

Pourquoi cette UE ?



Éléments constitutifs de l'UE

	coefficient	
INFRES_9_2_DL-1 Frontend	2	
INFRES_9_2_DL-2 Environnement ASP .Net Microsoft	3	
INFRES_9_2_DL-3 Intelligence artificielle : apprentissage profond	4	
Volume d'heures d'enseignement encadré	Volume d'heures de travail personnel	Nombre d'ECTS
101	37	5

Alignement curriculaire

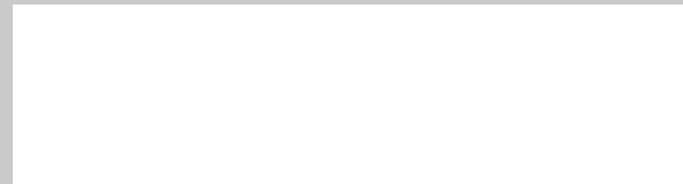
Parmi les compétences visées par la formation, lesquelles sont développées dans cette UE ?



Contexte et enjeux de l'enseignement

Ce cours initie les étudiants aux fondamentaux du développement frontend moderne, en mettant l'accent sur la bibliothèque React. Ils apprendront à construire des interfaces utilisateur dynamiques et interactives, à gérer l'état de vos applications, et à maîtriser les concepts clés de React.

Prise en compte des dimensions socio-environnementales



Prérequis

Maîtriser un langage de programmation

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	10
Cours intégré (cours + TD)	
TD	
TP	10
Projets	
Travail en autonomie encadré	
Contrôles et soutenances	
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

- Comprendre les principes fondamentaux du développement frontend.
- Maîtriser les bases de JavaScript ES6+.
- Se familiariser avec l'écosystème React et ses outils.
- Développer des applications web interactives et performantes avec React.
- Apprendre à gérer l'état et les données dans une application React.
- Comprendre les principes de base du routage et de la gestion des formulaires avec React.
- Découvrir les bonnes pratiques pour le développement frontend.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Cours et tp illustrant la conception d'un frontend moderne.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

TP ou écrit suivant le contexte.

INFRES_9_2_DL Développement

INFRES

INFRES_9_2_DL-1 Frontend

S9

Plan de cours

- Introduction au développement frontend
- Les langages de Javascript et typescript
- Les bases de React
- Concepts avancés de React

Ressources et références

Cours sur Campus.

Contexte et enjeux de l'enseignement

Présentation des technologies Microsoft .NET (ASP.NET MVC, Windows phone, XAML). • Découverte du framework .NET • Créer un projet ASP.NET MVC et Windows phone Maîtriser les patterns MVC et MVVM

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Programmation orientée objet

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	17
Cours intégré (cours + TD)	
TD	
TP	17
Projets	
Travail en autonomie encadré	
Contrôles et soutenances	2
Travail personnel	7

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

- S'adapter à plusieurs technologies
- Organiser son code avec différents patterns

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Cours et travaux pratiques en alternance.
2 projets fil-rouge pour appliquer chaque notion.

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Évaluation :
Contrôle et TP

Retour sur l'évaluation fait à l'élève :
Copies corrigées et consultables sur demande

Plan de cours

- La programmation orientée objet en c#
- Découverte d'ASP.NET + Architecture MVC
- Découverte XAML avec windows phone + Architecture MVVM

Ressources et références

Les supports pédagogiques sont disponibles en ligne sous Campus.

Contexte et enjeux de l'enseignement

Cette unité d'enseignement initie les étudiants aux fondements théoriques et aux pratiques avancées de l'apprentissage profond, un pilier de l'intelligence artificielle contemporaine. Elle couvre les architectures clés telles que les MLP, CNN, RNN et Transformers, tout en introduisant les techniques modernes d'entraînement, de régularisation et de spécialisation (fine-tuning) des modèles. Une attention particulière est portée à la compréhension fine des mécanismes d'apprentissage des réseaux de neurones, notamment la rétropropagation, le calcul des gradients, les fonctions de perte et les algorithmes d'optimisation. Les étudiants apprendront à utiliser PyTorch, l'une des bibliothèques les plus répandues en recherche et en industrie, pour concevoir, entraîner et évaluer des modèles sur des jeux de données réels (images, texte, séries temporelles). L'ECUE est structurée autour de cinq axes progressifs : fondements théoriques, architectures, entraînement de modèles, transfert et adaptation, et mise en œuvre pratique.

Prise en compte des dimensions socio-environnementales

ODD9 - Industrie, innovation et infrastructure

Prérequis

Bases en Python, Mathématiques pour l'IA : algèbre linéaire, dérivées partielles, probabilités Notions fondamentales d'apprentissage automatique (régression, classification, surapprentissage)

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	14
Cours intégré (cours + TD)	
TD	7
TP	14
Projets	7
Travail en autonomie encadré	
Contrôles et soutenances	3
Travail personnel	30

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

- Comprendre en profondeur les mécanismes fondamentaux des réseaux de neurones.
- Maîtriser les techniques modernes d'entraînement et les techniques d'optimisation des paramètres de modèle.
- Décrire et expliquer les architectures fondamentales de réseaux de neurones (MLP, CNN, RNN, Transformers).
- Implémenter, entraîner et évaluer des modèles avec PyTorch.
- Appliquer une démarche d'ingénieur rigoureuse et responsable.

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

Cours magistraux. Travaux dirigés, Travaux pratiques, Projet

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Evaluation individuelle (50%)
Projet en groupe + TP (50%)

Plan de cours

- Introduction aux concepts de base en deep learning: Descente de gradient, fonction de perte, Graphes computationnels
- Réseau de neurones artificiels: Neurone formel et architecture du perceptron multicouche (MLP), Fonctions d'activation , algorithme de rétropropagation
- Entraînement des réseaux: Régularisation (dropout, weight decay), Optimisation (SGD, Adam...), Recherche d'hyperparamètres, Techniques de normalisation (batch/layer norm)
- Implémentation avec PyTorch: Création, entraînement et évaluation de modèles, Gestion des données et du cycle d'entraînement
- Architectures spécialisées: Réseaux convolutifs (CNN), Réseaux récurrents (RNN, LSTM, GRU), Modèles basés sur les transformeurs (Transformers)
- Projet final

Ressources et références

Support de présentation