UNIVERSITY OF THE FRASER VALLEY

OFFICIAL UNDERGRADUATE COURSE OUTLINE (page 1)

COURSE IMPLEMENTATION DATE:	January 2004
COURSE REVISED IMPLEMENTATION DATE:	September 2011
COURSE TO BE REVIEWED:	January 2014
(four years after UPAC approval)	*(month, year)*

OFFICIAL UNDERGRADUATE COURSE OUTLINE INFORMATION

Students are advised to keep course outlines in personal files for future use.

Shaded headings are subject to change at the discretion of the department – see course syllabus available from instructor

CIS 371	**Computer Information Systems**	**3**
COURSE NAME/NUMBER	FACULTY/DEPARTMENT	UCFV CREDITS

Object Oriented Modeling and Design

COURSE DESCRIPTIVE TITLE

CALENDAR DESCRIPTION:

Using object-oriented techniques, students will learn how to design and implement high quality computer systems. Emphasis is placed on creating a requirements model and using design principles to create a working system (user interface, application, and database). The unified modeling language (UML) is used extensively throughout this course.

PREREQUISITES: Acceptance to the CIS degree program; CIS 230, CIS 270, and COMP 251. (Students accepted to a CIS or Computing Science minor may register with permission of the department.)

COREQUISITES:

PRE or COREQUISITES:

SYNONYMOUS COURSE(S):

(a) Replaces:

(b) Cross-listed with:

(c) Cannot take: for further credit.

SERVICE COURSE TO: *(department/program)*

TOTAL HOURS PER TERM: **45**

STRUCTURE OF HOURS:

Lectures:	**45**	Hrs
Seminar:		Hrs
Laboratory:		Hrs
Field experience:		Hrs
Student directed learning:		Hrs
Other (specify):		Hrs

TRAINING DAY-BASED INSTRUCTION:

Length of course:

Hours per day:

OTHER:

Maximum enrolment: **35**

Expected frequency of course offerings: Once a year

(every semester, annually, every other year, etc.)

WILL TRANSFER CREDIT BE REQUESTED? (lower-level courses only)	☐ Yes	☒ No
WILL TRANSFER CREDIT BE REQUESTED? (upper-level requested by department)	☐ Yes	☒ No
TRANSFER CREDIT EXISTS IN BCCAT TRANSFER GUIDE:	☐ Yes	☒ No

Course designer(s):	**Paul Rushton**		
Department Head:	**Ora Steyn**	Date approved:	**October 16, 2009**
Supporting area consultation (Pre-UPAC)		Date of meeting:	**November 6, 2009**
Curriculum Committee chair:	**Edward Lo**	Date approved:	**October 8, 2009**
Dean/Associate VP:	**Dan Ryan**	Date approved:	**January 18, 2010**
Undergraduate Program Advisory Committee (UPAC) approval		Date of meeting:	**January 29, 2010**

LEARNING OUTCOMES:

Upon successful completion of this course, students will be able to:

- Explain what constitutes a good requirement
- Use UML to model system requirements
- Use UML to model system design
- Apply design principles
- Translate requirements and design into good quality programs

METHODS: *(Guest lecturers, presentations, online instruction, field trips, etc.)*

- Lectures
- Review of case studies
- Programming assignments
- Formal examinations
- Student presentations

METHODS OF OBTAINING PRIOR LEARNING ASSESSMENT RECOGNITION (PLAR):

☒ Examination(s) ☒ Portfolio assessment ☒ Interview(s)

☐ Other (specify):

☐ PLAR cannot be awarded for this course for the following reason(s):

TEXTBOOKS, REFERENCES, MATERIALS:

[Textbook selection varies by instructor. An example of texts for this course might be:]

Larman, Craig. Applying UML and Patterns (3rd edition), Prentice Hall, 2005

SUPPLIES / MATERIALS:

None

STUDENT EVALUATION:

[An example of student evaluation for this course might be:]

Assignments – 70%
Examinations – 30%

COURSE CONTENT:

[Course content varies by instructor. An example of course content might be:]

- Object-oriented review
- Developing requirements
- Modeling with classes
- User centric design
- Modeling interactions and behaviours
- Architecting and designing software
- Testing and inspecting to ensure high quality