



# PLAN D'ÉTUDES

**Département**

Techniques de l'informatique

**Titre du cours**

Applications Web 2

**Numéro du cours**

420-G26-RO

**Pondération**

2-4-2

**Unités**

2 2/3 unités

**Session**

Hiver 2025

**Programme**

Techniques de l'informatique (420-B2) - Développement d'applications web et mobiles

**Type de formation**

Formation spécifique

**Prénom et nom de l'enseignant**

Dini Ahamada

**Bureau**

B-313

**Courriel**

dahamada@[crosemont.qc.ca](mailto:dahamada@crosemont.qc.ca)

# 1. PRÉSENTATION GÉNÉRALE DU COURS

## 1.1 Thématique générale

Depuis environ 2005, le développement d'applications web subit des transformations majeures. L'introduction de la technique AJAX a offert de nouvelles possibilités de collaboration entre un navigateur web et un serveur web pour faire fonctionner différemment les applications web. AJAX a ramené au premier plan le langage de programmation JavaScript, devenu incontournable. Cet apport a conduit à développer des applications web structurées en deux modules séparés et collaboratifs : le module coté client en HTML, CSS et JavaScript et le module coté serveur à l'aide d'une technologie serveur telle que Java, PHP, .Net ou Node JS. Pour optimiser le développement de ce deuxième module, on exploite des frameworks tels que Spring en Java, Laravel et PHP ou ASP.Net MVC.

Afin d'optimiser le module coté client, des frameworks tels que Angular, React JS et Vue JS sont apparus.

Pour faire fonctionner l'application sans risque de bloquer le navigateur en attendant que le serveur réponde à sa requête, des modèles de programmation asynchrone ont été introduits et sont utilisés.

Afin de permettre aux environnements de développement d'assister efficacement les développeurs, le langage TypeScript est apparu. TypeScript est un sur-ensemble de JavaScript incluant un système de typage. Il permet ainsi de développer en orienté objet de manière similaire aux langages classiques tels que C++, Java et C#.

Ce cours abordera tous ces aspects et permettra aux étudiants de s'initier au développement d'applications web transactionnelles asynchrones. Tous les concepts de programmation web en Java seront étudiés, complétés par l'apprentissage de l'exploitation du framework Spring. Le cours abordera aussi les modèles de programmation asynchrone et l'exploitation du framework Angular à l'aide du langage TypeScript pour développer des applications de type single-page.

## 1.2 Contribution et place du cours dans le programme

Le cours 420-G26-RO est le deuxième d'une séquence de 3 cours (*Applications Web 1, 2 et 3*) qui amèneront l'étudiant à maîtriser les technologies et concepts reliés à la programmation Web. Après avoir abordé, dans le cours Applications Web 1, les concepts et technologies fondamentales de programmation web (HTML, CSS, JavaScript, programmation web coté serveur en PHP, ...), ce cours poursuit les apprentissages en abordant une autre technologie de programmation web coté serveur (Java), l'exploitation d'un framework de programmation web (Spring) ainsi que les concepts et technologies de programmation web asynchrone. Finalement, le cours aborde l'étude d'un framework de programmation d'applications web asynchrones de type single-page (Angular).

### 1.3 Objectif intégrateur

Réaliser les étapes du cycle de développement d'applications web transactionnelles asynchrones.

### 1.4 Intentions éducatives visées par le cours (prescrites par le ministère ou établies localement)

- Rendre l'élève efficace dans l'exercice d'une profession
- Favoriser l'intégration de l'élève à la vie professionnelle
- Développer l'autonomie et favoriser la polyvalence.

### 1.5 Préalables

Les cours **420-G03-RO Introduction à la sécurité informatique** et **420-G16-RO Applications web 1** sont préalables relatifs à ce cours. Ils doivent donc être déjà suivis et une note supérieure ou égale à 50% doit être obtenue dans chacun de ces 2 cours pour s'inscrire à ce cours.

