



Université Mohammed Premier
Ecole Nationale des Sciences Appliquées
Al Hoceima



Chapitre 1 : Fonctionnement du Web

Plan

- I. Généralités
- II. Architecture Client/serveur du web
- III. Protocole HTTP
- IV. Accès aux Ressources Web
- V. Notions liées au web
 - ▶ Page Statique/Dynamique
 - ▶ FTP
 - ▶ DNS
 - ▶ Hébergement Web
 - ▶ Indexation

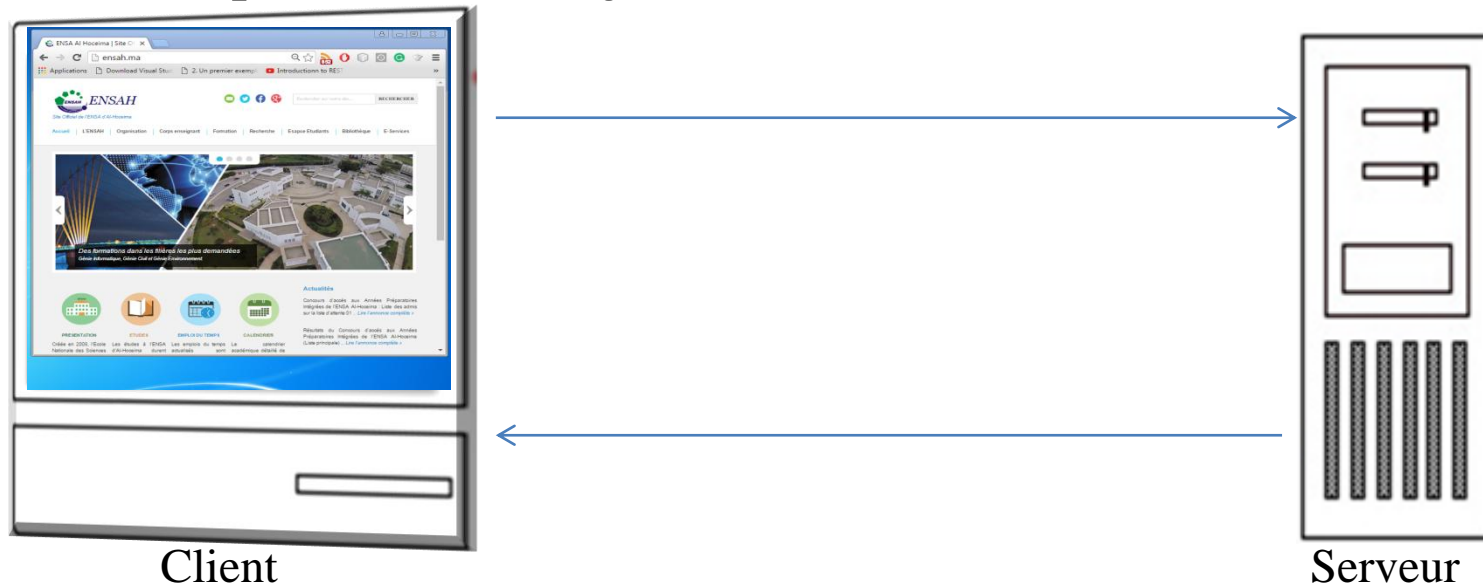
I. Généralités

1. Qu'est-ce que le WWW ?

- ▶ Le *World Wide Web* (WWW), littéralement la « **toile d'araignée mondiale** », communément appelé le **Web**, et parfois la **Toile**, est un vaste **ensemble de sources d'informations accessibles** à travers le **réseau Internet**.
- ▶ Il fut initialement construit par le CERN (Organisation Européenne pour la Recherche Nucléaire) pour la documentation des projets de recherches.
- ▶ Il est maintenant utilisé par tout le monde pour **mettre en ligne** (i.e. rendre accessible sur le Web via Internet) **des documents et des services de tous horizons**.
- ▶ La standardisation des principales technologies du WWW est assurée par **W3C** (World Wide Web Consortium), **un organisme de international de standardisation** à but non lucratif, chargé de promouvoir la compatibilité des technologies du WWW telles que HTML5, HTML, CSS, SOAP...etc

