COURSE DESCRIPTION

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Course Type</th>
<th>Course Title</th>
<th>Coordinator</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSC 340</td>
<td>Required</td>
<td>Software Engineering</td>
<td>Nancy Green</td>
</tr>
</tbody>
</table>

**Sem. Hours**: 3

**Current Catalog Description:**

Practical and theoretical concepts of software engineering

**Textbook:**


**References:**

None

**Course Outcomes:**

Upon completion of the course students should be able to

1. Demonstrate knowledge of principles and terminology of the field of Software Engineering
2. Demonstrate knowledge of object-oriented modeling techniques (UML)
3. Apply knowledge in 1-2 to the requirements, analysis, design, implementation, and testing of a software system in a team project; and present project deliverables in written form
4. Demonstrate knowledge of software teams and software engineering ethics
5. Demonstrate knowledge of software evaluation

**Activities Enabling Program Outcomes (POx refers to program student outcome x)**

*Instruction:* The lectures in this course cover techniques of software engineering for medium to large software projects including requirements-analysis-design-implementation-testing (POc), teamwork: how software teams are organized and how successful teams operate (POd), professional ethics (POe), components of requirements-analysis-design documentation (POf), classical and UML modeling techniques (POi), software lifecycle models and software architecture (POk).

*Student Activities and Assessment:* Every offering of this course will include (details of assessment criteria and expectations are in outcome rubrics):

- One or more assignments in which students identify, analyze, and clearly document system requirements and design, using current software engineering techniques (POc, POfw, and POi)
- One or more assignments in which students utilize classical and UML diagrams for software design and development (POk)
- One or more program in which students implement a system consistent with a design document, using current software engineering techniques (POc and POi)
- One or more assignment or quiz/test question in which students demonstrate an ability to evaluate a product in terms of requirements, design, and implementation, and testing (POc and POfw and POi)
- One or more test/quiz questions in which students to identify a professional, ethical, legal, security, or social issue and the stakeholders involved (POe)
- One or more test/quiz questions in which students demonstrate understanding of principles of effective teamwork (POd)
- A survey and peer assessment exercise in which students demonstrate an understanding of teamwork and an ability to function effectively on a team (POd)

**Prerequisites by Topic:**

<table>
<thead>
<tr>
<th>Students must have</th>
</tr>
</thead>
<tbody>
<tr>
<td>a grade of at least C (2.0) in CSC 330 (Advanced Data Structures)</td>
</tr>
</tbody>
</table>

**Major Topics Covered in the Course:**

- Introduction to Software Engineering including Software Engineering Ethics
- Software Life Cycle Models
- Software Process
- Software Teams
- Introduction to Testing
- Software Project Planning and Estimating
- Requirements and UML
- Object-oriented Analysis and UML
- Modules and Design
- Reuse and Portability
- Implementation
- Integration and Testing
- Maintenance
- Special Topics (e.g. Software Architecture, User Interface Design)

**Estimated Curriculum Category Content (Semester hours):**

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>0</td>
<td>0</td>
<td>Software design</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Data structures</td>
<td>0</td>
<td>0</td>
<td>Prog. Languages</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Comp Org &amp; Arch</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>