



Coversheet - Proposal for New and Revised Courses

(Use for non-Pathways courses)

For CLE/Pathways courses, form can be found here: <https://www.pathways.prov.vt.edu/proposal-forms.html>

General Information			
Proposal Date:	Fall 2023	Department:	Computer Science
Course Designator and Number (Cross-listed Course Designator and Number):		CS 5784	
Title of Course:	Software Project Management	Credit Hours:	3
Course Transcript (ADP) Title (30 Characters & Spaces Maximum):		Software Project Management	
Instructor and/or Departmental Contact:		Trey Mayo - Director of Graduate Programs	
Contact Phone:	X 0780	Contact E-mail:	treymayo@vt.edu
Please refer to Office of University Registrar for guidelines and policy requirements: https://registrar.vt.edu/governance.html			

Please count this course toward the following Scorecard Metrics areas:

- Study Abroad
 Service Learning
 Experiential
 Undergraduate Research
 Scorecard Metrics Definitions can be found here: <https://registrar.vt.edu/faculty-toolbox/scorecard-metrics.html>

Please insert an X if this course should count toward First Year Experience:



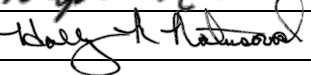
- First Year Experience (FYE) Include approval letter from FYE Director. More information can be found here: <http://www.fye.vt.edu>

Select ONE of the following boxes	
<input checked="" type="checkbox"/> New Course	<input type="checkbox"/> *Revised Course (Revision > 20% _____ Revision < 20% _____)

For CLE/Pathways courses, form can be found here: <https://www.pathways.prov.vt.edu/proposal-forms.html>

*Please include a summary of course revisions to the Justification section of proposal

A:	Attach statement from Dean or Departmental Representative as to whether teaching this course will require or generate the need for additional departmental resources.		
B:	Attach appropriate letters of support (e.g., prerequisite, corequisite, or cross-list memo) from affected departments and/or colleges.		
C:	Effective Semester:	Fall 2023	
D:	Change in Title From:		
	To:		
E:	Change in Transcript Title (ADP) From:	To:	
F:	Change in Credit Hours From:	To:	
G:	Change in Lecture and/or Lab Hours From:	To:	
H:	Course Number(s) and Title(s) to be deleted from the Catalog with <u>APPROVAL</u> :		

Approval Signatures			
Department Representative		Date	3/22/2023
College Curriculum Committee Rep		Date	3/23/2023
College Dean or Designee		Date	3/23/2023

Course Information

Catalog Description

Principles, activities, and tools relevant to the project management of software systems using agile methods. Differences between traditional and agile project management approaches and develop a minimum viable product (MVP) using agile project management techniques. Student participation in a team-based format focusing on end-to-end software/systems planning, technical analysis, and development. Pre: Graduate standing. (3H, 3C).

Learning Objectives

Having successfully completed this course, the student will be able to:

1. Differentiate between traditional and agile project management approaches.
2. Use agile methods to lead a software development effort.
3. Create agile project management metric artifacts.
4. Develop an application using agile methods.
5. Communicate about a developed software system.

Justification

Software engineering involves developing and maintaining complex software systems. However, software projects frequently experience a variety of issues, such as being over-budget, behind schedule, and containing numerous bugs and errors that need repair. Further, while software engineering is a technical endeavor, it is also a social activity, involving human programmers working together to develop a cohesive and complex system. In the software industry, project managers are necessary to support the technical and social challenges in software engineering.

Project management is the set of activities to help a team to achieve product goals within given constraints. Software project management consists of the processes and tools to support programmers collaborating to build and maintain complex software systems for clients and users. In addition, project managers are responsible for delivering software products that meet the needs of customers and other stakeholders while also creating conditions and an environment where programmers can be productive and thrive.

This course is offered at the 5000 level because it builds upon the programming skills that computer science students have prior to entering the graduate program. Additionally, students will gain skills necessary to lead project teams and understand management theories and practice to apply to future careers. This course will complement existing courses in the graduate curriculum for the department, such as CS5704 (Software Engineering), CS5714 (Usability Engineering), CS5744 (Software Design and Quality), and CS5934 (Capstone Project). These courses are also mostly tech-oriented for students to gain experience programming a system utilizing specific phases of the software development lifecycle. CS5784 will further emphasize methods for constructing and maintaining quality software applications, while also providing an alternative option for students more interested in management and leadership positions.

Prerequisites and Corequisites

Pre: Graduate standing.



Texts and Special Teaching Aids

No textbook will be required for this course. However the course readings and lecture materials rely on content from research papers from software engineering and management relevant to project management techniques, and the following textbooks.

Recommended books:

- *Building software together*. (n.d.) <https://github.com/gvwilson/buildtogether.tech#readme>
- Heagney, J. (2016). *Fundamentals of project management* (5th ed.). American Management Association. pp. 250.
- Oram, A. & Wilson, G. (2010). *Making software: What really works, and why we believe it*. O'Reilly Media. pp. 624.
- Pressman, R. S. (2004). *Software engineering: A practitioner's approach*. Boston. pp. 880.
- Sadowski, C. & Zimmerman, T. (2019). *Rethinking productivity in software engineering*. Open Access. <https://library.oopen.org/handle/20.500.12657/22839>

Examples of Relevant Research papers:

