

582 311 MO Web 3

Nombre d'heures d'enseignement :	60h
Pondération :	1-3-2
Programme :	Techniques d'intégration multimédia
Département du programme :	Techniques d'intégration multimédia
Session :	A2026
Professeure ou professeur :	Jean-François Cartier
Département de la professeure ou du professeur :	Techniques d'intégration multimédia
Courriel :	jfcartier@cmontmorency.qc.ca
Bureau :	C-1647
Plateforme pédagogique utilisée :	☒ Teams ☐ Timdoc ☒ GitHub ☒ Autre
Coordination :	Lora Boisvert
Contact de la coordination :	lora.boisvert@cmontmorency.qc.ca Bureau : C-1651

PRÉSENTATION DU COURS

DESCRIPTION DU COURS

L'élève apprend à appliquer les notions d'interactivité avancées afin de réaliser l'intégration d'un site Web audiovisuel.

Au terme de ce cours, l'élève combine les notions de mise en page avec feuilles de styles, d'interactivité et de manipulation des contenus par programmation. Il raffine ces notions plus avancées et bonifie un produit multimédia à l'aide de différentes librairies et cadriciels.

OBJECTIF INTÉGRATEUR

Programmer l'interface audiovisuelle d'un site Web à l'aide de librairies et d'un cadriciel.

COMPÉTENCE(S) MINISTÉRIELLE(S)

015L intégrer les médias pour la diffusion en ligne (éléments 4 et 5)

015P Programmer des produits multimédias (éléments 1 et 2)

VOLETS TRANSVERSAUX

- Compétence langagière : L'élève adapte son discours à la situation de communication, communique adéquatement à l'oral et à l'écrit, soigne ses productions écrites et utilise le lexique de la profession.
- Compétence numérique : L'élève exploite les TIC de manière efficace et responsable. Il recherche, traite et présente de l'information.

ATTITUDES PROFESSIONNELLES : Autonomie, Débrouillardise

OBJECTIFS D'APPRENTISSAGE

1. Intégrer les médias audiovisuels dans une interface Web.
2. Exploiter un cadriciel et des librairies permettant
3. d'augmenter l'efficacité de l'interactivité.

COURS LIÉS (PRÉALABLES ABSOLUS, RELATIFS, COREQUIS)

Les cours suivants sont préalables absolus au présent cours	Le présent cours est préalable absolu aux cours suivants
--	---

420 V11 MO Programmation interactive

582 411 MO Web 4

582 211 MO Web 2

582 541 MO Préparation au milieu de travail

CONTEXTE D'APPRENTISSAGE ET MÉTHODES PÉDAGOGIQUES

- ☒ Exposés magistraux
- ☒ En laboratoire informatique
- ☐ Apprentissage et utilisation de logiciels sous forme de démonstrations, d'exercices, de travaux pratiques
- ☒ Projets multimédias
- ☒ Exposés interactifs
- ☐ Écoute de pistes sonores
- ☐ Activités coopératives
- ☒ Tutorat individuel ou de petits groupes
- ☒ En présence
- ☐ En ligne
- ☐ Stage en milieu de travail
- ☐ Discussions en groupe (tables rondes)

MATÉRIEL, COMPTES ET VOLUMES REQUIS

- ☒ Disque dur portatif.
- ☐ Carte SD (pour les détails voir le [guide étudiants](#)).

- ☐ Clé USB.
- ☐ Cahier ou papiers divers, pour griffonner, conceptualiser, réaliser des croquis et noter vos inspirations.
- ☐ Pour le travail à la maison avec _____, il vous faut _____.
- ☐ Prévoir un budget d'environ ____\$ pour _____.
- ☐ Compte GitHub, Codepen.

RÉSERVATION DU MATÉRIEL SPÉCIALISÉ

Lorsque vous êtes autorisée par votre professeure ou professeur pour emprunter le matériel spécialisé, vous pouvez le réserver auprès des TTP en suivant la procédure suivante: [Procédure de prêt – TTP.pptx](#)

RÈGLES D'IDENTIFICATION DES TRAVAUX

À moins d'indication contraire, les remises de travaux doivent être nommées de la manière suivante :

Travail individuel

[nom de famille]-[prénom]_[nom du travail]_[#travail]_[#cours]
 Exemple : **gilbert-charlene_polychromes_01_582-314M0**

Travail d'équipe

[les noms de famille en ordre alphabétique séparés par -]_[nom du travail]_[#travail]_[#cours]
 Exemple : **boisvert-gilbert-hubert_polychromes_01_582-314M0**

Les travaux non identifiés ne seront pas corrigés et les pénalités de retard s'appliqueront.

Dates importantes

- **18 septembre** : Date limite de désinscription au cours sans mention au bulletin.
- **4 novembre** : Date limite d'abandon de cours sans échec entraînant une mention au bulletin.

