

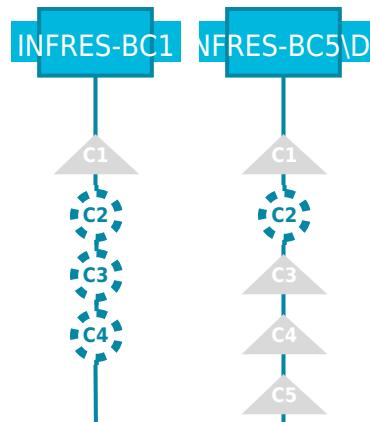
Pourquoi cette UE ?

Un module combinant un cours de frontend et un cours de programmation orientée objet permet de relier conception d'interfaces et développement structuré. Le frontend initie l'élève à la création d'interfaces claires et ergonomiques, directement perçues par l'utilisateur. Le langage orienté objet apporte une organisation du code favorisant modularité, réutilisation et maintenance. Leur complémentarité prépare l'élève à concevoir des applications complètes, à la fois fonctionnelles, robustes et adaptées aux besoins des utilisateurs.

Eléments constitutifs de l'UE

	coefficients	
INFRES_5_4-1 Frontend	1	
INFRES_5_4-2 Langage de développement orienté objet	1	
Volume d'heures d'enseignement encadré	Volume d'heures de travail personnel	Nombre d'ECTS
39	19	2

Alignement curriculaire

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

Contexte et enjeux de l'enseignement

JavaScript est un langage incontournable du développement web moderne. Ce cours vise à fournir aux étudiants les bases du langage afin de les amener à intégrer JavaScript dans un projet web concret. Cet enseignement aborde également des concepts avancés comme les closures, la notion de design pattern avec le modèle MVC et différentes notions algorithmiques.

Prise en compte des dimensions socio-environnementales**Modalités d'enseignement et d'évaluation**

	Nb d'heures
Cours	
Cours intégré (cours + TD)	15
TD	
TP	
Projets	3
Travail en autonomie encadré	
Contrôles et soutenances	
Travail personnel	10

Prérequis

Aucun

Objectifs pédagogiques

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

- Comprendre les fondamentaux de JavaScript (syntaxe, types de données, fonctions, événements).
- Manipuler le DOM et capter les interactions d'un utilisateur.
- Closure et manipulation de fonctions (fonctions en paramètre d'autres fonctions, retour de fonctions, etc.).
- Implémenter des concepts algorithmiques dans un projet web.
- Implémenter un modèle MVC à partir de fonctions de « binding ».

Activités

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

La première séance se concentre sur les fondamentaux de JavaScript. Cette séance est accompagnées d'exercices permettant aux élèves de commencer au mieux le projet de développement qu'ils devront réaliser le reste des autres séances. Les pages des projets sont consultables à cette adresse : https://github.com/PAJEAN/cours_javascript/tree/master/projects

Évaluations et retours faits aux élèves

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

L'évaluation repose sur la réalisation d'un projet de développement web intégrant HTML, CSS et JavaScript. Les critères d'évaluation incluent :

- Le respect du cahier des charges du projet.
- La qualité et la structuration du code.
- Une présentation orale du projet permettant aux étudiants d'expliquer leurs choix techniques et de démontrer le fonctionnement de leur application.

Lors des séances dédiées aux projets, chaque étudiant bénéficie d'un accompagnement personnalisé afin d'obtenir un résultat.

Plan de cours

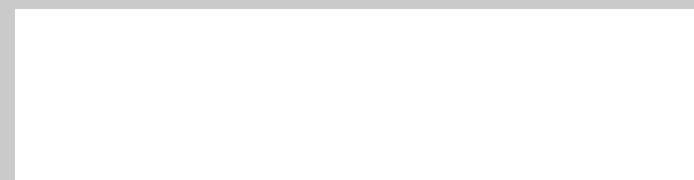
1. Introduction
1. JavaScript, c'est quoi ?
2. Principe client/serveur
2. Les bases du langage JavaScript
1. Hello World
2. Les variables
3. Les nombres
4. Les chaînes de caractères
5. Les fonctions
6. La manipulation de tableaux
7. Les Map
8. Les conditions
9. Les boucles
3. Les objets JavaScript
1. La création d'un objet
2. L'héritage et la chaîne de prototypage
3. La copie d'un objet
4. Le mot-clé this
4. L'interactivité
1. La programmation événementielle
2. La détection dans le code HTML
3. La détection par gestionnaire d'événements
4. La manipulation du document
5. Les appels AJAX
6. Le dessin et les canvas
5. Références

Ressources et références

https://github.com/PAJEUAN/cours_javascript

Contexte et enjeux de l'enseignement

Dans un contexte où les systèmes informatiques deviennent de plus en plus complexes, la programmation orientée objet constitue un paradigme incontournable. Elle offre un cadre conceptuel et pratique pour organiser, structurer et maintenir des programmes de grande taille. Ce cours permet aux élèves d'aborder les notions clés de modularité, d'abstraction et de réutilisation, indispensables pour concevoir des logiciels robustes et évolutifs. L'enjeu est de leur donner une compréhension solide des principes de l'objet afin de faciliter l'apprentissage ultérieur de divers langages modernes, mais aussi de les préparer à travailler en équipe sur des projets où clarté, évolutivité et qualité du code sont essentielles.

Prise en compte des dimensions socio-environnementales**Modalités d'enseignement et d'évaluation**

	Nb d'heures
Cours	5
Cours intégré (cours + TD)	
TD	
TP	15
Projets	
Travail en autonomie encadré	
Contrôles et soutenances	1
Travail personnel	9

Objectifs pédagogiques

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

Distinguer l'approche objet d'une approche procédurale. Reconnaître les concepts fondamentaux dans leurs mécanismes de modélisation et mise en œuvre

Activités

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

Cours magistral/TP.

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 : 1 écrit de 2 heures.
Retour sur l'évaluation fait à l'élève : Copies corrigées consultables sur demande

Plan de cours

- Type de base
- Structure de contrôle
- fonction
- type composé
- structure, méthode et interface
- serveur web.

Ressources et références

Cours et corrections fournit par l'enseignant.