

Sigle : INF1743 Gr. 20**Titre : Développement d'applications web****Session : Hiver 2026 Horaire et local****Professeur : Baratli, Karim****1. Description du cours paraissant à l'annuaire :****Objectifs**

Au terme de ce cours, l'étudiant.e sera en mesure de concevoir, programmer et déployer une application web complète comprenant client, serveur et stockage de données.

Contenu

Méthodologies de développement d'applications et de sites web utilisant une pile Full-Stack. Application du modèle MVC (Modèle-Vue-Contrôleur). Développement en parallèle des parties frontale et dorsale. Conception d'interfaces RESTFul. Interactivité et réactivité d'applications et de sites web adossés à des bases de données. Plateformes de développement avec pile MEAN, MERN, etc. Déploiement en mode production. Ce cours comporte des séances obligatoires de travaux dirigés (TD).

Descriptif – Annuaire**2. Objectifs spécifiques du cours :**

- Le premier objectif de ce cours est de permettre à l'étudiant de comprendre et maîtriser les principes fondamentaux du développement d'applications web modernes en mode Full-Stack.
- Être capable de concevoir, programmer et intégrer les trois couches essentielles d'une application web complète : client (front-end), serveur (back-end) et stockage de données (base de données).
- Appliquer le modèle MVC (Modèle-Vue-Contrôleur) dans la construction d'une application web structurée, modulable et maintenable.
- Développer en parallèle la partie frontale et la partie dorsale d'une application en utilisant une pile technologique contemporaine basée sur React (client) et Django/Django REST Framework (serveur).
- Concevoir et implémenter des interfaces RESTFul, incluant la définition des routes, la sérialisation des données, la validation, l'authentification et la gestion des erreurs.
- Développer des interfaces utilisateur interactives, réactives et dynamiques, adossées à une base de données relationnelle et alimentées par des appels API.
- Être capable de déployer une application web complète en mode « production », incluant la préparation des environnements, la configuration des variables d'environnement, le build du front-end et la mise en ligne du serveur.
- Mettre en pratique l'ensemble des apprentissages au travers d'un projet intégrateur couvrant l'analyse, la conception, l'implémentation, les tests, l'intégration et le déploiement d'une application web complète.

3. Stratégies pédagogiques :

Au cours de cette activité, diverses formules pédagogiques seront utilisées, notamment : cours magistraux, exemples et lectures personnelles. Les attentes sont que les étudiants et les étudiantes investissent au moins 90 heures de travail personnel en plus des 45 heures de cours.

Sept séances de travaux dirigés (2 h/semaine) viendront appuyer les séances de cours. Ces séances se font en présence d'un assistant. La présence des étudiants et les étudiantes à ces laboratoires est fortement recommandée.

Mode d'enseignement au trimestre d'hiver 2026 : **PRÉSENTIEL**.

4. Heures de disponibilité ou modalités :

Rendez-vous sur Zoom ou par courriel : karim.baratli@ugo.ca

5. Plan détaillé du cours sur 15 semaines :

Semaine	Thèmes	Dates
1	Introduction au cours & Architecture Web (Non présentiel) <ul style="list-style-type: none"> Présentation du cours Vue d'ensemble des principales piles Full-Stack (MERN, MEAN, Django, etc.) Justification du choix du stack du cours Présentation du stack officiel : React + Django REST + BD SQL Structure d'une application Web (client, API, BD) Présentation du projet intégrateur 	13 janv. 2026
2	React — Composants et état (Non présentiel) <ul style="list-style-type: none"> JSX, composants, props Gestion de l'état (useState) Événements et formulaires contrôlés <p>Travail dirigé #1 (Jeudi 22 janvier): State, gestion d'événements, formulaires contrôlés.</p>	20 janv. 2026
3	React — Navigation et appels API <ul style="list-style-type: none"> React Router Appels API (fetch / axios) Gestion du chargement et des erreurs <p>Travail dirigé #2 (Jeudi 29 janvier): React Router + API publique</p>	27 janv. 2026
4	Django REST — Modèles, ORM et API CRUD <ul style="list-style-type: none"> Structure d'un projet Django Models, migrations Serializers et Views (CRUD) Endpoints REST <p>Travail dirigé #3 (Jeudi 5 février): Django REST + CRUD + BD</p>	3 fév. 2026
5	Authentification et sécurité API <ul style="list-style-type: none"> Authentification JWT Endpoints protégés Permissions dans DRF Sécurité : validation, injections, CORS <p>Travail dirigé #4 (Jeudi 13 février): Auth + JWT + permissions</p>	10 fév. 2026
6	Intégration React - Django <ul style="list-style-type: none"> Appels API sécurisés depuis React Gestion du token Formulaire connecté et navigation protégée <p>Travail dirigé #5 (Jeudi 19 février): Intégration React - Django</p>	17 fév. 2026
7	Sécurité Web avancée (Non présentiel) <ul style="list-style-type: none"> XSS, CSRF Middleware et validation serveur Bonnes pratiques de sécurité 	24 fév. 2026
8	Semaine de relâche	03 mars 2026