DÉROULEMENT DU COURS

#	À PRÉPARER AVANT LA CLASSE	SAVOIRS ESSENTIELS / CONTENUS	ACTIVITÉS EN CLASSE
1	Créer un compte GitHub et un compte CodePen	CSS Grid et ligne de commande • CSS Grid : grid-template-columns/rows, zones nommées, gap, alignements, placement manuel • Le terminal : pwd, ls, cd, mkdir, touch, cp, mv, rm, echo • Présentation du plan de cours et du projet final	Atelier d'exercices assisté
2		Cadriciels CSS - Tailwind • Qu'est-ce qu'un cadriciel CSS. Exemple de Milligram • Tailwind : classes utilitaires, espacements, couleurs, typographie, bordures, tailles, effets, display, transform • Variantes responsives, états interactifs, mode sombre	Retour sur les exercices du cours précédent Atelier d'exercices assisté
3		Gestionnaire de paquets, Vite et DaisyUI • Du CDN à npm : package.json, node_modules, .gitignore • Vite : création de projet, serveur de développement, build • Tailwind via @tailwindcss/vite • DaisyUI : couleurs sémantiques, btn, badge, alert, card, navbar, menu	Retour sur les exercices du cours précédent Atelier d'exercices assisté
4		Composantes et thèmes • hero, tabs, dropdown, carousel, drawer • Surcharge et adaptation d'une composante avec Tailwind • Thèmes : activation, theme-controller, thème maison	Retour sur les exercices du cours précédent Atelier d'exercices assisté
5		Mise en ligne • Serveur, domaine, hébergement ; code source vs build • GitHub Pages / Codeberg : base dans vite.config.js, workflow d'automatisation • cPanel : gestionnaire de fichiers, courriels, FTP (ftp-simple) • Validation W3C du HTML et du CSS Énoncé de remise des exercices (1 ^{ère} partie)	Retour sur les exercices du cours précédent Atelier d'exercices assisté
6	Étudier pour l'examen  Déposer le dossier de remise des exercices des cours 1, 2, 3, 4 et 5 (10%)	 Examen 01 (25 %) Reproduction d'une maquette dans un projet Vite : Tailwind, composantes DaisyUI adaptées, thème, responsive, dépôt GitHub et mise en ligne.	Évaluation sommative : Examen

7		Alpine.js • Cadriciel ou librairie ? Installation • x-data, x-text, x-html, x-on/@, x-model, x-show/x-if, x-bind/::, x-for • Propriétés magiques • Persistance de l'état : localStorage et plugin \$persist	Retour sur l'examen 01 et sur la remise des exercices (1re partie) Atelier d'exercices assisté
8		Médias audiovisuels <i>Partie 1</i> - Formats d'image (WebP, AVIF), compression, dimensionnement, loading="lazy", vidéo (<source> de repli, poster), audio, classement et nomenclature, validation W3C <i>Partie 2</i> - API native <video>/<audio>, introduction aux classes JavaScript, Howler.js, DevTools	Retour sur les exercices du cours précédent Atelier d'exercices assisté Déclaration du volet de projet (A ou B)
9		GSAP - les bases • Installation ; le <i>tween</i> : to(), from(), fromTo() • Cibles, propriétés animables, paramètres • Courbes d'accélération (ease), décalage (stagger) • <i>Timelines</i> et contrôle de lecture	Retour sur les exercices du cours précédent Atelier d'exercices assisté
10		GSAP - ScrollTrigger et sprites • Activation d'un plugin • ScrollTrigger : start, end, toggleActions, scrub, marqueurs • Animation de sprite • Animation responsive : gsap.matchMedia()	Retour sur les exercices du cours précédent Atelier d'exercices assisté
11		GSAP - parallaxe, MotionPath et SVG • Épinglage de section (pin) • Parallaxe multicouches • MotionPathPlugin : déplacement le long d'un tracé • DrawSVGPlugin : dessin de tracés • snap Présentation du projet final et du plan de production	Retour sur les exercices du cours précédent Atelier d'exercices assisté
12		Autres librairies • Chart.js : <canvas>, structure des données, types de graphiques, options • Tone.js : notes, séquences, lecture de fichiers • Three.js (survol) Énoncé de remise des exercices (2 ^{ème} partie)	Retour sur les exercices du cours précédent Atelier d'exercices assisté Travail sur le projet final