2. Le Web

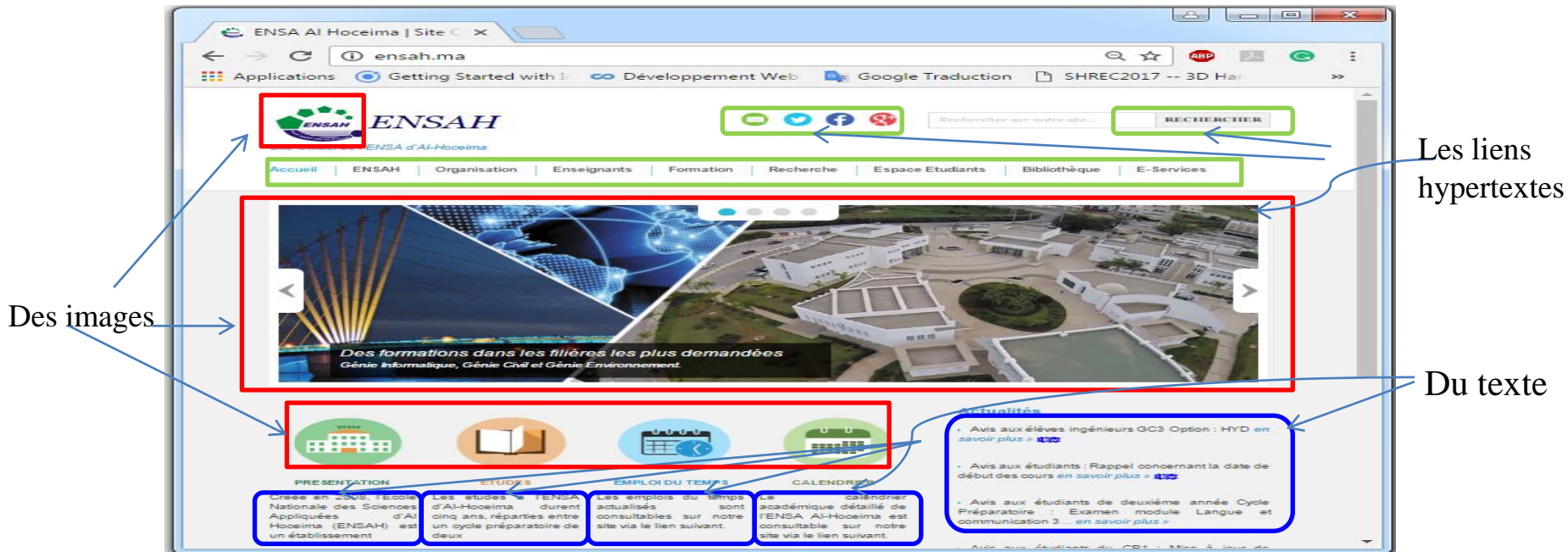
- Le Web permet à des utilisateurs de consulter, avec un navigateur, des pages web accessibles sur des sites hébergés sur des machines (serveurs) de l'Internet situant à des endroits géographiquement différents, parfois très éloignés.



- Le navigateur permet de demander et d'afficher une page Web en interprétant son code source correspondant.

