**World Wide Web** (WWW, toile d’araignée mondiale) **:**

Système hypertexte public fonctionnant sur Internet.

**1990 : WEB 1.0 – Relier les ordinateurs**

Le WEB est constitué de pages web et d’hyperliens, les sont dits « statiques »

**2000 : WEB 2.0 – Relier les utilisateurices**

Ou WEB social, l’ère de la connexion permanente entre les individus du monde entier (réseaux sociaux, blogs, plateforme vidéo, …)

**2010 : WEB 3.0 – Relier les objets entre eux et avec les utilisateurices**

Le WEB donne accès à une masse d’informations qui sont réunies par les entreprises. Ces données sont collectées, stockées et analysée grâce à l’I.A.

Sur le plan technique, ces applications doivent répondre à certains critères :

* Indépendante : doit être utilisable sur tous les supports (pc, téléphone, imprimanté, …),
* Universelle : ne doit être sous l’emprise d’aucun système d’exploitation (maque,fabricant de logiciel ou matériel)
* Accessible à toustes : en conformité avec les directives du W3C et ouvert à diverses bases de données

**2020 – 2030 : WEB 4.0**

La quatrième génération d’Internet sera ultra-intelligente, elle s’inscrira dans un contexte dans lequel l’I.A., la réalité augmentée, l’apprentissage automatique et l’Internet des Objets (IoT) seront importantes et feront partie de notre quotidien.

**Site statique :**

Les pages sont conçues principale en HTML, ainsi qu’en CSS et JavaScript---> 1 contenu = 1 page.  
Ces pages montre des informations telles que l’a conçu l’auteur, sans jamais changer en fonction de l’utilisateurice qui visite le site.  
C’est-à-dire que tous les utilisateurs voient le même contenu. La page peu avoir énormément d’animation, des vidéos et des images, ainsi que de la musique, mais cette page sera identique à chaque utilisateurice et donc, non personnalisée. C’est en ce sens qu’un site est statique.

**Liaison entre un site statique et un site dynamique :**

Les langages serveur : Au fil de l’évolution, les sites ont eu besoin de développer des technologies, notament les langages dits « serveur », comme le PHP ou le C#, qui ont permis de faciliter la conception de sites plus élaborés.  
Le stockage de données : Le PHP et le C# ont besoin de stocker de l’informaiton, par exemple pour une inscription de membres, il faut permettre à l’utilisateurice de s’authentifier. Ces données sont enregistrées sur une base de données. Pour le PHP, il est en lien avec du MySQL et pour le C#, il s’agit de SQL Server.

**Site dynamique :**

Les pages sont conçues principalement en PHP/MySQL ou C#/SQL Server. Elles varient en fonction des informations en fonction des données apportées ou demandées par l’ utilisateurice :

* Classement des produits sur un site de vente,
* Création d’un compte personnel sur un site social
* …

Ces sites nécessitent une base de données qui sera interrogée par les langages de programmation (Backend).

**Choisir entre statique et dynamique :**

Il est important de déterminer le rôle de la page. Tout système d’interaction avec l’utilisateurice nécessite une partie dynamique. Il conviendra de réfléchir à la fréquence des mises à jours. Si un contenu n’est modifié qu’1x par an, il n’est peut-être pas nécessaire de créer des pages dynamiques.

**Avantages et inconvénients :**

Site dynamique : l’utilisation des pages dynamiques est plus souple pour l’évolution du site mais nécessite plus de temps e travail et des compétences poussées en programmation.

Site statique : ces pages sont dites « figées » mais plus simples à réaliser.

**En bref :**

Un site statique est un ensemble de pages préformées en .html, stockées sur un serveur WEB. Aucun traitement n’est effectué côté serveur. Les pages sont conçues en HTML, CSS, JavaScript.

Un site dynamique est généralement associé à PHP – MySQL ou C# - SQL Serveur. Le contenu du site est stocké dans une base de données (MySQL par exemple), et est dissocié du frontend.  
Le serveur WEB reçoit une requête HTTP ou HTTPS émise par l’utilisateurice, qui interroge la base de données (MySQL), puis construit une page dynamique de résultat qu’il reconstitue à l’utilisateurice via le navigateur.

A diagram of a server

Description automatically generated

**Application WEB :**

Est un logiciel applicatif qui imite un site web et affiche une interface homme – machine dans un navigateur web.

Le logiciel est hébergé par un serveur web. Il est exécuté de façon partagée par les ordinateurs serveur et client (utilisateurice).

C’est donc une application manipulable directement en ligne grâce à un navigateur web et qui ne nécessite pas d’installation côté client (utilisateurice), contrairement aux applications mobiles.

Les usages populaires sont :

* Un moteur de recherche
* Une authentification
* Un jeu par navigateur
* Une messagerie instantanée

**Développeureuse full stack :**

Informaticien.ne capable de réaliser des tâches à n’importe quel niveau technique autant en frontend, backend qu’en base de données ainsi qu’en gestion de projet, en bref, tout ce qui constitue l’élaboration d’une application web.

**Serveur WEB :**

Est un ordinateur qui stocke les fichiers qui composent un site web (exemple : Apache, IIS, Nginx)

**SSR (server-side rendering / rendu côté serveur) :**

Consiste à créer les pages côté serveur pour les envoyer toutes faites au navigateur. Permet de fournir en temps réel les pages à l’utilisateurice. Optimisation du référencement.

**CSR (client-side rendering / rendu côté client) :**

Consiste à créer les pages côté navigateur. Même si elles sont plus fluides, elles possèdent un mauvais référencement et un temps de chargement plus long car dépendant de la connexion de l’utilisateurice.

Analyse d’un site dynamique :

<https://www.julienvanroy.fr/>

**Serveur WEB :**

Nuxt.js : est un framework SSR (server-side rendering) open-source reposant sur le framework Vue.js

**Frontend :**

Nuxt.js : framework reposant sur Vue.js

Vue.js : framework JavaScript open-source pour créer des interfaces utilisateurs et des SPA (single-page application / applications web monopages).

**Backend / langage de programmation :**

Node.js : Environnement de logiciel open-source en JavaScript

**Sécurité :**

HSTS : HTTP Strict Transport Security, permet à un serveur web de déclarer à l’utilisateurice d’interagir avec lui en utilisant une connexion sécurisée

**Librairies :**

Scrollreveal : Librairie JavaScript pour animer des éléments lorsque l’utilisateurice scroll.

**Autres :**

PWA (Progressive Wep App / application web progressive) : Elle se consulte comme un site web classique, depuis une URL sécurisée mais permet une expérience utilisateur similaire à celle d’une application mobile.