

# PROGRAMMATION WEB

Pr. OMARI Kamal

ENSA KHOURIBGA

21 février 2025

# Chapitre 1 : Introduction à la programmation Orientée Objet et web

- 1 Introduction
  - L'histoire de l'Internet
  - Introduction au Web
- 2 Word Wide Web (WWW)
  - Définition
  - les concepts clés
  - Évolution du Web
  - Architecture du Web
- 3 Architecture Client-Serveur
  - Définition
  - Modèles Client-Serveur
- 4 Ergonomie du Web
  - Définition
  - Les aspects de l'ergonomie du web
- 5 Les langages de programmation
  - Définition
  - Langages côté client
  - Langages côté serveur
- 6 Exercice
  - Exercice de résumé et de compréhension

# Objectifs du cours

Ce cours vise à fournir une compréhension approfondie des concepts de la programmation orientée objet (POO) et leur application dans le développement web. Nous allons explorer comment intégrer efficacement HTML, CSS, PHP, et JavaScript dans des projets web en utilisant des approches orientées objet pour structurer et organiser le code de manière modulaire et maintenable.

# Objectifs du cours

- **HTML et CSS** : Compréhension des bases du développement front-end pour créer des interfaces utilisateur (UI) attractives et fonctionnelles.
- **PHP orienté objet** : Utilisation de PHP pour le back-end, en mettant l'accent sur la POO pour structurer des applications web robustes.
- **JavaScript orienté objet** : Introduction aux objets en JavaScript et à la gestion des interactions dynamiques côté client.
- **Projet pratique** : Intégration de ces technologies dans un projet complet, démontrant l'usage de la POO dans le développement web.

# L'histoire de l'Internet

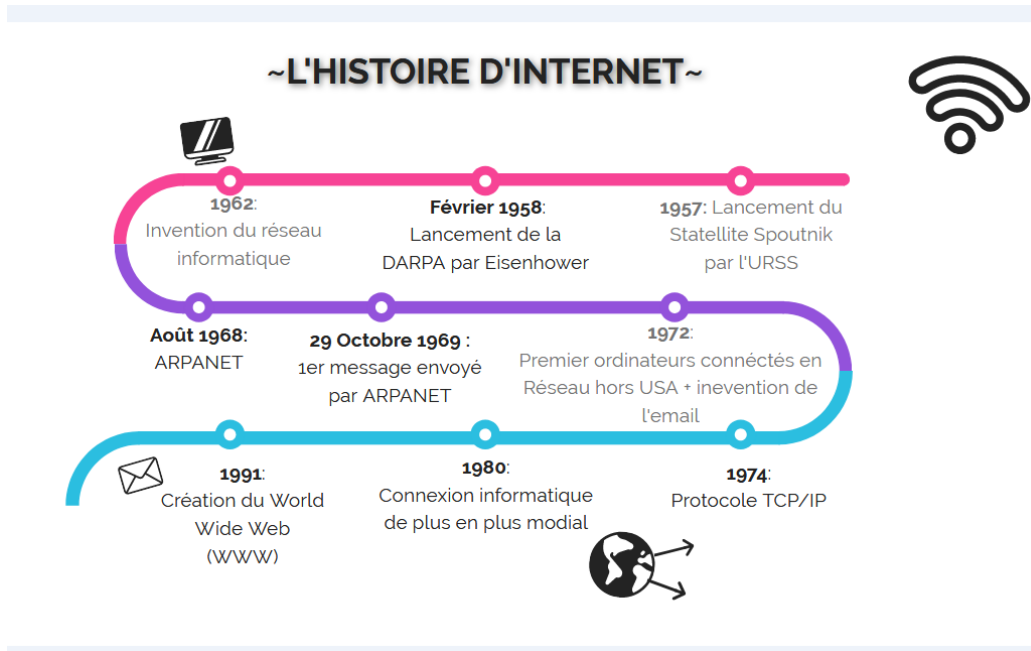


Figure 1 – L'histoire de l'Internet

# Définition

- Internet est le réseau informatique mondial accessible au public.
- Il fournit plusieurs services, tels que le courrier électronique, la messagerie instantanée, le pair-à-pair, et le World Wide Web.
- Chaque ordinateur sur Internet a une adresse IP, par exemple, 134.59.130.x.
- Les ordinateurs sont souvent identifiés par un nom de domaine, qui est plus facile à retenir pour les utilisateurs. Par exemple, le nom de domaine `www.google.com` correspond à l'adresse IP 216.58.215.46.
- La conversion entre les adresses IP et les noms de domaine se fait grâce au DNS (Domain Name System).
- Les informations sont transmises sur Internet à l'aide d'un ensemble standardisé de protocoles de transfert de données.