## 2. DESCRIPTION DE LA (OU DES) COMPÉTENCE(S) DÉVELOPPÉE(S)

### 2.1 Les compétences développées dans le cours

CODE DE COMPÉTENCE	ÉNONCÉ DE LA COMPÉTENCE	ÉTAT DE DÉVELOPPEMENT (A, P, V) <sup>1</sup>
00SU	Effectuer le développement d'applications Web transactionnelles	P

### 2.2 Le devis ministériel

ÉNONCÉ DE LA COMPÉTENCE	CONTEXTE DE RÉALISATION
Effectuer le développement d'applications Web transactionnelles	<ul style="list-style-type: none"> <li>• Pour des applications Web transactionnelles : réservations, inscriptions, travail collaboratif, gestion des stocks, commerce électronique, etc.</li> <li>• Pour de nouvelles applications et des applications à modifier.</li> <li>• À partir des documents de conception.</li> <li>• À l'aide d'images.</li> <li>• À l'aide de procédures de suivi des problèmes et de gestion des versions.</li> </ul>
ÉLÉMENTS DE COMPÉTENCE	CRITÈRES DE PERFORMANCE
1. Analyser le projet de développement de l'application	<ul style="list-style-type: none"> <li>• Analyse juste des documents de conception.</li> <li>• Détermination correcte des tâches à effectuer.</li> </ul>
2. Préparer l'environnement de développement informatique.	<ul style="list-style-type: none"> <li>• Installation correcte de la plateforme de développement Web et du système de gestion de base de données de développement.</li> <li>• Installation correcte des logiciels et des bibliothèques.</li> <li>• Configuration appropriée du système de gestion de versions.</li> <li>• Importation correcte du code source.</li> </ul>
3. Préparer la base de données	<ul style="list-style-type: none"> <li>• Création ou adaptation correctes de la base de données.</li> <li>• Insertion correcte des données initiales ou des données de tests.</li> <li>• Respect du modèle de données.</li> </ul>
4. Programmer l'interface Web.	<ul style="list-style-type: none"> <li>• Utilisation appropriée du langage de balisage.</li> <li>• Création et utilisation appropriées des feuilles de styles.</li> <li>• Intégration correcte des images.</li> <li>• Création appropriée des formulaires Web.</li> </ul>

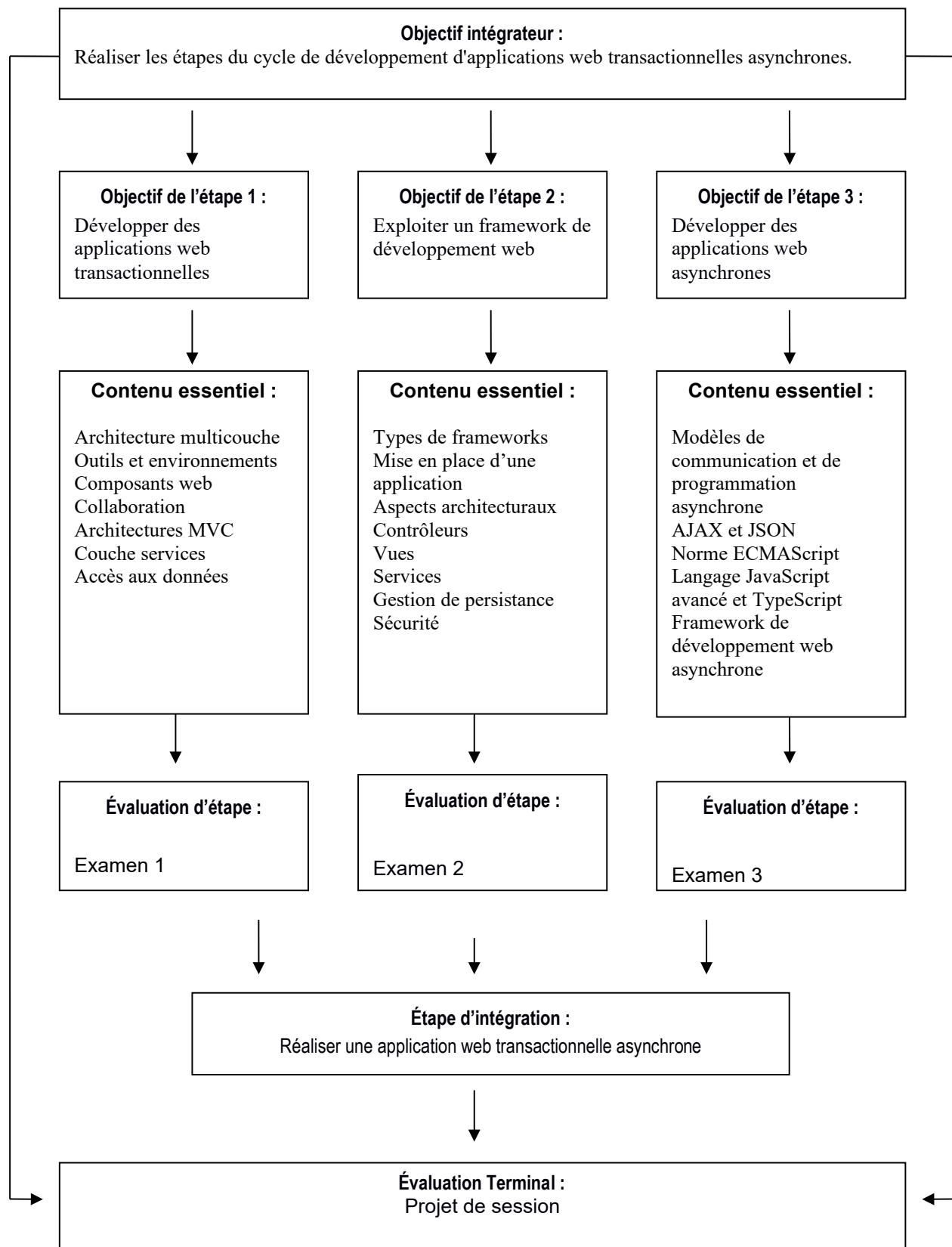
<sup>1</sup> A : amorce le développement de la compétence ;

