

Architecture Web

# Table des matières

Analogie : Au restaurant .....	1
Restaurant - commande .....	1
Restaurant - les allers retours .....	1
Restaurant - requêtes / réponses .....	2
Traduit pour un site web .....	3
Le Web, c'est quoi ? .....	4
Le Web - définition .....	4
Site Web - fichiers .....	4
URL .....	4
URL - Activité .....	5
Internet .....	6
Exemples de connexions à internet .....	6
Exemples de fournisseurs d'accès à internet français .....	6
Pause .....	7
TCP/IP .....	8
Adresse IP .....	8
Numéro de port .....	8
Aparté : Modèle OSI .....	8
Aparté : Paquets .....	9
Client - Serveur .....	11
Architecture Client-Serveur - définition .....	12
Client .....	13
Client informatique - définition .....	14
Serveur .....	15
Serveur informatique - définition .....	16
Hôte .....	16
DNS - Domain Name System .....	17
Nom de Domaine .....	17
Serveur DNS - résolution DNS .....	17
Activité - Bidouiller avec les devtools .....	19
HTTP/S (protocole) .....	20
Requête HTTP .....	20
Réponse HTTP .....	20
Quelques codes de statut HTTP .....	20
Activité - Des chorégraphies HTTP en vrai .....	21
Récapitulatif - Les parties du web .....	23

# Analogie : Au restaurant

→ Miro [[https://miro.com/app/board/uXjVJvpNT7s=?share\\_link\\_id=252293520316](https://miro.com/app/board/uXjVJvpNT7s=?share_link_id=252293520316)]

## Restaurant - commande

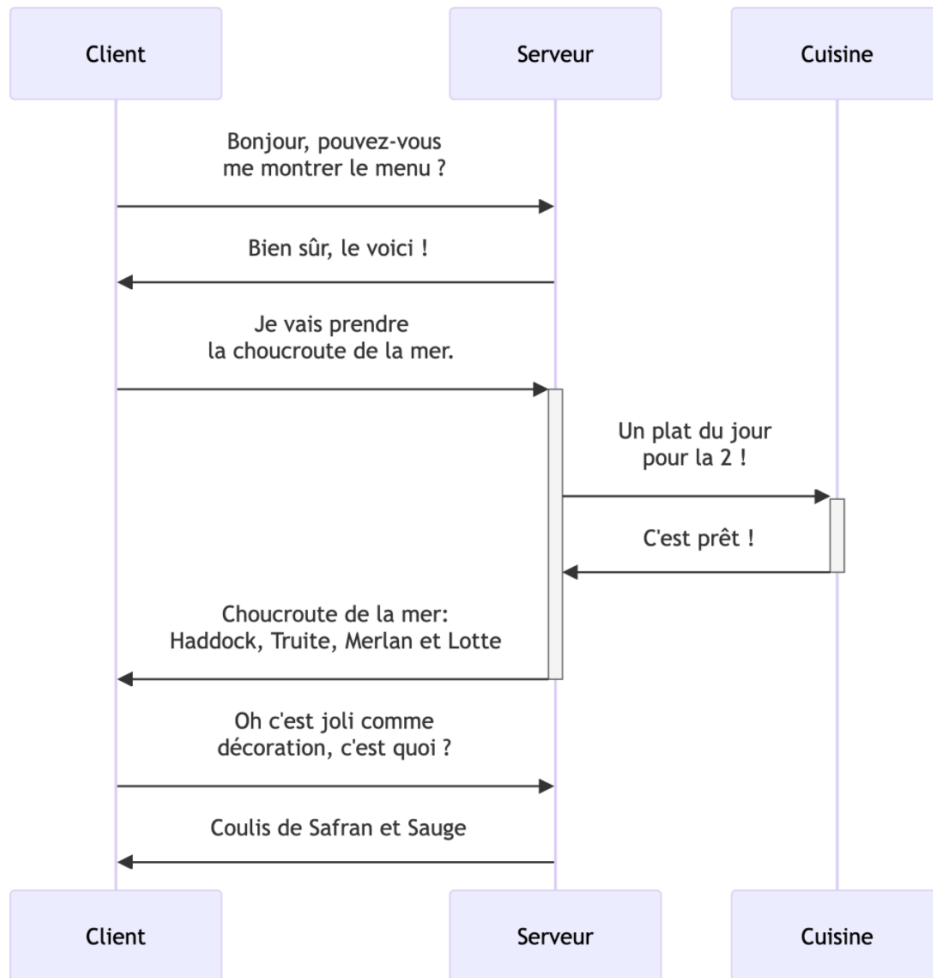
Imaginons un restaurant avec un-e client-e qui souhaite commander son plat. Il y a un serveur ainsi qu'une personne en cuisine.



