

## Bloc1

Développer la présence en ligne de l'organisation

3h

# Client Serveur Web - Architecture

### Pré-requis :



- **Avoir un compte Github.**
- Disposer d'une connexion internet et de droits administrateur sur son poste de travail.

### Repository

- Créer un repository github nommé **bloc1**.
- Ajouter **jcheron** à la liste des collaborateurs.

Ce repository va permettre :

- De déposer vos travaux en gérant les versions (au travers des commits dans un premier temps) ;
- De publier en ligne vos productions ;
- De suivre et de communiquer sur vos travaux.

### Markdown

**Markdown** est un langage de balisage léger permettant de présenter des documents (html, pdf...). Il est utilisé par github.

Vous allez utiliser markdown dans le cadre de ce TD pour présenter votre propre documentation, mise en ligne sur votre repository.

- [Apprendre le markdown en 60 s](#)
- [Tutoriel en 10 minutes](#)
- Référence

### Lecture(s)

Ne pas limiter votre lecture aux seules références citées, approfondissez en effectuant vos propres recherches, avec curiosité 😊.

A lire impérativement :

- [Vue d'ensemble de http \(MSDN\)](#)
- [Historique et évolution de HTTP](#)
- [Headers HTTP](#)
- [Méthodes http](#)
- [http response status](#)
- [Types mime](#)
- [Négociation](#)

Pour aller plus loin : [cours complet HTTP](#)

# Bloc1

Développer la présence en ligne de l'organisation

## Travaux

### A vos markdown :



- **Rédiger vos réponses dans des fichiers md, sur votre repository bloc1, dans un sous dossier documents.**
- Vous pouvez éventuellement créer plusieurs fichiers md, et faire le lien entre eux.

#### 1 - Méthodes GET et POST

Illustrez au travers d'un ou plusieurs exemples précis les différences entre les méthodes http GET et POST, mentionnez les URLs utilisées, les données échangées (requête)...

#### 2 – Comparaison méthodes

Utilisez les exemples précédents pour effectuer un comparatif (tableau avec critères) entre ces 2 méthodes.

#### 3 -Extensible

Expliquez précisément en quoi le protocole http est extensible.

#### 4 - Sans état

Que signifie le fait qu'http soit qualifié de protocole sans état. Quelles sont les conséquences pour la navigation Web de cette caractéristique ?

#### 5 – URL

Décomposer une URL en ces différentes parties et expliquez le rôle de chacune d'entre elles.

#### 6 - Codes Status

Lister les familles de code status de la réponse http et donnez un exemple pour chacun d'entre eux.

#### 7 – Négociation de contenu

Expliquez le principe de la négociation de contenu http entre client et serveur (Exemple, illustrations...).

#### 8 – Installation Apache & configuration

Installer Xampp et configurer un virtualhost sur votre poste local.

Voir <https://slamwiki2.kobject.net/web/server#xampp>

#### 9 – CURL

Pour chaque requête effectuée, afficher la commande, le résultat et vos commentaires.

Requêtes à effectuer en invite de commande, avec curl :

1. Effectuer une requête GET vers l'url <http://dev.local>
2. Afficher l'entête de la réponse pour cette URL
3. Effectuer une requête GET vers l'url <http://dev.local/notExisting>
4. Afficher l'entête de la réponse pour cette URL
5. Déposer un fichier localement dans le dossier download depuis la racine de votre virtualhost **dev.local**
6. Téléchargez-le depuis curl

#### 10 – Headers

Lister dans un tableau les principaux en-têtes de la requête http, expliquez et illustrez leur rôle.