P : poursuit le développement de la compétence amorcé dans un autre cours ;

V : vérifie le développement final de la compétence.

	<ul style="list-style-type: none"> <li>• Adaptation de l'interface en fonction du format d'affichage et de la résolution.</li> </ul>
5. Programmer la logique applicative du côté serveur.	<ul style="list-style-type: none"> <li>• Programmation ou intégration correctes de mécanismes d'authentification et d'autorisation.</li> <li>• Programmation correcte des interactions entre l'interface Web et l'utilisatrice ou l'utilisateur.</li> <li>• Choix approprié des clauses, des opérateurs, des commandes ou des paramètres dans les requêtes à la base de données.</li> <li>• Manipulation correcte des données de la base de données.</li> <li>• Utilisation appropriée des services d'échange de données.</li> <li>• Application correcte des techniques d'internationalisation.</li> <li>• Application rigoureuse des techniques de programmation sécurisée.</li> </ul>
6. Programmer la logique applicative du côté client.	<ul style="list-style-type: none"> <li>• Manipulation adéquate des objets du modèle DOM.</li> <li>• Programmation correcte des interactions entre l'interface Web et l'utilisatrice ou l'utilisateur.</li> <li>• Utilisation systématique des techniques de validation de données des formulaires Web.</li> <li>• Formulaires Web conformes aux exigences d'utilisabilité.</li> </ul>
7. Contrôler la qualité de l'application.	<ul style="list-style-type: none"> <li>• Application rigoureuse des plans de tests.</li> <li>• Revues de code et de sécurité rigoureuses.</li> <li>• Pertinence des correctifs.</li> <li>• Respect des procédures de suivi des problèmes et de gestion des versions.</li> <li>• Respect des documents de conception</li> </ul>
8. Participer au déploiement de l'application chez un hébergeur Web	<ul style="list-style-type: none"> <li>• Détermination judicieuse du nom de domaine.</li> <li>• Configuration appropriée de l'application chez l'hébergeur Web.</li> <li>• Application correcte de la procédure de migration de l'application chez l'hébergeur Web.</li> <li>• Application rigoureuse des mesures de sécurité.</li> <li>• Respect des exigences de référencement.</li> </ul>
9. Rédiger la documentation	<ul style="list-style-type: none"> <li>• Détermination correcte de l'information à rédiger.</li> <li>• Notation claire du travail effectué.</li> </ul>