Il existe de nombreux protocoles, notamment :

- **TCP/IP** : est un ensemble de règles qui permet la communication sur Internet et sur les réseaux informatiques.
- **HTTP** : utilisé pour la navigation sur les pages web dans votre navigateur.
- **POP/IMAP/SMTP** : protocoles pour l'envoi et la réception d'e-mails.
- **BitTorrent** : utilisé pour l'échange de fichiers de pair à pair.
- **FTP** : protocole pour le transfert de fichiers.
- **IRC** : utilisé pour la messagerie instantanée.

# Introduction au Web

Le Web est sans doute l'une des technologies les plus importantes du 21<sup>e</sup> siècle. Bien que sa structure, son fonctionnement et ses usages aient changé au fil du temps, il est évident que cette évolution a aussi transformé nos façons de faire des affaires et de communiquer. Dans ce chapitre, nous allons découvrir du Web, ainsi que les étapes de son développement, pour mieux comprendre comment il a évolué et ce qu'il représente aujourd'hui.



# Introduction au Web

- Les technologies web sont un ensemble de normes et de langages qui composent et alimentent le World Wide Web, communément abrégé en "Web". Ces technologies permettent la création, le partage et l'accès à des informations via Internet.
- Créé en 1990 pour faciliter le partage d'informations, le Web s'est rapidement transformé en une plateforme incontournable, où de nouvelles technologies sont constamment développées et intégrées.

# Introduction au Web

Les fondements de ces technologies reposent sur deux éléments clés :

- Le protocole HTTP (Hypertext Transfer Protocol), normalisé par l'IETF (Internet Engineering Task Force), qui permet la communication entre les serveurs web et les navigateurs.
- Le langage HTML (Hypertext Markup Language), standardisé par le W3C (World Wide Web Consortium), qui définit la structure des pages web.

# Introduction au Web

**Aujourd'hui, le Web est devenu l'un des protocoles de communication les plus utilisés sur Internet, grâce à sa simplicité et son adaptabilité.**

# Définition

Le World Wide Web (WWW) est un système global d'information hypertextuel qui permet de naviguer entre des documents et des ressources interconnectés sur Internet. Il utilise le protocole **HTTP** pour transférer les pages web et des **URLs** pour localiser ces ressources. Inventé en 1989 par Tim Berners-Lee, le Web est devenu essentiel pour partager et diffuser des informations à travers le monde.

Cette définition met l'accent sur l'aspect technique et fonctionnel du Web en tant que composant clé de l'Internet.

Voici les concepts clés du World Wide Web (WWW) :

- **Système hypertextuel global** : Organisation des informations sous forme de liens permettant la navigation entre différents documents.
- **Internet** : Le Web fonctionne sur Internet, qui fournit l'infrastructure de connexion.
- **Protocole HTTP** : Utilisé pour transférer des pages web.
- **URL (Uniform Resource Locator)** : Adresse unique permettant de localiser une ressource sur le Web.
- **Documents et ressources interconnectés** : Le Web permet de lier différents types de contenu multimédia (texte, images, vidéos, etc.).

# Évolution du Web

Pour comprendre le Web actuel, il est important de connaître son évolution, car chaque nouvelle version découle des limites de la version précédente (techniques, usages, etc.). De plus, chaque nouvelle version vient la compléter, et non la remplacer.

# Évolution du Web

L'évolution du Web peut être divisée en plusieurs phases clés :

- **Web 1.0 (années 1990) :** Le Web statique : pages web principalement en lecture seule, avec peu d'interaction utilisateur. Les contenus étaient créés par des développeurs, et les utilisateurs se limitaient à la consultation d'informations (textes, images).
- **Web 2.0 (années 2000) :** Le Web participatif et interactif : permet aux utilisateurs de contribuer en créant des contenus (blogs, réseaux sociaux, wikis). Les technologies comme AJAX et JavaScript ont amélioré l'interactivité, ouvrant la voie aux plateformes collaboratives et aux communautés en ligne.