3. Page et site Web

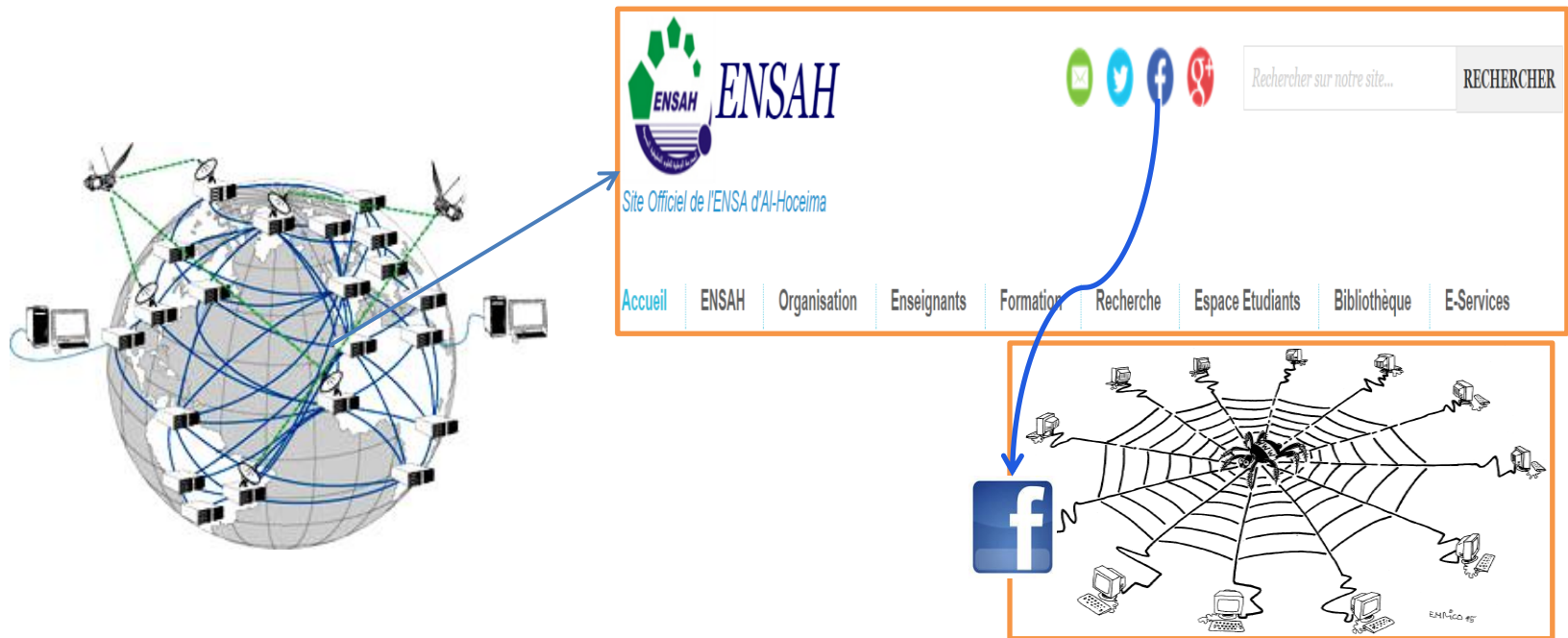
- Une page web est un hypermédia (hypertexte), un document électronique contenant des images, du son, du texte, des programmes, mais surtout des liens vers d'autres hyper-documents : des liens hypertextes.



- Un **site web**, est un ensemble de pages **web** liées ensemble par un domaine et accessible par **une** adresse **web**.

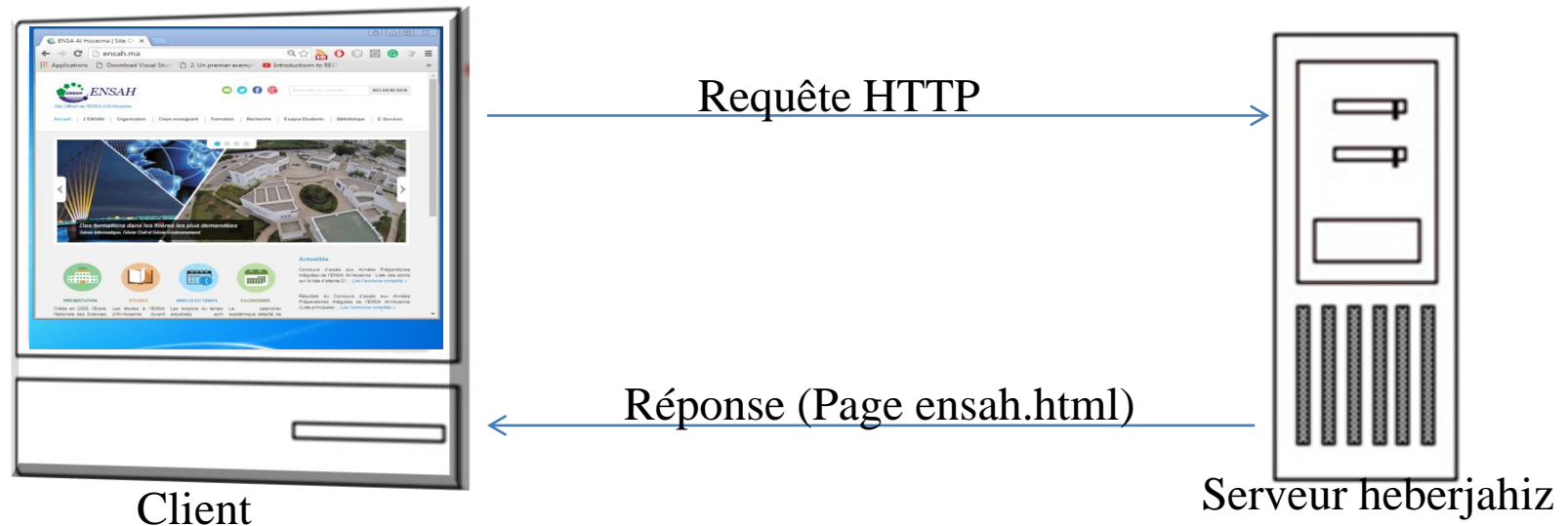
4. Web : la toile d'araignée

- ▶ L'image de **la toile d'araignée** vient des **hyperliens** qui **lient les pages web entre elles**.
- ▶ Ces hyperliens **permettent** de **passer d'un document à l'autre**. L'utilisateur peut passer en un clic d'une page placée, par exemple, sur un serveur à Rabat à une autre placée sur un serveur à Paris.



