

Les bases du Web

Pr. Amina BENGAG

Objectifs et contenu

- Le fonctionnement du Web
 - Architecture client/serveur
 - Protocoles HTTP, DNS
- La production des pages web statiques ➔ HTML/CSS
- La production des pages web dynamiques
 - Coté client ➔ Javascript
 - Coté serveur ➔ PHP/Python (Mini Projet)

Chapitre I : Le fonctionnement du Web

Plan

- Généralités
- Architecture Client/Serveur du web
- Protocole HTTP
- Notions liées au web
 - Page Statique/dynamique
 - FTP
 - DNS
 - Hébergement Web

Généralités

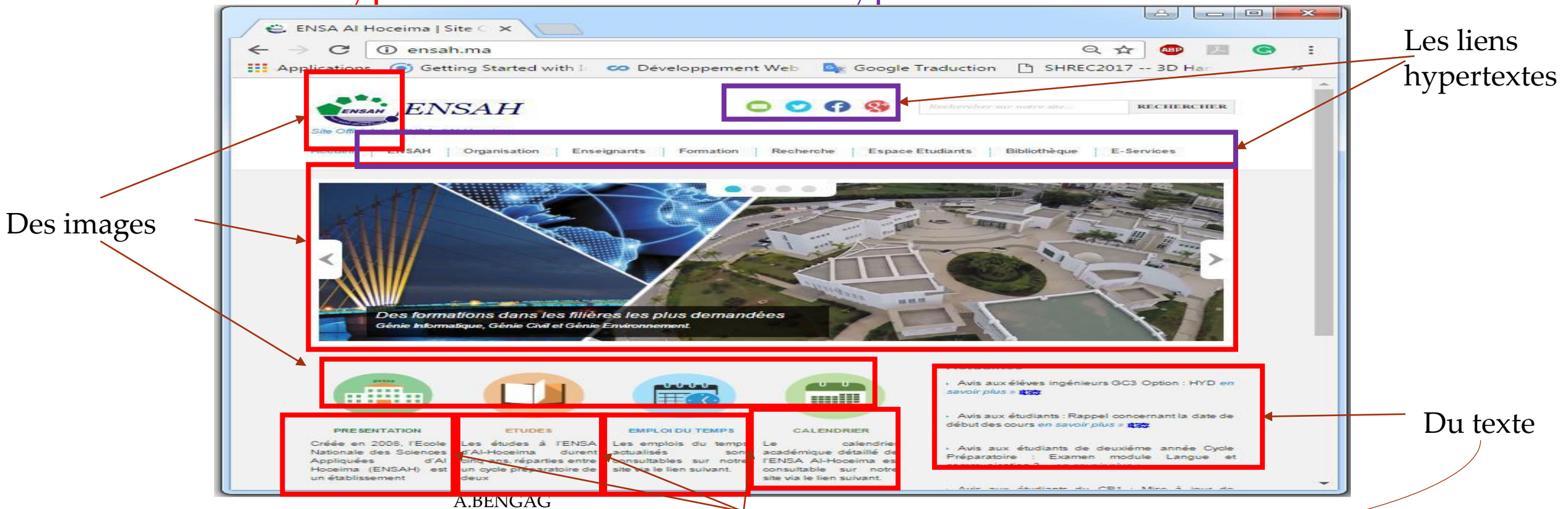
qu'est ce que le WWW ?

- 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 des documents et des services de tous horizons.
- La standardisation des principales technologies du WWW est assurée par **W3C** (World Wide Web Consortium).