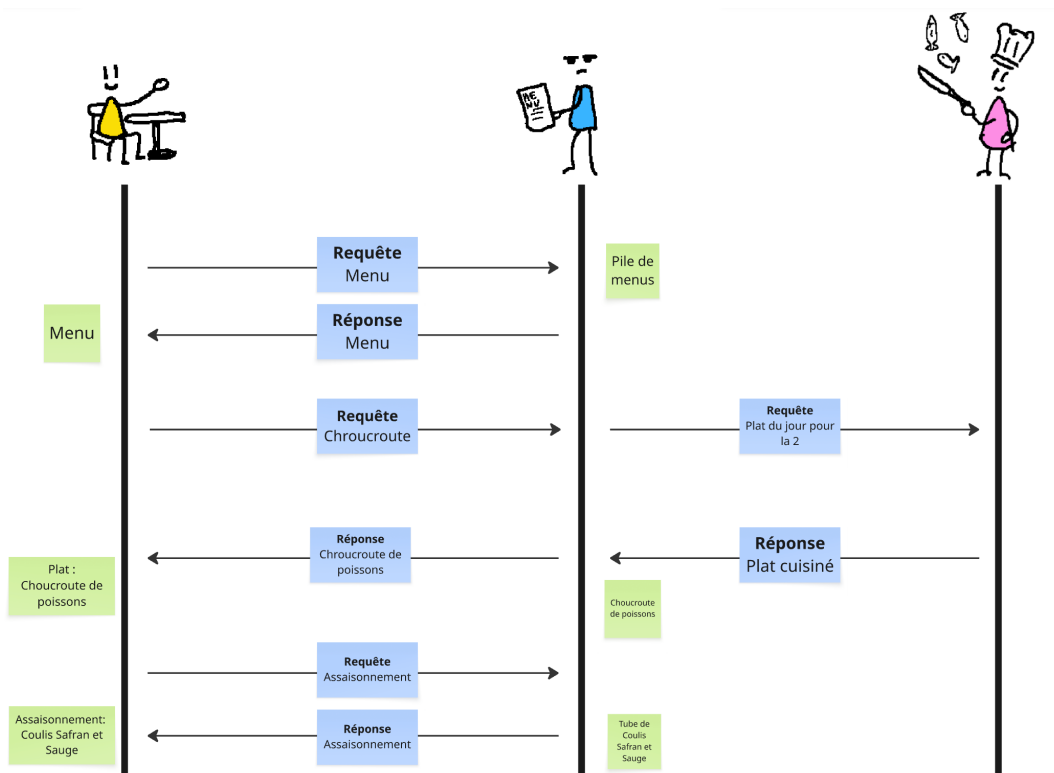
En général, le client, c'est **celui qui attend**.

## Restaurant - les allers retours

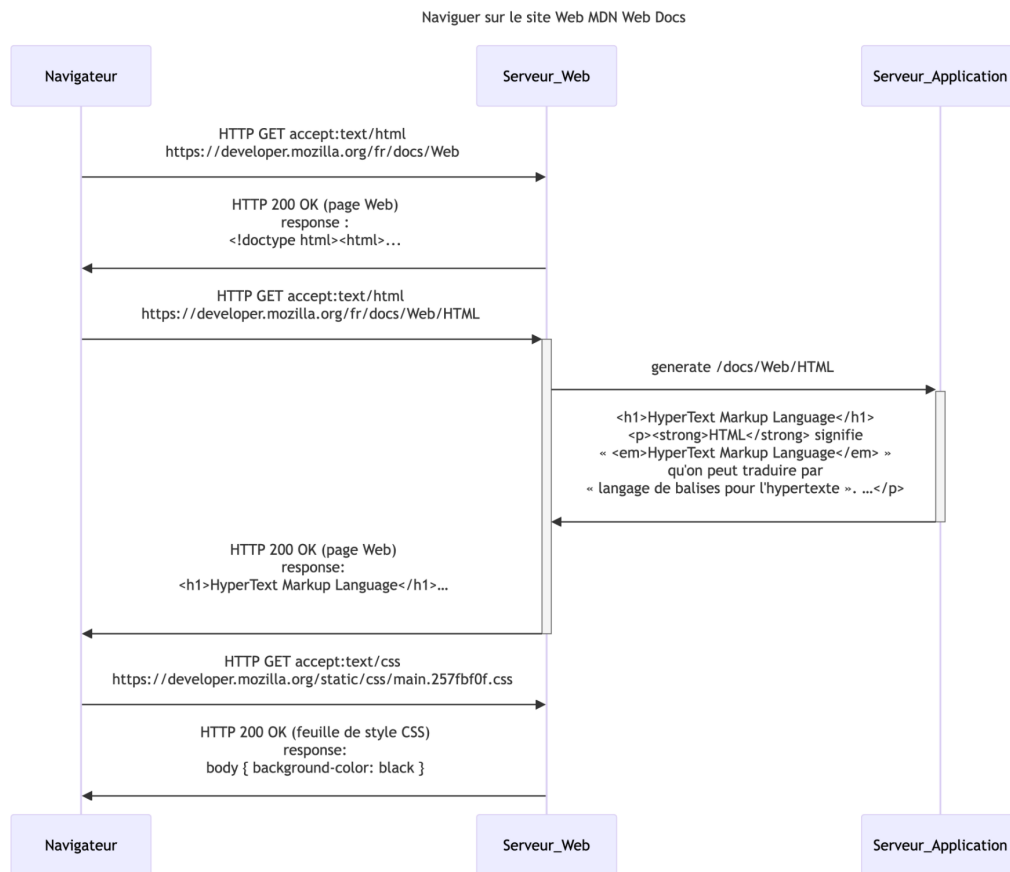
## Au restaurant "Meilleures Denrées du Nord"



## Restaurant - requêtes / réponses



# Traduit pour un site web



# Le Web, c'est quoi ?



## Le Web - définition

Le World Wide Web<sup>[1]</sup>, abrégé en WWW, W3, le Web.