5. Principe de fonctionnement

- Web = Ressource + Protocole HTTP + URI



- Le client accède à une page Web en utilisant son adresse, son URL. Exemple(<http://ensah.ma/pages/presentation.php>). L'url est composé de http (le protocole), le nom de domaine(ensah.ma) et l'endroit où se trouve la page dans le site (/pages/presentation.php)

II. Logiciels client/serveur

1. Présentation

- ▶ Le **WWW** s'appuie sur la notion d'architecture client/serveur, un couple de logiciels qui communiquent ensemble via le réseau, Internet ou un intranet.
- ▶ Un serveur permet de fournir plusieurs services (accès à des sources de données, applications...).
- ▶ Pour fournir ces services, il fait tourner en permanence, des programmes que l'on appelle aussi des serveurs en l'occurrence ce sont des serveurs Web ou serveurs HTTP.
- ▶ De l'autre côté les utilisateurs font tourner sur leur machine (machine cliente) un programme client qui, comme son nom l'indique va être demandeur de services, en l'occurrence ce client est un navigateur Web qui va demander des pages Web à un serveur Web.

2. Navigateur

- ▶ **Le navigateur web** (web browser en anglais) est un logiciel permettant d'accéder à une ressource sur le web (exemple: une page web) et de l'afficher sur l'écran de l'utilisateur. Techniquement, c'est au minimum un client HTTP.
- ▶ L'une des **fonctions principales** d'un navigateur web est d'**effectuer le rendu visuel d'une page web** à partir de son code HTML et des fichiers CSS associés. Cela consiste à **lire et interpréter le code HTML et CSS**.
- ▶ Le **composant principal** du navigateur est le **moteur de rendu**, il est responsable du rendu visuel. Par défaut, ce moteur peut afficher des documents HTML, XML et des images. Mais, il peut afficher aussi d'autres types avec un plug-in (ou extension de navigateur), par exemple, PDF s'affiche en utilisant un plug-in de visualisation de PDF.
- ▶ **Il existe de nombreux navigateurs web**, pour toutes sortes de matériels (ordinateur personnel, tablette tactile, smartphones, etc.) et pour différents systèmes d'exploitation (Linux, Windows, Mac OS, iOS et Android).

.... la suite

- Les plus utilisés à l'heure actuelle sont : Google Chrome, Mozilla Firefox, Internet Explorer (remplacé par Edge), Safari, Opera, etc...



- La différence entre ces navigateurs réside principalement dans le moteur de rendu utilisé. Safari et Chrome utilisent Webkit tandis que Firefox et Mozilla utilisent Gecko.
- Cette différence peut être constatée aussi au niveau de l'affichage. En effet, les navigateurs n'affichent pas toujours un même site web *exactement* de la même façon.
- Cela est dû au fait que les navigateurs ne connaissent pas toujours les dernières fonctionnalités de HTML et CSS (il faut faire des mises à jour). Par exemple, Internet Explorer a longtemps été en retard sur certaines fonctionnalités CSS (et paradoxalement, il a aussi été en avance sur quelques autres).
- En théorie, la manière dont elles sont affichées les pages est définie par W3C.

3. Serveur Web

- ▶ Un « serveur web » peut faire référence à des **composants logiciels** (*software*) **ou** à des **composants matériels** (*hardware*) **ou** à des composants **logiciels et matériels** qui fonctionnent ensemble.
 - ▶ *Au niveau des composants matériels*, un **serveur web** est un ordinateur **qui stocke les fichiers** qui composent un site web (par exemple les documents HTML, les images, les feuilles de style CSS, les fichiers JavaScript).
 - ▶ *Au niveau des composants logiciels*, un serveur web contient différents fragments qui **contrôlent la façon dont les utilisateurs peuvent accéder aux fichiers hébergés**. On trouvera *a minima* un serveur *HTTP*. Un serveur HTTP est un logiciel qui **comprend les URL et le protocole HTTP**.
- ▶ **Exemples** : Apache, Tomcat, Google Web Server, Internet Information Services (IIS), NodeJS...etc

