

# Lab 2 - Github and requirements

Lab outline:

- brief overview of GitHub's requirements support.
- discussion of how to write requirements statements.

GitHub is a tool that supports social coding. In this lab, I will show a few of the features it offers for agile requirements.

# Caveats

- GitHub is a tool for software development. It is not a tool for requirements engineering. As we discussed in class, it is not designed for Complicated domain requirements (e.g., where legal implications exist if requirements are not implemented).
- However, it does a great job of doing "just enough" for Simple and Complex domains, where the goal is to capture a starting point for building code, that will then be iterated.

# GitHub Issues

(walk thru the [Github tutorial](#))

# GitHub Project Boards

(walk thru [the Github tutorial](#))

# Structuring Assignment 2 Requirements

- acceptance criteria: consist of a checklist (use the GH notation) that a customer wants to see before agreeing the requirement has been met
- ☐ user can login
- ☐ user can upload a file
- ☐ the file appears in the user's list of files

Think of these (in our context) as the stuff the TA / Instructor wants to see to assign a mark. Missing all of them is like getting an F. Getting all of them is like getting an A.

(aside: this is not how we mark things!)

## Other aspects to include

- an early estimate (guess-timate) of the effort required to implement the requirement
- a set of assumptions that are required to be true for the requirement to be met
- a set of risks that could prevent the requirement from being met
- a set of evidence that could be used to verify the requirement is important or other "sources". Think of this like citations in a paper.