# Évolution du Web

- **Web 3.0 (années 2010) :** Le Web sémantique : introduction de l'intelligence artificielle et du traitement automatique des données pour comprendre et interpréter le contenu. L'objectif est de permettre aux machines de mieux comprendre les informations, facilitant ainsi la recherche, la personnalisation et l'interconnexion des données.
- **Web 4.0 (futur) :** Le Web intelligent et omniprésent : un Web hautement intégré dans la vie quotidienne, combinant intelligence artificielle, objets connectés, réalité augmentée et virtuelle, et des interactions de plus en plus naturelles entre humains et machines.



## En résumé

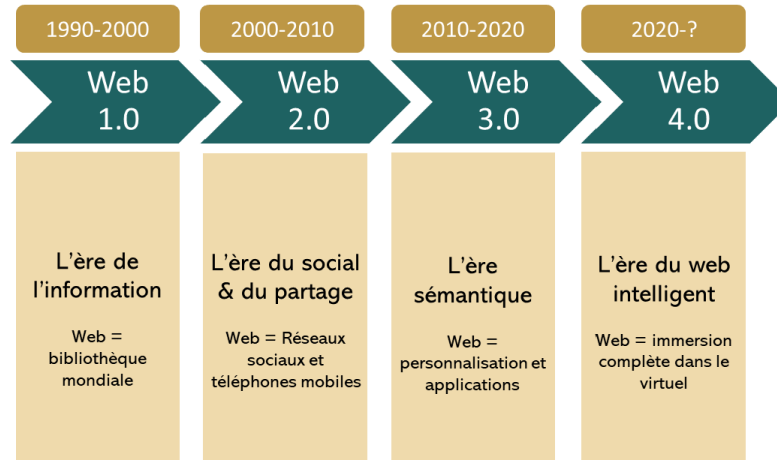


Figure 2 – Évolution du Web

Cette évolution montre comment le Web est passé d'une simple plateforme de consultation à un environnement interactif et intelligent.

# Architecture du Web

L'architecture du Web repose sur plusieurs couches et composants clés qui permettent le fonctionnement et l'accès aux ressources en ligne. Voici les principaux éléments :

- **Serveurs web** : Hébergent et stockent les pages web, les applications, et les données. Ils répondent aux requêtes des navigateurs en envoyant les fichiers demandés.
- **Navigateurs web** : Logiciels utilisés par les utilisateurs pour accéder aux pages web. Ils envoient des requêtes HTTP aux serveurs et interprètent les réponses (HTML, CSS, JavaScript) pour afficher le contenu.
- **Protocole HTTP/HTTPS** : Le HTTP (Hypertext Transfer Protocol) permet la communication entre les clients et les serveurs. Le HTTPS ajoute une couche de sécurité avec un chiffrement via SSL/TLS.

# Architecture du Web

- **Adresses URL (Uniform Resource Locator) :** Les URLs identifient et localisent les ressources sur le Web. Elles contiennent le nom de domaine, le chemin vers une page spécifique et parfois des paramètres supplémentaires.
- **DNS (Domain Name System) :** Système qui traduit les noms de domaine lisibles par les humains (comme `www.exemple.com`) en adresses IP, que les machines utilisent pour localiser les serveurs.

# Architecture du Web

- **Langages du Web :**

**Le HTML (Hypertext Markup Language)** structure les pages web.

**Le CSS (Cascading Style Sheets)** est utilisé pour la mise en forme et la présentation.

**Le JavaScript** permet l'interactivité et la dynamique des pages.

**Le PHP (Hypertext Preprocessor)** est un langage de script côté serveur, utilisé pour générer des pages web dynamiques. Avec l'approche orientée objet (POO).

- **Bases de données :** Stockent et gèrent les données dynamiques pour les sites web et les applications, comme les informations des utilisateurs, les articles de blog, etc.

# Architecture du Web

- **APIs (Application Programming Interfaces) :**

Permettent l'intégration et la communication entre différentes applications et services, facilitant le partage de données et la fonctionnalité sur le Web.

Cette architecture modulaire et en couches permet de créer, stocker, diffuser et afficher des informations sur le Web, assurant une interaction fluide entre les utilisateurs et les services en ligne.

# Internet VS Web

**Internet** est le réseau mondial d'ordinateurs interconnectés qui permet de transporter des données à travers divers protocoles comme TCP/IP. Il prend en charge de nombreux services comme le courrier électronique, la messagerie instantanée, et les transferts de fichiers.

**Le Web** (World Wide Web) est l'un des services d'Internet, permettant d'accéder à des pages web via le protocole HTTP. Il utilise des adresses appelées URL pour localiser des ressources en ligne et se compose de documents multimédias (texte, images, vidéos) accessibles grâce aux navigateurs web.