Généralités

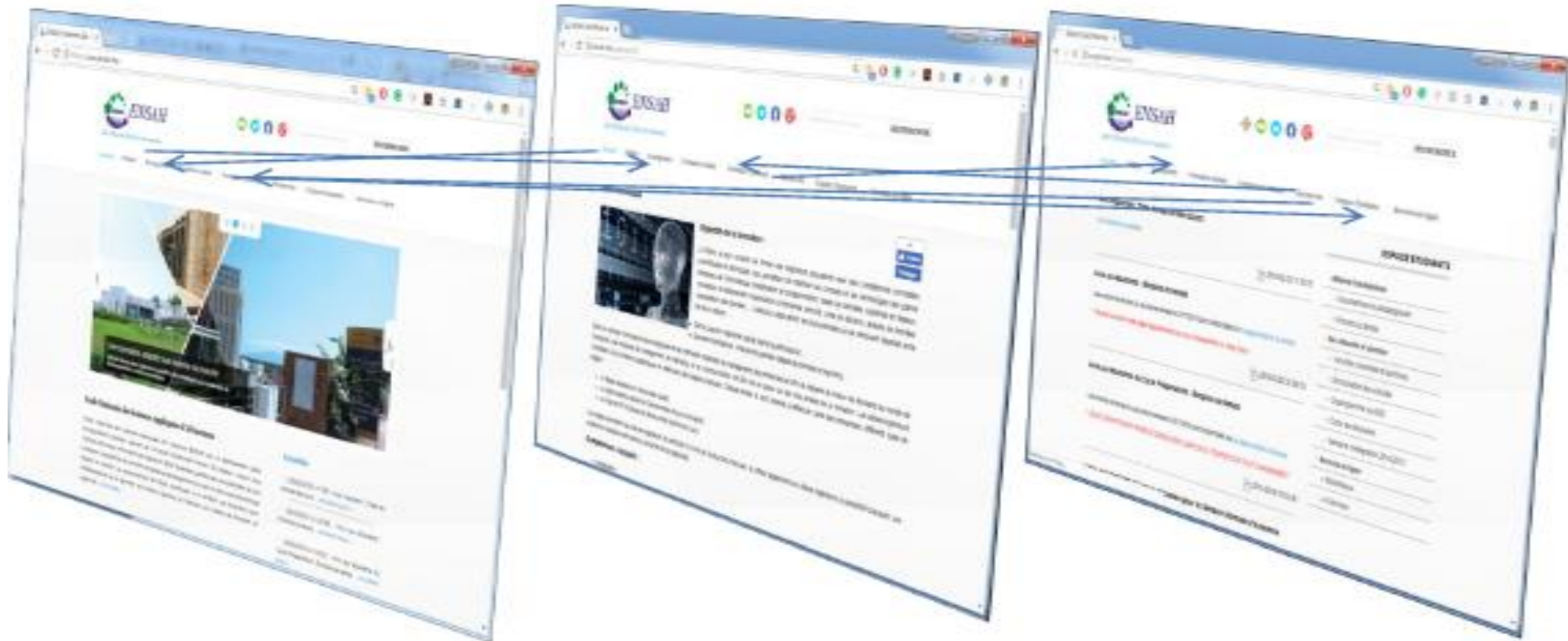
Page Web

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



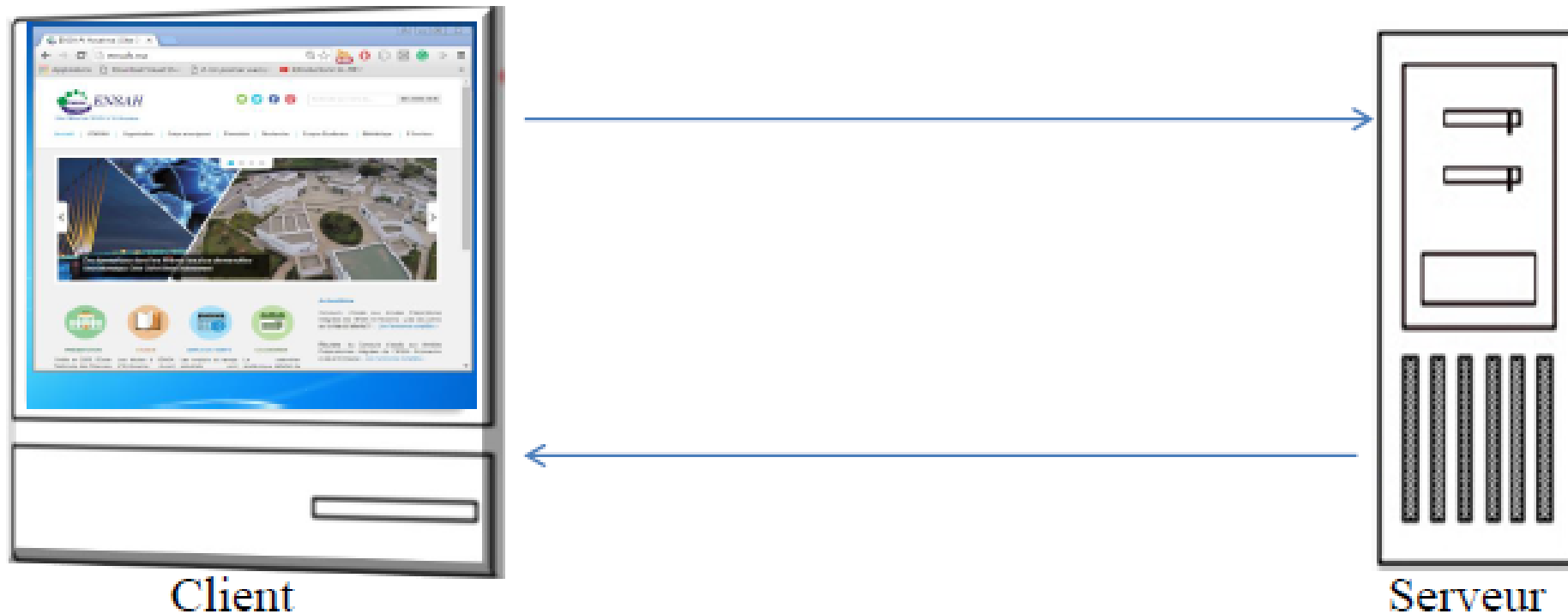
Généralités site Web

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



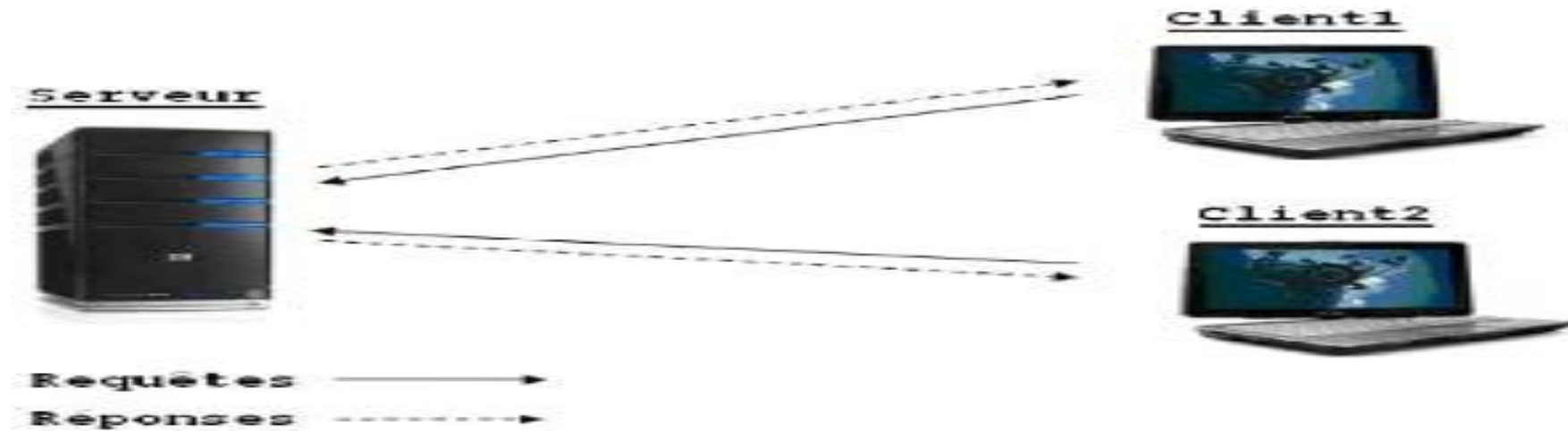
Généralités le Web

- 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.



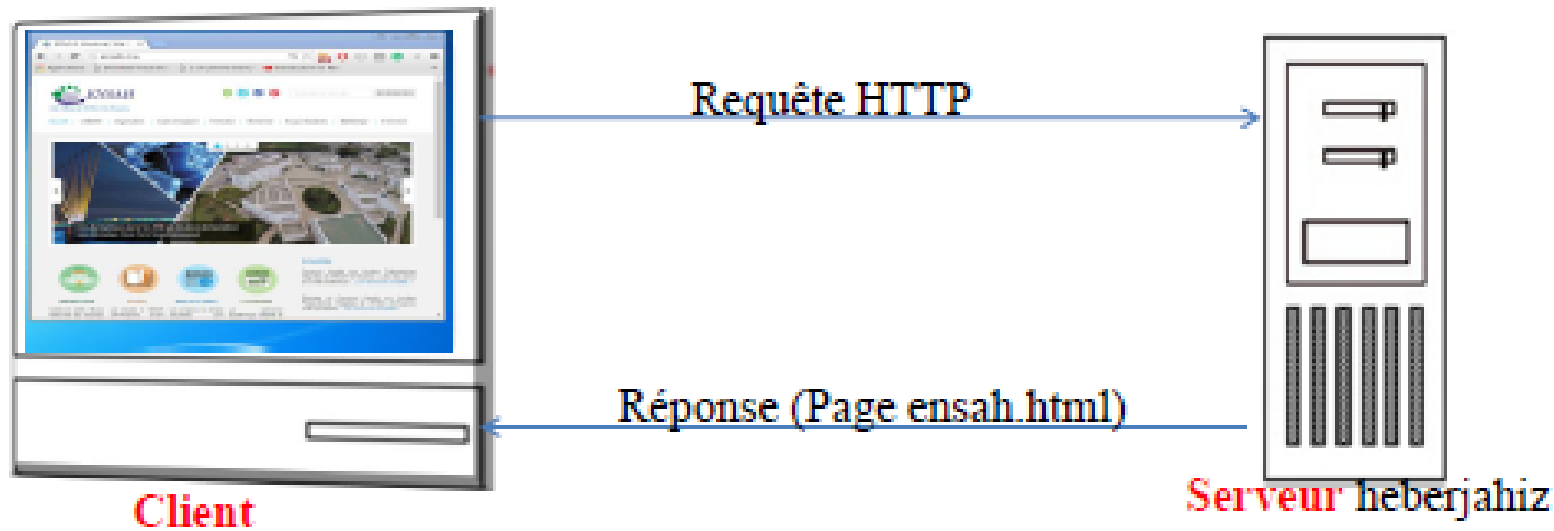
Modèle client/serveur

- Un système client/serveur fonctionne selon le schéma suivant:
 - **Le client** émet une requête vers le serveur grâce à son **adresse** et à son **port**, qui désigne un service particulier du serveur
 - **Le serveur** reçoit la demande et répond à l'aide de l'**adresse** de la machine client (et de son **port**)



Principe de fonctionnement

- Web = Ressource + Protocole HTTP + URL



- Le client accède à une page Web en utilisant son adresse, son **URL**.
Exemple(<http://ensah.ma/pages/presentation.php>).
- L'**url** est composé principalement du protocole (**http(s)**) et du **nom de domaine**(ensah.ma) mais aussi, elle peut contenir **l'endroit où se trouve la page** dans le dossier du site web ([/pages/presentation.php](#))

Navigateur web (Client)

- Le **navigateur web** 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 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**.

Navigateur web (Client)

- Il existe de nombreux navigateurs web :



- 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** est constaté 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 (il a aussi été en avance sur quelques autres).
- En théorie, la manière dont elles sont affichées les pages est définie par W3C.

