Cornell University
Computing and Information Science

CS 5150 Software Engineering

1. The CS 5150 Projects

William Y. Arms

Choosing a project

Client

A client can be any person except yourself (e.g., a member of a Cornell department, a local company, or other external organization, a member of faculty or staff, etc.).

The client should have a firm intention to use the software in production. Aim for a minimum of a three-year production life with many users.

Special types of client

See the Projects pages on the website for special factors for:

- Remote clients
- Student organizations
- Entrepreneurship projects

Choosing a project and forming a team

In selecting a project, think broadly.

The project can be an application, system software, or even a toolkit. Software engineering covers everything from smartphones to supercomputers.

The only conditions are that there must be a real client and real users.

- You are encouraged to identify your own project.
- Some potential projects and clients will be suggested by the course team.

At the beginning of the first few classes, there will be time to describe possible projects or to announce that you are setting up a team.

Milestones

The projects are divided into four parts, each of which ends in a milestone.

The first milestone is a feasibility report.

- For the second and third milestones, the team makes a presentation and submits a progress report to the client and the course team.
- At the fourth milestone, the team demonstrates the working software and makes a presentation to the client and the course team, followed by a final report and handover of the completed project to the client.

These are group projects, but you will also be rewarded individually for special contributions to the project, or failure to provide a fair share of the effort.

See the Assignments pages on the website for more information.

Overview

Software development is more than writing code.

Every CS 5150 project includes all aspects of software development:

- feasibility study
- requirements
- system and program design
- coding
- reliability and testing
- delivery
- documentation for future maintenance
- software license

Each of these topics is covered in the course lectures.

Sprint

By production standards, the CS 5150 projects are small, about the size of an agile sprint in most production environments.

Sprint

In agile terminology, a sprint is a fixed period of time during which a team completes part of a software project.

- Every sprint ends with code that is ready to put into production.
- A typical sprint might have a team of 4 to 9 people working for 2 to 4 weeks.
- If not released during the sprint, the software should be ready for release, or for integration into a large system.
- It should be fully tested, with documentation for maintenance.

Time box

Time box

A time box is a set period of time during which a development team completes part of a software project.

CS 5150

Time: The project must be completed within one semester.

Resources: The team size is fixed (between 6 and 8 people)

Scope: The scope must be chosen to fit within the time and resources available

The scope depends on the experience of the team and the complexity of the project. The project team must agree a project scope with the client.

If the project runs into difficulties (e.g., time delays), the scope has to be adjusted.

The CS 5150 project: an agile sprint

Within the time box, the CS 5150 project will complete an agile sprint. This can be:

A complete system

By the end of the time box, the system is either in operation, or fully tested and ready to be installed by the client.

Part of a larger system

The project will create a completed section of a larger system. It will deliver code, fully tested, and documented.

Team organization

A single time box can use lightweight project management, with minimal documentation and specification within the time box.

But a team of 6 to 8 people cannot function without organization.

Every project should have:

- Regular meetings with the client (at least once a week).
- Regular team meetings.
- A project plan, kept up to date (e.g., a Gantt chart).
- A project management system for code and documentation, chosen in conjunction with the client (e.g., GitHub).

Within the time box

The CS 5150 project is large enough that you will need a systematic process for developing the software.

Most projects will use one of the following processes:

- Iterative refinement
- Modified waterfall model

Some projects may use:

- An agile process with a sequence of short sprints

These processes are the subject of Lectures 2 and 3.

Organizing the project

The projects are team projects.

Everybody should aim to contribute to every aspect of the project:

- Meeting with the client
- Taking notes at meetings
- Requirements and design
- Presenting at the presentations (required)
- Coding and testing
- Project management
- Writing reports and documentation
- etc.

Leadership

Members of the team are jointly responsible for the success of the project.

Leadership

There is no team leader. Everybody shares in organizing the project.

Student contact

During the start up period, the student contact coordinates setting up the project team, but the student contact is not the the team leader.

Project management

Members of the team are jointly responsible for project management.

It is often useful to have a project manager, who manages the schedule, but the assignment of tasks is a shared activity.

Suggestion: Some teams change project manager at regular intervals, e.g., every two weeks.

Small groups

It is often useful to divided the team into small groups for some activities, but everybody should be prepared to help with the work of every group.

Special expertise

If a team member has special expertise, make use of it, but try to get other members of the team involved.

Cornell University

CS 5150 Software Engineering

1. The CS 5150 Projects

End of Lecture