III. Le protocole HTTP

1. Présentation

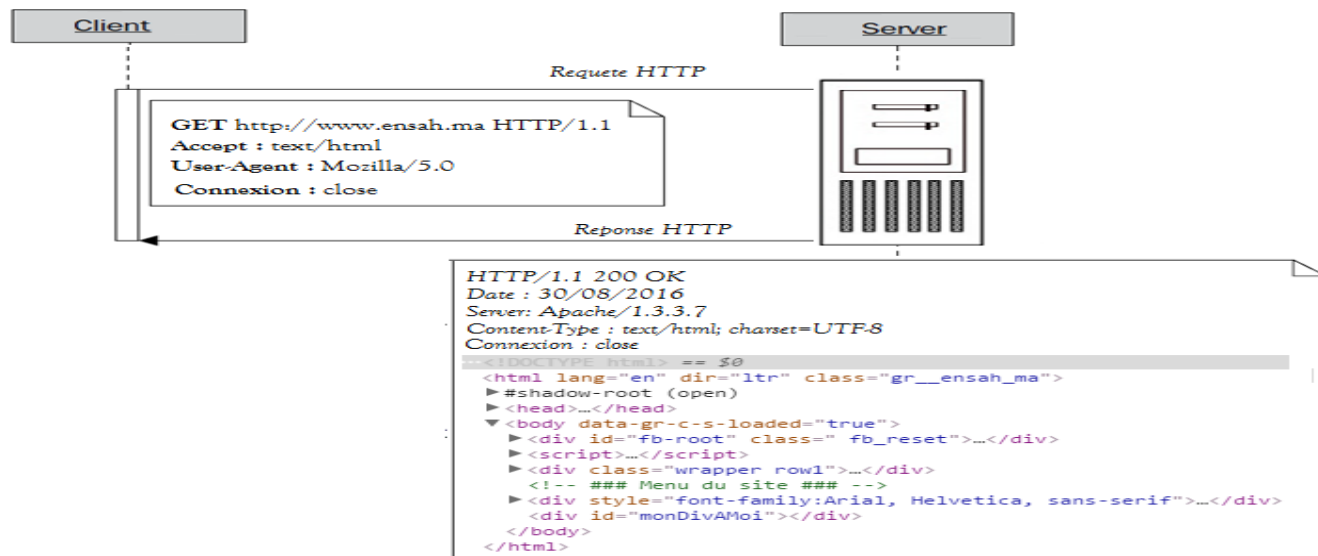
- ▶ La communication entre le client et le serveur sur le monde Web est assuré à l'aide de protocole HTTP(Hyper Text Transfer Protocol). Ce protocole est communément utilisé pour transférer les ressources du Web.
- ▶ Il est **capable d'assurer le transfert** de **texte, hypertexte, fichiers audio, images** ou **tout autre type d'information** pouvant se mettre sous la forme d'un fichier.
- ▶ Il est **utilisé pour la plupart des transactions du Web** : requête d'une ressource, envoi de données d'un formulaire, navigation...
- ▶ D'autres informations plus discrètes pour l'utilisateur sont transmises par HTTP : échange d'informations comme par exemple des dates de dernières mises à jour, un résumé de contenu d'une page, la configuration d'un serveur,...
- ▶ **Des informations sécurisés**, telles que les noms d'utilisateur et **les mots de passe**, peuvent être transmises grâce à la variante sécurisé de ce protocole qui est l'**HTTPS**. Il est la combinaison du HTTP avec une couche de chiffrement comme SSL ou TLS.

2. Principe de fonctionnement

- ▶ HTTP **se base** essentiellement **sur un dialogue texte** (les pages web sont constituées de code HTML).
- ▶ Ce protocole fonctionne en mode **sans état (*stateless*)**, c'est-à-dire qu'il **ne conserve aucune information entre deux transactions**, il faut donc tout reprendre depuis le début à la transaction suivante.
- ▶ **Le dialogue** entre le client et le serveur **se compose donc de requêtes** émises par le client et de réponses données par le serveur.
- ▶ **Une requête HTTP** est **un ensemble de lignes envoyé au serveur** par le navigateur.
- ▶ Le scénario correspondant à une requête de type « demande d'une page » entre un navigateur et un serveur Web est le suivant :
 - ▶ Le navigateur Web client établit une connexion TCP avec le serveur Web qui contient la page qui l'intéresse.
 - ▶ Une fois la connexion établie, le client émet une requête HTTP contenant une commande, le lien de la page, et parfois d'autres informations.
 - ▶ Lorsque le serveur Web reçoit la requête il essaie d'exécuter la commande qu'elle contient.
 - ▶ Il retourne ensuite comme réponse le résultat obtenu qui peut être des données, un message d'erreur, et d'autres informations.
 - ▶ Une fois que le client a reçu sa réponse la connexion est fermée et détruite.