Serveur web

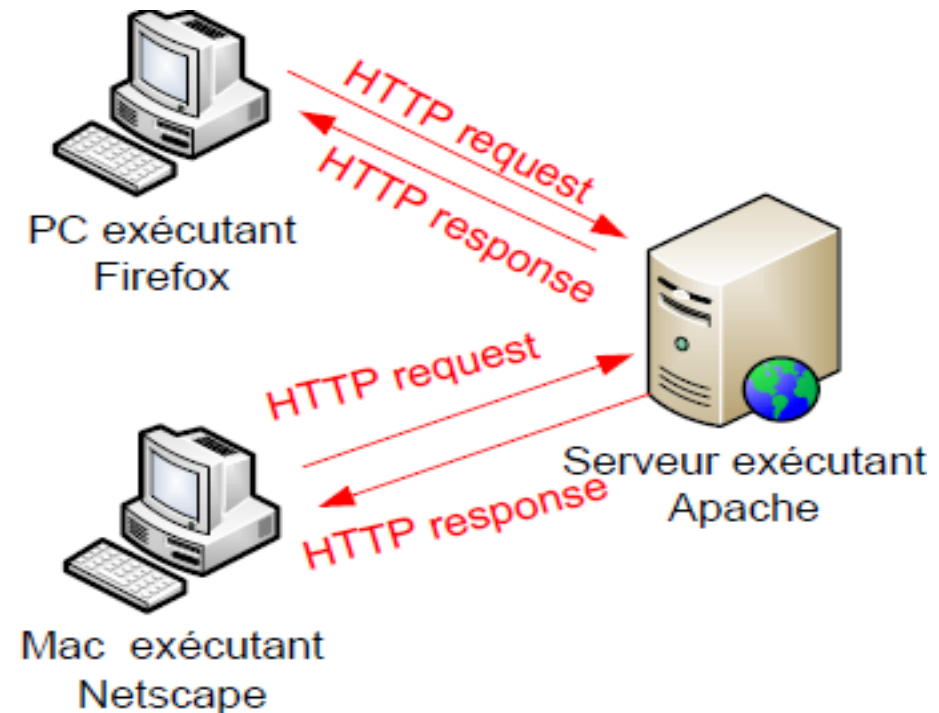
- Un « **serveur web** » peut faire référence à des **composants logiciels** (*software*) ou à des **composants matériels** (*hardware*) ou bien les **deux composants** qui fonctionnent ensemble.
- *Au niveau des composants matériels*, un serveur web est un **ordinateur** qui **stocke** les fichiers 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 est capable d'interpréter les **requêtes HTTP** arrivant sur le port associé au protocole HTTP (par défaut le port **80**), et de fournir une réponse avec ce même protocole.
- *Exemples* : Apache, Tomcat, Google Web Server, Internet Information Services (IIS), ...etc

HTTP

Hyper Text Transfer Protocol

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).
- Il est capable d'assurer le transfert de hypertexte, texte, fichiers audio, images ou tout autre type d'information pouvant se mettre sous la forme d'un fichier.

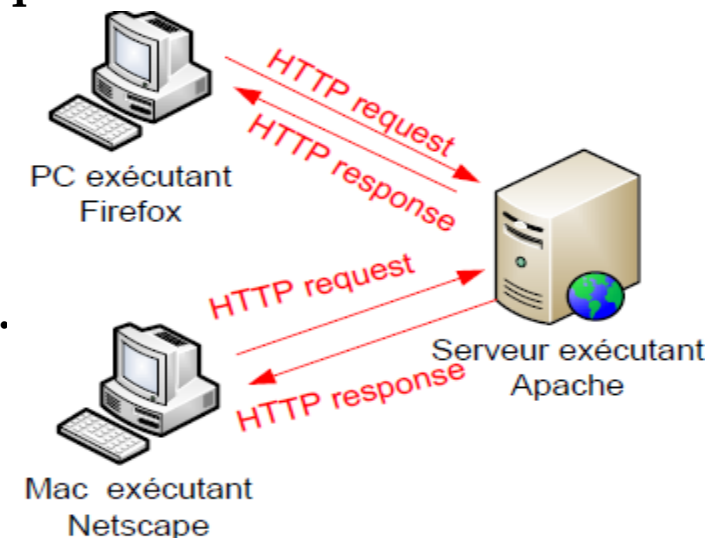


HTTP (Hyper Text Transfer Protocol)

- HTTP utilise les services de transport **TCP**
 - Le client initialise une **connexion TCP**
- HTTP est «**sans état**»
 - Le serveur **ne maintient** aucune information sur des requêtes clients **précédentes**
- Connexions HTTP
 - **Connexion non-persistante**
 - Au plus, **un objet** est transmis avec **une connexion TCP** (entre le client et le serveur)
 - Le serveur interprète la requête du client, répond et ferme la connexion TCP
 - HTTP/1.0 utilise ce type de connexion
 - **Connexion persistante**
 - **Plusieurs objets** peuvent être transmis avec une **seule connexion TCP** (entre le client et le serveur)
 - Le client envoie une requête pour tous les objets requis dès qu'ils sont référencés dans le fichier HTML
 - HTTP/1.1 utilise ce type de connexion par défaut

Principe de fonctionnement

- Le scénario correspondant à une requête de type « **demande d'une page** » entre un **navigateur** et un **serveur Web** est le suivant :
 1. Le navigateur Web **client établit une connexion TCP** avec le serveur Web qui contient la page qui l'intéresse.
 2. Une fois la connexion établie, le client émet une **requête HTTP** contenant une commande, le lien de la page, et parfois d'autres informations.
 3. Lorsque le serveur Web reçoit la requête il essaie **d'exécuter la commande** qu'elle contient.
 4. Il retourne ensuite comme **réponse** le résultat obtenu qui peut être des données, un message d'erreur, et d'autres informations.
 5. Une fois que le client a reçu sa réponse la connexion est fermée et détruite.



HTTP (Hyper Text Transfer Protocol)

- **Exemple d'une requête HTTP (GET)**

Question :

Quand on entre l'URL <http://www.utt.fr> dans la barre de navigation, quel est le msg est effectivement émis par le clt vers le srv ?