13	 Déposer le dossier de remise des exercices des cours 7, 8, 9, 10, 11 et 12 (15%) Médias du projet rassemblés	Atelier projet (1/2) • Assemblage des sections avec Tailwind + DaisyUI • Interactivité Alpine, animations GSAP • Formulaire de contact fonctionnel avec EmailUS • Intégration de la démo jouable (volet A) ou de la bande-annonce (volet B)	Retour sur les exercices du cours précédent Travail sur le projet final
14	Poursuivre le projet final	Atelier projet (2/2) et gel • Vérification responsive (mobile, tablette, bureau) • Validation W3C, tests multinavigateurs, retrait des traces de débogage • npm run build et déploiement (GitHub Pages ou cPanel) • Gel des fonctionnalités : correctifs et polissage seulement • Préparation de la présentation orale	Atelier encadré et rencontres individuelles concernant le projet final
15	Terminer le projet final	 Remise et présentation orale du projet final (50%)	Évaluation sommative : présentation orales et post-mortem

ÉVALUATION DES APPRENTISSAGES

Les **évaluations formatives** jouent un rôle crucial dans le processus d'apprentissage en offrant à l'étudiant une occasion continue de perfectionner ses connaissances, de comprendre ses points forts et ses domaines à améliorer. Ces évaluations visent à soutenir activement le développement des compétences en offrant des conseils personnalisés pour renforcer la compréhension des concepts enseignés et améliorer la capacité de les appliquer de manière pratique.

D'un autre côté, les **évaluations sommatives** sont utilisées pour évaluer les acquis et les connaissances de l'étudiant à un moment donné. Elles servent à mesurer la réussite et la maîtrise des objectifs d'apprentissage à la fin d'une période déterminée. Les résultats des évaluations sommatives fournissent une évaluation globale du niveau de compétence atteint par l'étudiant.

Dans le cas où des étudiants auraient formulé une demande en raison de besoins spécifiques pour bénéficier de temps supplémentaire lors des évaluations, l'enseignant essaiera de respecter les recommandations émises pour favoriser le succès de l'étudiant. Cette mesure vise à garantir l'équité et l'accessibilité, permettant à tous les apprenants de démontrer leur compréhension de manière juste et équitable. Cependant c'est la responsabilité de l'étudiant de faire une demande au SAA (Service d'aide à l'apprentissage) au moins 7 jours avant l'évaluation sommative afin de disposer de leur temps supplémentaire accordé dans leurs locaux.

ÉVALUATIONS FORMATIVES

Le cours suit chaque semaine le même cycle : un exposé magistral en première partie de séance, puis un atelier d'exercices pratiques qui occupe le reste du cours.

Tous ces ateliers sont assistés. Pendant que la classe travaille, l'enseignant circule, répond aux questions, commente le code en cours d'écriture et débloque les étudiants qui butent sur un problème.

Le retour sur les exercices

Chaque séance de 2 à 12 s'ouvre par un retour de 10 à 15 minutes sur les exercices de la semaine précédente, avant le nouvel exposé :

1. l'enseignant présente la solution de référence et les deux ou trois pièges classiques de l'exercice ;
2. chaque étudiant ouvre sa propre version en parallèle et relève les écarts ;
3. les questions sur ces écarts sont traitées devant le groupe.

Auto-évaluation au fil de la session

Au moment des retours, l'étudiant remplit une colonne Auto-évaluation dans son README.md de son dépôt web3-exercices :

- ☒ Réussi seul
- ☐ Réussi avec de l'aide
- ☒ Pas réussi, à reprendre

ÉVALUATIONS SOMMATIVES

DESCRIPTION ET FORME DE L'ÉVALUATION	SAVOIRS ESSENTIELS / PRINCIPAUX CRITÈRES D'ÉVALUATION	ÉCHÉANCE	%
Atelier Exercices des cours 1, 2, 3, 4 et 5 ☒ Individuel ☐ Équipe	Cette évaluation mesure l' autonomie , la débrouillardise et la régularité du travail sur le bloc <i>outillage et intégration</i> . Positionnement des éléments selon les maquettes (CSS Grid) ; prise en charge d'un cadriciel CSS permettant de faciliter l'intégration Web (Tailwind) ; réutilisation et adaptation des composantes régulières d'une interface Web (DaisyUI) ; réactivité ; compatibilité sur les navigateurs modernes ; validation du code et repérage d'erreur.	Veille du cours 6 à 23h59	10%
Examen Examen 01 ☒ Individuel ☐ Équipe	Positionnement des éléments selon les maquettes ; prise en charge d'un cadriciel CSS pour faciliter l'intégration Web (système de grille, conteneurs, points d'arrêt, typographie, couleurs) ; réutilisation et adaptation des composantes régulières d'une interface Web (navbar, hero, carte, tiroir, onglets, formulaire) par surcharge, variables et thèmes ; réactivité ; structure et nomenclature des fichiers dans un projet avec cadriciel ; compatibilité sur les navigateurs modernes ; validation du code et repérage d'erreur.	Cours 6	25%
Atelier Exercices des cours 7, 8, 9, 10, 11 et 12 ☒ Individuel ☐ Équipe	Cette évaluation mesure l' autonomie , la débrouillardise et la régularité du travail sur le bloc <i>interactivité, médias et animation</i> . Introduction à un cadriciel JavaScript (Alpine) ; programmation fonctionnelle et intégration de l'interactivité ; gestion de sauvegarde dynamique de données côté client ; intégration, contrôle, classement et optimisation des médias audiovisuels ; classe JavaScript et classes sur mesure ; structure logique d'une programmation basée sur des événements ; animation à l'aide d'une librairie ; réactivité.	Veille du cours 13 à 23h59	15%

