COURSE DESCRIPTION

Department and Course Number: COMP 4710
Course Title: Senior Design Project
Total Credits: 3
Required: Yes (SWEN, WIRS)
Prerequisites: COMP 3700 or COMP 3710
Class meetings per week: 3 hours
Lab meetings per week: 0 hours
Course Coordinator: Dr. David Umphress
Date Prepared: February 18, 2004

Current Catalog Description:
Development of requirement definitions, architectural design specification, detailed
design specification, testing plan and documentation for the software and/or hardware
components of a comprehensive project.

Textbooks:
None.

References:
None.

Course Objectives:
1. To experience an end-to-end software development project using sound software
   engineering practices and process.
2. Work effectively as members of a software development team.
3. Communicate professional work to others in written, oral, and electronic forms.

Prerequisites by Topic:
1. Software design and construction
2. Fundamentals of computer science

Topics Covered: (specify number of hours on each)
There are no lecture topics per se, owing to the nature of the course. However,
approximately four to five hours of instruction in Extreme Programming (XP) is provided
during the first two weeks of the semester.

Laboratory Projects: (specify number of weeks on each)
All fifteen weeks of the semester are devoted to completing a non-trivial software
engineering project. A typical schedule of activities follows.
1. XP introduction (1.5 weeks)
   During this time the instructor will provide a detailed description of the XP-based
   process adapted for the Senior Design course. This description will be delivered to
   the students in written form and explained/reviewed during scheduled class meetings.
   Additionally, the customer will have the opportunity to meet her team, explain the
vision of the project informally, and prepare an initial set of User Stories with the team.

2. **Ad hoc project (2 weeks)**
   In parallel to the “XP Introduction”, each team must experiment/research/develop as deemed necessary by each member (including customer) in an ad hoc manner. This may include developing micro-applications that perform some aspects of the project, familiarizing the team with development environments/languages, and reviewing existing applications that provide for similar needs as the project. The primary stipulations are 1) the team attempt to accomplish as much of the project as possible, 2) all work can be mapped to the vision of the customer, 3) each element of the customer vision can be mapped to some work, and 4) all work must have some deliverable form (i.e., it cannot be stated that “Java was learned”, rather Java applications demonstrating those concepts learned much be presented). The intent is to allow each team to focus on what they do not know, in any-order, by any means, without the burden of formality. This will hopefully afford insights that might otherwise be undiscovered due only to formal scheduling and process.

3. **Delivery and demonstration of ad hoc project (1 class meeting)**
   To close the “Ad Hoc Project” period, each team will deliver a report of works completed along with a brief, in-class presentation of these works in their demonstrable form. The works demonstrated here will account for a substantial portion of the final grade (15%) to ensure the work is addressed with an appropriate degree of professionalism. This grade cannot be determined upon delivery; rather the demonstrated works must be compared against the work and efforts that follow over the development cycles to determine a grade based on 1) the usefulness of the work to the project, 2) whether appropriate decisions are made relative to gained insight, and 3) whether there is an explicit mapping between these works and components of the project.

4. **Design cycles (3 – 3.5 weeks each, total of four cycles during semester)**
   There will be four XP development cycles consisting primarily of user story design, analysis, tracking and refactoring, test code development, product code development, and project document development. At each class meeting during a cycle, team status will be assessed through an interview with the instructor and/or TA.

5. **Customer installations (2 class meetings)**
   As part of the customer acceptance test, the “Final Delivery” for each project will be evaluated at a high level, in-class, during the final two class meetings. This evaluation will concentrate on deployment and basic usage issues. This includes (but is not limited to) installation to a clean target device, uninstall, user documentation (both paper and electronic), etc. The protocol for this will be simple... the instructor/TA/customer attempt to install the product on a target device, use the product in some typical manner, and uninstall it from the device, all the while referencing the user documentation for help. No team members will be allowed to address the evaluator without first being prompted.

6. **Final Presentations (1 class meeting)**
   All teams must make a formal, oral presentation of their work. This is to include, but is not limited to, a presentation of requirements, analysis, design, implementation, and testing issues.
Oral and Written Communications:
Written: There is substantial written communication required of all students. Students construct a software development notebook describing the technical details of their software for each design cycle. They also document in written form project status reports, process descriptions, and lessons learned.
Oral: There is substantial oral communication required of all students. Students report weekly on the status of their projects. They also make a formal oral presentation describing the progress of the project at multiple points in the semester.

Social and Ethical Issues:
None.

Theoretical Content:
None.

Problem Analysis and Solution Design:
This course is designed to ensure that students experience an end-to-end software development project using sound software engineering practices and process. There is substantial analysis and design content.