

Computer Science Technology Program

COURSE OUTLINE

420-410-DW Individual Project Java IV

Ponderation: 3-3-3

Credits: 3

Hours:90

Prerequisites:

420-310-DW Programming III - Java III
420-320-DW Web Development II – Client-side programming
420-331-DW Database II – Database Applications

Co-requisites:

420-440-DW Infrastructure III - Linux II
420-420-DW Web Applications III – Server-side programming

Note: You must pass this course as well as the two co-requisite courses in order to advance to the fifth term of the Computer Science Technology Program.

Domain:

Programming, Data Structures and Algorithms

Description:

This course exposes the students to the process of requirement gathering and verification. Students will learn about advanced design patterns and how to implement and apply them. Students will also learn about different testing types and techniques and will use them. Each student will apply the concepts learnt in the course in the development of a significant project with a GUI throughout the semester. The project is required to connect to a database to store, retrieve, and update the data.

Competency	Statement
00SR	Develop native applications without a database
00SS	Develop native applications with a database

Semester Winter 2021 Revision: 2021-01-27

Teacher(s)	Office	e-mail	Section(s)
Nagi Basha	3H.14	nbasha@dawsoncollege.qc.ca	01,02,03

Teachers will respond to students' inquiries in a timely manner.

Teachers' schedules and availabilities will be posted on Moodle during the first week of classes.

Pedagogical approach

The course consists of two 1½-hour lectures and one 3 hour lab session per week. Concepts and techniques will be presented in lectures. The lectures will be pre-recorded. It is expected that students watch the videos before coming to class on Zoom for discussions. Forums will also be posted on Moodle to facilitate asynchronous discussions and communications between all students and the instructor.

A big project will start at the beginning of the semester and will keep going on till the end. Each student will go through the different phases of the software development cycle starting from requirements gathering, analysis, and verification. Then software design followed by software coding. Special emphasis will be placed on testing. Testing is not only for the code. The requirements and the design will be tested too.

The lab periods are designated to complete the work related to the project. It is expected the project will involve additional time (averaging 3 hours per week) outside of lab/class time to complete.

Required materials:

Computer with internet access as well as the following software installed. Note: More detail will be provided during the lectures/lab periods about these pieces of software.

1. NetBeans: This is an IDE where you will write, compile, debug, and run your programs. NetBeans is free and platform independent (there are versions for mac, linux, windows, etc). You should download it here <https://netbeans.apache.org/download/index.html>.
The most up to date version is needed.
2. Git and GitBash: This is a version control system which you will use to keep track of edits of your code. For windows users, download git here: <https://git-scm.com/download/win> .
For OS X and Linux users, please refer to <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git> .
3. A GitLab account with a private repository.

Supplementary resources

- Gaddis, Tony Starting Out With Java From Control Structures Through Objects, 6th edition, Pearson ISBN-13: 978-0-13-395705-1
- Savitch, Absolute Java, 6th edition, Pearson, ISBN-13: 978-0-13-404167-4
- Bloch, Effective Java, 3rd edition, Addison-Wesley Professional, ISBN-13: 978-0-13-468599-1

Assessment of student performance

Project: 60%

The project will be done in phases. Each phase is graded separately.

Midterm: 20%

Final: 20%

Exact project milestone dates will be posted on Moodle.

Tentative Schedule

Week	Lecture Topics	Tentative Project Milestones
1	Explain the project	
2	Understand the software development phases Learn about the different software development models Identify the pros and cons of each development model	
3	Understand the process of creating the requirement specifications document <ul style="list-style-type: none">• Elicit requirements• Analyze requirements• Verify requirements consistency• Verify requirements feasibility• Understand the benefits of detecting defects at the requirement gathering stage	
4	Examine the Extreme Programming (XP) paradigm and its characteristics <ul style="list-style-type: none">• Understand the benefits of simple design• Identify the long-term gains of continuous testing• Use refactoring to improve the quality of the code• Identify the benefits of pair programming	Finalize requirements
5	Review OOP principles <ul style="list-style-type: none">• Review inheritance• Review polymorphism• Review interfaces• Review UML	
6	Identify the benefits of using a versioning control system like Git Identify the benefits of adhering to the same coding convention within a software development company	Preliminary design + ERD for the database
7	Explain the importance of testing Software testing is a career on its own Identify and use the different types of testing <ul style="list-style-type: none">• Black box testing• White box testing	Test cases and JUnit tests

	<ul style="list-style-type: none"> • Peer review • Boundary testing • Integration testing • Performance testing <p>Use unit tests</p>	
8	Midterm	
9	<p>Understand what refactoring is</p> <ul style="list-style-type: none"> • Identify the benefits of refactoring • Consolidate conditional expressions • Consolidate duplicate conditional fragments • Rename methods, classes, variables, etc.. • Pull up repetitive fields from derived classes to base classes 	Adapt to changing requirements
10-14	<p>Understand and apply design patterns</p> <ul style="list-style-type: none"> • Understand and apply some of the creational patterns such as: The abstract factory pattern The builder pattern The prototype pattern The singleton pattern • Understand and apply a range of the structural patterns such as: The adapter pattern The decorator pattern • Understand and apply a range of the behavioral patterns such as: The observable pattern The strategy pattern 	Final design using appropriate design patterns
15	Finalize project	Working software

Program rules and regulations:

For 420 career program courses, students must receive at least 60% on the assignment component and at least 60% on the examination component in order to obtain a passing grade for the course. Failure to do so will result in a maximum grade of 50 (the lower of 50 or the grade earned). The examination component is the summative assessment, and does not include quizzes. A minimum grade of 60% is required to pass the course.

