COURSE SYLLABUS

Course Number:
CSC490

Course Title:
Senior Project

Credits:
3:3

Prerequisites/Co-requisites:
Pr. CSC340 (and senior standing) or permission of instructor

For Whom Planned:
Computer Science undergraduate students

Instructor Information:
Dr. Shan Suthaharan, 161 Petty Building, (Phone: 256-1122, email: ssuthaharan@uncg.edu)

Catalog Description:
Application of fundamental knowledge and skills in computer science to solve real-world problems and to develop research and development skills

Course Objectives:
Students in the Senior Project course will:

1. develop openness to new ideas in computer science, develop the ability to draw reasonable inferences from observations and learn to formulate and solve new computer science problems and situation using analytical and problem-solving skills;

2. develop the ability to synthesize and integrate information and ideas, develop the ability to think creatively, develop the ability to think holistically and develop the ability to distinguish between facts and opinion;

3. develop the ability to work individually and as part of a team, develop a commitment to accurate work, develop management skills, improve speaking and writing skills, improve the ability to follow directions, instructions and plans, and improve the ability to organize and use time effectively;

4. develop a commitment to personal achievement, the ability to work skillfully, informed understanding of the role of science and technology, a lifelong love of learning, and cultivate a sense of responsibility for one’s own behavior and improve self-esteem/self confidence.

Student Learning Outcomes:
On completion of the senior projects students will

1. (Knowledge, Comprehension) identify project/research problems; understand information and grasp meaning; translate knowledge into new context; use information, methods, concepts, and theories of fundamental topics in computer science in new situations;

2. (Application and Evaluation) apply computer science principles and practices to a real-world problem; demonstrate in-depth knowledge in the area of the project they have undertaken; solve problems using required knowledge and skills; implement and test solutions/algorithms;
3. **(Analysis)** identify potential solutions/algorithms for the project problem; see patterns and modularize the problem, recognize hidden meanings and identify components, show proficiency in software engineering principles;

4. **(Synthesis)** create new ideas using the old ones; generalize from given facts in the project they undertake, relate knowledge from several areas in systematic scientific approach, predict and draw conclusions relevant to the project they undertake;

5. **(Team Work)** show evidence (group collaboration, regular meetings, email communications, significant knowledge and skills contributions, etc.) of working productively as an individual and in team on a project that produces a significant software product;

6. **(Communications)** show evidence of competency in oral and written communications skills through oral presentations (project presentation, department seminar or conferences), technical reports and/or published research papers in conferences and/or journals.

7. **(Lifelong Learning)** use modern techniques, skills and tools necessary for computer science practices relevant to the project they undertake; use techniques in the recent research papers to solve problems.

**Teaching Strategies:**

Students are expected to choose an appropriate project/research topic in consultation with their instructor. Students must carry out a requirements elicitation/analysis or literature survey and then identify potential solutions to the problems stated in the project. Students are expected to attend regularly assigned class meetings and individualized conference sessions. Students are also expected to attend their regular group meetings. Students must develop projects that will demonstrate that they have a working knowledge of the basic and advanced concepts in computer science and also demonstrate a reasonable knowledge of recent development in computer science. Each project should include software development that has been approved by the instructor. **Students write preliminary reports and present seminars that describe the project background, proposed solutions and plans to implement and test the solutions.** The next major phase of the project is to implement and test the solutions to the identified problems. On completion students will produce a technical report and deliver an oral presentation. In case of a group project, every member of the group must present (oral and technical report) the entire project activities pointing out their contribution to make the project successful.

**Evaluation Methods and Guidelines for Assignments:**

The project will be graded for content, correctness, method of presentation (oral and technical report), team work (in case of group project) and the demonstration of the student’s knowledge in the computer science field.

<table>
<thead>
<tr>
<th>Topics: WI-writing intensive, SI-speaking intensive and TC-technical content</th>
<th>WI</th>
<th>SI</th>
<th>TC</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project definition or Requirements Specification (Progress Report 1):</td>
<td>3%</td>
<td>3%</td>
<td>4%</td>
<td>10%</td>
</tr>
<tr>
<td>System/Algorithm Analysis (Progress Report 2):</td>
<td>2%</td>
<td>2%</td>
<td>6%</td>
<td>10%</td>
</tr>
<tr>
<td>System/Research Design (Progress Report 3):</td>
<td>3%</td>
<td>3%</td>
<td>4%</td>
<td>10%</td>
</tr>
<tr>
<td>Coding and Testing (Progress Report 4):</td>
<td>2%</td>
<td>4%</td>
<td>4%</td>
<td>10%</td>
</tr>
<tr>
<td>Implementation and Conversion (Progress report 5):</td>
<td>-</td>
<td>2%</td>
<td>3%</td>
<td>5%</td>
</tr>
<tr>
<td>Evaluation (Progress Report 6) will be divided into:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Final Presentation:</td>
<td>-</td>
<td>8%</td>
<td>2%</td>
<td>10%</td>
</tr>
<tr>
<td>Source Code and User Manual</td>
<td>5%</td>
<td>3%</td>
<td>7%</td>
<td>15%</td>
</tr>
<tr>
<td>Technical Report/Research Papers:</td>
<td>8%</td>
<td>-</td>
<td>2%</td>
<td>10%</td>
</tr>
<tr>
<td>Client Evaluation:</td>
<td>3%</td>
<td>3%</td>
<td>4%</td>
<td>10%</td>
</tr>
<tr>
<td>Instructor Evaluation:</td>
<td>4%</td>
<td>2%</td>
<td>4%</td>
<td>10%</td>
</tr>
</tbody>
</table>
Speaking and Writing Requirements

