***CSE 611 – MS Project Development***

**Course Information**

Lectures and project during a 14-week semester

Instructor(s): Alan Hunt (Office 354 Davis Hall) / Michael Buckley (327 Davis Hall)

|  |
| --- |
|  |

**Course Description**

This course is intended to help MS students select, plan, execute, document, and demonstrate a nontrivial programming or computer engineering project using robust software development lifecycle (SDLC), project management, and hybrid agile development methodologies. The course will contain a limited lecture component to introduce students to the appropriate methodologies, and then focus on establishing project success factors, developing a project plan, and holding project management and review sessions to help ensure tracking to overall goals.

The projects available for the students to select will be crowd sourced from CSE faculty and/or relevant industry, with guidelines provided to ensure an equitable level of complexity is encompassed in order to provide an effective experiential learning outcome. Students may also propose their own project to pursue as long as it meets the overall scope and complexity requirements of the course as defined by the instructor. All projects considered for this course will be from the computer science and engineering discipline, and will require the application of software and or computer engineering techniques to produce a tangible product, such as a web, mobile, or database driven application.

There will also be a presentation component at the end of the course where students demonstrate their projects and summarize their learnings in a formal presentation to a wider audience.

While it is certainly the best outcome that the course end with a working piece of software that accomplishes a specific aim, a well documented presentation that captures any failures with suggestions on how they might be remediated can also lead to a successful outcome.

This course can be used to satisfy the CSE MS Program project requirement, provide the student achieves at least a B+ grade in the course.

**Course Prerequisites**: Appropriate programming or engineering background to execute a larger scale project successfully. Students should have taken at least two programming oriented classes before attempting this course.

Students also need to have taken at least one CSE 500 level course.

**Learning Outcomes**

At the conclusion of this course, students should be able to do the following:

* Write and revise a detailed requirements document capturing both user and technical implementation details
* Write and revise a project plan, including creation of milestones, assignment of project tasks, and identification of incremental deliverables
* Write and revise a test plan, indicating how requirements will be tested to ensure the final product confirms to specifications
* Collaboratively implement a complex software or computer engineering project
* Create and present project outcomes in written and oral form, including review of any outstanding issues or nonconformance to specifications

**Textbook**

None.

**Course Schedule**

**Course Requirements**

* Meeting attendance and participation is expected
* There will be one project
* Regular deliverables on the project will be graded during the course
* Nontechnical deliverables such as requirements, documentation, and presentation materials will also be part of the course requirements

**Grading Policy**

All deliverables are due on the date and time specified. Late assignments can be submitted at 20% penalty. A late assignment is accepted up to 24 hours after the original deadline.

Learning assessments will be graded based on rubric criteria and weighted according to the following break-down.

|  |  |
| --- | --- |
| Weighting | Assessment / Assignment |
| 40% | Project documentation and meetings |
| 60% | Project deliverables |
| 100% |  |

Final Grades:

|  |  |  |
| --- | --- | --- |
| Grade | Quality Points | Percentage |
| A | 4.0 | 92.0% -100.00% |
| A- | 3.67 | 86.0% - 91.9% |
| B+ | 3.33 | 80.0% - 85.9% |
| B | 3.00 | 76.0% - 79.9% |
| B- | 2.67 | 72.0% - 75.9% |
| C+ | 2.33 | 66.0% - 71.9% |
| C | 2.00 | 62.0% - 65.9% |
| C- | 1.67 | 58.0% - 61.9% |
| D+ | 1.33 | 50.0% - 57.9% |
| D | 1.00 | 45.0% - 49.9% |
| F | 0 | 44.9% or below |

Incompletes (I/IU)\*: A grade of incomplete (“I”) indicates that additional course work is required to fulfill the requirements of a given course. Students may only be given an “I” grade if they have a passing average in coursework that has been completed and have well-defined parameters to complete the course requirements that could result in a grade better than the default grade. An “I” grade may not be assigned to a student who did not attend the course.

Prior to the end of the semester, students must initiate the request for an “I” grade and receive the instructor’s approval. Assignment of an “I” grade is at the discretion of the instructor.

The instructor must specify a default letter grade at the time the “I” grade is submitted. A default grade is the letter grade the student will receive if no additional coursework is completed and/or a grade change form is not filed by the instructor. “I” grades must be completed within 12 months – see the [Incomplete Grade Policy](http://undergrad-catalog.buffalo.edu/policies/grading/explanation.shtml%20-%20incomplete) for the schedule. Individual instructors may set shorter time limits for removing an incomplete than the 12-month time limit. Upon assigning an “I” grade, the instructor shall provide the student specification, in writing or by electronic mail, of the requirements to be fulfilled, and shall file a copy with the appropriate departmental office.

**Academic Integrity**

Academic integrity is a fundamental university value. Through the honest completion of academic work, students sustain the integrity of the university while facilitating the university's imperative for the transmission of knowledge and culture based upon the generation of new and innovative ideas.

Academic dishonesty will not be tolerated in this course and may result in grade reductions, failed assignments, failed classes, or expulsion from UB according to the university and CSE policies.

<http://undergrad-catalog.buffalo.edu/policies/course/integrity.html>

<http://www.cse.buffalo.edu/undergrad/policy_academic.php>

No behavior that compromises academic honesty (such as use of someone else's work or code, using prohibited materials during tests, or making your work available to others) will be tolerated in this course. You are expected to complete assignments on your own. You are encouraged to collaborate on the approach and discuss ideas to complete assignments, however it is essential that you complete all programming and writing assignments on your own. The submitted assignments will be checked for plagiarism, which will be triggered by programs written in close collaboration. It is expected that your work represents your own understanding of the problem. If work of others is used, it must be properly cited. Use of properly cited material is acceptable, but no referencing is treated as claiming the work as your own. If you need assistance with anything, do not hesitate to contact the instructor.

**Accessibility Resources**

If you have any disability which requires reasonable accommodations to enable you to participate in this course, please contact the Office of Accessibility Resources, 25 Capen Hall, 645-2608, and also the instructor of this course .. The office will provide you with information and review appropriate arrangements for reasonable accommodations. http://www.student-affairs.buffalo.edu/ods/

The University at Buffalo and the School of Engineering and Applied Sciences are committed to ensuring equal opportunity for persons with special needs to participate in and benefit from all of its programs, services and activities.