Summer 2008

CEG 460/660-01: Computer Engineering

Eric Maston
Wright State University - Main Campus

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi

Part of the Computer Engineering Commons, and the Computer Sciences Commons

Repository Citation
https://corescholar.libraries.wright.edu/cecs_syllabi/1256

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.
Course Description

This course is concerned with the techniques of designing and constructing large programs. Some of the required basic concepts necessarily have to be developed using small programs as examples. To this extent, we also study programming-in-the-small. The overall objectives are to present an overview of issues in the development of software, to discuss terminology, to illustrate via example case studies, and to give sufficiently detailed advice on how to develop quality software. Hands-on experience is emphasized through the use of homework and a class project.

Goals

There are several goals in CEG 460/660:
1. Develop understanding of software engineering processes.
2. Learn how to use software engineering tools (Netbeans/UML).
3. Understand the big picture of software development.
4. Understand all of the things that are not in the book!
5. Have some fun!

Class Details

Lecturer: Eric Matson
Office: 336 Russ Engineering Center
Phone: 937-775-5108
Office Hours: M/W 3:00-4:00 at Russ 336 or by appt.
Email: eric.matson@wright.edu
Web: WebCT
Class: 6:05 - 7:20 (pm) MW, Health Sciences 140
Platform: Netbeans 6.0 with UML, available for free from netbeans.org
Prerequisites

For this class, the official prerequisite is CS 400 or CS 600. Please let me know the first lecture if you do not meet this prerequisite, and we can talk about your preparation if it differs.

Grading

Homework 15%
Projects 35%
Midterm Exam 25%
Final Exam 25%

The base scale is: A: 90-100, B: 80-89, C: 70-79, D: 60-69, F: 0-59. This is the highest requirement that will be used. The scales may be lowered or revised if necessary.

Project

There will be a term project in class in which you will develop a significant application. The project will be worked in teams. You may pick your partners or I will pick them. More details on the project will be handed out later.

Policies and Notes

• Attendance: Attendance is not required, nor will it be taken after the first couple of lectures. If you are not a regular attendee, it will be your responsibility to seek out what material was covered in the lecture and learn it. Most of my exam questions will be taken directly from ideas covered during the lecture, so it greatly helps if you attend!

• I will utilize WebCT to post updates to the course, sample code, projects, announcements, schedule, etc. Get in the habit of checking it regularly.

• The prerequisites of the course are basic understanding of high-level development in some object language and object oriented concepts. If you are not confident in your skills or do not have the required prerequisites, then visit with me and I can evaluate how to catch you up to the appropriate level and develop a plan to do so.

• Always make back ups of all of your work. Never have just one copy of anything! This way, when your dog eats your laptop the day before the final project is due, you will not have a problem (except getting a new laptop).

• If you are going to miss an exam, for any reason, discuss it with me in advance. If it is an emergency situation, please notify me as soon as possible.

• You can reach me a number of ways. Email is normally the best as I check it about 18 hours a day normally. You can also reach me by phone during the day at 775-5168. If you need human contact either stop in during my office hours, make an appointment, or just come by
my office. If I am in and not on a deadline to get something else completed, I will normally try to help as much as possible.

- There are technologies we will use in this class that you may not already know, such as UML, file transfer, command line, text editors, file systems, etc. We will cover some of these technologies as we go.

- The key to learning in this class will be spending time working through the problems. Do not wait until 2 hours before something is due to try to learn the concept and then complete a design. This normally ends in a disaster! Stay up with the readings and spend lots of time thinking through problems.

**Academic Misconduct**

In this class, the only way to truly learn the concepts to is do the work yourself. You will work with others in the class. Teamwork is an essential element of the course, but you are still expected to contribute and add to the class and your team.

Work that has obviously been copied or in the more extreme case, when the original authors name has not even been changed, both parties will receive a 0 grade for that assignment. Both parties will also be turned over to the Office of Judicial Affairs.
## Schedule

(always subject to changes) Always have readings scheduled for that day complete prior to the class meeting.

<table>
<thead>
<tr>
<th>#</th>
<th>Date</th>
<th>Topic</th>
<th>Reading</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jun 16</td>
<td>Introduction</td>
<td>i</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Jun 18</td>
<td>Software Lifecycles</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Jun 23</td>
<td>Software Lifecycles</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Jun 25</td>
<td>Software Process, Teams</td>
<td>3, 4</td>
<td>Abstract</td>
</tr>
<tr>
<td>5</td>
<td>Jun 30</td>
<td>Requirements</td>
<td>10, 15</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>July 2</td>
<td>Requirements</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Jul 7</td>
<td>Analysis</td>
<td>11</td>
<td>Requirements</td>
</tr>
<tr>
<td>8</td>
<td>Jul 9</td>
<td>Analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Jul 14</td>
<td>MIDTERM EXAM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Jul 16</td>
<td>Design</td>
<td>12</td>
<td>Analysis</td>
</tr>
<tr>
<td>11</td>
<td>Jul 21</td>
<td>Design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Jul 23</td>
<td>Implementation</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Jul 28</td>
<td>Implementation</td>
<td></td>
<td>Design</td>
</tr>
<tr>
<td>14</td>
<td>Jul 30</td>
<td>Testing</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Aug 4</td>
<td>Tools</td>
<td>5</td>
<td>Construction</td>
</tr>
<tr>
<td>16</td>
<td>Aug 6</td>
<td>Planning, Estimating</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Aug 11</td>
<td>Projects</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Aug 13</td>
<td>Projects</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Aug 18</td>
<td>Projects</td>
<td></td>
<td>Testing/Final</td>
</tr>
<tr>
<td>20</td>
<td>Aug 25</td>
<td>FINAL EXAM</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>