Students are required to write multiple reports and give multiple presentations during the course of the semester. While the instructor will give feedback and assist students in improving the quality of their written and oral communication, students are advised that additional help is available from the following university centers:

- The University Writing Center provides assistance to students in writing tasks from organizing thoughts to the mechanics of effective writing. The Writing Center provides individual consulting for students through either face-to-face or online sessions. For more information, please refer to the Writing Center website at http://www.uncg.edu/eng/writingcenter/

- The University Speaking Center provides tutoring and services that help students improve their oral communication skills. Consultants at the Speaking Center can provide assistance in the preparation and delivery of speeches, as well as assistance in developing group or team communication skills. For more information, their website is http://speakingcenter.uncg.edu

Required Texts/Readings/References:

To be determined by student or group of students, with approval of instructor based on the research or project topic.

As an example, for a computer system design project the following text books may be used:

2. Red Hat Linux for Dummies, Jon Hall and Paul G. Sery, IDG Books, 2000. (or similar reference)

For a software engineering project the following books may be used:


ETS Exam:

This course has been designed to satisfy many of the ABET accreditation requirements. Therefore students are required to take the ETS exam to pass this course. ETS exam fees will be paid by the department.

Mandatory Requirement:

All projects must have at least two design choices and analyze tradeoffs between these choices. These design choices must be implemented and evaluated by means of subjective or objective methodologies.

The project that does not satisfy this requirement will receive 0% and the student will fail the course.
## Topical Outline:

<table>
<thead>
<tr>
<th>Project Stage</th>
<th>Submission Requirements</th>
<th>Date Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Proposal Outline of proposal</td>
<td>Week 1</td>
</tr>
</tbody>
</table>
| 1             | Project Definition, & Requirements Specification Progress Report 1 and **Oral Presentation**  
  - Approach and System profile  
  - Feasibility and Draft models | Week 3   |
| 2             | System/Algorithm Analysis Progress Report 2 and **Oral Presentation**  
  - Systems/Algorithm Analysis report with data & process models  
  - Data Dictionary | Week 5   |
| 3             | System/Research Design Progress Report 3 and **Oral Presentation**  
  - Amended models  
  - Detailed designs and controls | Week 7   |
| 4             | Coding & Testing Progress Report 4 and **Oral Presentation**  
  - Plan and Testing | Week 10  |
| 5             | Implementation & Conversion Progress Report 5 and **Oral Presentation**  
  - Conversion plan  
  - Training plan  
  - Preliminary demonstration | Week 12  |
| 6             | Evaluation Progress Report 6 and **Oral Presentation**  
  - **Final presentation**  
  - Source code and User Manual  
  - Technical report  
  - Evaluation by Client and Instructor | Week 13  |
| 7             | Final Report Due | Week 14  |

### Useful Steps to Accomplish the Senior Project Requirements:

- Obtain a course syllabus from the Instructor;
- Set up an email group and project home page;
- Submit an initial project proposal describing in detail your project/research topic and how you plan to treat this topic;
- Submit to the Instructor progress reports;
- Attend all assigned class meetings, conferences and oral presentations;
- Submit the final findings and results as technical report (in case of group project every member of the group must write papers on their work as part of the technical report);
- Give oral presentation on completion of the project (in group project every member of the group must give presentation)
Academic Honor Code:

Each student is required to sign the Academic Integrity Policy on all major work submitted for this course. Your work in this course is bound by the following policies: Students should note that cheating is regarded as a very serious offence which is likely to lead not only to failure in the course but also to additional penalties, refer http://studentconduct.uncg.edu/policy/academicintegrity/. Students should carefully note that the taking of any unauthorized material into examination, such as notes, unauthorized materials and copying from other students would be regarded as cheating. Students should also note that essays, assignments and other work are generally understood to be the student’s own work and where such work is identical with, or similar to, another student’s work, an assumption of cheating may arise. Where students wish to undertake work in conjunction with other students, it is suggested that the matter be discussed with the instructor.

Attendance Policy:

Regular attendance is required. You may be dropped from the course for missing more than two meetings.

Additional Requirements:

Collaboration on the group project is encouraged. However, it is student responsibility to highlight their work in the documents presented for assessment. Students are expected to be familiar with and to follow the UNCG Integrity Policy at http://studentconduct.uncg.edu/policy/academicintegrity/.

Due dates: Late work will not be accepted. Make arrangements with the instructor to turn in work early if you will not be in class on the due date.

Missed exams (if any) may be taken only if the student's absence has been excused by the instructor and if the exam is made up on the make-up exam time announced by the instructor.