C'est un système **hypertexte** public s'appuyant sur **les infrastructures d'internet**.

Permet de consulter, à l'aide d'un **navigateur Web**, des **pages Web**, regroupées en **sites Web**.



### Signification de Web

Web signifie "toile" en anglais, une métaphore pour représenter les liens hypertexte qui interconnectent les pages Web entre elles.

## Site Web - fichiers

Un site Web est composé de **fichiers**

### Code

HTML, CSS, Javascript, **interprétés** par le navigateur et **assemblés** en une **page Web** à afficher à l'utilisateur-ice

### Ressources

terme générique pour **les autres éléments** d'un site Web (ex. images, vidéos, audio...) qui ne sont **pas interprétés** par le navigateur.

## URL

### Uniform Resource Locators (URL)

Emplacements de **ressources uniques** sur **internet**

**https://developer.mozilla.org/**

Protocole

Nom de domaine

Ressource

Figure 1. Anatomie d'une URL

Exemple 1. URLs d'un site Web

- `https://developer.mozilla.org/fr/`
  - → Page HTML accueil en français (code)
- `https://developer.mozilla.org/style.css`
  - → Feuille de style CSS (code)
- `https://developer.mozilla.org/logo.png`
  - → Image du logo (ressource)

## URL - Activité

Voici l'arborescence d'un site web dont l'adresse est `https://monsite.fr`

Pour chaque fichier

1. **donne l'URL** correspondant
2. indique s'il s'agit de **code** ou d'une **ressource**.

```
.
├── index.html
├── apropos.html
├── style.css
├── img
│   └── logo.png
├── recettes
│   ├── index.html
│   └── tarfilette
│       └── index.html
```

Ex. pour le fichier `apropos.html`

- URL : `https://monsite.fr/apropos.html`
- C'est du **code**.

[1] [https://fr.wikipedia.org/wiki/World\\_Wide\\_Web](https://fr.wikipedia.org/wiki/World_Wide_Web)

# Internet

Vaste réseau **d'ordinateurs** qui communiquent les uns avec les autres.

Internet est un réseau informatique qui est **l'infrastructure sous-jacente du Web**.

Ils le font par l'intermédiaire **d'une connexion à internet** fournie par un **fournisseurs d'accès à internet** (FAI).



*Pour creuser plus loin*

MDN - Le fonctionnement d'internet [[https://developer.mozilla.org/fr/docs/Learn\\_web\\_development/Howto/Web\\_mechanics/How\\_does\\_the\\_Internet\\_work](https://developer.mozilla.org/fr/docs/Learn_web_development/Howto/Web_mechanics/How_does_the_Internet_work)]

## Exemples de connexions à internet

- 3G, 4G, 5G (connexions à internet mobile)
- Fibre, ADSL, Satellite (connexions à internet domestiques)



Selon la nature de la connexion à internet, la vitesse, le débit (la quantité d'information pouvant être envoyée), et la latence (le temps de réponse) peut varier.

## Exemples de fournisseurs d'accès à internet français

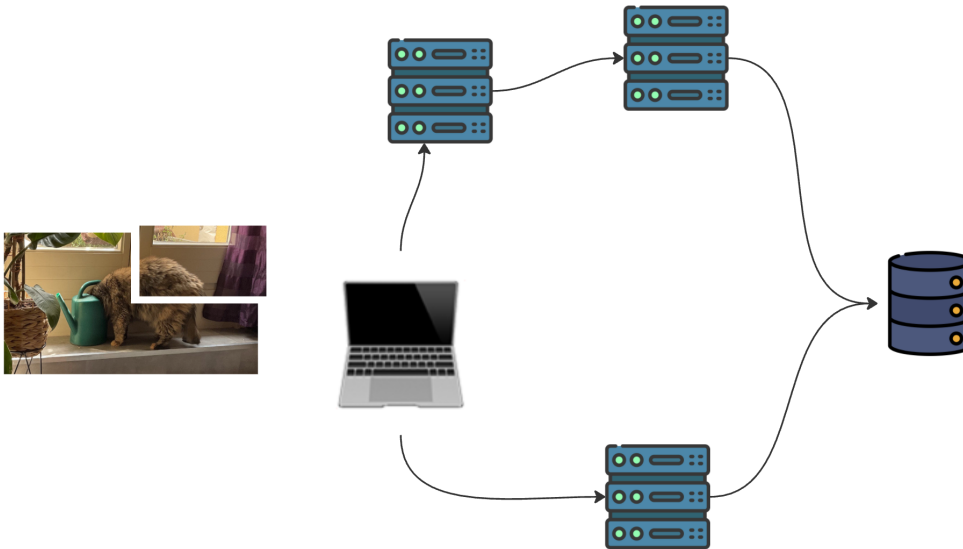
- SFR
- Free
- Orange
- Bouygues Telecom



# Pause

Miro - aide Nounou à être envoyée à destination [<https://miro.com/app/board/uXjVJvpNT7s=?moveToWidget=3458764647187324194&cot=14>]