INTÉGRATION DES APPRENTISSAGES <i>Activité d'évaluation démontrant l'atteinte de l'objectif intégrateur du cours</i> Projet Projet final - Site promotionnel ☒ Individuel ☐ Équipe	Création d'un site Web qui intègre l'interactivité des interfaces, des médias audiovisuels et des animations, dont la mise en page est facilitée par l'utilisation d'un cadriciel. Prise en charge d'un cadriciel pour faciliter l'intégration et pour réutiliser les composantes régulières en les adaptant ; positionnement des éléments selon les maquettes ; réactivité ; introduction à un cadriciel JavaScript ; programmation fonctionnelle et intégration de l'interactivité ; sauvegarde dynamique de données côté client ; structure logique d'une programmation basée sur des événements ; animation à l'aide d'une librairie ; intégration, contrôle, classement et optimisation des médias audiovisuels ; classes JavaScript et classes sur mesure ; compatibilité sur les navigateurs modernes ; validation du code et repérage d'erreur.	Cours 15	50%
Total :			100 %

Médiagraphie

Références transversales

- MDN Web Docs (Mozilla) | Référence HTML, CSS, JavaScript et API Web : <https://developer.mozilla.org/fr/>
- World Wide Web Consortium (W3C) | Organisme de normalisation du Web : <https://www.w3.org/>
- Can I use... - Compatibilité des navigateurs, propriété par propriété : <https://caniuse.com/>
- Baseline (Google / W3C WebDX) | Savoir si une fonctionnalité est largement disponible : <https://web.dev/baseline>
- Chrome DevTools (Google) | Documentation des outils de développement : <https://developer.chrome.com/docs/devtools>
- Chrome DevTools | Console : <https://developer.chrome.com/docs/devtools/console>
- Chrome DevTools | Réseau : <https://developer.chrome.com/docs/devtools/network/overview>
- Chrome DevTools | Performance : <https://developer.chrome.com/docs/devtools/performance/overview>

Mise en page | CSS Grid

- Grilles CSS (CSS Grid) | MDN | Portail du module : https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Grid_layout
- Les concepts de base des grilles CSS | MDN : https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Grid_layout/Basic_concepts
- Définir des zones sur une grille | MDN | `grid-template-areas` :
https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Grid_layout/Grid_template_areas
- Placer les éléments sur les lignes d'une grille | MDN : https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Grid_layout/Line-based_placement
- Le placement automatique sur une grille | MDN : https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Grid_layout/Auto-placement
- L'alignement des boîtes avec une grille | MDN : https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Box_alignment/In_grid_layout
- Les grilles CSS et l'accessibilité | MDN : https://developer.mozilla.org/fr/docs/Web/CSS/Guides/Grid_layout/Accessibility
- CSS Grid Layout Module Level 1 | W3C : <https://www.w3.org/TR/css-grid-1/>
- Grid Garden | Pratique du placement sur grille : <https://cssgridgarden.com/>

Ligne de commande et environnement de travail

- Cours express sur l'invite de commande | MDN :
https://developer.mozilla.org/fr/docs/Learn_web_development/Getting_started/Environment_setup/Command_line
- Node.js : <https://nodejs.org/fr>
- Gestionnaire de paquets npm : <https://docs.npmjs.com/>
- Créer un fichier package.json | npm : <https://docs.npmjs.com/creating-a-package-json-file>
- Gestion sémantique de version | npm : <https://docs.npmjs.com/about-semantic-versioning>
- Git | Documentation : <https://git-scm.com/docs>
- gitignore.io | Génération d'un `.gitignore` : <https://www.toptal.com/developers/gitignore>

Outil de construction | Vite

- Vite | Guide : <https://vite.dev/guide/>
- Vite | Compilation pour la production : <https://vite.dev/guide/build.html>
- Vite | Déployer un site statique (section GitHub Pages, réglage de `base`) : <https://vite.dev/guide/static-deploy.html>
- Vite | Gestion des ressources statiques (dossier `public`) : <https://vite.dev/guide/assets.html>
- Vite | Référence de configuration : <https://vite.dev/config/>