3. Types de Requêtes HTTP

- ▶ Il existe de nombreuses méthodes HTTP (GET, HEAD, POST, OPTIONS, CONNECT, TRACE, PUT, PATCH, DELETE).
- ▶ Les quatre méthodes correspondent aux opérations CRUD (Create Read Update and Delete) sont GET, POST, PUT, et DELETE.
 - ▶ *Requête « GET »* : Elle permet de consulter et de retrouver l'information. Un exemple de déroulement de dialogue pour consulter le site de www.ensah.ma (demander la page d'accueil du site) :



- *Requête « POST »* : Envoi de données au programme situé à l'URL spécifiée. Exemple d'une requête POST

```
POST /eservices/login.php HTTP/1.1
Host: https://ensah.ma/apps/eservices/login.php
User-Agent: Mozilla/5.0
Accept: text/html
Accept-Language: fr,fr-fr;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip,deflate Accept-Charset: ISO-8859-1,utf-8
Content-Type: application/x-www-form-urlencoded
Content-Length: 40
Keep-Alive: 300
Connection: keep-alive
login=test&motpass=test
```

- *Requête « PUT »* : permet de remplacer ou d'ajouter une ressource sur le serveur. L'URI fourni est celui de la ressource en question.
- *Requête « DELETE »* : Cette méthode permet de supprimer une ressource du serveur

4. Test HTTP avec Telnet

- ▶ Pour tester http on va utiliser le protocole telnet.
- ▶ Exemple d'une requête GET avec telnet:
 - ▶ Sur l'invite de commande on lance :
 > telnet www.ensah.ma 80
 GET / HTTP/1.1
 HOST : www.ensah.ma
- ▶ Essayez de tester pour voir le retour de cette requête

IV. Accès aux Ressources Web

1. Représentation de la ressource

- ▶ Une représentation désigne les données échangées entre le client et le serveur pour une ressource.
- ▶ Une ressource du Web c'est généralement un document HTML, mais peut aussi être un PDF, une image ou un autre type.
- ▶ Les données échangées sont spécifiées par le type MIME(Multipurpose Internet Mail Extensions) présenté dans l'entête de protocole http.
- ▶ Le type MIME sert à identifier le type de ressource, le format de fichiers qui s'échangent sur le Web.
- ▶ Il est initialement prévu pour l'envoi de ressources par mail, il est à présent utilisé par de nombreux services d'Internet.
- ▶ Un type MIME se compose de deux parties séparées par un slash /. La première partie indique une catégorie générale : image, text,... Le deuxième spécifie en général le nom du format de fichier.
- ▶ Exemples: text/html, application/json, application/xml, application/pdf, application/zip, image/jpeg, audio/x-wav ...etc.

2. Localisation de la ressource

- ▶ Il est spécifié par l'utilisateur à l'aide d'un URI (Uniform Resource Identifier) qui est l'identifiant de la ressource sur le réseau, sa syntaxe respecte une norme d'Internet mise en place par W3C.
- ▶ Les URL (Uniform Resource Locator) sont une forme particulière d'URI qui permettent de définir une localisation pour une ressource (les liens hypertexte du web). Il est utilisé pour accéder à une page web.
- ▶ Un URL peut désigner un serveur ftp, un fichier sur le disque dur, une image, une adresse courrier, un serveur de News, un serveur telnet et bien sûr une page Web publiée par un serveur http, c'est-à-dire un serveur de Web.
- ▶ l'URL contient le nom du protocole d'accès au fichier (HTTP, HTTPS), le nom du serveur (adresse IP ou nom symbolique), le chemin d'accès au fichier et bien sûr le nom du fichier : **<Protocole>://<nom serveur>/<chemin>**. Exemple : **http://www.ensah.ma/**