## ***TCP / IP - PAQUETS***



# TCP/IP

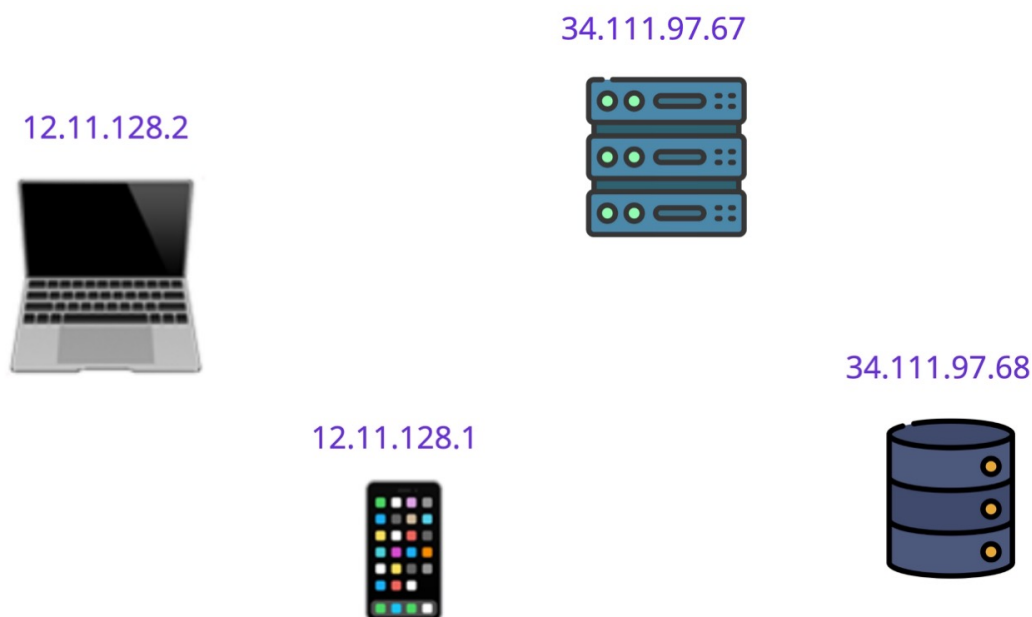
## TCP/IP

Transmission Control Protocol / Internet Protocol

- Protocoles de communication
- Comment les données circulent **sur internet**
- Les données sont **découpées en paquets**

## Adresse IP

Une **adresse IP** permet d'**identifier de façon unique** un ordinateur (hôte) sur un **réseau informatique** au niveau de la couche matérielle.



## Numéro de port

Permet à un **ordinateur hôte** de distinguer les interlocuteurs qui écoutent ( serveur) ou émettent (client) des informations sur ces ports<sup>[1]</sup>.

Exemples de ports courants

- 80 : HTTP (non sécurisé)
- 443 : HTTPS (sécurisé)
- 21 : SSH (pour se connecter à un ordinateur distant à l'aide d'un terminal)