Cadriels CSS | Tailwind et DaisyUI

- Tailwind CSS | Documentation : <https://tailwindcss.com/docs>
- Tailwind | Installation avec Vite : <https://tailwindcss.com/docs/installation/using-vite>
- Tailwind | Classes utilitaires : <https://tailwindcss.com/docs/styling-with-utility-classes>
- Tailwind | Design responsive : <https://tailwindcss.com/docs/responsive-design>
- Tailwind | États interactifs (`hover`, `focus` ...) : <https://tailwindcss.com/docs/hover-focus-and-other-states>
- Tailwind | Mode sombre : <https://tailwindcss.com/docs/dark-mode>
- Tailwind | Variables de thème : <https://tailwindcss.com/docs/theme>
- Tailwind | Couleurs : <https://tailwindcss.com/docs/colors>
- Tailwind CSS v4.0, ce qui a changé depuis la v3 : <https://tailwindcss.com/blog/tailwindcss-v4>
- Tailwind Play | bac à sable officiel : <https://play.tailwindcss.com/>
- daisyUI | Documentation : <https://daisyui.com/docs/>
- daisyUI | Installation : <https://daisyui.com/docs/install/>
- daisyUI | Composantes : <https://daisyui.com/components/>
- daisyUI | Couleurs sémantiques : <https://daisyui.com/docs/colors/>
- daisyUI | Thèmes : <https://daisyui.com/docs/themes/>
- daisyUI | Personnalisation : <https://daisyui.com/docs/customize/>

- daisyUI | Générateur de thèmes : <https://daisyui.com/theme-generator/>
- Milligram | cadriciel minimaliste, mise en bouche : <https://milligram.io/>
- Bootstrap | point de comparaison : <https://getbootstrap.com/docs/>

Mise en ligne et hébergement

- GitHub Pages | Documentation : <https://docs.github.com/fr/pages>
- Démarrer avec GitHub Pages : <https://docs.github.com/fr/pages/getting-started-with-github-pages>
- GitHub Actions | Documentation : <https://docs.github.com/fr/actions>
- cPanel | Gestionnaire de fichiers : <https://docs.cpanel.net/cpanel/files/file-manager/>
- cPanel | Comptes FTP : <https://docs.cpanel.net/cpanel/files/ftp-accounts/>
- FileZilla | Documentation : <https://wiki.filezilla-project.org/Documentation>

Validation, débogage et compatibilité

- Valideur HTML du W3C : <https://validator.w3.org/>
- Valideur CSS du W3C (Jigsaw) : <https://jigsaw.w3.org/css-validator/>
- W3C Link Checker | repérage des liens morts : <https://validator.w3.org/checklink>
- Google Lighthouse | Introduction : <https://developer.chrome.com/docs/lighthouse/overview>
- Fast load times | web.dev | poids des pages et Core Web Vitals : <https://web.dev/explore/fast>

JavaScript | événements, classes et stockage

- Guide JavaScript | MDN : <https://developer.mozilla.org/fr/docs/Web/JavaScript/Guide>
- Classes | MDN : <https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Classes>

- Les bases de JavaScript orienté objet | MDN : https://developer.mozilla.org/fr/docs/Learn_web_development/Core/Scripting/Object_basics
- Introduction aux évènements | MDN : https://developer.mozilla.org/fr/docs/Learn_web_development/Core/Scripting/Events
- `addEventListener()` | MDN : <https://developer.mozilla.org/fr/docs/Web/API/EventTarget/addEventListener>
- Évènement `scroll` | MDN : https://developer.mozilla.org/fr/docs/Web/API/Element/scroll_event
- API Web Storage | MDN : https://developer.mozilla.org/fr/docs/Web/API/Web_Storage_API
- `Window.localStorage` | MDN : <https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>
- `JSON` | MDN | `stringify()` et `parse()` : https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Global_Objects/JSON
- Méthodes de tableau (`map`, `filter`, `reduce`) | MDN : https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Global_Objects/Array
- `Window.matchMedia()` | MDN : <https://developer.mozilla.org/fr/docs/Web/API/Window/matchMedia>

Cadriciel JavaScript | Alpine.js

- Alpine.js | Start Here : <https://alpinejs.dev/start-here>
- Alpine.js | Installation : <https://alpinejs.dev/essentials/installation>
- Alpine.js | State : <https://alpinejs.dev/essentials/state>
- Alpine.js | Templating : <https://alpinejs.dev/essentials/templating>
- Alpine.js | Events : <https://alpinejs.dev/essentials/events>
- Alpine.js | Lifecycle : <https://alpinejs.dev/essentials/lifecycle>
- Alpine.js | Directives (référence) : <https://alpinejs.dev/directives/data>
- Alpine.js | Propriétés magiques : <https://alpinejs.dev/magics/el>
- Alpine.js | `Alpine.store()` : <https://alpinejs.dev/globals/alpine-store>
- Alpine.js | Greffon Persist (`\$persist`) : <https://alpinejs.dev/plugins/persist>
- Alpine.js | Greffon Intersect (`x-intersect`) : <https://alpinejs.dev/plugins/intersect>
- Alpine.js | Dépôt GitHub : <https://github.com/alpinejs/alpine>

