Minnesota State University, Mankato

Curriculum Proposal

Please type or select the requested information. Print completed forms, add appropriate paper attachments, and route through MSU's curricular process for recommendations and decisions.

(Check all that apply):
College: Science, Engineering and Technology
Department: Computer and Information Sciences
Program: Computer Information Science
Type of Change: COURSE PROPOSALS
Proposed Title: Information Technology Capstone

Course Designator and Number: IT 498
Number of Credits: 4

Proposal #: 111
Effective Date of Change: N(4-07)
Academic Year: N(4-07)
(For Office Use Only)

Include a course or program description for the Bulletin (30-40 words maximum for courses, 100 for programs):
Develop high quality software application researching and applying fundamental software engineering techniques. Several advanced development and test tools, human factors of interface design and a team approach, each student controlling only a part of the system.
Pre: Permanent admission to IT, completion of all core courses, and consent.
Fall, Spring

Rationale or Justification for change:
The CIS major is being redesigned and name changed to Information Technology (IT). The above change is proposed to be consistent with the IT major, it's designator and focus.

***For General Education or Cultural Diversity Courses Only***

General Education Course:

<table>
<thead>
<tr>
<th>GE Category #</th>
<th>GE Category Name (Maximum of 3 Categories)</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td></td>
</tr>
</tbody>
</table>

Cultural Diversity Course:

(Please check one.)

- Core (At least 75% devoted to topics of race, gender, sexual orientation, age, class, and disabilities as they occur in United States Society.)
- Related (At least 25% devoted to the above topics or to a global perspective on topics related to African American, Asian, Hispanic, and Native American inhabitants of the United States.)

Attach paper copies of the following:

a. Syllabus or course outline.
b. Course's student learning outcomes associated with each GE competency or CD designation.
c. List of strategies to be used to assess students' achievement of each GE competency or CD designation.

***For New Courses***

(See all that apply): Instructional Type: Lecture
Course is an elective.
Course is required for program
Pre- or Co-requisites: Permanent admission to IT, completion of all listed core courses.
Other courses are being changed or eliminated. (Explain.)

Course will be offered:

- Fall Semester
- Spring Semester
- Summer Session

- Course content or title is similar to courses in other departments. (Attach copy of letter of agreement with other program(s) contacted. Indicate the nature of the discussions and/or resolution of differences or potential conflicts.)

Attach paper copies of the following:

a. Syllabus or course outline.
b. Course's student learning outcomes.
c. A list of resources required to offer and support this course.
d. A description of how teaching this course will affect department staffing.
e. If 400/500 level course, an explanation of added expectations of graduate students.

Revised September 2002
# Signature Page

**Department**
- [ ] Recommended
- [ ] Not Recommended

**College Curriculum Committee**
- [ ] Recommended
- [ ] Not Recommended

**College Dean**
- [ ] Recommended
- [ ] Not Recommended

**General Education Subcommittee**
- [ ] Recommended
- [ ] Not Recommended

**Undergraduate Curriculum and Academic Policy Committee**
- [ ] Recommended
- [ ] Not Approved

**Faculty Association Graduate Committee**
- [ ] Recommended
- [ ] Not Recommended

**Graduate Dean**
- [ ] Recommended
- [ ] Not Recommended

**Academic Affairs Council**
- [ ] Recommended
- [ ] Not Recommended

**Senior Vice President and Vice President for Academic Affairs**
- [ ] Approved
- [ ] Not Approved

---

*Revised September 2002*
IT 498 (4) Information Technology Capstone

a. SYLLABUS

Textbook:


The Mythical Man Month, Fredrick Brooks, Addison-Wesley, ISBN: 0201835959


Text book related to the area of specialization in which the project is to be completed.

Prerequisites:

All core courses listed in the major, at least two courses from the specialization, consent

Course Objectives:

The purpose of this course is to teach students how to develop a significant software application in the area of their specialization by applying fundamental software engineering techniques. In this course the students will apply the knowledge of the IT fundamental courses to the development and delivery of a high quality and secure software project. In this course, the student will:

- Understand the state of modern software development
- Research on a topic of software development
- Derive and specify software requirements
- Design a software system using UML
- Build a software system from the design
- Verify the software system against requirements
- Use several advanced software system development and test tools
- Document the software system for users
- Explore other important topics in Software Engineering including J2EE programming, Software Performance Engineering, Component Based Development, Configuration Management
- Apply human factors of user interfacing
- Include security aspect
- Demonstrate Programming Skills (Java, .NET etc)
- Work as part of an organized software team to design and develop a large multi- step project in which each person has control of only part of the system.
- Follow professional codes of ethics
- Develop effective communication skills
- Present work in a professional manner.
Course Coverage:

Software Design
Programming Languages
Software engineering process.
Software team organization and management
Software development environments and tools
Application of Unified Modeling Language (UML).
Data structures and specification.
Advanced object-oriented design.
Testing and debugging.

Expected specialization area coverage: The students will cover one or more the following topic areas depending on the project selected.

Modern Programming Languages
Networking and Telecommunications
Data Management
Data mining
Artificial Intelligence
Information Assurance and Security
Analysis and Design:

Theoretical Content: Depends on project

Analysis and Design: Design and implementation of project

Computing Platform: To be determined by student, with approval of instructor.

Course Contents:

<table>
<thead>
<tr>
<th>Number of Weeks</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project related activities (in and outside class hours):</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Introduction, finalization of topic area</td>
</tr>
<tr>
<td>3</td>
<td>Research on the topic area, derive and specify requirements</td>
</tr>
<tr>
<td>3</td>
<td>Design the software system</td>
</tr>
<tr>
<td>3</td>
<td>Implementation and testing of individual component</td>
</tr>
<tr>
<td>3</td>
<td>Integration of the individual components and testing</td>
</tr>
<tr>
<td>2</td>
<td>Documentation</td>
</tr>
<tr>
<td>In class lecture:</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Review and guide to software engineering principles, tools and methodologies</td>
</tr>
<tr>
<td>2</td>
<td>Human factors of user interfacing</td>
</tr>
<tr>
<td>2</td>
<td>Security aspects in the software project</td>
</tr>
</tbody>
</table>
2 Professional codes of ethics
2 Documentation Guide
2 Oral Presentation
2 Individual and team activity

**Catalog Description:**
In this course students will develop a significant and high quality software application in their area of specialization by researching and applying fundamental software engineering techniques, several advanced development and test tools, human factors of interface design and a team approach with each student controlling only a part of the system.

**Grading:**
- Homeworks & Quizzes 30%
- Final Project 70%
  - Requirements Document 20% of project grade
  - Design Document 20% of project grade
  - User Manual 10% of project grade
  - Delivery 50% of project grade

Assessment criteria include: group meetings, oral presentations, written reports,

**b. LEARNING OUTCOMES**
After completing this course the student will be able to:
- Understand the state of modern software development
- Derive and specify software requirements
- Identify security vulnerabilities
- Design a software system using UML
- Build a software system from the design
- Verify the software system against requirements
- Use several advanced software system development and test tools
- Document the software system for users
- Apply human factors of user interfacing to software development
- Work as part of an organized software team to design and develop a large multi-step project in which each person has control of only part of the system.
- Understand professional codes of ethics
- Develop effective communication skills
- Present work in a professional manner.

**c. RESOURCES REQUIRED TO OFFER AND SUPPORT THIS COURSE**
Resources currently in place within the department and the University Library will support this new course. No new resources are required.

**d. IMPACT ON STAFFING IN THE DEPARTMENT**
There is no impact on department staffing.