## Aparté : Modèle OSI

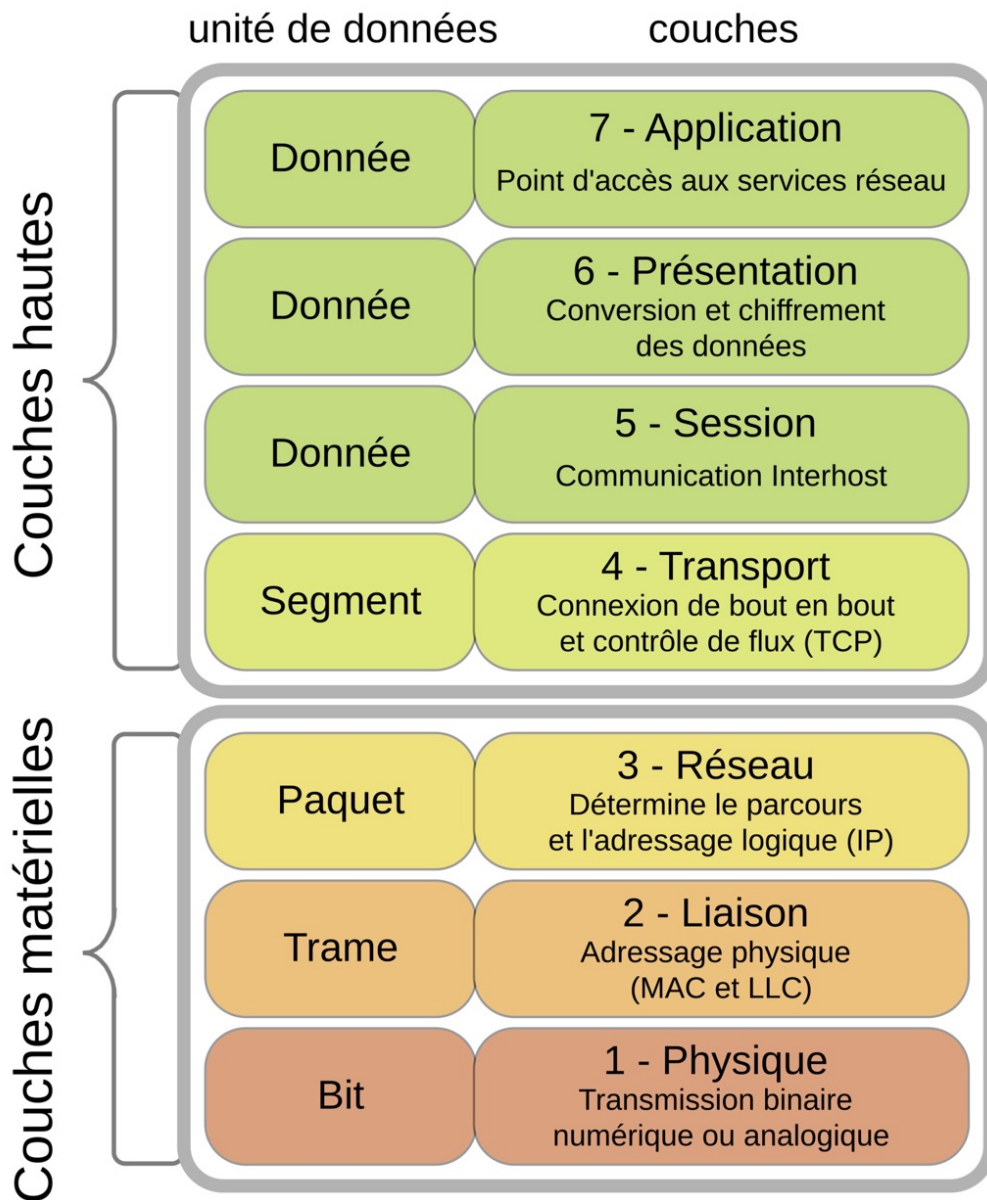


Figure 2. Diagramme du modèle OSI<sup>[2]</sup>

## Aparté : Paquets

Les **données** envoyées sur le **Web** le sont sous forme de **petits morceaux appelés paquets**.

Les paquets ont **une en-tête** contenant les informations d'ordre d'assemblage, d'origine, de destination (un peu comme une enveloppe postale) et **une charge utile** (payload) un peu comme le contenu d'une lettre.

Ils peuvent être **acheminés par différents chemins** et **dans le désordre**, pour transmettre l'information de façon plus efficace.

À l'arrivée, les données sont **reconstituées** et remises dans l'ordre grâce à leur en-tête.



### La raison

Les paquets sont parfois perdus ou corrompus, et dans ce cas, il est plus rapide et plus simple pour le client de redemander les paquets manquants plutôt que

l'ensemble du fichier.

[1] [https://fr.wikipedia.org/wiki/Port\\_\(logiciel\)](https://fr.wikipedia.org/wiki/Port_(logiciel))

[2] Par Offnfopt — Travail personnel, Domaine public, <https://commons.wikimedia.org/w/index.php?curid=39917431>

# Client - Serveur



Que savez-vous de l'architecture Client - Serveur en vous appuyant sur l'exemple du restaurant qu'on a vu en intro de ce cours?

5 min - Explique à ta/ta voisin-e

# Architecture Client-Serveur - définition

On retrouve l'**architecture Client-Serveur** très souvent sur **internet** et en particulier sur **le Web**.

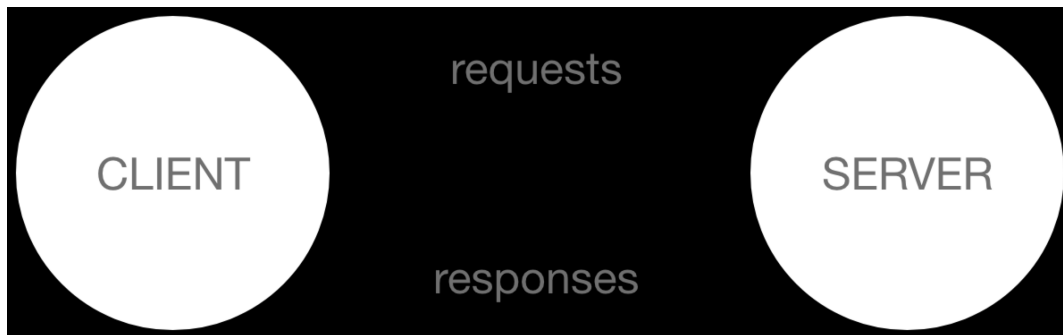


Figure 3. Schéma simplifié d'une interaction entre client et serveur<sup>[1]</sup>

[1] [https://developer.mozilla.org/fr/docs/Learn\\_web\\_development/Getting\\_started/Web\\_standards/How\\_the\\_web\\_works](https://developer.mozilla.org/fr/docs/Learn_web_development/Getting_started/Web_standards/How_the_web_works)

Client



Client

# Client informatique - définition

Un **logiciel** (ou une application) permettant d'accéder au **Web** sur un ordinateur connecté à internet.

Son rôle: enregistrer les interactions des utilisateur-ice-s et les traduire en **requêtes** envoyées à un autre ordinateur appelé un **serveur Web**.

Chaque client a une **adresse unique** appelée **adresse IP** utilisée par les autres ordinateurs **pour l'identifier**.