GET /nomrepertoire/page.html HTTP/1.1

Host: www.utt.fr

Connection: close

User-agent: Mozilla

Accept-language: fr

Méthode

Nom de la ressource

Version d'HTML

Ligne de la requête

Nom du serveur qui contient la ressource

Pas de demande de connexion persistante

Version de la page si disponible

HTTP (Hyper Text Transfer Protocol)

- Exemple d'une réponse HTTP

entête

HTTP/1.1 200 OK → Ligne de statut (protocole, Code de statut, Etat)

Connection: close → La connexion est coupée par le serveur après l'envoi des données

Date : Thu, 15 Jun 2004 09:00:15 GMT

Server: Apache/1.3.0 (Unix)

Last-Modified Mon, 22 Jun 1998 20:17:51 GMT

Content-Length: 6599

Content-Type: text/html

(données) → L'objet demandé est dans la réponse

HTTP (Hyper Text Transfer Protocol)

- *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.
- *Requête « DELETE »* : permet de supprimer une ressource sur le serveur.

Notions liées aux web

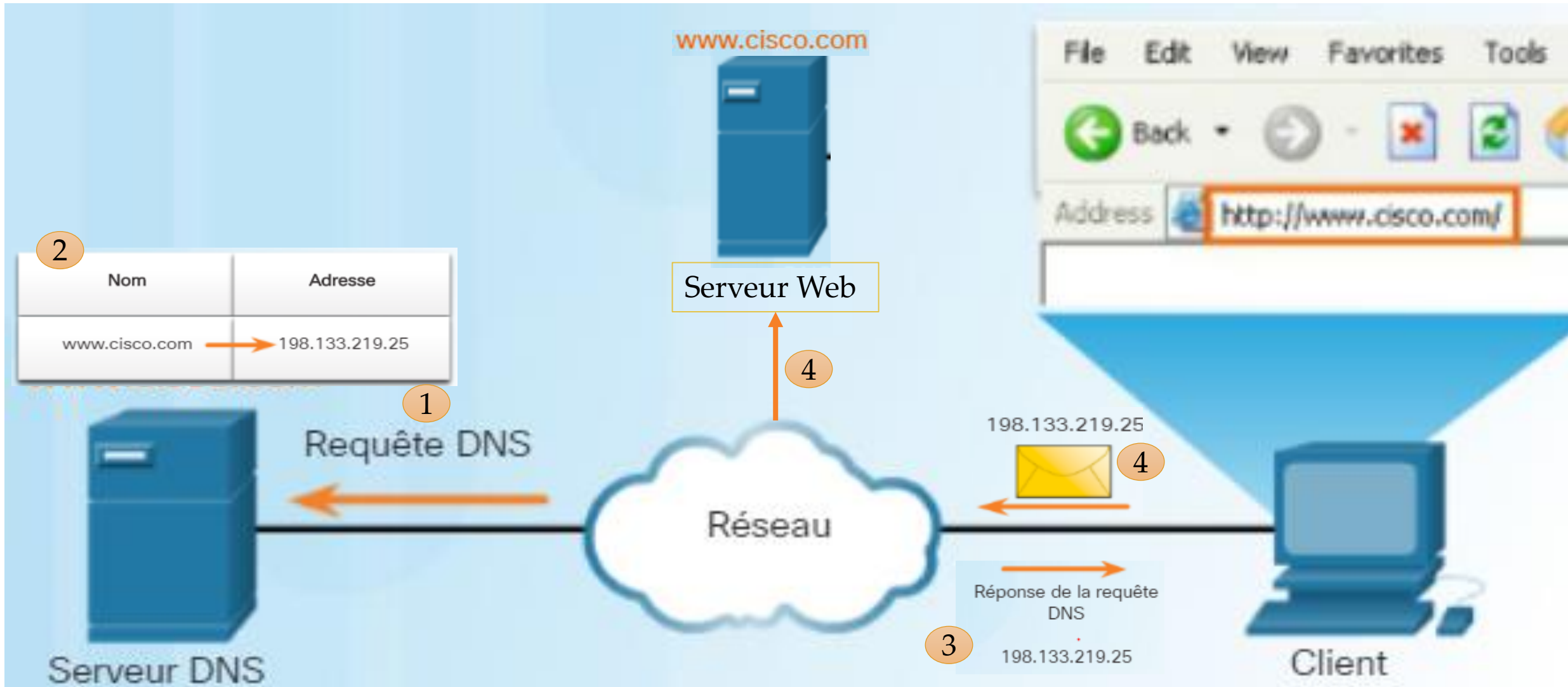
Contenu statique / dynamique

- **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**. 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.