### 3. DÉROULEMENT DU COURS : VUE SYNOPTIQUE



### 3.1 Échéancier détaillé du cours pour la session

SEM.	CONTENUS	ACTIVITÉS D'ENSEIGNEMENT ET D'APPRENTISSAGE	TRAVAUX ET ÉVALUATIONS <i>FORMATIVES ET SOMMATIVES</i>
<b>OBJECTIF DE L'ÉTAPE 1 : DÉVELOPPER DES APPLICATIONS WEB TRANSACTIONNELLES</b>			
1	Introduction Présentation du cours Installation et configuration Présentation des outils nécessaires Premier application Web Conception du model des données Interaction avec une base de données avec JDBC (rappels) Pattern DAO et implémentation d'une couche d'accès aux données Pattern Singleton Tests Dao		Installer et tester les outils et les environnements  Laboratoire :  Révision Java, HTML5 et JavaScript
2	Introduction : Applications distribuées Architectures client-serveur Technologie Java-EE Les servlets Les pages JSP		Laboratoire
3	Gestion de Sessions Gestion de Cookies Actions et Javabeans EL et JSTL		Laboratoire
4	Actions et Javabeans EL et JSTL Architecture MVC Gestion de persistance et communication avec la couche DAO		Laboratoire
5	Collaboration Mise en place d'une architecture MVC Model 2 Injection de dépendances Servir du contenu statique Couche métier		Laboratoire
<b>OBJECTIF DE L'ÉTAPE 2 : EXPLOITER UN FRAMEWORK DE DÉVELOPPEMENT WEB</b>			
6	Pourquoi des frameworks? Types de frameworks Caractéristiques du framework Spring Installation et configuration de Spring boot		Laboratoire

	Création d'une application Spring Boot		
7	Utilisation de JPA pour communiquer avec une base de données Utilisation de Spring Data JPA		
8	Création du contrôleur Spring MVC Création des vues Spring MVC à l'aide Thymeleaf Les pages Thymeleaf		Laboratoire
9	Création de l'API REST Teste de l'API grâce à Postman Communication entre une application Spring Boot et une application externe		Laboratoire
<b>OBJECTIF DE L'ÉTAPE 3 : DÉVELOPPER DES APPLICATIONS WEB ASYNCHRONES</b>			
10-11	JavaScript (rappels) Norme ECMA Script JavaScript avancé et TypeScript Bibliothèques JavaScript  Programmation asynchrone AJAX et JSON		Laboratoire
12-15	Applications de type <i>single-page</i> Le framework Angular		Laboratoires



## 3.2 Modalités d'encadrement et somme de travail personnel

### 3.3 Disponibilités

Vous pouvez communiquer avec moi par courriel à l'adresse : dahamada@crosemont.qc.ca ou par MIO; je réponds normalement durant la semaine dans les 24 h.

#### Bureau : B-313

Mes disponibilités sont les suivantes.

	LUNDI	MARDI	MERCREDI	JEUDI	VENDREDI
AM			10h à 12h		
PM		11h30 à 13h30			

## 4. ÉVALUATION DES APPRENTISSAGES

### 4.1 Tableau-synthèse des évaluations

SEMAINE	TYPE D'INSTRUMENT D'ÉVALUATION	NATURE DE L'ÉVALUATION FORMATIF/SOMMATIF	PONDÉRATION
6 (3 ou 4 mars)	Examen 1	Sommatif	20%
10 (14 avril)	Examen 2	Sommatif	20%
15 (20 ou 23 mai)	Examen 3	Sommatif	25%
Durant toute la session	Laboratoires	Formatif	Bonus*
Durant toute la session	Projet de session	Sommatif	35%
Présentation des projets de Session (21 et 22 mai)			

\*Les laboratoires, exercices, activités et devoirs formatifs réalisés en classe ou en dehors de la classe peuvent, à la discrétion du professeur, être utilisés pour bonifier la moyenne des étudiants méritants qui effectuent régulièrement leurs activités afin de consolider leurs apprentissages.

## 4.2 Description des évaluations sommatives

### 1. Examen 1

L'examen portera sur la programmation web avec Java (étape 1 du cours). L'étudiant.e doit démontrer qu'il/elle maîtrise les concepts, standards et techniques de programmation d'applications web transactionnelles à l'aide de la technologie Java.

La documentation et Internet sont permis.

L'utilisation de l'IA durant l'examen est **strictement interdite**.

Une **partie ou la totalité** de l'examen pourrait être à **faire sur papier** avec ou sans un résumé de la documentation fourni par l'enseignant, et pourra contenir aussi des questions théoriques.