Assignments are due at the due date specified by the instructor. Assignments received after this time are late. Assignments will not be accepted after due date set by instructor. In exceptional circumstances, such as illness, etc. labs submitted late will be graded provided that the student has received an extension from the instructor in advance.

Each student must submit their own individual answers to assignments, and shared solutions are not acceptable. Students must protect their work from being easily copied by others by introducing uniqueness into their work in the choice of images used, the display styles used in the presentation of documents, the originality of the written descriptive passages, and by placing digital signatures on their original images.

Students are reminded that they are responsible for all material presented in the lecture and during the lab and are expected to obtain material they have missed.

Students must follow exactly the instructions in the lab exercises about how and where to store all computer files in order for their work to be accepted and properly evaluated.

“This course outline has been prepared in conformity with the **Institutional Student Evaluation Policy (I.S.E.P.)**. The “policy is designed to promote equitable and effective evaluation of student learning” and is **therefore a crucial policy for you to read and understand**. The policy describes the rights and obligations of students, faculty, departments, programs, and the College administration with regard to evaluation in all your courses. **I.S.E.P.** is published in the College Calendar and reference copies are available in the Library and in the offices of program chairs and the Dean of Social Science & Business Technologies”.

DAWSON COLLEGE COMPUTER SCIENCE DEPARTMENT POLICIES

1. ACADEMIC INTEGRITY

The Computer Science Department adheres to the Dawson College Academic Conduct policy. Students have an obligation to inform themselves of all aspects of this policy. According to ISEP, the teacher is required to report to the Sector Dean all cases of cheating and plagiarism affecting a student's grade. (ISEP Section V-A) Students may appeal any decision of the teacher regarding cheating and plagiarism according to the methods established by the College. The penalty for cheating or plagiarism may range from a zero on the work to a failure in the course.

2. ATTENDANCE AND LATENESS

Students are responsible for all material covered in classes and labs, whether or not they are present. Students have an obligation to arrive on time and to remain for the duration of scheduled classes, labs and activities. Students who disregard this obligation may be asked to leave the class, lab or activity. Students should refer to the Institutional Student Evaluation Policy (ISEP Section IV-C) regarding attendance.

3. RELIGIOUS HOLIDAY OBSERVANCE

Class period(s) may be cancelled in order that the teacher can observe religious holidays. Any material missed as a result, will be made up during labs, class periods and alternate tutorial sessions.

Students observing religious holidays must inform their teachers, in writing, as prescribed in the ISEP Policy on Religious Observances, no later than the end of the second week of the impacted semester or term. This applies both to the semester or term, as well as to any final examination period. (ISEP Section IV-D). Students absent from classes

because of observance of religious holidays will not be penalized. It must be emphasized, however, that this College policy should not be interpreted to mean that a student can receive credit for work not performed. It is the student's responsibility to fulfill the requirements of the alternative arrangement.

4. SOFTWARE

The Computer Science Department forbids the use of the computer labs it uses to make any copies of any software without the explicit authorization of the Department. Apart from any legal action that might arise from such unauthorized copying, the Computer Science Department reserves the right to discipline any student involved in such activity.

Students who borrow software from the College and fail to return it will be placed on the defaulter list and be subject to the appropriate penalties.

Viruses are programs that attach themselves to a computer system without the permission of those to whom the system belongs. They are deliberately written to be, at worst, harmful (e.g. destroying the contents of disks) and, at best, bothersome (e.g. disturbing the image shown on the screen). They can cause serious losses of time and effort for students, staff and faculty. Any student involved in the

deliberate spreading of viruses is subject to the most severe penalties prescribed by College regulations, apart from any legal action that might arise from such acts.

5. LITERACY

The Computer Science Department recognizes that literacy in all its forms (read, written, spoken) is essential to our students in their careers.

Teachers may choose to incorporate a literacy component into the marking scheme for any piece of work. Teachers may use their discretion to insist that any piece of work submitted for credit is revised by the student if it is unsatisfactory with regard to literacy.

Teachers will inform all students in their courses of this policy at the beginning of each semester either by including it in the course description or otherwise.

6. CELL PHONES

The use of cell phones in all Computer Science lectures, labs and exams is prohibited.

7. PORTABLE COMPUTERS

The use of portable computers in all Computer Science lectures for purposes other than note-taking is prohibited.

8. STUDENT CONDUCT

Everyone has the right to a safe and non-violent environment. Students are obliged to conduct themselves as stated in the Student Code of Conduct and in the ISEP section on the roles and responsibilities of students. (ISEP section II-D). This policy includes online conduct.

9. PROFESSIONAL CONDUCT POLICY

Students who are enrolled in the Computer Science Technology career program must conduct themselves according to the Professional Conduct Policy as described in the Student Handbook. (ISEP Section IV-O.2). This professional conduct policy includes online conduct.

10. INTENSIVE COURSE CONFLICTS

If a student is attending an intensive course, the student must inform the teacher, within the first two weeks of class, of the specific dates of any anticipated absences.

11. ISEP

The Institutional Student Evaluation Policy (ISEP) is designed to promote equitable and effective evaluation of student learning and is therefore a crucial policy to read and understand. The policy describes the rights and obligations of students, faculty, departments, programs, and the College administration with regard to evaluation in all your courses, including grade reviews and resolution of academic grievance. ISEP is available on the Dawson website.