Médias audiovisuels | formats et intégration

- Guide des types et formats d'image | MDN (WebP, AVIF, JPEG, PNG, SVG) : https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Formats/Image_types
- Guide des formats de conteneurs média | MDN (MP4, WebM, Ogg) : <https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Formats/Containers>
- Guide des codecs vidéo | MDN (H.264, VP9, AV1) : https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Formats/Video_codecs
- Guide des codecs audio | MDN (MP3, AAC, Opus, Vorbis) : https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Formats/Audio_codecs
- ``<video>`` | MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/video>
- ``<audio>`` | MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/audio>
- ``<source>`` | MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/source>
- ``<picture>`` | MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/picture>
- Images adaptatives (`srcset`, `sizes`) | MDN : https://developer.mozilla.org/fr/docs/Web/HTML/Guides/Responsive_images
- ``<iframe>`` | MDN : <https://developer.mozilla.org/fr/docs/Web/HTML/Reference/Elements/iframe>

Optimisation, classement et nomenclature des médias

- Learn Images | web.dev (Google) | cours complet sur l'image sur le Web : <https://web.dev/learn/images/>
- Chargement différé (lazy loading) | MDN : https://developer.mozilla.org/en-US/docs/Web/Performance/Guides/Lazy_loading
- Squoosh (Google) | conversion et compression locales : <https://squoosh.app/>
- TinyPNG | compression par lots : <https://tinypng.com/>
- FFmpeg | Documentation : <https://ffmpeg.org/documentation.html>
- FFmpeg | Wiki d'encodage H.264 : <https://trac.ffmpeg.org/wiki/Encode/H.264>

- FFmpeg | Wiki d'encodage AAC : <https://trac.ffmpeg.org/wiki/Encode/AAC>
- HandBrake | Documentation | équivalent graphique de FFmpeg : <https://handbrake.fr/docs/>
- Gérer les fichiers | MDN | arborescence et conventions de nommage : https://developer.mozilla.org/fr/docs/Learn_web_development/Getting_started/Environment_setup/Dealing_with_files

Contrôle programmatique des médias

- `HTMLMediaElement` | MDN | `play()`, `pause()`, `currentTime`, `volume` : <https://developer.mozilla.org/fr/docs/Web/API/HTMLMediaElement>
- Audio and video delivery | MDN : https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Audio_and_video_delivery
- Créer un lecteur vidéo multinavigateur | MDN : https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Audio_and_video_delivery/cross_browser_video_player
- Politiques de lecture automatique | MDN : <https://developer.mozilla.org/en-US/docs/Web/Media/Guides/Autoplay>
- Howler.js | site officiel : <https://howlerjs.com/>
- Howler.js | Documentation de l'API : <https://github.com/goldfire/howler.js#documentation>
- Web Audio API | MDN : https://developer.mozilla.org/fr/docs/Web/API/Web_Audio_API

Animation | GSAP (Depuis 2025, GSAP est entièrement gratuit, plugins bonus compris (ScrollTrigger, MotionPath, DrawSVG). Les tutoriels qui parlent d'un « Club GreenSock » payant sont périmés.)

- GSAP | Docs & Learning : <https://gsap.com/docs/v3/>
- GSAP | Installation : <https://gsap.com/docs/v3/Installation>
- GSAP | Your first animation : <https://gsap.com/resources/get-started>
- GSAP | Tween (`to`, `from`, `fromTo`) : <https://gsap.com/docs/v3/GSAP/Tween>
- GSAP | Timeline : <https://gsap.com/docs/v3/GSAP/Timeline>
- GSAP | Le paramètre de position : <https://gsap.com/resources/position-parameter>

