



Plan d'études

Techniques d'intégration multimédia

Programmation Web 2

582-223-HU

Pierre Roy

Bureau : 2537D

Téléphone : 770-4012 poste 3820

Courriel : pierre.roy@cegepoutaouais.qc.ca

Campus Félix-Leclerc

Hiver 2026

1. INTRODUCTION

1.1 Intentions pédagogiques

À cette session, le cours porteur est 582-225-HU - Intégration Web 2 et les notions vues dans le cours 582-223-HU - Programmation Web 2 y seront mises en contexte et intégrées dans un projet pour diffusion dans le cours porteur.

Le cours 582-223-HU - Programmation Web 2 vise à permettre aux étudiants d'approfondir leur compréhension du vocabulaire de la programmation et des concepts d'un langage de programmation.

L'étudiant poursuit son développement d'apprentissage des outils, de la manipulations des langages balisés et des scripts nécessaires à la diffusion en ligne pour le volet client d'un environnement client/serveur.

L'étudiant sera capable de créer des scripts optimisés dans le langage choisi et d'utiliser les bons outils en fonction de l'environnement de diffusion en ligne.

1.2 Brève description du cours

L'étudiant apprendra à manipuler la programmation dans un contexte de réalisation de projets interactifs en multimédia.

Dans ce cours, l'étudiant approfondira sa compréhension d'un langage balisé, d'un langage de programmation de scripts et de l'utilisation de feuilles de style. L'étudiant sera en mesure de structurer, d'organiser et de présenter les différents éléments constituant un document dynamique. Il produira ainsi des projets dynamiques pour une diffusion en ligne.

1.3 Pondération

La pondération du cours (1-2-2) correspond à une heure d'activités théoriques en classe, à deux heures de laboratoire et à deux heures de travail personnel en dehors des périodes de cours.

1.4 Relation entre le cours et d'autres cours du programme

Le cours 582-223-HU Programmation Web 2 est le deuxième cours de programmation de la formation. Tout au long de la formation, l'étudiant aura un minimum d'un cours de programmation par session.

Il est un préalable relatif au cours 582-323-HU Programmation multiplateformes.

1.5 Profils de sortie

Lorsqu'il est question de l'exploitation de langages de programmation utilisés en multimédia, l'employeur s'attend à ce que l'intégrateur soit **responsable** de programmer les fonctionnalités permettant l'interactivité **d'application multimédia** en plus d'assembler et d'intégrer les différents éléments de contenu.¹

2. COMPÉTENCES À DÉVELOPPER

2.1. Énoncé de la compétence

(015F) Exploiter les langages de programmation utilisés en multimédia.

2.2. Éléments de la compétence

Choisir un outil de programmation multimédia.
Produire des algorithmes.
Programmer l'interactivité.
Adapter la programmation aux différentes plates-formes.
Documenter la programmation.
Résoudre des erreurs de programmation.

2.3. Les critères de performance

- 1.1 Reconnaissance exacte des différents langages de programmation et de leurs applications en production multimédia.
- 1.2 Reconnaissance exacte des différents logiciels utilisés en programmation multimédia.
- 1.3 Relevé exhaustif des sources d'information relatives à la programmation multimédia.
- 1.4 Utilisation pertinente de la documentation technique relative aux langages et aux logiciels de programmation multimédia.
- 2.1 Reconnaissance exacte de la structure logique des langages de programmation.
- 2.2 Analyse détaillée de la problématique à résoudre.
- 2.3 Analyse détaillée des opérations d'entrée et de sortie des routines du programme.
- 2.4 Choix approprié de la structure de programmation.
- 2.5 Développement pertinent de la logique des opérations.
- 2.6 Utilisation judicieuse de différents modes de représentation de l'algorithme.
- 2.7 Vérification complète des algorithmes.
- 3.1 Codification correcte de l'algorithme.
- 3.2 Exploitation pertinente des fonctions propres aux langages de programmation événementielle.

¹ Gouvernement du Québec, Ministère de l'Éducation, Description générale de la profession, *Technicienne et technicien en intégration multimédia : Rapport d'analyse de situation de travail*, La formation professionnel et technique, 1998, p.3

- 3.3 Exploitation correcte des bibliothèques de codes et d'extensions au langage fournis.
- 3.4 Programmation fonctionnelle des liens dynamiques et des effets sur les éléments de navigation.
- 3.5 Création précise des routines générales et de gabarits.
- 3.6 Adaptation correcte de la programmation aux paramètres techniques du projet.
- 3.7 Adaptation judicieuse des codes fournis.
- 3.8 Programmation modulaire du code.
- 3.9 Aménagement ergonomique du poste de travail.
- 4.1 Modifications fonctionnelles des routines de programmation pour les plates-formes utilisées.
- 4.2 Programmation correcte de l'affichage des médias en fonction des différentes plates-formes.
- 5.1 Lisibilité des sources.
- 5.2 Clarté et précision des scripts.
- 6.1 Test rigoureux de validation.
- 6.2 Repérage exact des erreurs de syntaxe, de logique et d'implantation.
- 6.3 Emploi efficace des fonctions de déminage des logiciels de programmation.
- 6.4 Analyse exacte des problèmes rencontrés.
- 6.5 Solutions appropriées aux problèmes de codification.

