Working with Your Capstone Team:

Guidelines for Sponsors

The Computer Science and Engineering department thanks you for offering this opportunity to allow CSE students to work on your product. The capstone project course CSE5911 is the culminating event for graduating students: a chance for them to put their skills together and demonstrate that they are effective in developing an industrial-grade product before they join the professional workforce after graduation.

The students are not the only ones to benefit. Your company will benefit by having a working product, as defect-free as we can make it, delivered to you with all source code, test cases, and technical documents. There is no cost to you. Your company gets working software, the students get graduating course credit, and the instructor ensures that the processes goes smoothly. I personally run the class as if I were a program manager of many professional project teams to ensure consistency and high product quality.

The guidelines below are intended to make the capstone experience go smoother, to let you know what to expect and what not to expect, when working with your capstone team. To enhance communication and understanding, please copy any emails to the team to me also.

1. General

Working with your capstone team is no different than working with an outside vendor. You could think of this effort as “outsourcing to OSU”. You should expect a professional level of service and product. You have signed the **OSU Capstone Agreement**, which ensures that your company maintains all intellectual property rights. It also contains a non-disclosure agreement with your capstone team: no knowledge is to be shared outside of the project team (except for brief class presentations), and all work done by the team is turned over to you at the end of the semester.

2. Project Scoping

You submitted a capstone **project charter** and have been selected as a capstone sponsor. Your project information, containing purpose, overview, milestones, technical requirements, enables teams to be assigned for the best set of skills to apply to your business problem. The high-level information you presented will be broken down into smaller objectives based on the key milestones you provide. Within two weeks, the capstone team will set up a **business team kick-off meeting**, provide a summary of project objectives (**project abstract**), and will identify preliminary dates of when each objective will be done (**preliminary release plan**). Each product artifact is a refinement of the previous one. Coding will start shortly after. Each month the capstone team will demonstrate working software of your objectives (**user demo**); the partial product will be ready for you to accept.

Project Charter: This initial document defines what you want for your project: mission statement, overview, milestones, technical factors and risks, and all contact information. This document is

refined with the capstone team in the business kick-off meeting so they can come to a full understanding of your project problem.

Business team kick-off meeting. The team will set up this meeting with you and any other people you delegate to the project. You should see an agenda before the meeting, and minutes afterwards. The meeting gives all project members a chance to meet each other, to confirm that the capstone team understands the product well-enough to proceed, and confirm that the project can be completed within the semester time frame. It is also necessary for you and the team to reserve three dates for the three monthly user demos, and a regular, one-hour weekly team meeting.

Project Abstract: This document defines the project scope further. All objectives for the project are defined, stakeholders identified, and how the users (or external systems) will interact with the future product at the high level.

Preliminary Release Plan. All your milestones will be decomposed into smaller objectives or features, and in what order they will be completed. You must help the team decide how to prioritize those features by business importance—what do you want built first, then second, etc.? At this point, the team can only estimate how fast they can build your product, but after three weeks, they will have a measured speed of development (*team velocity)* with which to revise the preliminary release plan into a *final release plan* with assured dates of completion.

User Demos. Each month at the user demo, the team will demonstrate working software of those features that was promised by that time. Your role at the demo is critical: it confirms that the team progress is on-track, and allows you to make corrections and revisions once you see it. The team will also demonstrate test cases running against the software, and a few project management statistics, such as project progress, defect repair rates, and other items needed for grading. User demos typically take between 20 – 45 minutes.

3. Project Development Process

The capstone teams use a best-practice development methodology called *agile* (of which there are many variations). The key to this practice is **iterative, thin-thread,** and **incremental development**. In addition to refining the problem statement and project scope, the capstone team will spend the first week setting up their development infrastructure, learn whatever technical requirements you defined (tools, languages, databases) before writing code.

Iterative Development: The team breaks down the semester into one-week iterations. Each week they decide what they can accomplish that week, including detailing requirements, design, coding, defect repair, and all testing. The team reviews each aspect of their work for consensus. Weekly meetings between the capstone team and your business team are important to clarify or revise detailed requirements.

Thin-Thread Development: Each element of the product's functionality can be viewed as happening as a user transaction, a thin-thread triggered by a user's action, processed by the system, and then returned back to a user. The team will not build "components", such as full-scale databases, large GUI components, or large processing components in large blocks of time and effort before they are needed. They build only what they need to enable the specific thin-thread user interaction. This approach avoids the time-intensive integration of components

afterwards, and wasting time building parts of components that are never or rarely used (which studies have shown are well over 60% of the work with a component-based approach).

Incremental Development: Thin-thread development enables a partial product to be built in the first or second iteration. Each partial product is part of the core of the whole product. Each iteration's code is built on the accumulated code of previous iterations, and thus the product is built incrementally. The product is not built in pieces and then cobbled together in one integrative step, but is integrated continually throughout the project. Each iteration, a partial product is delivered with all coding complete and all tests pass. You should not need a QA team to test the product after it is delivered to you (unless you prefer).

4. Client Grading

Near the end of the semester, you will be asked for your feedback on the team and product itself. This feedback constitutes a significant portion on the students' grade. The form is a simple questionnaire about the product (deliverables, quality, timeliness, thoroughness) and about the capstone team (professionalism, quality, timeliness, and thoroughness).

5. Further Questions or Problems

The weekly meetings and user demos are meant to maximize communication and feedback for developing the best possible product. However, sometimes things don't happen as they should. For example, if the team misses a weekly meeting or fail to deliver a project deliverable on time, please contact them first. Like anyone else, they might have thought they communicated but didn't. If there is an inadequate response, then contact me. No week should go by before these kinds of issues are resolved.

If you have an unsatisfactory experience, such as you see a trend with some effect of the team, like being 20 minutes late to several meetings, then contact me. I will try to resolve it as swiftly as possible and get the team or project back on track.

In general, if there are any questions at all about what is happening on the project, or something you would like to talk about outside your capstone team, please feel free to contact me at any time.

Alan B. Cline
Sr. Lecturer
Computer Science and Engineering Dept.
The Ohio State University
Dreese Labs Rm 248
Email: cline.4@osu.edu
Office: 614.292.1109
Mobile: 614.832.0300