- GSAP | Easing : <https://gsap.com/resources/getting-started/Easing>
- GSAP | Visualiseur d'ease : <https://gsap.com/docs/v3/Eases>
- GSAP | Staggers : <https://gsap.com/resources/getting-started/Staggers>
- GSAP | Control and Callbacks : <https://gsap.com/resources/getting-started/control>
- GSAP | Keyframes : <https://gsap.com/resources/keyframes>
- GSAP | Cheat sheet : <https://gsap.com/cheatsheet>
- GSAP | Common GSAP mistakes : <https://gsap.com/resources/mistakes>
- GSAP | Notes de version 3.13 (gratuité) : <https://gsap.com/blog/3-13/>
- ScrollTrigger : <https://gsap.com/docs/v3/Plugins/ScrollTrigger/>
- ScrollTrigger tips & mistakes : <https://gsap.com/resources/st-mistakes>
- Démonstrations ScrollTrigger : <https://demos.gsap.com/?category=Scroll>
- `gsap.matchMedia()` | animation responsive : [https://gsap.com/docs/v3/GSAP/gsap.matchMedia\(\)](https://gsap.com/docs/v3/GSAP/gsap.matchMedia())
- Plugins GSAP (index) : <https://gsap.com/docs/v3/Plugins>
- MotionPathPlugin : <https://gsap.com/docs/v3/Plugins/MotionPathPlugin>
- MotionPathHelper | éditeur de tracé dans le navigateur : <https://gsap.com/docs/v3/Plugins/MotionPathHelper>
- DrawSVGPlugin : <https://gsap.com/docs/v3/Plugins/DrawSVGPlugin>
- `SteppedEase` | animation de sprite image par image : <https://gsap.com/docs/v3/Eases/SteppedEase>
- SVG | MDN : <https://developer.mozilla.org/fr/docs/Web/SVG>
- `background-position` | MDN | l'autre moitié de la technique du sprite : <https://developer.mozilla.org/fr/docs/Web/CSS/background-position>
- `prefers-reduced-motion` | MDN : <https://developer.mozilla.org/fr/docs/Web/CSS/Reference/At-rules/@media/prefers-reduced-motion>

Autres librairies

- Chart.js | Documentation : <https://www.chartjs.org/docs/latest/>

- Chart.js | Getting Started : <https://www.chartjs.org/docs/latest/getting-started/>
- Chart.js | Samples : <https://www.chartjs.org/docs/latest/samples/>
- Canvas API | MDN : https://developer.mozilla.org/fr/docs/Web/API/Canvas_API
- Tone.js | site officiel : <https://tonejs.github.io/>
- Tone.js | Documentation de l'API : <https://tonejs.github.io/docs/>
- Tone.js | Exemples : <https://tonejs.github.io/examples/>
- Three.js | Documentation : <https://threejs.org/docs/>
- Three.js | Manuel (disponible en français) : <https://threejs.org/manual/>
- Three.js | Exemples : <https://threejs.org/examples/>

Formulaire de contact | EmailJS

- EmailJS | Documentation : <https://www.emailjs.com/docs/>
- EmailJS | Installation du SDK : <https://www.emailjs.com/docs/sdk/installation/>
- EmailJS | Créer un formulaire de contact : <https://www.emailjs.com/docs/tutorial/creating-contact-form/>
- Validation des données de formulaires | MDN :
https://developer.mozilla.org/fr/docs/Learn_web_development/Extensions/Forms/Form_validation
- Envoyer et extraire les données des formulaires | MDN :
https://developer.mozilla.org/fr/docs/Learn_web_development/Extensions/Forms/Sending_and_retrieving_form_data

Médias libres de droits

- Pexels | photos et vidéos : <https://www.pexels.com/fr-fr/>
- Unsplash | photos : <https://unsplash.com/fr>
- Pixabay | photos, vidéos, illustrations, musique : <https://pixabay.com/fr/>
- Freesound | effets sonores, licences CC variées à vérifier fichier par fichier : <https://freesound.org/>

- Free Music Archive | musique : <https://freemusicarchive.org/>
- Kenney | ressources de jeu en domaine public (CC0) : <https://kenney.nl/assets>
- OpenGameArt | ressources de jeu, licences variées : <https://opengameart.org/>
- Google Fonts | polices libres : <https://fonts.google.com/>
- Font Awesome | icônes : <https://fontawesome.com/icons>
- Heroicons | icônes : <https://heroicons.com/>

Licences, droit d'auteur et crédits

- Les licences Creative Commons | les six licences et ce qu'elles autorisent : <https://creativecommons.org/share-your-work/cclicenses/>
- Openverse (WordPress Foundation) | recherche d'œuvres sous licence libre : <https://openverse.org/>
- Le droit d'auteur | Gouvernement du Canada : <https://ised-isde.canada.ca/site/office-propriete-intellectuelle-canada/fr/droits-dauteur>
- Utilisation équitable | Université de Montréal : <https://bib.umontreal.ca/gerer-diffuser/droit-auteur/utilisation-equitable>

Accessibilité et expérience utilisateur

- Web Accessibility Initiative (W3C-WAI) : <https://www.w3.org/WAI/>
- Comment satisfaire aux WCAG 2.2 (Quickref) : <https://www.w3.org/WAI/WCAG22/quickref/>
- Accessibilité | MDN : <https://developer.mozilla.org/fr/docs/Web/Accessibility>
- WebAIM | articles et listes de vérification : <https://webaim.org/>
- WebAIM Contrast Checker | contraste des couleurs d'un thème : <https://webaim.org/resources/contrastchecker/>
- Les 10 heuristiques d'utilisabilité | Nielsen Norman Group : <https://www.nngroup.com/articles/ten-usability-heuristics/>
- Responsive design | MDN : https://developer.mozilla.org/fr/docs/Learn_web_development/Core/CSS_layout/Responsive_Design

Guides et bases de connaissances

- Guide pour les étudiants dans le programme TIM : [guide_etudiants.docx](#)
- [Carrefour de l'information étudiante](#)
- [Le métier étudiant, ça s'apprend!](#)
- [Écran : Base de connaissances | étudiantes et étudiants](#)

Règles d'évaluation des apprentissages

Tous les articles de la [Politique institutionnelle d'évaluation des apprentissages \(PIÉA\)](#) s'appliquent à ce cours.

