

# Course Outline

---

School:	Eng. Tech. & Applied Science
Department:	Information and Communication Engineering Technology (ICET)
Course Title:	Computer Programmer Proj
Course Code:	COMP 231
Course Hours/Credits:	56
Prerequisites:	COMP 214, COMP 228, COMP 229, COMP 246
Co-requisites:	N/A
Eligible for Prior Learning, Assessment and Recognition:	Yes
Originated by:	A. Donin, G. Leitch
Creation Date:	Winter 2014
Revised by:	Hao Lac
Revision Date:	Summer 2016
Current Semester:	Winter 2017
Approved by:	<i>p pesikan</i> <i>l c/o</i>

---

Chairperson/Dean

---

*Students are expected to review and understand all areas of the course outline.*

*Retain this course outline for future transfer credit applications. A fee may be charged for additional copies.*

*This course outline is available in alternative formats upon request.*

## Course Description

This is the capstone course for Software Engineering Technician program and the first software development project course for all software engineering technology programs. The students are required to work in teams to design, implement, and document an application or a component for a hypothetical organization. This enables the student to simulate the activities of the software engineering processing using a variety of collaborative tools used in the computer industry.

## Program Outcomes

Successful completion of this and other courses in the program culminates in the achievement of the Vocational Learning Outcomes (program outcomes) set by the Ministry of Advanced Education and Skills Development in the Program Standard. The VLOs express the learning a student must reliably demonstrate before graduation. To ensure a meaningful learning experience and to better understand how this course and program prepare graduates for success, students are encouraged to review the Program Standard by visiting <http://www.tcu.gov.on.ca/pepg/audiences/colleges/progstan/>. For apprenticeship-based programs, visit <http://www.collegeoftrades.ca/training-standards>.

## Course Learning Outcomes

The student will reliably demonstrate the ability to:

1. Gather requirements for the proposed system.
2. Analyze and estimate the requirements using appropriate Agile methodologies.
3. Apply Agile methods and if applicable, non-Agile methods to solve design and implementation issues.
4. Design and utilize the appropriate database structures and tools.
5. Code and test the application.
6. Produce both user and system documentation.
7. Document the programming aspects of a system.
8. Produce timely and accurate reports on project status.
9. Demonstrate and defend the project produced in this course.

## Essential Employability Skills (EES)

The student will reliably demonstrate the ability to\*:

1. Communicate clearly, concisely and correctly in the written, spoken, and visual form that fulfills the purpose and meets the needs of the audience.
2. Respond to written, spoken, or visual messages in a manner that ensures effective communication.
4. Apply a systematic approach to solve problems.
5. Use a variety of thinking skills to anticipate and solve problems.
6. Locate, select, organize, and document information using appropriate technology and information systems.
7. Analyze, evaluate, and apply relevant information from a variety of sources.
9. Interact with others in groups or teams in ways that contribute to effective working relationships and the achievement of goals.
10. Manage the use of time and other resources to complete projects.
11. Take responsibility for one's own actions, decisions, and consequences.

\*There are 11 Essential Employability Skills outcomes as per the Ministry Program Standard. Of these 11 outcomes, the following will be assessed in this course.

# Global Citizenship and Equity (GC&E) Outcomes

N/A

## Text and other Instructional/Learning Materials

### Text Book(s):

Cohn, Mike. 2004. User Stories Applied: For Agile Software Development, Addison-Wesley Professional. ISBN-10: 0321205685

ISBN-13: 978-0321205681

### Online Resource(s):

Google+ and others.

## Evaluation Scheme

- ✦ User Roles: Identify and consolidation of user roles.
- ✦ User Stories: Generate user stories for the system.
- ✦ Release Planning: Create a release plan.
- ✦ Iteration Planning: Create an iteration plan per iteration.
- ✦ Measuring, Monitoring, and Electronic Maintenance of Project: Create charts to monitor project progress and enhance team collaboration using a software project management system.
- ✦ Release 1.0: Production code meets the project requirements with test code. All related document that supports the code-base.
- ✦ Release 2.0: Production code meets the project requirements with test code. All related document that supports the code-base.

Evaluation Name	CLO(s)	EES Outcome(s)	GCE Outcome(s)	Weight/100
User Roles	1, 2, 6	1, 2, 4, 5, 6, 7, 9, 10, 11		10
User Stories	1, 2, 3, 6, 8	1, 2, 4, 5, 6, 7, 9, 10, 11		15
Release Planning	1, 2, 3, 6, 8	1, 2, 4, 5, 6, 7, 9, 10, 11		10
Iteration Planning	1, 2, 3, 6, 8	1, 2, 4, 5, 6, 7, 9, 10, 11		15
Measuring, Monitoring, and Electronic Maintenance of Project	1, 2, 3, 6, 7, 8	1, 2, 4, 5, 6, 7, 9, 10, 11		10
Release 1.0	2, 3, 4, 5, 6, 7, 8, 9	1, 2, 4, 5, 7, 9, 10, 11		15
Release 2.0	2, 3, 4, 5, 6, 7, 8, 9	1, 2, 4, 5, 7, 9, 10, 11		25
<b>Total</b>				<b>100%</b>

If students are unable to write a test they should immediately contact their professor or program Chair for advice. In exceptional and well documented circumstances (e.g. unforeseen family problems, serious illness, or death of a close family member), students may be able to write a make-up test.

