

Présentation du bloc développement web	Projet	AA : Workflow GIT	AA : SCRUM	Maquette statique	Apache/ HTTP	PHP	MVC/ Composer/ moteur de template	AA : Tests	PDO/ Auth/ mobile	Javascript
<p>[2] Comprendre les principes de base de la configuration d'Apache, y compris la définition du répertoire racine des sites web et ses implications en matière de sécurité.</p> <p>[3] Configurer plusieurs hôtes virtuels (vhosts) sur un même serveur Apache.</p> <p>[4] Localiser et interpréter les journaux (logs) d'Apache pour diagnostiquer des problèmes spécifiques du serveur web et en tirer des conclusions pour des actions correctives.</p> <p>[2] Comprendre en détail les en-têtes des requêtes et réponses HTTP, ainsi que leur impact sur la communication entre le navigateur et le serveur.</p> <p>[4] Analyser les différentes méthodes (verbes) HTTP et leur utilisation pour effectuer diverses actions sur le serveur web.</p> <p>[3] Comprendre la signification des codes d'état HTTP (statuts) et leur utilisation pour gérer les réponses du serveur de manière efficace.</p> <p>[4] Savoir interpréter un fichier .htaccess et configurer la redirection.</p> <p>[1] Connaître les fondements de HTTPS.</p>										

## SUJET

### Énoncé

## MISSION APACHE

Le client est satisfait des modifications effectuées par l'agence *Web4All* concernant le formulaire pour postuler à une offre.

Il faut maintenant passer à la vitesse supérieure, l'équipe doit rapidement configurer un environnement de développement local efficace et sécurisé.

Jean-Marc réunit l'équipe et expose les enjeux.

**Jean-Marc** : « Bonjour à tous. Nous avons un délai très court pour mettre en place notre environnement de développement. Rodriguez, peux-tu nous expliquer comment configurer notre serveur Apache local avec plusieurs hôtes virtuels ? »

**Rodriguez**, l'admin Sys/DevOps, répond : « Bien sûr. Nous allons définir le répertoire racine du site principal en utilisant la directive `DocumentRoot` d'Apache. Cela est crucial pour la sécurité, car nous devons nous assurer que seuls les fichiers nécessaires sont accessibles publiquement. »

**Estelle**, l'architecte logiciel, ajoute : « En plus de cela, je recommande de configurer deux hôtes virtuels distincts : un pour le site principal (*cesi-site.local*) et un autre pour nos ressources statiques (*cesi-static.local*) comme les fichiers JS et les images. Cela améliorera les performances et facilitera la gestion des fichiers. »

**Dominique**, la développeur frontend, interroge : « Et comment allons-nous gérer les erreurs, surveiller les requêtes, et déployer le frontend ? »

**Rodriguez** : « Pour le frontend, nous devons migrer tous les fichiers qui ne sont pas encore sur le serveur afin de garantir un déploiement complet et fonctionnel. Ensuite, nous utiliserons les journaux (logs) d'Apache pour suivre les requêtes et identifier les erreurs grâce aux codes d'état *HTTP* comme les *404* ou *500*. De plus, nous configurerons les en-têtes *HTTP* pour améliorer la communication entre le navigateur et le serveur. »

**Sami**, le développeur backend, poursuit : « Qu'en est-il des méthodes *HTTP*, des redirections et des outils de diagnostic ? »

**Rodriguez** : « Nous devons maîtriser les différentes méthodes *HTTP* comme *GET* et *POST* pour gérer les interactions serveur-client. Pour les redirections, nous utiliserons des fichiers *.htaccess* afin de configurer les redirections nécessaires sans modifier la configuration principale d'Apache. Concernant le diagnostic, nous utiliserons les outils de développement intégrés au navigateur. L'onglet *Network* permet d'inspecter les requêtes *HTTP*, leurs réponses, et de déboguer efficacement. »

**Jean-Marc** conclut : « Parfait. N'oublions pas d'implémenter *HTTPS* pour sécuriser les communications, même en local. Rodriguez, assure-toi que tout soit prêt dans les deux jours et demi impartis. Pensez également à documenter dans le livrable la migration complète du frontend et l'utilisation des outils de diagnostic pour les requêtes *HTTP*. »

L'équipe s'engage à suivre ces directives pour livrer un serveur Apache local configuré avec plusieurs hôtes virtuels, garantissant sécurité et performance.

### Ressources pour les étudiants

#### Installation de LAMP (Linux, Apache, MySQL et PHP) sur Linux Mint

Installation de LAMP sous Linux Mint - JGT v1.3.pdf [pdf] ↗

#### Documentation Officielle Apache

<https://httpd.apache.org/docs/> ↗

#### Guide sur les Hôtes Virtuels