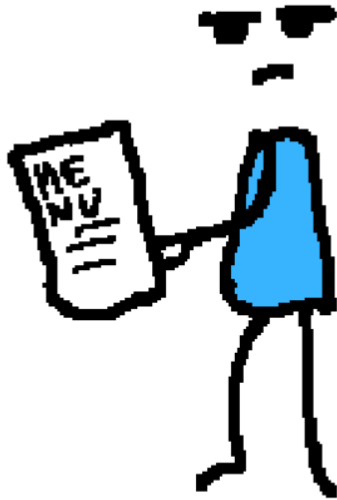
Par extension, on appelle aussi client les appareils utilisant ces logiciels.

## Exemples

- Un ordinateur connecté au Wi-Fi
- Un navigateur Web



Serveur



Serveur

# Serveur informatique - définition

Des ordinateurs qui **stockent** des **pages web**, des **sites** ou des **applications**.

Lorsqu'un **client** souhaite accéder à une **page web**, une **copie du code de la page** est téléchargée depuis le **serveur** vers la **machine cliente** pour être **interprétée par le navigateur** et affichée à l'utilisateur·rice.

## Hôte

Un ordinateur relié à un **réseau informatique** (client et/ou serveur)

Wikipedia - Hôte (informatique) [[https://fr.wikipedia.org/wiki/H%C3%B4te\\_\(informatique\)](https://fr.wikipedia.org/wiki/H%C3%B4te_(informatique))]

# DNS - Domain Name System

Comme quand tu cherches le numéro de téléphone de tes contacts de ton téléphone, le **DNS** permet de retrouver l'**adresse IP** d'un **hôte internet** à partir de son **nom de domaine**.

*Exemple 2. Résolution d'adresse IP pour des noms de domaines*

**google.com**

172.217.20.174

**mozilla.org**

35.190.14.201

**youtube.com**

142.251.39.206

## Nom de Domaine

C'est le **nom lisible** d'une adresse qui permet d'accéder à un **serveur** sur un réseau comme internet.

Lorsqu'un navigateur se rend sur une **adresse web** composée d'un **nom de domaine**, il doit résoudre à quelle **adresse IP** elle correspond.

Il fait appel à un **serveur DNS** pour récupérer l'**adresse IP** correspondant au **nom de domaine**.

*Exemple 3. Noms de domaines*

**google.com**

**mozilla.org**

**youtube.com**

## Serveur DNS - résolution DNS

Adresse initiale

**https://developer.mozilla.org/**

Protocole

Nom de domaine

Ressource

Adresse remplacée avec l'IP et le port

**https://34.111.97.67:443/**

Protocole

Adresse IP

Numéro  
de port

# Activité - Bidouiller avec les devtools

1. Ouvrir le navigateur
2. Ouvrir les outils de développeur (F12)
3. Ouvrir l'onglet de network
4. Saisir `https://developer.mozilla.org` dans la barre d'adresse
5. Observer ce qui se passe dans la zone de l'onglet "network"



Que comprends-tu de ce que tu vois ? Résume en quelques phrases ce que tu comprends.

