

Capstone Project Guidelines for Computer Science

While capstones in computer science may take many different forms, most have been internships or programming projects. These guidelines are intended to give general guidance to students on expectations for these projects. Other projects are possible; students should consult with faculty regarding the expectations in those cases. Double majors are encouraged to have a capstone project that bridges the two areas.

Projects usually fall into one of several broad categories:

- internships in companies with established internship programs
- employment with a company (usually for a summer)
- software development projects for a faculty member or for a campus organization or office
- software development projects following a student's interests
- participation in an established research program
- research undertaken following a student's interests

In all cases, it is expected that the capstone be something appropriate for a computer science professional. The complexity and scope of the project must go significantly beyond the topics covered in any relevant computer science course, regardless of whether or not the student has taken that course.

A primary distinction made is the degree of supervision by the organization for the student. In companies that employ and supervise computer science professionals, standards are generally established by the organization and the student will be expected to follow those standards.

In smaller organizations, including smaller businesses and campus offices and organizations, there is often not much professional supervision of the projects. It is expected that students will follow generally accepted standards of professional computer science program development. At a minimum, this will follow one of the models for software development life cycles, and will include documentation for requirements analysis, specification, design, implementation, and integration, and the project should be deployed for the end user. Programs should be documented to facilitate maintenance by another programmer. It is expected that the student will bring expertise to the project even where the person supervising the project does not have experience in supervising computer program development. In these cases, the computer science faculty member supervising the project will consult with the student on progress of the project, and will approve the documents for each of the stages of software development.

Projects will usually involve two supervisors. A computer science faculty member will serve as a resource and consultant on the project, and will approve intermediate stages of the project. Another supervisor will come from the office or organization for which the project is being developed; that supervisor must be a non-student client or customer who submits a detailed requirements document prior to capstone pre-approval, equal in scope to the project description that we require for an internship. Projects for student clubs or organizations must interact with a faculty or staff mentor or adviser. Projects developed by an individual student following their own interests will usually have only a computer science faculty member as supervisor, and will generally require more interaction with the adviser.

The project should include the capability for the client or customer to be able to maintain and modify the operation of the system, after the student has completed the project, without editing source code.

In the case of participation in an established research program, the research must either be in a computer science-related field or have a significant computer science or software development component.

In the case of research undertaken for a student's own interest, the amount of software development will be decided on a case-by-case basis. Some amount of software development is expected, whether it results in simulation systems or end-user products, or is used to generate or collect data.

Depending on the area, some projects may require more extensive guidelines. In particular, projects that involve construction of a web page should respect the following guidelines:

- The web pages, both static and dynamic, should validate to one of the standards published by w3.org.
- There must be full consideration given to security, including
 - the system must be proof against sql- and html-injection attacks
 - any stored passwords should be encrypted or otherwise secured
 - any sensitive, non-public data should only traverse a network in encrypted form
- Any database backend that forms a significant part of the project must be subjected to a normal-form analysis; this analysis is a part of the project.
- It should be possible to add and modify text, alter images, maintain the database, reset passwords, etc., all from a secure administrative interface (via web, email, or local program) which has clear instructions and which does not require knowledge of sql or html.

- The project should include the installation of the system at the final server location. For example, if the system was developed on the student's T: drive, the project is not complete until installation at the client's server location.

For all capstones:

It is expected that students will seek approval before starting work on the capstone. All students should keep a daily log of their progress and keep copies of the significant documents in the project. Following the completion of the project, each student is expected to write to the following prompts:

- Describe in detail your capstone experience (at least one page).
- Describe the physical, social, and development environment of your project or internship.
- Characterize the extent to which you worked with other people.
- Describe the computer hardware and software platform.
- Describe the communications skills you used.
- Describe the CS courses or topics which were particularly useful.
- Describe the non-CS courses or topics which were particularly useful.
- List and briefly describe three things that you learned.
- Describe a problem you encountered and its solution.
- Indicate if there is anything you now realize that you should have known or studied before starting your capstone experience.
- Explain whether the experience turned out the way that you anticipated that it would.