All submitted work may be reviewed for authenticity and originality utilizing Turnitin®. Students who do not wish to have their work submitted to Turnitin® must, by the end of the second week of class, communicate this in writing to the instructor and make mutually agreeable alternate arrangements.

When writing tests, students must be able to produce official College photo identification or they may be refused the right to take the test or test results will be void.

## Student Accommodation

Students with permanent or temporary accommodations who require academic accommodations are encouraged to register with the Centre for Students with Disabilities (CSD) located at Ashtonbee (L1-04), Progress (C1-03), Morningside (Rm 190), and Story Arts Campus (Rm 284). Documentation outlining the functional limitations of a disability is required; however, interim accommodations pending receipt of documentation may be possible. This service is free and confidential. For more information, please email [csd@centennialcollege.ca](mailto:csd@centennialcollege.ca).

## Use of Dictionaries

- Any dictionary (hard copy or electronic) may be used in regular class work.
- Dictionary use is not permitted in test or examination settings.
- Any dictionary (hard copy or electronic) may be used in regular class work.

## Program or School Policies

N/A

## Course Policies

N/A

## College Policies

Students should familiarize themselves with all College Policies that cover academic matters and student conduct.

All students and employees have the right to study and work in an environment that is free from discrimination and harassment and promotes respect and equity. Centennial policies ensure all incidents of harassment, discrimination, bullying and violence will be addressed and responded to accordingly.

Academic honesty is integral to the learning process and a necessary ingredient of academic integrity. Academic dishonesty includes cheating, plagiarism, and impersonation. All of these occur when the work of others is presented by a student as their own and/or without citing sources of information. Breaches of academic honesty may result in a failing grade on the assignment/course, suspension or expulsion from the college.

For more information on these and other policies, please visit [www.centennialcollege.ca/about-centennial/college-overview/college-policies](http://www.centennialcollege.ca/about-centennial/college-overview/college-policies).

Students enrolled in a joint or collaborative program are subject to the partner institution's academic policies.

## PLAR Process

This course is eligible for Prior Learning Assessment and Recognition (PLAR). PLAR is a process by which course credit may be granted for past learning acquired through work or other life experiences. The PLAR process involves completing an assessment (portfolio, test, assignment, etc.) that reliably demonstrates achievement of the course learning outcomes. Contact the academic school to obtain information on the PLAR process and the required assessment.

**This course outline and its associated weekly topical(s) may not be reproduced, in whole or in part, without the prior permission of Centennial College.**

Semester: Winter 2017  
 Section Code: S001,S002,S003,S004  
 Meeting Time & Location: As Per Published Teaching Schedule on myCentennial

Professor Name: Bob Pajkowski  
 Contact Information: ICET Faculty Office  
 Delivery Method: Blended

Topical Outline (subject to change):

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
1	Introduction, Project Plan, User Roles	Chapters 1, 3, 5 (Cohn) Software Project Management System (SPMS)	-Develop high-level project description. -Apply user role modeling and the user proxy for selecting the project customer. -Understand how to use a SPMS to facilitate communication and development.	Class discussions Independent work		
2	Introduction, Project Plan, User Roles	Chapters 1, 3, 5 (Cohn) Software Project Management System (SPMS)	-Develop high-level project description. -Apply user role modeling and the user proxy for selecting the project customer. -Understand how to use a SPMS to facilitate communication and development.	Class discussions Independent work		
3	User Stories	Chapters 2, 4, 7, 8 (Cohn)	-Develop user stories using appropriate method(s). -Estimate user stories based on a metric proposed by the team.	Class discussions Independent work	User Role Modelling Due (10%)	
4	Planning a Release, Acceptance Testing, Planning an Iteration, Measuring and Monitoring Velocity	Chapters 9, 10, 11 (Cohn)	-Use themes per release. -Prioritize user stories (and constraints) into iterations. -Disaggregate a story into tasks -Select an iteration length and the velocity. -Create the release plan -Understand how to create burndown charts. -Understand how to create tables showing changes and progress of user story development by the entire development team.	Class discussions Independent work	User Stories Gathering Due (15%)	
5	Planning a Release, Acceptance Testing, Planning an Iteration, Measuring and Monitoring Velocity	Chapters 9, 10, 11 (Cohn)	-Use themes per release. -Prioritize user stories (and constraints) into iterations. -Disaggregate a story into tasks -Select an iteration length and the velocity. -Create the release plan -Understand how to create burndown charts. -Understand how to create tables showing changes and progress of user story	Class discussions Independent work	First Release Plan Due (10%)	