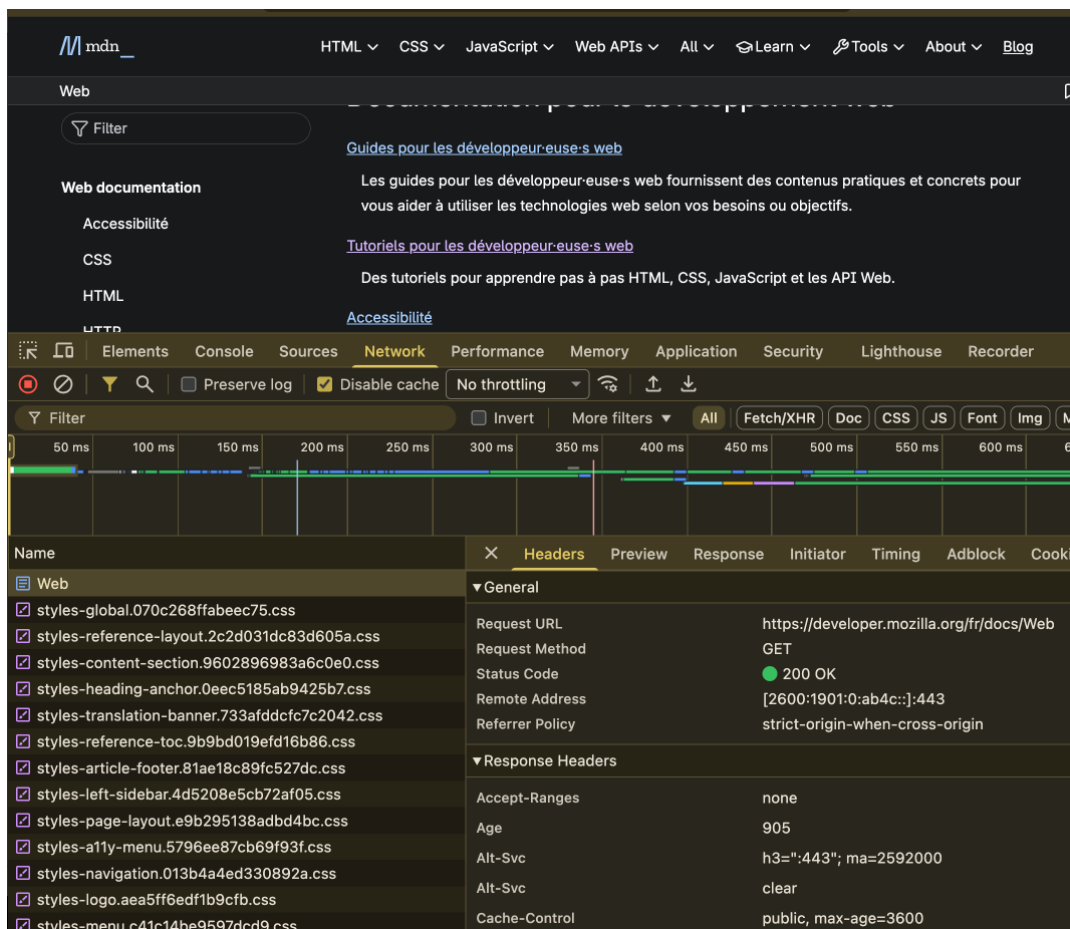


Figure 4. Analyse des communications réseau dans les outils de développeur de Chromium

# HTTP/S (protocole)

## HTTP

Hypertext Transfer Protocol (HTTP)

- Un **protocole applicatif**
- Définit **un langage** pour que les **clients** et les **serveurs** puissent communiquer

C'est comme la langue que vous utilisez pour passer commande.

## Requête HTTP

```
GET /fr/ HTTP/2
```

```
Host: developer.mozilla.org
```

## Réponse HTTP

*Réponse HTTP (simplifiée pour l'exemple)*

```
HTTP/2 200
```

```
date: Tue, 11 Feb 2025 11:13:30 GMT
expires: Tue, 11 Feb 2025 11:40:01 GMT
server: Google frontend
last-modified: Tue, 11 Feb 2025 00:49:32 GMT
ETag: "65f26b7f6463e2347f4e5a7a2adcee54"
content-length: 45227
content-type: text/html
```

```
<!doctype html> ... (les 45227 octets du HTML de la page demandée)
```

