Guidelines:
Information regarding CS 36000 and CS 46000 and CS 46500
April 29, 2011

The courses CS 36000 Software Engineering (4 cr.), CS 46000 Senior Capstone Project I (3 cr.) and CS 46500 Senior Capstone Project II (3 cr.) are required to be completed successfully for your BS or BA in Computer Science. Students who are majoring in computer science are required to take CS 36000 during their junior year. After successful completion of CS 36000, students must take CS 46000 during the Fall and CS 46500 during the Spring semester, of their senior year, doing the same project under a faculty member as their Senior Project Advisor. Students begin work in teams on their selected senior projects in the very beginning of course CS 46000. The project could be a substantial application-oriented or research-oriented software project. Projects should be completed at the end of course CS 46500.

Both CS 36000 and CS 46000 are listed as part of the CS core courses. CS 46500 can be counted toward any of the concentration areas of the program. The concentration area will be selected as the one best accommodating the project’s topic. For example, if your project is dealing with network security or database, then your CS 46500 will be counted toward the concentration area of Network Computing or Informatics, respectively.

The course CS 3600 (4 cr. hrs.) will be taught by CS faculty members. The classes will meet 3 hours 45 minutes per week. CS 46000 and CS 46500 will be administered by a Course Director (CS faculty). For CS 46000 and CS 46500, the Course Director will begin by conducting the first class-meeting (or the first two class-meetings if needed) and will provide the designated days of class for presenting their on-going projects throughout the semester. Students must attend these designated days of class to formally present their project and to conduct a final formal review and demonstration to the Course Director, project stakeholders and other interested persons.

Students in CS 46000 and CS 46500 must begin, develop, and complete their project under the supervision of their designated Senior Project Advisor. Students are advised to meet with their advisor regularly.

The Department of Computer Science introduces this above-mentioned new structure to assist our students in achieving their academic goals by providing a more challenging, real-life experience in software development and software engineering methodology. For this purpose the students will be engaged in major senior projects with real-life applications. The Department would like the new structure applied with an effective date of the Spring Semester 2012.

For current students, if you have taken CS 36000 (3 cr. hrs.) or if you are taking CS 36000 (3 cr. hrs.) in the Fall 2011, then you will need to take CS 46000 (3 cr. hrs.) with CS 494 (1 cr. hr.) in the Spring 2012 or Fall 2012.
CS 36000 Software Engineering (4 cr.) [Fall, Spring]
CS 36000 gives an introduction to software engineering using an object-oriented approach. Topics include the software development process; iterative and incremental development; team organization and project management; object-oriented analysis and design; representation of software models using UML: use cases, class and interaction diagrams, and state-charts; metrics for design evaluation; software quality assurance; testing planning and specification; unit and integration test methods; software tools for analysis and design. Ethics and professionalism will also be covered. The prerequisites for the course are the completion of both CS 26000 and ENG W234. It is recommended that students take CS 36000 their Junior year.

Students will work on assignments which will help with understanding the concepts and activities of the methodology applied. Students will not start their work on a senior capstone project during this course.

Course learning outcomes include students gaining knowledge of the historical background and fundamental principles of software engineering, an understanding of the process of software development and the variety of contemporary development models. Students will learn and apply skills critical to software development with a focus on object-oriented analysis, design, implementation, and testing. This will help students understand the advantages and disadvantages of alternative team organizations for software development as well as issues of ethics and professionalism and the impact of computing on individuals and society.

While taking course CS 36000, students are advised to begin to contemplate and plan what and how they would conduct their senior capstone project.

For the successful completion of CS 36000 and before entering CS 46000 Senior Capstone Project 1, the first course of a two-semester capstone sequence, a student must:
- Consult and invite a faculty member, with acceptance, to be their Senior Project Advisor.
- Select one or two other students to form a team. It is advisable to create a team that will not consist of more than three students.
- Submit a proposal of the intended project to the Senior Project Advisor (see Appendix) (who advises his students to plan, develop and complete the team project) and the Course Director (who administers the entire senior capstone projects for the department).

The Course Director will arbitrarily assign a student to a Senior Project Advisor prior to their registration of the course CS 46000, if they fail to find a faculty member with acceptance as their Senior Project Advisor.

After successfully completing CS 36000, students will need to ensure that they will have a smooth and continuous flow from CS 46000 into CS 46500 with the SAME project under a faculty member as their Senior Project Advisor.
CS 46000 Senior Capstone Project I (3 cr.) [Fall]
CS 46000 is the first course of a two-semester sequence. Student teams will participate in the development of a substantial application-oriented or research-oriented software project utilizing a formal software process model. Emphases are on teamwork, project management, and oral and written communication. Student teams will conduct review activities and develop artifacts appropriate for the software project and process model chosen.

This is the first course of a two-semester capstone sequence. Based on their studies in 36000, students begin work in teams on their selected senior projects under their Senior Project Advisor’s supervision. The prerequisite for taking CS 46000 is the completion of the CS 36000. All students (who register for CS 46000) must attend the first class-meeting (or first two class-meetings if needed) of the course CS 46000 and the designated days of classes for presenting their on-going projects.