9	Examen intra	10 mars 2026
10	Déploiement Full-Stack (React + Django) <ul style="list-style-type: none"> • Build React • Déploiement Django • Variables d'environnement • BD distante Travail dirigé #6 (Jeudi 19 mars) : Exercices de déploiement (React + Django)	17 mars 2026
11	Tests front-end et back-end <ul style="list-style-type: none"> • Tests Django (unit tests, API tests) • Tests React (Jest, RTL) 	24 mars 2026
12	Optimisation et performance <ul style="list-style-type: none"> • Optimisation React (memo, lazy) • Optimisation Django (select_related, pagination) • Audit performance Travail dirigé #7 (Jeudi 2 avril) : Audit performance front/back	31 mars 2026
13	Finalisation du projet intégrateur <ul style="list-style-type: none"> • Vérifications API et front • Préparation au déploiement final 	7 avr. 2026
14	Révision <ul style="list-style-type: none"> • Exercices de pratiques • Préparation à l'examen final 	14 avr. 2026
15	Examen Final	21 avr. 2026

6. Évaluation du cours :

L'évaluation est l'appréciation du niveau d'apprentissage atteint par l'étudiant(e) par rapport aux objectifs des cours et des programmes

L'attribution des notes se fera selon la répartition suivante :

- Examen intra : 30 %
- Examen final : 40 %
- Projet : 30 %

Pour le projet d'équipe, les notes peuvent être attribuées d'une manière individuelle selon la contribution de l'étudiant(e).

Attention : La présence aux cours est fortement recommandée.

7. Politiques départementales et institutionnelles :

- Politiques relatives à la tenue des examens
- Note sur le plagiat et les fraudes
- Politique relative à la qualité de l'expression française écrite chez les étudiants et les étudiantes de premier cycle à l'UQO
- Absence aux examens : cadre de gestion, demande de reprise d'examen (formulaire)

Tolérance **ZÉRO** en matière de violence à caractère sexuel.

Le Bureau d'intervention et de prévention en matière de harcèlement (BIPH) a pour mission d'accueillir, soutenir et guider toute personne vivant une situation de harcèlement, de discrimination ou de violence à caractère sexuel. Le BIHP oriente ses actions afin de prévenir les violences à caractère sexuel pour que nous puissions étudier, travailler et s'épanouir dans un milieu sain et sécuritaire.

Vous vivez ou êtes une personne témoin d'une situation de violence à caractère sexuel ? Vous êtes une personne membre de la communauté étudiante ou une personne membre du personnel, autant à Gatineau qu'à Ripon et St-Jérôme, l'équipe du BIHP est là pour vous, sans jugement et en toute confidentialité.

Ensemble, participons à une culture de respect.

Pour de plus amples renseignements consultez UQO.ca/biph ou écrivez-nous au Biph@uqo.ca

8. Principales références :

Volume de référence :

- William S. Vincent. *Django for APIs: Build Web APIs with Python & Django*. 4th Edition, 2022.

Ouvrages suggérés :

- Alex Banks, Eve Porcello. *Learning React*, O'Reilly, 2020.
- William S. Vincent. *Django for Beginners*, 2023.

9. Page Web du cours :

<https://moodle.uqo.ca>