CONESTOGA

COURSE OUTLINE

Course Systems Development: Analysis (2013-2014)

Code / Version INFO2070 (101)

Total Hours 60

Credits 4

PreRequisite(s) PROG1780 (100) Programming: Fundamentals
or PROG1783 (100) IT Support Prog Fundamentals

CoRequisite(s)

Course Description

Upon completion of this course, the student will understand the Systems Development Life Cycle and the Object-Oriented Analysis methodology using the Unified Modeling Language. An object-oriented approach will be applied to a business system case study utilizing a Computer Assisted Systems Engineering tool to analyse the systems requirements. Emphasis will be placed on typical industry practices, documentation and presentation skills in a team environment.

PLAR Eligible: Yes

Course Outcomes

Successful completion of this course will enable the student to:

1. Describe and differentiate between major analysis and design methodologies
2. Explain Scrum and eXtreme Programming
3. Explain the agile software development process
4. Use a Computer-aided Software Engineering (CASE) tool to draw Unified Modeling Language (UML) diagrams
5. Discover and document customer requirements

Unit Outcomes

Successful completion of the following units will enable the student to:

1.0 History of Analysis and Design

- 1.1 List the phases in the classic systems development lifecycle (SDLC)
- 1.2 List the phases in the classic systems development lifecycle (SDLC)
- 1.3 Describe the Waterfall methodology
- 1.4 Describe the Unified Process methodology
- 1.5 Describe the Agile approach to system development
- 1.6 Distinguish between adaptive and predictive methodologies
- 1.7 Compare Waterfall, Unified Process and Agile approaches to analysis and design

2.0 Agile Methodologies

- 2.1 Describe the Scrum methodology
- 2.2 Describe eXtreme Programming (XP)

3.0 The Agile Software Development Process

- 3.1 Write a Project Charter
- 3.2 Describe the staffing of an Agile project
- 3.3 Identify stakeholders

3.4 Explain user stories
3.5 Explain the team backlog
3.6 Explain iterations
3.7 Explain acceptance testing
3.8 Explain incremental releases

4.0 Deriving and Documenting Requirements

4.1 Analyze a case study and identify functional and non-functional requirements
4.2 Analyze a case study and write user stories
4.3 Analyze a case study and write acceptance tests
4.4 Analyze a case study and discuss iteration and release planning
4.5 Analyze a case study and write use cases
4.6 Analyze a case study and write pseudocode to express logic
4.7 Use a Computer-aided Software Engineering (CASE) tool to draw Unified Modeling Language (UML) diagrams
4.8 Analyze a case study and draw use case diagrams
4.9 Analyze a case study and draw activity diagrams
4.10 Analyze a case study and draw domain class diagrams
4.11 Analyze a case study and draw system sequence diagrams

Required Student Resources

Leffngwell, Dean. Agile Software Requirements (1st). Addison Wesley.

Optional Student Resources

Evaluation

The minimum passing grade for this course is 55 (D).

In order to successfully complete this course, the student is required to meet the following evaluation criteria:

Tests	50.00
Quizzes/Assignments	50.00
	100.00 %

A passing grade in both the tests and quizzes/assignments portions independently is required in order to attain standing in this course. If the student fails one or both portions, then the lowest failing mark is submitted.

Other

Conestoga College is committed to providing academic accommodations for students with documented disabilities. Please contact the Accessibility Services Office.

Students will be required to work in groups on the assignments in this course.

Prepared By John McKay

School Information Technology

CONESTOGA

COURSE OUTLINE

Course Systems Development: Analysis (2013-2014)

Code / Version INFO2070 (101)

Date 2013-12-17

© Conestoga ITAL