For CS 46000, students learn how to apply software engineering principles and skills to a team-oriented software project; construct a software project schedule and track its progress; construct artifacts appropriate to demonstrate completion of each phase of the software process; conduct formal technical reviews; and utilize a repository for project artifacts. An Assessment Plan for the course will be developed.

The course utilizes two tools for assessment: student performance on the first part of the project, which may be solicited from an external client, and the results of a student survey. The project is evaluated by the Course Director and other stakeholders (e.g., client, Departmental faculty). This provides an objective measure of level-of-achievement for course outcomes. The survey asks students to judge how well they achieved each learning outcome on a 5-pt Likert scale. The Course Director summarizes these evaluation results, which are given as recommendations to the Senior Project Advisor. The Senior Project Advisor then generates his/her student grade for the course at the end of semester. Then the Course Director will electronically enter their final student grade for the course given by their Senior Project Advisor, into the grade system. Furthermore, the Senior Project Advisor is encouraged to make recommendations for the next course offering, especially for those course outcomes which average objective or subjective scores fall below 3.

The results of the instructor and client project evaluation and student survey are the principal means by which program outcomes are assessed. Thus, at the end of the semester, the Department will receive a copy of the summarized results and the student surveys from the Course Director for conducting the course/program assessments.

CS 46500 Senior Capstone Project 2 (3 cr.) [Spring]
CS 46500 is the second course of a two-semester sequence. Student teams will complete the development of a substantial application-oriented or research-oriented software project begun in CS 46000. This course places an emphasis on teamwork, project management, and oral and written communication. Student teams will conduct review activities and develop artifacts appropriate for the software project and process model chosen. Students will be required to conduct a final formal review and demonstration to project stakeholders and other interested persons. The prerequisite for taking CS 46500 is the completion of CS 46000.
Students will learn how to apply software engineering principles and skills to a team-oriented software project; construct a software project schedule and track its progress; construct artifacts appropriate to demonstrate completion of each phase of a software process model; conduct formal technical reviews; and utilize a repository for project artifacts.

The course utilizes two tools for assessment: student performance on the completed project begun in CS 46000 and a student survey. For assessing student performance on the completed project, students must attend designated days of class to formally present their project and to conduct a final formal review and demonstration to the Course Director, project stakeholders and other interested persons. The Course Director will invite stakeholders and others to attend these designated days of class for their formal presentation, review and demonstration of their project, as well as organize and preside over these classes.

The project is evaluated by the Course Director and other stakeholders. This provides an objective measure of level-of-achievement for course outcomes. The survey asks students to judge how well they achieved each learning outcome on a 5-pt Likert scale. The Course Director summarizes these evaluation results, which are given as recommendations to the Senior Project Advisor. The Senior Project Advisor then generates his/her student grade for the course at the end of semester. The Course Director will electronically enter the student’s final grade, given by the Senior Project Advisor, into the grade system. Furthermore, the Senior Project Advisor is encouraged to make recommendations for the next offering, especially for those course outcomes which average objective or subjective scores fall below 3.

The results of the instructor and client project evaluation and student survey are the principal means by which program outcomes are assessed. Thus, at the end of the semester, the Department will receive a copy of the summarized results and the student surveys from the Course Director for conducting the course/program assessments. The senior projects will be kept in a secure place in the Department.

**Non Computer Science Majors:**
If you are not majoring in computer science (that is BS in CS or BA in CS), and if you do not have to take CS 46000 for your degree program, then you ONLY need to be responsible for CS 36000. Please let your instructor know before you enter your project for CS 36000.

If you have any concerns, please consult your instructor first. Our priority is how students can gain experience or learn the software development life cycle for successfully implementing a large scale software project or learn from a failure in development and implementation of a software project.

Peter A. Ng, Ph.D.
Chair and Professor of Computer Science
Department of Computer Science
College of Engineering, Technology, and Computer Science
Indiana University-Purdue University Fort Wayne
Appendix

I. Senior Research/Project Proposal

There will be no standard format for a proposal. Below is a recommendation for the proposal.

1. Title page

Title of the Proposal (front size 18)

Proposed by:
Chris Alfra
Betty White

Senior Project Advisor: Dr. Grin Kimbly

May 15, 2011

• The title of the proposal should give readers the idea of what this senior research/project will be.
• The author(s) of the proposal
• Name of a faculty who will serve as their Senior project Advisor
• Date of completion/submission

2. Table of Contents (of the Proposal)

3. Body of the Proposal
   I. Introduction
   II. Existing Work
III. Multiple sections/chapters for describing the proposed project to be accomplished

IV. Conclusion

V. Appendix

VI. References

4. Adopt the format of the text book for CS 36000 [Ian Sommerville, *Software Engineering*, Addison-Wesley] for writing the proposal. This proposal could be used as a base that could be extended to be a full project write-up for CS 46500. If the proposed project is planned to be published, then it is advisable to use the format for that publication. Quotations in the text should be matched with the references listed in the References.