### 2. Examen 2

L'examen portera sur la programmation web avec le framework *Spring* (étape 2 du cours). L'étudiant.e doit démontrer qu'il/elle maîtrise les concepts et les aspects architecturaux propres au framework et qu'il/elle est capable d'exploiter le framework pour développer une application web transactionnelle.

La documentation et Internet sont permis.

L'utilisation de l'IA durant l'examen est **strictement interdite**.

Une **partie ou la totalité** de l'examen pourrait être à **faire sur papier** avec ou sans un résumé de la documentation fourni par l'enseignant, et pourra contenir aussi des questions théoriques.

### 3. Examen 3

L'examen portera sur la programmation web asynchrone (étape 3 du cours). L'étudiant.e doit démontrer qu'il/elle maîtrise les concepts, normes et techniques de programmation asynchrone dans le web. Il/elle doit aussi démontrer qu'il/elle est capable d'exploiter le framework *Angular* pour développer des applications de type *single-page* avec le langage *JavaScript/TypeScript*.

La documentation et Internet sont permis.

L'utilisation de l'IA durant l'examen est **strictement interdite**.

Une **partie ou la totalité** de l'examen pourrait être à **faire sur papier** avec ou sans un résumé de la documentation fourni par l'enseignant, et pourra contenir aussi des **questions théoriques**.

### 4. Projet de session

Le projet de session constitue l'évaluation finale du cours. Tout au long de la session, l'étudiant.e doit développer une application web transactionnelle asynchrone intégrant tous les apprentissages du cours.

Le projet est divisé en trois étapes distinctes, chacune avec une pondération spécifique :

1. **Version 1** : Développement de l'application en **Java JEE** – cette étape représente **5 %** de la note totale du projet.
2. **Version 2** : Migration ou développement en **Spring Boot** – cette étape constitue **25 %** de la note totale du projet.
3. **Version 3** : Intégration d'une interface utilisateur en **Angular** couplée à **Spring Boot** – cette étape vaut **5 %** de la note totale du projet.

À la fin de la session, l'étudiant.e devra présenter son projet, démontrant les fonctionnalités développées et la compréhension des concepts abordés.

### Critères d'évaluation et pondération

- Implémentation correcte de la logique applicative coté client
- Implémentation correcte de la logique applicative coté serveur
- Exploitation judicieuse des mécanismes de programmation asynchrone
- Respect des standards architecturaux
- Respect des standards des interfaces pour le web
- Évaluation individuelle et par équipe

## 1. MÉDIAGRAPHIE

### 1.1 Documents obligatoires

Aucun

### 1.2 Documents de référence

à fournir en classe

### 1.3 Référence à la PIEA diffusé sur le site du Collège de Rosemont

Lien : [http://www.crosemont.qc.ca/public/bd6fad11-ee4c-4d0c-a5d7-c094a8a27d06/mes\\_documents/college\\_reglements\\_politiques\\_programmes/piea\\_version\\_ca\\_22\\_fevrier\\_2010.pdf](http://www.crosemont.qc.ca/public/bd6fad11-ee4c-4d0c-a5d7-c094a8a27d06/mes_documents/college_reglements_politiques_programmes/piea_version_ca_22_fevrier_2010.pdf)

### 1.4 Modalités départementales d'application des règles de la PIEA.

#### 1.4.1 Modalités de participation au cours

- La présence au cours et en laboratoire est requise et **obligatoire**.
- Un étudiant qui ne se présente pas à un cours doit se renseigner sur la matière vue lors de son absence ainsi que sur les travaux à faire pour le prochain cours.
- Tout étudiant en retard doit attendre la pause (50 min après le début du cours) pour entrer dans la salle du cours, et ceci afin de ne pas perturber l'enseignement aux étudiants présents. Ce retard sera considéré comme une absence à moins qu'il ne soit justifié par une raison valable.
- Un retard et un départ hâtif peuvent être considérés par l'enseignant comme une absence au cours.
- Les étudiants doivent respecter les règles de fonctionnement, de bienséance et d'éthique particulières à chacun des cours. Celles-ci seront énoncées par l'enseignant lors du premier cours.
- **Les jeux sont prohibés dans les salles de cours et laboratoires.**

