From College To Workforce - A Computer Science Capstone[*]

Nifty Assignment

Ryan Florin
Department of Computer Science
Georgia Southern University
Statesboro, GA 30458
rflorin@georgiasouthern.edu

Capstone courses are not a new type of course or a new type of assignment; however, I come from industry where I have 16 years of experience as a software engineer and technical project manager. In this talk I will provide my unique insight into the Capstone course experience that I provide for students that help them evolve from senior computer science students into junior developers with the confidence to enter the workforce.

A Computer Science Capstone is a final course where senior students use the skills and knowledge they have been accumulating in their college experience to solve practical problems. It is meant to provide an experience to evolve students from working on individual-based, short-term class projects into working on team-based, long-term, industry projects.

In one semester, students are organized into teams of 3-6 students. As seen in Table 1, projects for Fall 2024 and Spring 2025 are comprised of four or five members each. Each team must complete four stages of the Software Development Life Cycle to develop a prototype for their project. The four stages are Requirements Elicitation, Software Design, Software Development and Testing and Deployment. In the Requirements Elicitation stage the teams meet with their respective clients to learn the requirements for the project. During this stage the meetings are used to gather an understanding of the project and to establish informal requirements. The deliverable for this stage is a Project Summary document that describes the requirements for the project in an informal manner. This document has taken place of a formal requirements

Table 1: Capstone Projects - Fall 2024 and Spring 2025

Semester	Project Name	Students	Languages
Fall '24	Digital Twin	4	87.7% Python 12.3% C++
Fall '24	Discussion Board Analytics	5	48.2% JavaScript 30.8% Java 12.1% Python 8.9% CSS and HTML
Fall '24	Automated Warehouse	4	67.2% Python 30.0% JavaScript 2.8% CSS and HTML
Fall '24	LLM Onboarding Resource	5	38.3% JavaScript 37.8% Python 23.9% CSS and HTML
Fall '24	M-Kart	5	100.0% Python
Fall '24	Robot Demonstration	4	67.8% Java 17.0% JavaScript 15.2% CSS and HTML
Spring '25	Digital Twin	4	91.6% Python 8.4% Shell
Spring '25	Data Literacy Explorer	5	95.5% JavaScript 4.5% CSS and HTML
Spring '25	Privacy Policy Analytics	4	60.7% Python 35.8% JavaScript 3.5% Other
Spring '25	CV-Enhanced Game Board	5	100.0% Python
Spring '25	Interactive Lesson Viewer	5	78.0% Javascript 22.0% CSS and HTML
Spring '25	Mini-Game Console	5	89.2% GDScript 9.5% Python 1.0% Shell

document, because at this point in the semester some teams do not have a solid understanding of the project.

In the Design stage, teams are encouraged to split the project into parts and individually research and develop a proof of concept that will later be used in the project. This is meant to get the students actively working on a meaningful aspect of the project in the project early in the semester. Next in the design stage, the teams must develop a design for their project and create a sprint plan for the remaining 10 weeks of the semester assuming a 2-week sprint cycle. The Design Document includes their high-level design as well as design for individual components of the projects. As shown in Table 1, it is interesting to note that most teams favor the use of scripting languages to develop their prototypes.

In the Development stage teams must follow their sprint plans to develop the prototype. Typically, teams learn that their sprint plans are flawed and must adjust it throughout the semester to handle their client's changing requirements and the discovery of problems in their assumptions. Every two weeks, teams prepare and present a Sprint Demo for the class to show off the current status of their project and to highlight any problems they are having. Additionally, at the end of each sprint a Sprint Report is due which requires them to list what they have done in the sprint, what they will accomplish in the next sprint, and a retrospective on what they did well as a team and what they need to improve on.

At the end of the semester, teams prepare a final presentation where they demo their prototype and present their semester's work to the Computer Science Department faculty for evaluation. The Final Documentation includes a full description of their project, an updated design document, and a retrospective on how they improved over the semester.