



Collège de  
**Maisonneuve**

# **ATTESTATION D'ETUDES COLLÉGIALE**

## **CONCEPTION ET PROGRAMMATION DE SITES WEB (NWE.0F)**

**Cadriciel Web  
582-41B-MA**

**1 – TP 1**

**Créer un site web dynamique en utilisant le cadriciel Laravel**

**20%**

**Professeur:  
Marcos Sanches  
msanches@cmaisonneuve.qc.ca**

**COLLÈGE DE MAISONNEUVE  
2030, boul. Pie-IX, bureau 430  
Montréal (Québec) H1V 2C8  
Téléphone : 514 254-7131**



Félicitations, vous avez reçu un nouveau mandat et vous êtes prêt à mettre en pratique toutes les compétences acquises lors du cours de Cadriciel Web.

Ce mandat implique la création d'un site Internet visant à collecter des informations auprès des étudiants du Collège Maisonneuve. De plus, il est envisagé de développer un réseau social pour les étudiants à l'avenir.

La première étape consiste à collecter les informations nécessaires, à remplir la base de données avec des données aléatoires, et à créer des interfaces fonctionnelles permettant de visualiser, saisir, mettre à jour et supprimer les informations des étudiants.

La base de données initiale comprendra deux tables : "Étudiant" (id, nom, adresse, téléphone, email, date de naissance, ville\_id) et "Ville" (id, nom).

### **Étapes du Projet :**

#### **1. Création du Projet Laravel (1 pt)**

- Utilisez les lignes de commande pour créer un nouveau projet Laravel nommé "Maisonneuve{votre matricule}".

#### **2. Création des Modèles**

- Utilisez les lignes de commande Laravel pour créer les modèles nécessaires.

#### **3. Création des Tables**

- Utilisez les lignes de commande Laravel pour créer les tables de la base de données.

#### **4. Saisie des Villes**

- À l'aide des lignes de commande Laravel, ajoutez 15 nouvelles villes à la base de données.

#### **5. Saisie des Étudiants**

- Utilisez les lignes de commande Laravel pour ajouter 100 nouveaux étudiants à la base de données en utilisant des propriétés de "Factory" pour remplir les valeurs.

#### **6. Création des Contrôleurs**

- Utilisez les lignes de commande Laravel pour créer les contrôleurs nécessaires.

#### **7. Création du Layout**

- Créez un fichier `layout.blade.php` avec votre propre CSS et importez Bootstrap (ou un CSS personnalisé). Concevez le layout selon vos préférences.

## 8. Conception Ergonomique

- Utilisez Bootstrap (ou un CSS personnalisé) pour respecter les principes d'ergonomie et soyez créatif dans la conception de l'interface utilisateur.

## 9. Affichage de la Liste des Étudiants

- Utilisez la méthode "index" avec une vue pour afficher tous les étudiants, avec un lien permettant de sélectionner un étudiant et de le mettre à jour.

## 10. Création d'un Nouvel Étudiant

- Utilisez la méthode "create" avec une vue pour saisir un nouvel étudiant. Le formulaire doit inclure un champ "select" avec toutes les villes disponibles dans la base de données.
- Valider les données

## 11. Affichage d'un Étudiant Sélectionné

- Utilisez la méthode "show" avec une vue pour afficher les détails d'un étudiant sélectionné.

## 12. Mise à Jour d'un Étudiant

- Utilisez la méthode "edit" avec une vue pour afficher un formulaire pré-rempli avec les détails d'un étudiant sélectionné et permettre sa mise à jour.
- Valider les données.

## 13. Suppression d'un Étudiant

- Utilisez la méthode "destroy" pour supprimer un étudiant sélectionné.

## 14. Publication sur GitHub

- Publiez votre projet sur GitHub en le rendant public et envoyez le lien dans la documentation.

## 15. Envoi du Projet

- Enregistrez le projet avec une extension ZIP et ajoutez la documentation dans la racine. Déposez le fichier ZIP sur Léa - Omnivox.

## Documentation des Commandes Utilisées:

- Inclure une documentation détaillée des lignes de commande utilisées tout au long du projet, notamment pour la création du projet, des modèles, des contrôleurs, des tables, l'ajout de données, etc.

## Liens d'Accès:

Fournir le lien d'accès à votre site sur webdev et/ou tout autre serveur utilisé pour le déploiement du projet, ainsi que le lien GitHub (publique) du projet.