V. Notions liées au web

1. Contenu statique /dynamique

- ▶ Une page web statique c'est une page qui est visible telle qu'elle a été conçue. Ce type de pages peuvent présenter toute forme de contenu, animations flash, images, musique, vidéo etc... mais elles sont toujours présentées de la même façon. Elles ne changent pas et c'est en ce sens qu'elles sont statiques.
- ▶ Les pages dynamiques permettent de présenter les informations de différentes manières selon l'interaction avec le visiteur. Les pages sont alors construites "à la volée" grâce à une programmation conçue par le webmaster. Le contenu est issu d'une base de données en fonction de critères établis par l'internaute puis mis en page en temps réel.
- ▶ Un serveur web peut « servir » du contenu statique ou dynamique.
 - ▶ Pour un contenu « statique », le serveur envoie les fichiers hébergés « tels quels » vers le navigateur.
 - ▶ Pour un contenu « dynamique », le serveur hébergeur contient un serveur d'application qui tire les données d'une base de données, le formate et l'insère dans différents modèles HTML. Une fois ce traitement effectué, le serveur envoie le fichier vers le navigateur.

2. DNS

- ▶ Pour accéder à un site web il faut préciser l'adresse IP de la machine d'hébergement. **Le problème est que ces adresses sont numériques ce qui rend difficile leur mémorisation.**
- ▶ Pour faciliter l'accès, **un mécanisme** a été mis en place permettant d'associer à une adresse IP un nom plus simple à retenir, **appelé DNS** pour DNS(Domain Name System). Ce dernier permet de nommer les machines plutôt que d'avoir à mémoriser leur adresse IP.
- ▶ **Exemples** : l'adresse IP correspondant au site de l'ENSAH : www.ensah.ma est 5.153.23.8. Pour l'université Mohammed Premier www.ump.ma <-> 196.200.156.5
- ▶ L'opération qui consiste à **retrouver l'adresse IP associée à un nom de domaine** s'appelle **la résolution du nom**.
- ▶ Lorsqu'un visiteur demande une page à son navigateur Web, celui-ci interroge des serveurs DNS pour connaître l'adresse IP du serveur hébergeant ce site. Dès qu'il obtient la réponse, le navigateur va interroger ce serveur et lui demander cette page.

3. FTP

- ▶ FTP(File Transfer Protocol) c'est le protocole de transfert de fichier sur Internet. Il permet, depuis un ordinateur, de copier des fichiers vers un autre ordinateur du réseau, ou encore de supprimer ou de modifier des fichiers sur cet ordinateur.
- ▶ Il est utilisé souvent pour **le transfert de l'ensemble du site vers un hébergeur** ou bien pour l'alimenter.
- ▶ Il existe plusieurs logiciels FTP à savoir FileZilla, WinSCP, Core FTP LE, etc.
- ▶ **FileZilla** est l'un des clients FTP les plus connus, il propose un client FTP permettant aux utilisateurs de se connecter à distance sur un serveur afin d'y télécharger des fichiers. Cette application particulièrement riche en fonctionnalités **supporte le glisser-déposer**.

4. Hébergement d'un site web

- ▶ Le site web une fois développé, il faut choisir comment le mettre accessible sur le web. Ceci est réalisé par la mise en place d'un serveur web.
- ▶ Ce serveur doit stocker les fichiers nécessaires au fonctionnement du site web : tous les documents HTML et les ressources liées dont les images, les fichiers JavaScript, les feuilles de styles, les fichiers de fontes, les vidéos, etc...
- ▶ D'un point de vue technique, il serait tout à fait possible de stocker tout ces éléments sur un ordinateur personnel. Toutefois, **il est beaucoup plus pratique d'utiliser un serveur destiné spécifiquement** à cela car il devra :
 - ▶ Toujours être en fonctionnement;
 - ▶ Toujours être connecté à Internet;
 - ▶ Conserver la même adresse IP au cours du temps;
 - ▶ Etre maintenu par un fournisseur tiers.

6. Le choix de mode d'hébergement

- ▶ Les entreprises font face à de nombreux challenges au niveau de l'hébergement de leurs infrastructures et applications informatiques.
- ▶ Pour héberger un site web, il existe plusieurs solutions :
 - ▶ **Héberger mutualisé** : un prestataire est une personne ou une entreprise (physique ou morale) capable d'offrir un service appelé prestation.
 - ▶ **Hébergement Cloud** : de plus en plus en vogue, cela consiste à héberger le site web sur des serveurs virtuels. En fait, c'est l'équivalent d'un hébergement dédié virtuel, mais avec tout un tas de services autour permettant de gérer plus facilement le réseau, les bases de données, etc. C'est la tendance pour de plus en plus de moyens et gros sites. Parmi les hébergeurs cloud, on peut citer Amazon Web Services, Google Cloud, Microsoft Azure, etc...
 - ▶ **Hébergement dédié** : avoir son propre serveur d'hébergement.

