown (who did what), and the link to your github repo (see below).

Recommendation: You will get feedback on your proposal from one of the TAs. That specific TA will also act as a Project Manager for your team. We recommend meeting with them often (at least once a week) to update them on progress you've made so far and ask them for help if needed.

As a reminder, each project must contain the following:

- At least 2 of the following:
  - Formal Design (1 CRC or Class diagram, 2 Sequence Diagrams)
  - Interfaces (Define and use)
  - Design Patterns
- Both of the following:
  - Data Structures (at least three different kinds)
  - Advanced Topics (Java Graphics, Multithreading, etc.)

Note: Late submissions will not be allowed for the project part.

## GitHub

We will use github for the team projects. Please sign up using the link below. You will be able to create a new team and/or add yourself to an existing team and share a private repository for the project.

<https://classroom.github.com/group-assignment-invitations/2c50854f5b01c1d42d731b15efe15009>

## Grading Criteria

0% for compilation

70% for functionality – Does the code work as required? Does it crash while running? Are there bugs? ...

10% for design – Is your code well designed? Does it handle errors well? Are exceptions handled well? ...

10% for style – Do you have good comments in the code? Are your variables named appropriately? ...

10% for git/github usage – Did you use git well? Do you have good commit messages? Did you resolve merge conflicts well? ...

## Programming – General Comments

Here are some guidelines with respect to programming style.

Please use Javadoc-style comments.

For things like naming conventions, please see Appendix I (Page A-79) of the Horstmann book. You can also install the Checkstyle plugin (<http://eclipse-cs.sourceforge.net/>) in Eclipse, which will automatically warn you about style violations.

## Submission Instructions

Please create a zip file containing all the deliverables (code, user manual) and summary.txt file. Submit this zip file via canvas.