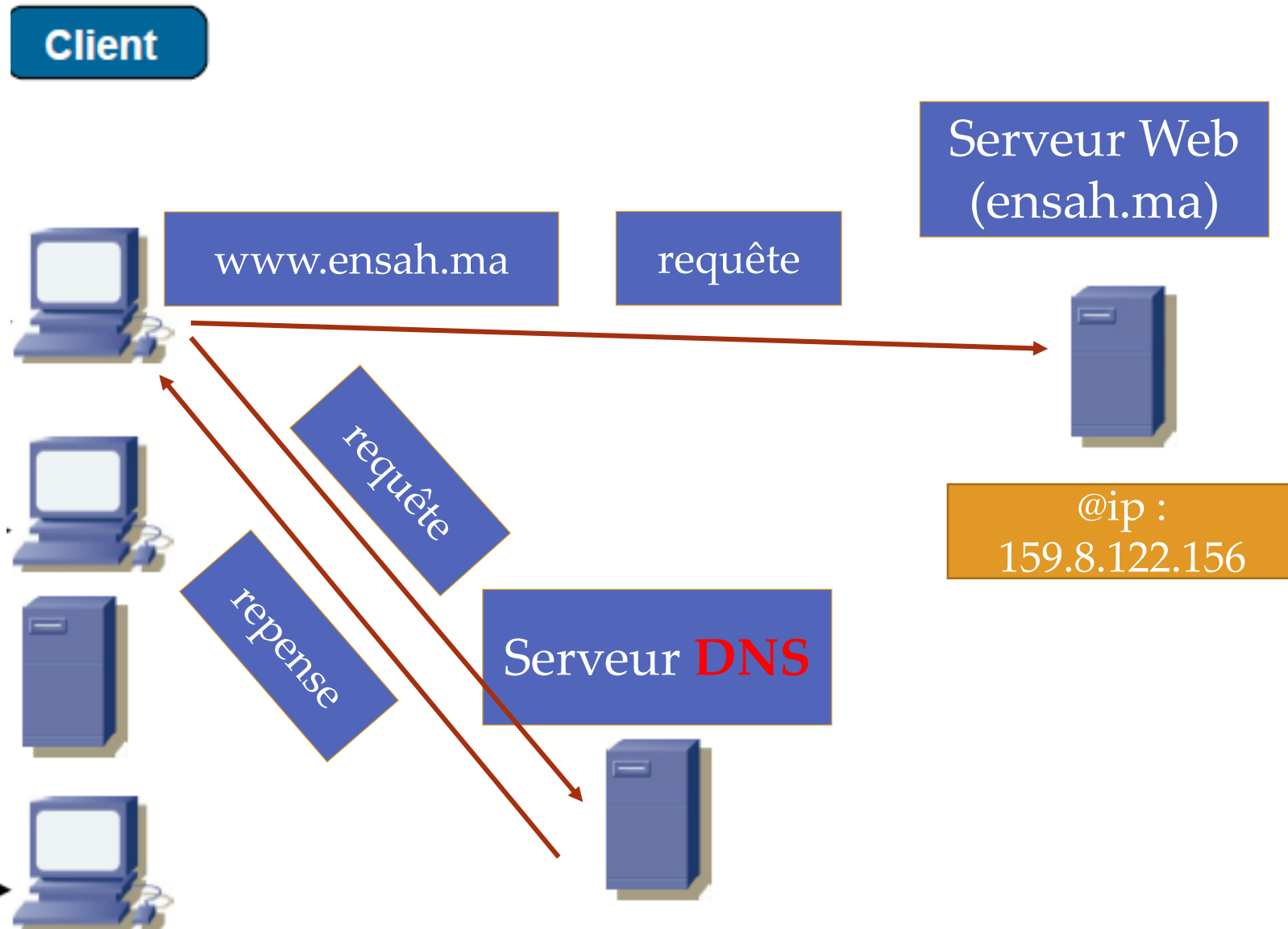
DNS

- 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 159.8.122.156.
- 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.

DNS



DNS



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(mise à jour).
- 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'en uploader/télécharger des fichiers.

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;
 - Être maintenu par un fournisseur tiers.

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 :
 - *Hébergement mutualisé* : héberger le site chez un prestataire de service qui offre un mode d'hébergement dans un environnement technique partagé (processeur, mémoire vive, espace disque, débit) par plusieurs utilisateurs.
 - *Hébergement dédié* : avoir son propre serveur d'hébergement.
 - *Hébergement Cloud* : héberger le site 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. Parmi les hébergeurs cloud, on peut citer: Google Cloud, Amazon Web Services, Microsoft Azure, etc...

Le choix de mode 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 fréquemment d'héberger ce type de site :
 - Sur son **propre machine**.
 - Chez un prestataire **d'hébergement mutualisé**.
 - Pour les sites de grandes tailles et à fort trafic :
 - **Avoir son propre serveur physique** qui possède des caractéristiques spéciales pour gérer une grande nombre de connexion à la fois. Le problème de ce type est quelque soit ses capacités physique ça reste limité par sa réalité physique, car on peut estimer apriori le nombre d'utilisation (problème de scalabilité).
 - **Allouer les ressources matérielles et logicielles** nécessaires sur un service **Cloud**. Cette solution permet d'avoir une flexibilité et elle évolue en fonction de l'utilisation. C'est la tendance pour de plus en plus de moyens et gros sites.

Chapitre II : Langage HTML

HyperText Markup Language

Plan

- HTML
 - Structure de base
 - Les liens
 - Les listes / les listes imbriquées
 - Les images
 - Les formulaires

HTML: Hyper Text Markup Language

1. **Hypertexte** : Hypertexte signifie simplement "plus que texte". Exemple un texte qui contient un lien, c'est un hypertexte. Chaque fois que vous cliquez sur un lien qui vous amène à une nouvelle page Web, vous avez cliqué sur un hypertexte. HyperText est un moyen de lier deux ou plusieurs pages Web (documents HTML) entre elles.
2. **Markup Language** : un langage de **balisage** qui est un langage informatique utilisé pour appliquer des conventions de mise en page et de formatage d'un document texte. Le langage de balisage **rend le texte plus interactif et dynamique**. Il peut transformer du texte en images, tableaux, liens, etc.
3. **Page Web** : Une page Web est un document généralement écrit en **HTML** et traduit par un **navigateur Web**. Une page Web peut être identifiée en saisissant une **URL**. Une page Web peut être de type statique ou dynamique. Avec l'aide de HTML uniquement, nous pouvons créer des pages Web **statiques**.