Français, méthodologie et plagiat

[Article 5.4 Évaluation de la langue française](#)

[Article 6.1.2 Sanction pour manquement à l'intégrité intellectuelle](#)

L'étudiante ou l'étudiant doit garder une copie de sécurité de ses travaux tant qu'elle ou il n'a pas reçu le corrigé de son travail.

L'étudiante ou l'étudiant doit aussi conserver tous les fichiers sources et ses originaux qui lui ont permis d'effectuer son travail. Si l'étudiante ou l'étudiant n'a pas les fichiers sources entre les mains, le professeur se réserve le droit d'attribuer la note zéro pour le travail concerné.

Comme les fichiers sources des travaux remis par la population étudiante doivent pouvoir être consultés par le corps professoral, les professeurs peuvent refuser tout travail réalisé à partir de bibliothèques, logiciels ou plugiciels piratés.

Le matériel visuel, sonore ou textuel créé par une IA est considéré comme ayant été « créé par l'IA » et dans ce sens doit être cité correctement en mentionnant le nom et la version de l'IA. Ne pas le mentionner constitue du plagiat. Il faut aussi mentionner textuellement la requête utilisée pour générer le contenu.

Climat en classe, éthique et travail en équipe

[Article 6.2.1 Manquement à la sécurité](#)

[Article 6.3.1 Manquement à l'éthique](#)

Le travail et l'esprit d'équipe est un pilier important de la création en multimédia. Lors des périodes d'atelier, les étudiants sont invités à se comporter comme s'ils étaient dans le milieu professionnel.

- Lors des périodes d'atelier, les étudiants sont invités à discuter entre eux tout en maintenant un niveau de voix normal pour échanger sur la matière et développer des relations. Pour aider à la concentration pendant les périodes d'atelier, les étudiants sont invités à utiliser des écouteurs.
- Aucune forme d'agressivité, d'intimidation ou de violence ne sera tolérée en classe et résultera au minimum en une expulsion. Si des étudiants ne se sentent pas confortables en classe, ils sont invités à en discuter avec le professeur et à sortir de classe pour prendre un moment pour eux.

Seuls les logiciels et les équipements enseignés peuvent être utilisés dans le cadre du cours. Les téléphones cellulaires et appareils mobiles sont interdits en classe virtuelle et en présentiel et ne font pas partie du matériel enseigné en cours, à moins d'indications contraires de la part du professeur. Lors du non-respect de ces règles, l'étudiante ou l'étudiant est considéré absent et l'enseignant peut lui demander de quitter la classe.

Lors d'un refus d'assumer sa part de travail pendant un travail d'équipe noté, l'étudiante ou l'étudiant peut être pénalisé par la professeure ou le professeur.

Présence et évaluations

[Article 7.1 Présence en classe](#)

[Article 7.2 Absence à une évaluation sommative](#)

[Article 7.3 Absence à un stage](#)

[Article 7.4 Retard](#)

[Article 7.4.1 Retard à une évaluation sommative](#)

[Article 7.4.2 Retard dans la remise des activités d'évaluation](#)

Les présences aux remises de chacune des évaluations sont obligatoires et essentielles à la réussite des activités sommatives. Une absence lors d'une évaluation entraînera la note 0, et ce même si le travail a été remis en ligne ou que le travail a été présenté par le reste de l'équipe.

La présence en classe lors des remises, des évaluations et des présentations des travaux est très importante pour bénéficier des commentaires du professeur, afin de réussir la session.

Pour les **stages**, les 225 heures de présence sont nécessaires pour la réussite du stage. Si vous devez vous absenter, vous devrez avertir votre employeur et voir comment reprendre les heures. Les jours de congés permis sont ceux de l'employeur et non ceux du collège.

Corrections

[Article 8.1 Délais de correction des activités d'évaluation](#)

Les étudiantes et étudiants peuvent rencontrer le professeur pour discuter des forces et faiblesses d'un travail corrigé 48 heures après sa remise.

La professeure ou le professeur peut être joint par courriel ou par Teams. **Teams est privilégié et plus rapide.** Comptez un délai de deux journées ouvrables pour obtenir un retour, sauf lors de cas exceptionnels.