## Quelques codes de statut HTTP

- 200 : OK
- 404 : NOT FOUND
- 500 : SERVER ERROR

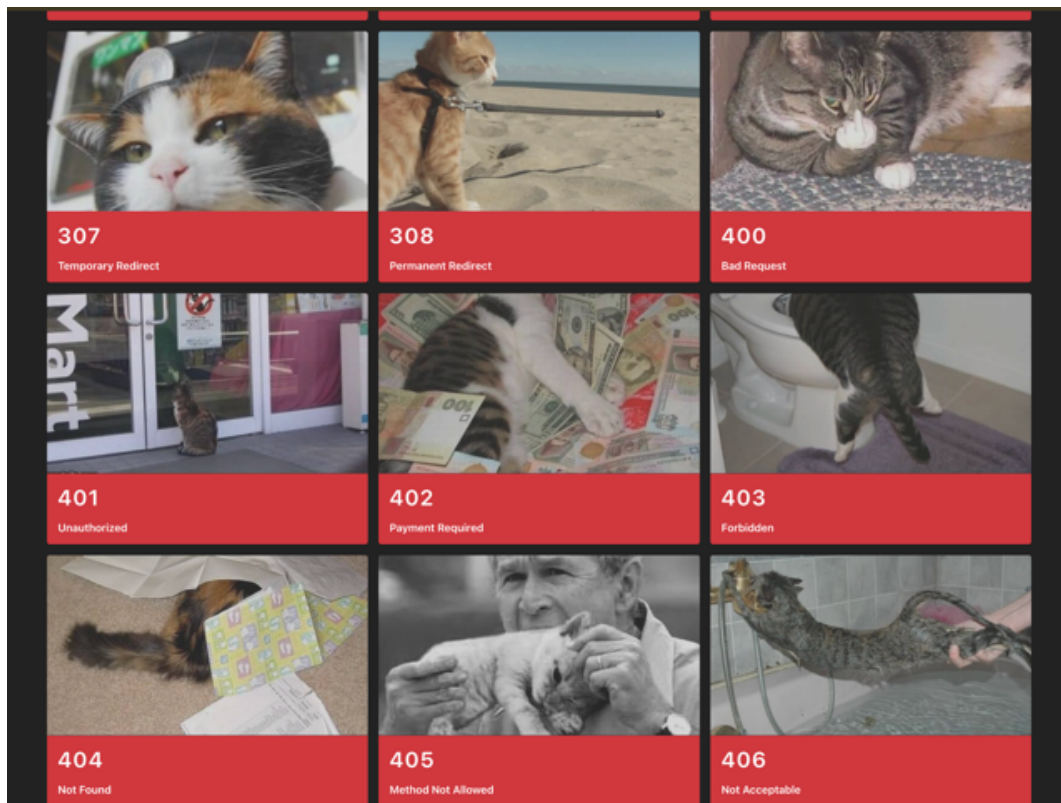


Figure 5. <http://http.cat>

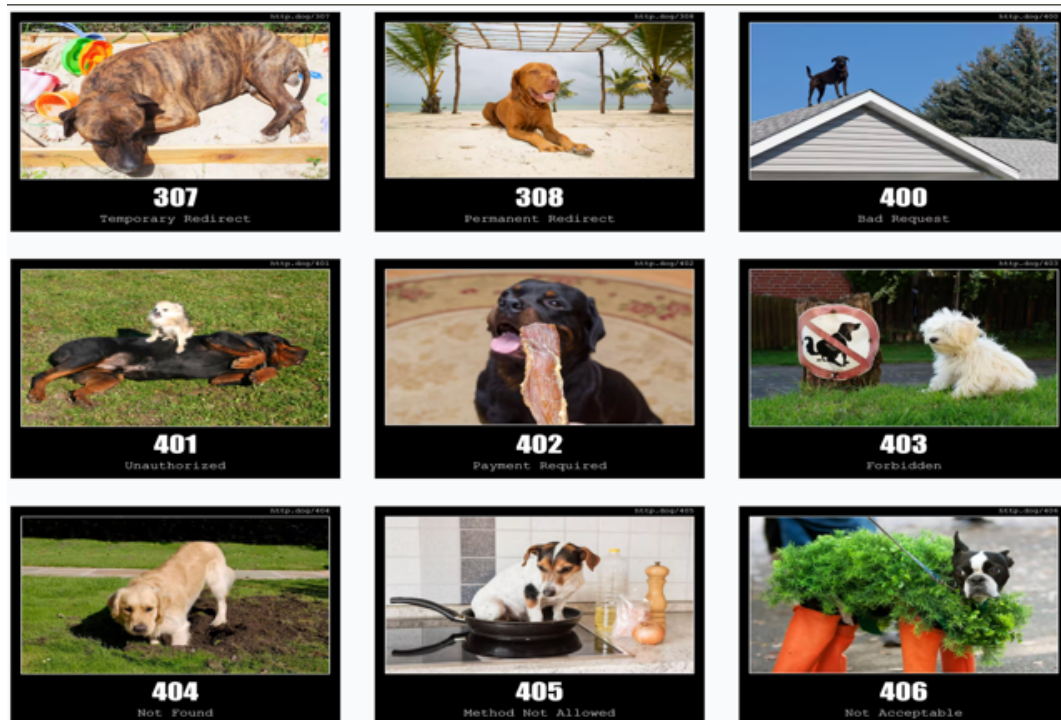
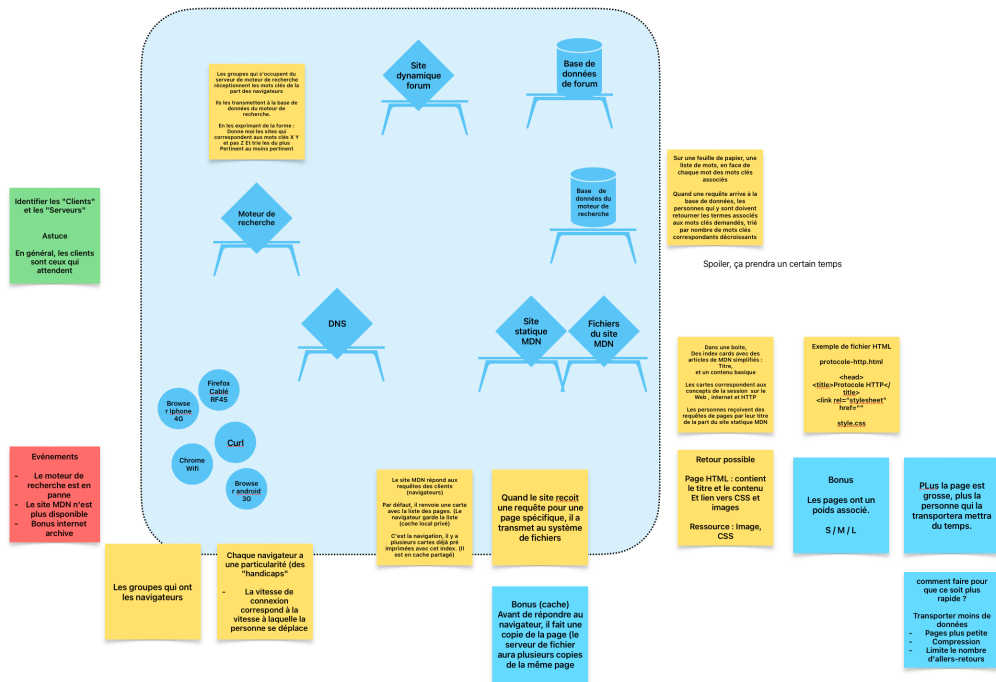


Figure 6. <http://http.dog>

## Activité - Des chorégraphies HTTP en vrai

Activité en salle avec des post-its





# Récapitulatif - Les parties du web

- Web (World Wide Web)
  - Fichiers : code, ressources
  - URL
- Internet
  - FAI - Fournisseur d'accès à internet
- TCP/IP
  - Adresse IP
  - Numéro de port
- Architecture Client-Serveur
  - Client informatique
  - Serveur informatique
  - Hôte
- DNS - Domain Name System
  - Nom de Domaine
  - Serveur DNS
- HTTP/S (protocole)
  - Requête HTTP
  - Réponse HTTP