UNIVERSITY OF THE FRASER VALLEY

ORIGINAL COURSE IMPLEMENTATION DATE: September 1996
REVISED COURSE IMPLEMENTATION DATE: September 2019
COURSE TO BE REVIEWED (six years after UEC approval)**:** May 2018
Course outline form version: 05/18/2018

OFFICIAL UNDERGRADUATE COURSE OUTLINE FORM

Note: The University reserves the right to amend course outlines as needed without notice.

Course Code and Number: COMP 351	**Number of Credits:** 3 Course credit policy (105)

Course Full Title: Advanced Website Programming
Course Short Title:
(Transcripts only display 30 characters. Departments may recommend a short title if one is needed. If left blank, one will be assigned.)

Faculty: Faculty of Professional Studies	**Department (or program if no department):** Computer Information Systems

Calendar Description:

This course covers current technologies for client and server-side programming of dynamic websites, web-enabled applications, and web services. Internet protocols, security issues, and database connectivity will be examined. The course will also introduce common design architectures used for the technologies.

Prerequisites (or NONE):	COMP 251 and admission to the Bachelor of Computer Information Systems degree or the Bachelor of Science with Computing Science major. Note: Students accepted to a CIS or Computing Science minor may register with permission of the department.
Corequisites (if applicable, or NONE):	
Pre/corequisites (if applicable, or NONE):	

Antirequisite Courses *(Cannot be taken for additional credit.)*
Former course code/number:
Cross-listed with:
Dual-listed with:
Equivalent course(s):
(If offered in the previous five years, antirequisite course(s) will be included in the calendar description as a note that students with credit for the antirequisite course(s) cannot take this course for further credit.)

Typical Structure of Instructional Hours

Lecture/seminar hours	45
Tutorials/workshops	
Supervised laboratory hours	
Experiential (field experience, practicum, internship, etc.)	
Supervised online activities	
Other contact hours:	
Total hours	**45**

Labs to be scheduled independent of lecture hours: ☒ No ☐ Yes

Special Topics *(Double-click on boxes to select.)*
This course is offered with different topics:
☒ No ☐ Yes *(If yes, topic will be recorded when offered.)*

Independent Study
If offered as an Independent Study course, this course may be repeated for further credit: *(If yes, topic will be recorded.)*
☒ No ☐ Yes, repeat(s) ☐ Yes, no limit

Transfer Credit
Transfer credit already exists: *(See **bctransferguide.ca**.)*
☒ No ☐ Yes
Submit outline for (re)articulation:
☒ No ☐ Yes *(If yes, fill in transfer credit form.)*

Grading System
☒ Letter Grades ☐ Credit/No Credit

Maximum enrolment (for information only): 35

Expected Frequency of Course Offerings:
Annually *(Every semester, Fall only, annually, etc.)*

Department / Program Head or Director: Talia Q	**Date approved:**	December 2028
Faculty Council approval	**Date approved:**	December 7, 2018
Dean/Associate VP: Tracy Ryder Glass	**Date approved:**	December 7, 2018
Campus-Wide Consultation (CWC)	**Date of posting:**	February 22, 2019
Undergraduate Education Committee (UEC) approval	**Date of meeting:**	March 1, 2019

Learning Outcomes:

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

- Assess current technologies for web programming.
- Design and implement client and server-side programs.
- Explain the protocols used.
- Explain security issues.
- Explain the design architecture.
- Use web, database, and other servers for the technologies.

Prior Learning Assessment and Recognition (PLAR)

☒ Yes ☐ No, PLAR cannot be awarded for this course because

Typical Instructional Methods *(Guest lecturers, presentations, online instruction, field trips, etc.; may vary at department's discretion.)*

The course will be delivered in lecture-lab format, with numerous demonstrations and hands-on activities. The lab portion gives students and the instructor the ability to view and interact with current projects.

NOTE: The following sections may vary by instructor. Please see course syllabus available from the instructor.

Typical Text(s) and Resource Materials *(If more space is required, download Supplemental Texts and Resource Materials form.)*

	Author (surname, initials)	Title (article, book, journal, etc.)	Current ed.	Publisher	Year
1.			☐		
2.			☐		
3.			☐		
4.			☐		
5.			☐		

Required Additional Supplies and Materials *(Software, hardware, tools, specialized clothing, etc.)*

Typical Evaluation Methods and Weighting

Final exam:	30%	Assignments:	45%	Field experience:	%	Portfolio:	%
Midterm exam:	%	Project:	%	Practicum:	%	Other:	%
Quizzes/tests:	25%	Lab work:	%	Shop work:	%	Total:	100%

Details (if necessary):

Typical Course Content and Topics

- **The Java Language:** History, syntax, current trends, security issues
- **Applets and applications:** Building and running applications; the applet lifecycle
- **Java Web Start:** Download, install, and run applets outside the web browser
- **Graphical user interfaces:** Using the AWT, JFC, Swing
- **Threads and concurrency issues:** Multiprocessing concepts; extending Thread, implementing Runnable
- **Communications:** Streams, Packets, URL, Sockets, Client/Server programming
- **Database connectivity:** JDBC
- **Java Beans:** Event model, Bean properties, custom source components and event sets
- **IDE and Web Server:** Netbeans IDE, Tomcat and GlassFish servers
- **Architecture:** Variations on the Model View Controller
- **Servlets:** Servlet API, processing forms, sending images, cookies, sessions
- **Java Server Pages:** JSP constructs, JavaBeans components, predefined variables, properties
- **Java Server Faces:** JSF UI components, JSF containers, message components, virtual forms
- **Web Services:** Role of WSDL, SOAP requests and responses