#### 1.4.2 Règles départementales et politiques en vigueur

- Tout **plagiat, toute tentative de plagiat, toute collaboration à un plagiat** entraîne la note zéro « 0 » pour l'évaluation en cause.
- Les consignes et les délais de remise de travaux doivent **impérativement** être respectés.
- Les travaux demandés doivent être réalisés en respectant les règles énoncées dans le respect des particularités de chacun des cours.
- L'étudiant doit se tenir informé des dates fixées pour les examens et est tenu de se présenter à ceux-ci. En cas d'absence, seules les situations jugées exceptionnelles par l'enseignant pourraient être considérées en vue d'une reprise d'un examen. Un étudiant qui se présente à une activité d'évaluation n'a pas le droit de reprise. Toute absence non justifiée par un motif exceptionnel entraîne la note zéro (0). En cas d'absence justifiée par des motifs exceptionnels, l'étudiant doit faire une demande par écrit au bureau du registraire dans les cinq (5) jours ouvrables suivant la date de l'activité d'évaluation. Le dernier examen de session ne peut être reporté sauf dans le cas de mesures exceptionnelles et avec présentations d'une pièce justificative.

- Un travail perdu, pour quelque raison que ce soit, est considéré comme un travail non remis. Pour éviter qu'une telle situation ne survienne, l'étudiant a la responsabilité de se garder des copies de sauvegarde de la copie originale. Un travail envoyé par courrier électronique doit obtenir un accusé de réception de la part de l'enseignant dans les 24 heures suivant son envoi. Si aucun accusé de réception n'est reçu à l'intérieur de ce laps de temps, l'étudiant a la responsabilité de contacter l'enseignant le plus rapidement possible afin de s'assurer de la réception du travail. Seul un accusé de réception de l'enseignant peut servir de preuve pour la remise d'un travail.
- L'étudiant qui ne respecte pas les délais imposés pour la remise d'un travail, **perd dix pour cent (10 %) des points par jour de retard**, à moins d'entente préalable avec l'enseignant.
- 
- Un travail **ne peut plus être remis après la correction** de celui-ci en classe avec l'enseignant ou lorsque les copies corrigées ont été retournées aux étudiants.
- Aucun travail ne doit être accepté par l'enseignant après le dernier jour de la session, à moins d'une entente établie à titre exceptionnel.
- Pour la réussite de certains cours comportant des travaux pratiques, un **seuil de 50% pour les examens** est exigé, sans quoi les travaux pratiques seront **pondérés à la baisse**, ce qui amènera par conséquent un échec. Votre enseignant vous précisera au début de votre cours si cette règle s'applique à ce cours.
- Les travaux et les examens faits en cours de session, excluant l'examen final, doivent être corrigés dans les deux semaines suivant la remise des copies à l'enseignant. Les étudiants doivent avoir accès au corrigé ou recevoir des explications sur la nature de leurs erreurs. Si un étudiant n'est pas satisfait de l'évaluation de son travail ou de son examen, il en informera son enseignant dans le même cours où il reçoit sa copie. Celui-ci vérifiera s'il y a une erreur dans l'application du barème de correction et modifiera, s'il y a lieu la note attribuée. Si l'étudiant demeure insatisfait, il sera invité à présenter une demande de révision de note selon la procédure établie.
- Dans le cas des travaux d'équipe, l'enseignant peut attribuer une note différente aux étudiants d'une même équipe afin de refléter leur apprentissage individuel. Pour sa part, l'étudiant peut demander à l'enseignant de faire en sorte que sa note reflète bien ses propres apprentissages.
- En conformité avec la Politique de valorisation de la langue française adoptée par le Collège, dans tous les cours de tous les programmes, la qualité de la langue écrite ou parlée fait partie intégrante des critères d'évaluation. En conséquence, un pourcentage de 10 % peut être retranché relativement à la qualité du français écrit dans une activité d'évaluation. Un demi-point est enlevé (0,5%) pour chaque erreur de vocabulaire, de syntaxe, de ponctuation d'orthographe d'usage et d'orthographe grammaticale jusqu'à un maximum de 10% de la note. Au-delà de ce seuil, votre professeur(e) peut refuser un travail.

- Les dates sont à titre indicatif et pourraient être ajustées en fonction de la progression du groupe et/ou d'événements majeurs.
- Celui qui s'absente à **plus de 20%** des cours ne sera pas autorisé à se présenter à l'examen final.

### 1.4.3 La trousse Je réussis

Lien : <https://www.crosemont.qc.ca/je-reussis/>