

|   |                       |
|---|-----------------------|
| COURSE IMPLEMENTATION DATE:             | <u>January 2004</u>   |
| COURSE REVISED IMPLEMENTATION DATE:     | <u>September 2008</u> |
| COURSE TO BE REVIEWED:                  | <u>February 2012</u>  |
| <i>(four years after UPAC approval)</i> | <i>(month, year)</i>  |

**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                                    | CIS                | 3            |
| COURSE NAME/NUMBER                         | FACULTY/DEPARTMENT | UCFV CREDITS |
| <b>Object Oriented Modeling and Design</b> |                    |              |
| 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:** CIS 270. Acceptance to the CIS degree program. (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:                  | <u>45</u> | 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: January 11, 2008

Supporting area consultation (UPACA1)

Date of meeting: February 1, 2008

Curriculum Committee chair: Edward Lo

Date approved: January 10, 2008

Dean/Associate VP: Ian McAskill

Date approved: January 23, 2008

Undergraduate Program Advisory Committee (UPAC) approval

Date of meeting: February 29, 2008

**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 (3<sup>rd</sup> 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