HTML: un langage de balisage

- **Une balise** HTML est un code entre crochets de la forme: `<code>` qui s'applique à un objet de type **texte**, **image**... et doit se terminer par `</code>` pour marquer sa fin d'affectation à l'objet concerné.
- **Exemple :**
 - pour mettre un texte en gras on utilise le code `b`. La balise ouvrante est `` et la balise fermante est ``. Donc si, dans la phrase "Je suis étudiant à l'ENSAH.", je veux mettre le mot 'ENSAH' en gras comme ceci "Je suis élève ingénieur à l'**ENSAH**.", le code HTML correspondant sera :

Je suis élève ingénieur à l'`ENSAH`.

Les outils nécessaires pour développer en HTML

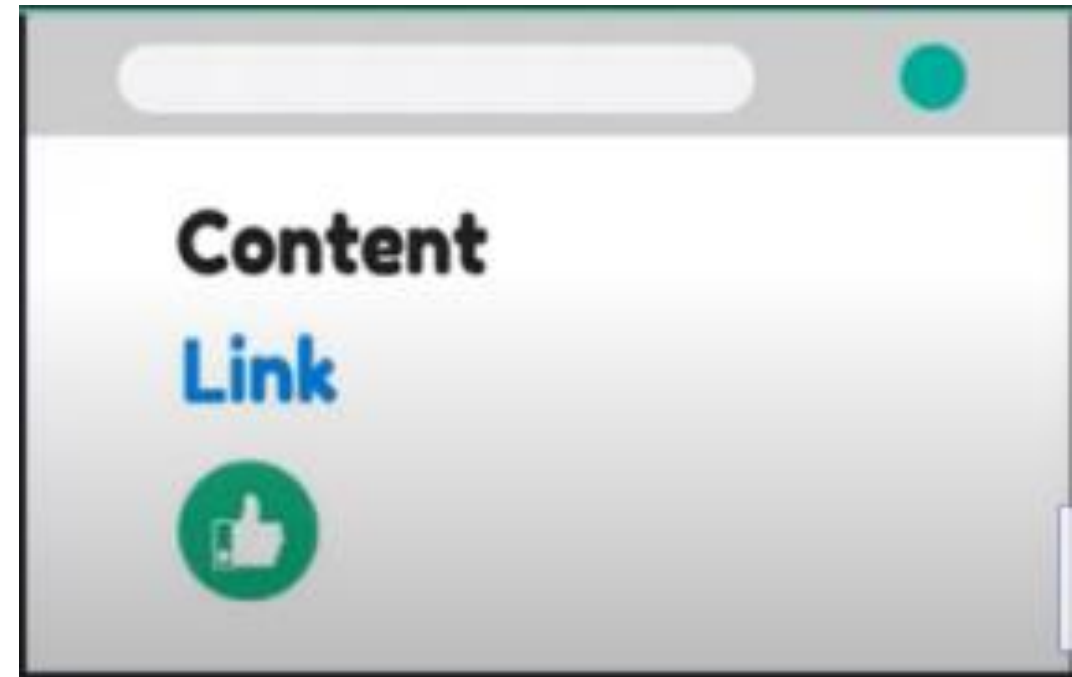
- Le développement en HTML ne demande pas assez d'outils, rien qu'un **navigateur web** ce qui est déjà livré avec un système d'exploitation et un **éditeur HTML** ou simplement un **éditeur de texte**:
 - **Un navigateur web** compatible avec **HTML5**: actuellement presque tous les navigateurs web sont compatibles avec HTML5 tels que : Google Chrome, Mozilla, Opera, Microsoft Edge...
 - **Un IDE HTML**: pour développer des pages web en HTML, il existe de nombreux IDE :
 - Visual Studio Code
 - Atom
 - Sublime Text
 - **Notepad++**

Exemple HTML

- Pour décrire un hypertexte, le langage HTML insère des **balises** dans le texte du document; un fichier portant l'extension « .html »

```
<p>content</p>  
<a >link</a>  
<img>
```

Code HTML



Son interprétation par le navigateur

Structure basique d'un document HTML

- Un document HTML (ou page Web) commence toujours par la balise de début `<!DOCTYPE html>` et se termine toujours par la balise de fin de l'élément `</html>` comme suit :

```
<!DOCTYPE html>
```

```
<head>
```

```
    <title>Titre du document</title>
```

```
</head>
```

```
<body>
```

```
...Contenu de la page HTML ...
```

```
</body>
```

```
</html>
```

Structure basique d'un document HTML

1. `<!DOCTYPE html>` : élément qui indique essentiellement à votre ordinateur qu'il s'agit d'un **document HTML**. Toutes les autres balises d'élément sont "imbriquées" dans les balises HTML de début et de fin. La page Web est ensuite subdivisée en deux sections principales qui sont le **HEAD** et **BODY**.
2. `<head> ... </head>`: définit la **section d'en-tête**. Cette section contient le **titre** du document, les informations concernant **l'auteur**, les balises méta utilisées par les moteurs de recherche...
3. `<body> ... </body>`: immédiatement après l'entête `<head>` et `</head>` vient les balises qui définissent le **corps du document** `<body>` et `</body>` et juste avant la balise de fin `html` vient la balise de fin `</body>`.

