



IDENTIFICATION	
Intitulé de l'Unité d'Enseignement : PROJET DE DÉVELOPPEMENT WEB	Niveau d'études : A – B – C - D
Intitulé du cours : PROJET DE DÉVELOPPEMENT : PROGRAMMATION CÔTÉ CLIENT ET PROGRAMMATION CÔTÉ SERVEUR	Nombre de crédits ECTS : 10
Nombre de périodes : 100 (50 → programmation côté client ; 50 → programmation côté serveur)	Code : 7534 30 U32 D1 UF : 1102

DESCRIPTION
<p>Capacités préalables requises</p> <p>1. En « Web principes de base » :</p> <p>Face à une structure informatique opérationnelle connectée à Internet, disposant des logiciels appropriés et de la documentation nécessaire, en utilisant le vocabulaire technique et l'orthographe adéquate, en respectant les normes et standards en vigueur, sur base d'un cahier des charges technique proposé par le chargé de cours, comprenant au moins l'intégration des feuilles de styles et le respect de la sémantique,</p> <ul style="list-style-type: none">• réaliser des pages statiques, compatibles avec au minimum un navigateur récent du marché (Internet Explorer, Safari...);• transférer et tester ces pages statiques sur un serveur. <p>2. Titre pouvant en tenir lieu :</p> <p>Attestation de réussite de l'unité de formation « Web principes de base » (UF 1101), n° de code 7534 29 U32 D1, classées dans l'enseignement supérieur économique de type court.</p>
<p>Documents de référence pour une préparation préalable au cours :</p> <p>Voir la bibliographie ci-dessous</p>

Objectifs :

Face au matériel et au logiciel adéquats et face à une structure informatique opérationnelle connectée à Internet, disposant des logiciels appropriés et de la documentation nécessaire, en utilisant le vocabulaire technique et l'orthographe adéquate, et en respectant les normes et standards en vigueur, l'étudiant sera capable :

En programmation côté client :

- d'identifier, dans une page web, les éléments impliquant l'usage d'un script client ;
- d'analyser un script client en termes de:
 - o définition des variables et des objets,
 - o structures conditionnelles et itératives,
 - o fonctions et de procédures,
 - o structures interactives (gestion des événements,...),
 - o etc. ;
- d'exploiter un script client dans une page web ;
- de modifier et de créer un script et de l'intégrer dans une page web ;
- de décrire et de caractériser objets, propriétés et méthodes ;
- de déterminer les événements auxquels les éléments de la page doivent réagir ;
- de traduire sous formes de commentaires, de schémas, de dessins, etc., les éléments nécessaires à la résolution d'un problème posé (structures procédurales, interactives, animations, objets, etc. ;
- de mettre en œuvre la résolution d'un problème posé au moyen du langage client choisi ;
- d'exploiter le côté orienté objet du langage choisi :
 - o les classes prédéfinies et leurs composants (window, document, cookie, etc.),
 - la définition de classes et leur instanciation,
 - o etc. ;
- d'utiliser, dans le langage choisi, les variables, les structures conditionnelles, les structures itératives, les tableaux, l'affichage dans une page web, etc. ;
- d'exploiter la notion d'expression régulière (validation de formulaires, etc.) ;
- d'exploiter des données structurées en XML (Extensible Markup Language), en JSON (JavaScript Object Notation), etc., contenues dans un fichier externe ;
- de décrire et de mettre en œuvre des technologies entrant dans le développement d'applications web dynamiques et animées tel que AJAX (Asynchronous Javascript and XML), etc. ;
- de choisir et d'exploiter une bibliothèque tierce telle que jQuery, MooTools, etc., en vue du développement de scripts spécifiques pour RIA (interfaces riches), transmissions asynchrones, etc. ;
- d'identifier des erreurs de programmation au moyen d'outils ou de techniques de
- débogage et d'y apporter une solution pertinente ;

En programmation côté serveur :

- d'identifier différents langages utilisés pour la programmation côté serveur ;
- d'installer les services nécessaires à l'exécution de scripts côté serveur ;
- d'analyser un script serveur en termes de:
 - o définition des variables,
 - o structures conditionnelles et itératives,
 - o fonctions et de procédures,
 - o etc. ;

- d'exploiter le modèle MVC (Model View Controller) ;
- d'exploiter un script serveur dans une page web ;
- d'utiliser, dans le langage choisi, les variables de programmation et d'environnement (session, application, cookies, etc.) ;
- de transférer des données entre pages et scripts (méthodes GET et POST, etc.) ;
- de traduire sous formes de commentaires, de schémas, de dessins, etc. les éléments nécessaires à la mise en œuvre d'une application dynamique (structure procédurale, transfert et conservation des données, interaction avec des données externes, etc.) ;
- de mettre en œuvre une application dynamique ;
- d'exploiter le système de gestion de fichiers du serveur (se déplacer dans l'arborescence, créer et modifier un fichier, créer un dossier et en gérer les droits d'accès, etc.) ;
- d'identifier les principales failles de sécurité d'une application dynamique ;
- d'identifier des erreurs de programmation au moyen d'outils ou de techniques de débogage et d'y apporter une solution pertinente.

Capacités terminales :

Pour atteindre le seuil de réussite, l'étudiant sera capable :

au départ d'un cahier des charges proposé par le chargé de cours, face au matériel et au logiciel adéquat et face à une structure informatique opérationnelle connectée à Internet, disposant des logiciels appropriés et de la documentation nécessaire, en utilisant le vocabulaire technique et l'orthographe adéquate, et en respectant les normes et standards en vigueur,

- de créer et d'exploiter des scripts clients basés sur des classes prédéfinies ;
- de créer et d'exploiter ses propres classes ;
- de créer et d'exploiter des scripts basés sur une bibliothèque tierce ;
- d'envoyer des informations venant du client vers le serveur et de les traiter ;
- de générer un ensemble de pages web contenant un système de navigation et un contenu dynamiques intégrant formulaires et résultats.