- Boehm, B W. (1984). Software engineering economics. *IEEE Transactions on Software Engineering*, 10(1). <https://doi.org/10.1109/TSE.1984.5010193>
- Fernandez, D. J. & Fernandez, J. D. (2008). Agile project management - Agilism versus traditional approaches. *Journal of Computer Information Systems*, 49(2). <https://doi.org/10.1080/08874417.2009.11646044>
- Kalliamvakou, E. et al., (2018). What makes a great manager of software engineers? *IEEE Transactions on Software Engineering*, 45(1), 87-106. doi: 10.1109/TSE.2017.2768368
- Levitt, R. E. (2011). Towards project management 2.0. *Engineering Project Organizational Journal*, 1(3). <https://doi.org/10.1080/21573727.2011.609558>
- Oliveira, E. et al., (2016). Software project managers' perceptions of productivity factors: Findings from a qualitative study. *ESEM '16: Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 15, 1-6. <https://doi.org/10.1145/2961111.2962626>



Topic Syllabus

<u>Topic</u>	<u>Percent of Course</u>
Traditional and Agile Software Engineering/Development <ul style="list-style-type: none"> - Traditional (Waterfall) - Agile (SAFe, eXtreme programming, Scrum, Kanban) 	10%
Traditional and Agile Product Management Techniques <ul style="list-style-type: none"> - Program/Project Manager (Traditional) - Product Owner (Agile) 	10%
Agile Planning <ul style="list-style-type: none"> - Product Backlog Planning - Release Backlog Planning - Spring Backlog Planning 	25% 8% 8% 9%
Application Development with Agile Methods/Product Management <ul style="list-style-type: none"> - Developing a working application in two time-boxed sprints that will be demoed to the class at the end of the session. - Managing the development of the application using agile reporting metrics to track and automate the application progress in real-time using real-world cloud-based agile project management tools utilized in the software industry (i.e. Monday.com, Trello, Jira, etc.) - Working as a self-organizing and self-managing agile development team 	30% 5% 10% 15%
Agile Requirements <ul style="list-style-type: none"> - Developing user stories - Developing Acceptance Criteria - Developing a Definition of Done 	13%
Agile Estimation <ul style="list-style-type: none"> - Estimating user stories using agile techniques (e.g., planning poker) - Identifying the difference between relative estimation and traditional estimation 	12%
Total	100%

Old (Current) Topic Syllabus

N/A for new courses.



Department of Computer Science
620 Drillfield Drive
Torgersen Hall, Suite 3210
Blacksburg, Virginia 24061
P: (540) 231-0780
treymayo@vt.edu

March 22, 2023

To: Course Approval Committees

From: Trey Mayo, Ed.D. **Trey Mayo**
Director of Graduate Programs
Computer Science

Digitally signed by Trey Mayo
Date: 2023.03.22 11:10:07
-04'00'

The Department of Computer Science is requesting approval of a new course proposal CS 5784-Software Project Management. No new resources will be required in order to offer this course.

February 24, 2023

To: College and University Curriculum Committees
RE: Computer Science Course Revisions Package

The Department of Computer Science presents a coordinated package of new course proposals and course revisions for the purpose of reorganizing our graduate-level offerings in the core Computer Science domain of Machine Learning and related topics. Our current offerings are the result of uncoordinated individual actions often made in conjunction with other departments over several years. The result is a collection of courses with overlaps and inefficiencies that lead to confusion for our students.

This package is centered around a two-semester sequence directly covering the core of traditional Machine Learning topics, with the first course in the sequence also serving as the prerequisite for the other, related courses. This allows us to avoid duplicating background material across these courses. We support the core with a collection of three courses (one existing, two new) that span the generally recognized major topics related to Machine Learning: Natural Language Processing, Learning-based Computer Vision, and Deep Learning. As part of the overhaul, we break some existing cross-listings to courses long recognized as duplicative.

Our package includes the following new courses and course revisions.

- Two course sequence CS 5805-6 Machine Learning. CS 5805 partially duplicates existing course CS/STAT 5525. However, we do not currently seek any changes to CS/STAT 5525 since this course is presently integrated into other certificates and programs. Instead, we have an agreement with Statistics to support their future changes to CS/STAT 5525 to bring it more in line with their needs. CS 5806 will (within our curriculum) replace the role currently held by ECE 5424/CS 5824.
- We request to break the cross-listing agreement for CS 5824, leaving this as ECE 5424.
- We include a revision to CS 5814 Introduction to Deep Learning. In addition to minor topics list changes, CS 5805 will become the prerequisite course for CS 5814.
- We request to break the cross-listing agreement for ECE 6524/CS 6524 Deep Learning. Despite the names, this is largely duplicative with CS 5814. Breaking this cross listing will reduce existing confusion for both CS and ECE students.
- A new course proposal for CS 5624 Natural Language Processing. This course will have CS 5805 as a prerequisite.
- A new course proposal for CS 5864 Learning-based Computer Vision. This course will have CS 5805 as a prerequisite.

Implementing these proposals will leave our department with a collection of courses that properly represents

the major sub-fields within the broad area of Machine Learning and Artificial Intelligence, as is typical for major Computer Science Departments across the US.

We note that while there is duplication between CS 5805 and CS/STAT 5525, and between CS 5806 and ECE 5424, this should have relatively minor impact on the teaching loads for the departments or the health of any of these courses. Historically, we have taught multiple sections of each of these courses every year, involving instructors from each of the three departments. None of these courses will lack for students for the foreseeable future.

Implementing this package will require no new resources. The NLP and Learning-based Computer Vision courses are already in our teaching rotation having been run as special topics courses. So, all of these courses have already been taught in various forms, and are already built into our course offering structure.

Unrelated to the described collection of proposals, we also offer a new course proposal for CS 5784 Software Project Management. This course has been piloted twice already.

Sincerely,

A handwritten signature in black ink, appearing to read "Cliff Shaffer", with a long horizontal flourish extending to the right.

Clifford A. Shaffer
Professor and Associate Department Head
for Graduate Studies