Ajouter un titre à un document HTML

- Un document HTML (ou une page web) possède généralement un titre qui apparaît dans la **barre de titre** qui s'étend tout en haut de la page Web.

```
<!DOCTYPE html>
<head>
  <title>Ma Page Web</title>
</head>
<body>
...Contenu de la page HTML ...
</body>
</html>
```



Mise en Forme d'un texte en html

- Le langage HTML offre la possibilité de mettre en forme un texte en toute fluidité grâce à des balises spécifiques. Ainsi pour mettre des titres en avant, on utilise les balises <h1>, <h2>, <h3>, <h4>, <h5> et <h6>. Chacune de ces balises correspond à un niveau de titre, allant du **titre principal** <h1> aux **titres secondaires** <h2> jusqu'au titres minuscules <h6>.

```
<h1>Titre 1</h1>
```

```
<h2>Titre 2</h2>
```

```
<h3>Titre 3</h3>
```

```
<h4>Titre 4</h4>
```

```
<h5>Titre 5</h5>
```

```
<h6>Titre 6</h6>
```



Titre 1

Titre 2

Titre 3

Titre 4

Titre 5

Titre 6

Mise en Forme d'un texte en html

- Tableau récapitulatifs des balises de mise en forme en html


Balise	Effet	Résultat
<code> Texte </code>		Texte
<code><i> Texte </i></code>		<i>Texte</i>
<code><u> Texte </u></code>		<u>Texte</u>
<code><s> Texte </s></code>		Texte
<code> Texte </code>	Famille & couleur de police	
<code><h1> Texte </h1></code>	h1(Titre), h2(sous titre) h6 (minuscule)	
<code>
 ou
</code>	revenir à la ligne	
<code><p> Paragraphe </p></code>	début et fin d'un paragraphe	
<code><sup> Text</sup></code>	Texte en exposant	AAA ^{Text}
<code><sub> indice </sub></code>	Texte en indice	AAA _{Text}
<code><blockquote> </...></code>	Texte en citation	

Mise en Forme d'un texte en html

- Exemples

Voici un exemple de texte en indice A₂₃ `₂₃
`

Voici un exemple de texte en exposant: A²³ `²³`



Voici un exemple de texte en indice A₂₃

Voici un exemple de texte en exposant: A²³

```
<font size="1"> Taille minuscule </font>
```

```
<br />
```

```
<font size="2"> Taille petite </font>
```

```
<br />
```

```
<font size="3"> Taille normale </font>
```

```
<br />
```

```
<font size="4"> Taille large </font>
```

```
<br />
```


```
<font size="5"> Grande Taille </font>
```

```
<br />
```

```
<font size="6"> Très grande Taille </font>
```

```
<br />
```

```
<font size="7"> Très très grande Taille </font>
```



Taille minuscule

Taille petite

Taille normale

Taille large

Grande Taille

Très grande Taille

Très très grande Taille

Mise en Forme d'un texte en html

- Exemples

```
Avec HTML on peut écrire en <font color="red"> rouge</font>,  
en <font color="blue"> bleu </font> en <font color="green"> vert </font>...
```



Avec HTML on peut écrire en rouge, en bleu en vert ...

Lien Hypertexte

Lien hypertexte

- Dans le but de **facilité la navigation de page en page** sur le web, le langage HTML nous offre la possibilité de créer des liens qu'on appelle **liens hypertextes**.
- Il s'agit d'un élément (image, texte, ...) qui comporte un lien vers autre objet spécifique (page interne au site, page externe, une image, un autre texte...), et dès que vous cliquez sur ce dernier, il vous envoie vers cet objet spécifique.

Contenu du cours

1. [A propos du langage HTML](#)
2. [Pourquoi apprendre le langage HTML ?](#)
3. [Les principaux avantages de l'apprentissage](#)



- Un lien hypertexte est définie par les balises :

```
<a href = " ... "> et </a>
```

Lien vers une adresse URL

- Un lien hypertexte vers une adresse URL est définie par la syntaxe :

```
<a href="adresse URL de destination">...</a>
```

- Exemple :

```
<a href="https://www.wikipedia.org/">Cliquez ici pour accéder à wikipedia</a>
```

[Cliquez ici pour accéder à wikipedia](https://www.wikipedia.org/)

- Info-bulle

```
<a href="https://www.wikipedia.org/" title="Il s'agit d'un lien vers wikipedia">Cliquez ici pour accéder à wikipedia</a>
```

[Cliquez ici pour accéder à wikipedia](https://www.wikipedia.org/)

Il s'agit d'un lien vers wikipedia



Lien vers un fichier

- Le langage HTML offre aussi la possibilité de créer un **lien hypertexte vers un fichier local** ou **distant** et ce en indiquant le chemin complet du fichier incluant son nom et son extension. Ainsi le code Html du lien vers un fichier sera de la forme :

```
<a href="chemin/nomdufichier.extension">...</a>
```

Lien vers boîte E mail

- Avec le langage HTML, on peut aussi créer des liens vers une **boîte email** via la syntaxe suivante :

```
<a href="mailto:adresse Email de destination ?">Contact</a>
```

- Exemple :

```
<a href="mailto:a.bengag@ump.ac.ma?" title="Ecrire au Prof">Ecrire au Prof</a>
```

Ecrire au Prof

Lien dans la même page

- On veut parfois créer un **lien hypertexte** vers une **autre partie** du texte **située sur la même page**.

1. Pour réaliser cela, positionnez vous à l'endroit que vous souhaitez atteindre, par exemple haut de la page Introduisez le balisage:

```
<div id="identifiant">votre texte ou vous pouvez le laisser vide</div>
```

2. Introduisez ensuite le balisage :

```
<A href="#identifiant">debut de la page</a>
```