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
			development by the entire development team.			
6	Constraints, Bugs, User Stories and the User Interface.	Chapters 1, 2, 3 (Chacon)	-Handling non-functional requirements -Stories for bugs -Basics of Version Control (Git)	Class discussions Independent work	First Iteration Plan Due (15%)	
7	First Release, User Stories, Planning the Second Release, Acceptance Testing, Planning an Iteration	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	-Estimate user stories based on a metric proposed by the team. -Use themes per release. -Prioritize user stories (and constraints) into iterations. -Disaggregate a story into tasks -Select an iteration length and the velocity. -Create the release plan -Create burndown charts. -Create tables showing changes and progress of user story development by the entire development team.	Class discussions Independent work		
8	First Release, User Stories, Planning the Second Release, Acceptance Testing, Planning an Iteration	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	-Estimate user stories based on a metric proposed by the team. -Use themes per release. -Prioritize user stories (and constraints) into iterations. -Disaggregate a story into tasks -Select an iteration length and the velocity. -Create the release plan -Create burndown charts. -Create tables showing changes and progress of user story development by the entire development team.	Class discussions Independent work		
9	First Release, User Stories, Planning the Second Release, Acceptance Testing, Planning an Iteration	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	-Estimate user stories based on a metric proposed by the team. -Use themes per release. -Prioritize user stories (and constraints) into iterations. -Disaggregate a story into tasks -Select an iteration length and the velocity. -Create the release plan -Create burndown charts. -Create tables showing changes and progress of user story development by the entire development team.	Class discussions Independent work		
10	Second Release, User Stories,	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	-Estimate user stories based on a metric proposed by the team.	Class discussions Independent work	Release 1.0 Due (15%)	

Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
	Planning an Iteration, Acceptance Testing		<ul style="list-style-type: none"> <li>-Use themes per release.</li> <li>-Prioritize user stories (and constraints) into iterations.</li> <li>-Disaggregate a story into tasks</li> <li>-Select an iteration length and the velocity.</li> <li>-Create the release plan</li> <li>-Create burndown charts.</li> <li>-Create tables showing changes and progress of user story development by the entire development team.</li> </ul>			
11	Second Release, User Stories, Planning an Iteration, Acceptance Testing	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	<ul style="list-style-type: none"> <li>-Estimate user stories based on a metric proposed by the team.</li> <li>-Use themes per release.</li> <li>-Prioritize user stories (and constraints) into iterations.</li> <li>-Disaggregate a story into tasks</li> <li>-Select an iteration length and the velocity.</li> <li>-Create the release plan</li> <li>-Create burndown charts.</li> <li>-Create tables showing changes and progress of user story development by the entire development team.</li> </ul>	Class discussions Independent work		
12	Second Release, User Stories, Planning an Iteration, Acceptance Testing	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	<ul style="list-style-type: none"> <li>-Estimate user stories based on a metric proposed by the team.</li> <li>-Use themes per release.</li> <li>-Prioritize user stories (and constraints) into iterations.</li> <li>-Disaggregate a story into tasks</li> <li>-Select an iteration length and the velocity.</li> <li>-Create the release plan</li> <li>-Create burndown charts.</li> <li>-Create tables showing changes and progress of user story development by the entire development team.</li> </ul>	Class discussions Independent work		
13	Second Release, User Stories, Planning an Iteration, Acceptance Testing	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	<ul style="list-style-type: none"> <li>-Estimate user stories based on a metric proposed by the team.</li> <li>-Use themes per release.</li> <li>-Prioritize user stories (and constraints) into iterations.</li> <li>-Disaggregate a story into tasks</li> <li>-Select an iteration length and the velocity.</li> <li>-Create the release plan</li> <li>-Create burndown charts.</li> </ul>	Class discussions Independent work		



Week	Topics	Readings/Materials	Weekly Learning Outcome(s)	Instructional Strategies	Evaluation Name	Evaluation Date
			-Create tables showing changes and progress of user story development by the entire development team.			
14	Second Release, User Stories, Planning an Iteration, Acceptance Testing	Chapters 2, 4, 7, 8, 9, 10 (Cohn)	<ul style="list-style-type: none"> <li>-Estimate user stories based on a metric proposed by the team.</li> <li>-Use themes per release.</li> <li>-Prioritize user stories (and constraints) into iterations.</li> <li>-Disaggregate a story into tasks</li> <li>-Select an iteration length and the velocity.</li> <li>-Create the release plan</li> <li>-Create burndown charts.</li> <li>-Create tables showing changes and progress of user story development by the entire development team.</li> </ul>	Class discussions Independent work	Release 2.0 Due (25%) and Progress Monitoring (10%)	