Pour la détermination du degré de maîtrise, il sera tenu compte des critères suivants:

- les techniques de programmation utilisées,
- le degré de pertinence des commentaires dans le code,
- la lisibilité du code,
- le niveau de fiabilité des tests,
- le degré d'autonomie atteint.

Contenu du cours :

Programmation côté client :

Cette partie du cours développe les notions spécifiques au développement de projet web au moyen de technologies s'exécutant dans le navigateur.

I. Théorie

1 Introduction et notions de syntaxe JavaScript

- 1.1 Distinction front-end / back-end
- 1.2 Introduction au Javascript
- 1.3 Présentation des outils
- 1.4 Syntaxe Javascript
- 1.5 Types de données
- 1.6 Introduction aux variables, fonctions et portée
- 1.7 Expressions, opérateurs
- 1.8 Instructions, conditions, boucles

2 Etude approfondie des fonctions

- 2.1 Fonctions pures et impures
- 2.2 Portée d'une fonction
- 2.3 Fonctions pré-définies

3 Programmation orientée objet et prototypes

- 3.1 Introduction et terminologie classique de la POO
- 3.2 Programmation orientée prototype
- 3.3 La programmation orientée objet avec JavaScript
- 3.4 Objets essentiels et objets sur mesure
- 3.5 Classes, objets, propriétés (ou attributs)
- 3.6 Constructeurs, méthodes
- 3.7 Pourquoi utiliser les prototypes
- 3.8 Héritage, encapsulation, abstraction & polymorphisme

4 Dom, XML, Lecture de fichiers

- 4.1 DOM : Document Object Model
- 4.2 Structure habituelle d'une hiérarchie DOM
- 4.3 Types de nœuds
- 4.4 Navigation et accès aux nœuds
- 4.5 Notions d'éléments enfants et parents
- 4.6 Introduction au XML
- 4.7 Lecture de fichier XML
- 4.8 Requête XML http

5 Introduction à jQuery

- 5.1 Présentation de jQuery et des frameworks de développement JavaScript
- 5.2 Diversité des navigateurs, complexité d'écriture

- 5.3 Installation et usage
- 5.4 Fonction globale "\$"
- 5.5 Sélecteurs
- 5.6 Fonctions et extension de fonctions natives
- 5.7 Filtres, traversée & parcours
- 5.8 Évènements et gestionnaires d'évènements
- 5.9 Attributs, getters/setters et manipulation d'objets du DOM

II. Pratique

1 Mise en application et exercices.

2 Laboratoires

- 2.1 Manipulation de texte et événements sur boutons dans des pages web générique.
- 2.2 Laboratoire : Validation de formulaire côté client.
- 2.3 Laboratoire : Lecture de fichiers XML et insertion.
- 2.4 Laboratoire : Réalisation d'un chat multi-utilisateurs et interaction asynchrones avec des scripts PHP, XML, JSON.
- 2.5 Laboratoire : Reproduction de « Tetris » en JavaScript.

Programmation côté serveur :

Cette partie du cours aura pour sujet l'ensemble méthodes de développement actuelles permettant le développement d'applications côté serveur, au moyen du langage PHP.

I. Théorie

1 Les bases de la programmation Web

- 1.1 Site statique vs site dynamique
- 1.2 Architecture n-Tiers
- 1.3 Fonctionnement d'un site Web
- 1.4 Le Web dynamique

2 Le langage PHP

- 2.1 Bref historique
- 2.2 Services nécessaires à l'exécution d'un script PHP
- 2.3 Structure de base d'une page PHP
- 2.4 Les bases du langage
- 2.5 Structures conditionnelles et itératives
- 2.6 Fonctions
- 2.7 Transmission de données de page en page
- 2.8 Cookies
- 2.9 Session
- 2.10 Inclure un fichier
- 2.11 Gestion de fichiers du serveur

3 Modèle MVC (Model View Controller)

- 4 **Stockage des informations dans une DB**
 - 4.1 Connexion à une base de données (interface PDO)
- 5 **L'orienté objet avec PHP**
- 6 **MVC-Factory-DAO**
- 7 **Débogueur : Xdebug**
- 8 **Les principales failles de sécurité**

II Pratique

- 1 **Installation des services**
 - 1.1 WAMP / MAMP / LAMP
 - 1.2 Editeur de code
 - 1.3 Configuration de PHP
- 2 **Mise en application / Exercices**

Bibliographie :

Liste non-exhaustive :

- HEURTEL O., PHP et MySQL : Maîtrisez le développement d'un site Web dynamique et interactif , Eni éditions, Ressources Informatiques (Mai 2008)
- Différents cours présents sur <https://openclassrooms.com/> (voir syllabus)

PERSONNEL ENSEIGNANT

Programmation côté client : CARLIER Romain

Programmation côté serveur : PIERON Benjamin

METHODOLOGIE

Programmation côté serveur : sur la plateforme Claroline, les étudiants disposent de notes de cours au format PDF ainsi que des illustrations de la théorie et les corrigés des exercices réalisés durant le cours. → Langage étudié : PHP

A la fin des 2 parties du cours, les étudiants sont amenés à réaliser un projet (regroupant la partie « client » et la partie « serveur »).

MODES D'EVALUATION

L'évaluation continue est réalisée sous forme de devoirs à difficultés croissantes.

A la fin des 2 parties du cours, les étudiants sont évalués sur la partie « Théorie » de la matière. Pour la partie « Pratique », les étudiants sont amenés à réaliser un projet, dont ils reçoivent le cahier des charges.