3. LE CONTENU DU COURS

Les savoirs

- Le vocabulaire utilisé en programmation, les concepts de base de la programmation, les structures décisionnelles, concept d'événement interactif, concept d'entrées / sorties des données.

Les savoir-faire

- Techniques de programmation, l'utilisation de fonctions et méthodes propre à un langage, construction d'une fonction personnalisée, lecture et compréhension d'un scénario interactif, utilisation d'une grille test, lecture des messages d'erreurs.

Les savoirs être

- Autonomie et débrouillardise, capacité d'analyse et de synthèse, rigueur et précision, effort et travail constant.

4. LES MÉTHODES D'ENSEIGNEMENT ET LES ACTIVITÉS D'APPRENTISSAGE

Les activités d'enseignements seront variées. Sous forme de présentations et/ou de démonstrations, l'enseignant assistera l'étudiant dans l'apprentissage des techniques de programmation d'interactivité en lui donnant des exemples précis sur l'application de méthodes de travail prescrite par le cours.

Cela se fera par l'utilisation d'une animation, pour ensuite explorer le code, le lire, le comprendre, le commenter et l'adapter. Puis les différents éléments de programmation seront vus en détails.

Des exercices pratiques suivront chaque apprentissage théorique afin de permettre à l'étudiant d'intégrer les notions apprises et de maîtriser les outils et les méthodes enseignées chaque semaine.

5. L'ÉVALUATION DES APPRENTISSAGES

5.1. Les modalités des évaluations formatives

Les exercices en classe et les devoirs à la maison pourront être évalués par l'enseignant à la demande de l'étudiant à condition que l'étudiant soit présent au cours et fait preuve d'efforts soutenus lors de la réalisation des activités d'apprentissages prescrit par l'enseignant.

5.2. Les modalités des évaluations sommatives

Les travaux devront rendre compte de l'acquisition adéquate, par l'étudiant, des éléments de compétences rattachés au cours. L'enseignant peut refuser un travail qu'il considère incomplet ou qui n'entre pas dans le contexte de réalisation prescrit par l'activité. Aucun délai ne sera accordé pour la remise des travaux. Les travaux doivent être remis aux cours prévus dans le calendrier des activités.

Évaluation	Pondération	Semaine de remise
Examen 1	30 %	6
Examen 2	30 %	11
Projet synthèse	40 %	15
Total	100 %	

5.3. Critères d'évaluation de l'épreuve certificative finale du cours

Exploiter les langages de programmation utilisés en multimédia (015F)

Critères	Pondération
1 Structure des fichiers et sémantique du langage utilisé	40 %
2 Produire des algorithmes	10 %
3 Adaptation et modification du code	35 %
4 Respect du contexte de réalisation et du mandat	15 %

5.4. Révision de la note de l'évaluation certificative finale du cours

Toute demande de révision de la note de l'évaluation certificative finale du cours doit obligatoirement respecter la procédure suivante :

- avant d'entreprendre une demande officielle de révision de note dans Omnivox, l'étudiante ou l'étudiant doit d'abord communiquer avec l'enseignante ou l'enseignant responsable du cours afin de comprendre les motifs de la note obtenue. Cette étape vise à assurer une compréhension claire de l'évaluation, des critères et de l'application de la grille de correction.
- si, à la suite de cette discussion, l'étudiante ou l'étudiant estime toujours que la note ne reflète pas adéquatement son travail, une demande officielle de révision de note peut alors être déposée par l'entremise du module **Demande de révision de notes** dans Omnivox, dans les délais prescrits par le Cégep.

Toute demande soumise sans avoir préalablement communiqué avec l'enseignante ou l'enseignant pourrait être jugée irrecevable par le département.

6. LE CALENDRIER DES ACTIVITÉS²

Semaines 1, 2, 3, 4 et 5		
Module 1	Le langage JavaScript	
Section 1	Variables, types de données	15 heures
Section 2	Opérateurs	
Section 3	Instructions conditionnelles	
Section 4	Dépannage	
Section 5	Évènements	
Semaine 6		
Examen 1 sur le module 1		3 heures
Semaines 7 et 8		
Module 2	Fonctions	
Section 1	Concevoir des fonctions	6 heures
Semaine 9, 10 et 11		
Module 3	Mise en pages	

² Le calendrier des activités est indicatif et s'ajuste en cours de session en fonction du rythme d'apprentissage des étudiantes et des étudiants

Section 1	Positionnement CSS	9 heures
Section 2	Attributs CLASS	
Section 3	Balises DIV	
Semaine 12		
Examen 2 sur les modules 2 et 3		3 heures
Semaines 13, 14 et 15		
	Finalisation du travail synthèse	9 heures
	Remise du travail synthèse	

7. LES RÈGLES DÉPARTEMENTALES

Tous les règlements adoptés par le département de techniques d'intégration multimédia s'appliquent en tout temps. Voir le document « Règlements et remarques » remis lors du premier cours de la session.

8. SITES WEB DU COURS

www.w3schools.com/js/default.asp