### **Importance de l'Information Manquante:**

Souligner que l'absence de ces informations dans la documentation et/ou la non-publication du projet (fonctionnel) sur WebDev ou tout autre serveur sera considérée comme un projet inachevé, entraînant une diminution de 5 points sur 20 de la note finale.

### **Support et Doutes:**

Préciser qu'une journée d'atelier sera dédiée aux doutes concernant le TP. Toutes les questions relatives aux travaux pratiques doivent être posées pendant les heures de cours ou dans le cours d'atelier. Aucun débogage ne sera réalisé par Mio.

Vous devez fournir les lignes de commandes utilisées pendant le projet, le lien vers votre site sur webdev et/ou serveur et GitHub (publique) dans la documentation. Assurez-vous de respecter les délais de soumission pour éviter toute pénalité.

### **Pénalité pour retard.**

- Jusqu'à 24 heures : réduction de 1 point sur la note finale sur 20.
- Entre 24 et 48 heures : réduction de 2 points sur la note finale sur 20.
- Après 48 heures de retard, le projet ne serait pas accepté et la note zéro serait accordée.

**Critères d'évaluation :**

Item	Points
Création du Projet Laravel	0.5
Création des Modèles	0.5
Création des Tables	2
Saisie des Villes	1
Saisie des Étudiants	1
Création des Contrôleurs	1
Création du Layout	1
Conception Ergonomique	1
Affichage de la Liste des Étudiants	2
Création d'un Nouvel Étudiant	2
Affichage d'un Étudiant Sélectionné	2
Mise à Jour d'un Étudiant	2
Suppression d'un Étudiant	1
Publication sur GitHub - Publique	1
Qualité et complexité du projet pour répondre aux exigences du projet.	2
<b>Total</b>	<b>20</b>

**Critères pour chaque point attribué :**

1. Création du Projet Laravel :
  - Excellent : Le projet Laravel est créé avec succès, en respectant les normes et les conventions de Laravel.
  - Réussit : Le projet Laravel est créé mais quelques normes et conventions de Laravel ne sont pas respectées.
  - Pas Réussit : Le projet Laravel n'est pas créé ou ne correspond pas aux normes et conventions de Laravel.
2. Création des Modèles :
  - Excellent : Les modèles sont créés avec précision et en respectant les bonnes pratiques.
  - Réussit : Les modèles sont créés mais présentent quelques erreurs mineures.
  - Pas Réussit : Les modèles ne sont pas correctement créés ou comportent des erreurs majeures.
3. Création des Tables :
  - Excellent : Les tables de la base de données sont correctement créées selon les spécifications.

- Réussit : Les tables sont créées mais quelques spécifications ne sont pas entièrement respectées.
  - Pas Réussit : Les tables ne sont pas correctement créées ou ne respectent pas les spécifications requises.
4. Saisie des Villes :
- Excellent : Les données des villes sont saisies avec précision et conformément aux attentes.
  - Réussit : Les données des villes sont saisies mais quelques informations peuvent manquer ou être incorrectes.
  - Pas Réussit : Les données des villes ne sont pas correctement saisies ou ne sont pas conformes aux attentes.
5. Saisie des Étudiants :
- Excellent : Les données des étudiants sont saisies avec précision et conformément aux attentes.
  - Réussit : Les données des étudiants sont saisies mais quelques informations peuvent manquer ou être incorrectes.
  - Pas Réussit : Les données des étudiants ne sont pas correctement saisies ou ne sont pas conformes aux attentes.
6. Création des Contrôleurs :
- Excellent : Les contrôleurs sont créés selon les bonnes pratiques et en respectant les spécifications.
  - Réussit : Les contrôleurs sont créés mais quelques spécifications ne sont pas entièrement respectées.
  - Pas Réussit : Les contrôleurs ne sont pas correctement créés ou ne respectent pas les spécifications requises.
7. Création du Layout :
- Excellent : Le layout est conçu de manière esthétique et ergonomique, en utilisant des techniques avancées de design.
  - Réussit : Le layout est conçu mais peut présenter quelques lacunes en termes d'esthétique ou d'ergonomie.
  - Pas Réussit : Le layout n'est pas correctement conçu ou ne répond pas aux critères esthétiques et ergonomiques.