

Présentation générale

16 séances de 1h45 sur le vaste sujet des “applications web”

Application web

- clients/serveur
- communication par http entre les clients et le serveur

cas classique : serveur = serveur HTTP, client = navigateur web

cas moderne : serveur = serveur HTTP, client = application javascript

Architecture HTTP

- le serveur contenant l'application est également client d'un autre serveur

Exemple : authentification décentralisée

- le client fait appel à plusieurs serveurs et agrège les réponses obtenus.

1

1

Objectifs du cours

Au niveau théorique :

- maîtriser HTTP et les différentes notions sous-jacentes : ressources, clients/serveur, en-têtes, cookies ...
- API REST
- architecturer une application en faisant collaborer des composants par dialogue HTTP
- indépendance du langage de programmation

Objectifs du cours

Au niveau pratique :

- architecture MVC
- pratique du développement orienté par les TESTS : BDD, mockup, stubs
- ...
- écriture de clients et serveurs HTTP complexes
- pratique de HTML
- javascript : coffeescript et jQuery
- utilisation d'un outil de versionning : git

Les exemples du cours seront donnés en Ruby et en javascript.

Évaluation de l'enseignement

- projet en contrôle continu, au fur et à mesure des séances (à partir de la séance 6)
- examen final pour l'aspect théorique

Le projet : une application web faisant intervenir plusieurs serveurs.

Déroulement des séances

Pour chaque séance :

- une partie théorique
- la mise en pratique sur un cas simple
- à faire à la maison : la (re)mise en pratique sur un cas différent

De semaine en semaine :

- avancement du projet