- ▶ **Le mode d'hébergement** peut être envisagé principalement **en fonction de l'importance du site web** :
 - ▶ Pour un petit site :(un site d'importance et d'audience faibles ou moyennes), il est fréquent d'être hébergé chez un prestataire qui propose un type ***d'hébergement dit « mutualisé »***, un mode d'hébergement dans un environnement technique (processeur, mémoire vive, espace disque, débit) dont la caractéristique principale est d'être partagé par plusieurs utilisateurs.
 - ▶ Pour les sites de grandes tailles et à fort trafic : dans cas il faut faire appel un type hébergement dit dédié, cela consiste à louer ou acheter un serveur dédié pour le site ou l'application en question.

7. Indexation Web

- ▶ L'indexation automatique de documents est le processus qui **permet de décrire** de manière compacte **le document**, cela consiste à coder dans une description synthétique le contenu du document en sélectionnant les données qui le caractérisent le mieux.
- ▶ Pour le web, l'indexation permet de créer une liste de descripteurs à chacun desquels est associée une liste des pages et/ou parties de pages auxquels ce descripteur renvoie.
- ▶ Le but est de permettre à des systèmes de recherche d'information(moteur de recherche) de retrouver des informations(exemple site web), dans des larges bases de données, d'une façon plus rapide et facile.
- ▶ Pour faire l'indexation, les **moteurs de recherche utilisent** un **robot d'indexation** (ou littéralement **araignée** du Web; en anglais **web crawler** ou **web spider**) qui est un logiciel qui explore automatiquement le Web. Il est généralement conçu pour collecter les ressources (pages Web, images, vidéos, documents Word, PDF etc.), afin de permettre à un moteur de recherche de les indexer. Exemple de Googlebot de Google.

- ▶ Il est possible d'empêcher ces robots de crawling d'accéder à tout ou une partie d'un site web en utilisant le protocole d'exclusion des robots, plus connu sous le nom de **robots.txt**.
- ▶ Le fichier robots.txt, à placer à la racine d'un site web, contient une liste de ressources du site qui ne sont pas censées être explorées par les moteurs de recherches.
- ▶ Par convention, les robots consultent le fichier robots.txt avant d'explorer puis d'indexer un site Web. Lorsqu'un robot tente d'accéder à une page web, comme par exemple <http://www.ensah.ma>, il tente d'accéder en premier lieu au fichier robots.txt situé à l'adresse : <http://www.ensah.ma/robots.txt>
- ▶ Exemple d'utilisation de robots.txt :

```
User-agent: *  
Disallow: https://ensah.ma/apps/
```

► Web: la qualité

► Besoin de Performance :

- Permettre la montée en charge (**scalabilité**). Il faut toujours considérer dès le départ les problèmes d'échelle liés à la capacité de croissance surtout pour le Web qui toujours en plein expansion et dimensionnement. **Les qualités d'un système ne doivent pas se dégrader en cas de croissance.**
- Partage de la charge (la **répartition de charge** anglais load balancing), en répartissant la charge entre plusieurs serveurs redondants (*reverse proxy*).

► Besoin de puissance de calcul :

- Exploiter les puissances de calcul disponibles.
- Coopérer plusieurs machines pour diminuer le temps d'exécution.
- Répartir les données pour un traitement parallèle.

► **Besoin de tolérance aux fautes**

- **Duplication (redondance) d'une application** afin de diminuer le taux de pannes et augmenter la fiabilité
- **Duplication des machines** (serveurs, unité de stockage,) **et des données**. En effet, N machines plus fiable qu'une seule

► **Besoin de communication et de partage d'information** : utiliser et partager des ressources distantes.

- **Services décentralisés**: création des applications de type « Mashup »; application composite.
- **Données délocalisées**: utiliser ses données à partir de n'importe quelle machine (Principe des Cloud computing. Exemple : iCloud, Dropbox....)
- **Imprimer à distance**: imprimante partagée entre toutes les machines.

► **Besoin d'un système évolutive**: Ce système doit avoir la capacité d'évolution car il faut toujours rester en veille face aux nouvelles technologies. Afin de permettre cette possibilité d'évolution, les applications réparties doivent avoir :

- **Une architecture modulaire** (composants)
- **Un découplage fort**