## En résumé

**En résumé, Internet est l'infrastructure, tandis que le Web est un service qui fonctionne sur cette infrastructure pour permettre la navigation sur des sites web.**

# Définition

L'architecture client-serveur désigne la structure générale d'une application où des clients et des serveurs interagissent. Elle décrit comment les composants d'une application sont organisés et interconnectés.

- **Client :**

**Rôle :** Le client est l'interface utilisateur qui envoie des requêtes au serveur. Il peut s'agir d'un navigateur web, d'une application mobile, ou d'un logiciel de bureau.

**Fonctionnalités :** Reçoit et affiche les données envoyées par le serveur. Permet à l'utilisateur d'interagir avec l'application (saisie de données, navigation, etc.). Effectue parfois des traitements locaux, selon qu'il s'agisse d'un client lourd ou léger.



# Définition

- **Serveur :**

**Rôle :** Le serveur est responsable du traitement des requêtes des clients, de l'exécution de la logique métier, et de la gestion des données.

**Fonctionnalités :** Traite les requêtes reçues (calculs, accès aux bases de données, etc.). Renvoie les résultats ou les ressources demandées (pages web, données). Peut héberger plusieurs applications ou services.

## En résumé

L'architecture client-serveur est essentielle pour le développement d'applications modernes, permettant une interaction efficace entre utilisateurs et systèmes distants. Elle s'adapte à différents besoins et types d'applications, des simples sites web aux systèmes d'entreprise complexes.

# Définition

Les modèles de client-serveur désignent des approches spécifiques au sein de l'architecture client-serveur. Ils se concentrent sur la manière dont les interactions entre clients et serveurs sont organisées et comment le traitement est réparti.

# Types

- Comprend des modèles tels que le client lourd (thick client), le client léger (thin client), le modèle N-tiers, les web services, et les microservices.
- Chaque modèle a des caractéristiques et des scénarios d'utilisation différents, par exemple, la répartition du traitement entre le client et le serveur.

## En résumé

Le modèle client-serveur est fondamental dans l'architecture des applications web, permettant une interaction efficace entre les utilisateurs et les systèmes distants. Chaque modèle a ses propres avantages et inconvénients, et le choix dépend souvent des exigences spécifiques du projet.

# Définition

L'ergonomie du web désigne l'ensemble des principes et pratiques visant à améliorer l'expérience utilisateur lors de la navigation sur un site web. Elle concerne la facilité d'utilisation, l'efficacité et la satisfaction de l'utilisateur lorsqu'il interagit avec une interface web.

# Les aspects de l'ergonomie du web

Voici les principaux aspects de l'ergonomie du web :

- **Accessibilité** : Le site doit être accessible à tous les utilisateurs, y compris ceux ayant des handicaps (visuels, auditifs, moteurs). Cela inclut l'usage de textes alternatifs pour les images, une navigation facile avec le clavier, et la compatibilité avec les lecteurs d'écran.
- **Simplicité de la navigation** : Les utilisateurs doivent pouvoir trouver rapidement ce qu'ils recherchent. Cela implique une organisation claire du contenu, des menus bien structurés, des liens explicites et un plan de site pour aider à la navigation.

# Les aspects de l'ergonomie du web

Voici les principaux aspects de l'ergonomie du web :

- **Temps de chargement** : Le temps de chargement des pages web doit être minimal pour éviter que les utilisateurs ne quittent le site. Cela implique une optimisation des images, des scripts et des éléments multimédia pour garantir des performances rapides.
- **Conception responsive** : L'ergonomie inclut la capacité du site à s'adapter à différents types d'appareils (ordinateurs, tablettes, smartphones). Un design responsive permet de garantir une expérience cohérente quel que soit l'appareil utilisé.



# Les aspects de l'ergonomie du web

Voici les principaux aspects de l'ergonomie du web :

- **Clarté visuelle** : La lisibilité est essentielle. Le contraste entre le texte et l'arrière-plan doit être suffisant, les polices doivent être facilement lisibles, et la hiérarchie des informations (titres, sous-titres, paragraphes) doit être évidente.
- **Cohérence** : Les éléments du site (boutons, menus, typographies, couleurs) doivent être cohérents sur l'ensemble des pages. Cela permet aux utilisateurs de naviguer sans confusion et avec un sentiment de continuité.

# Les aspects de l'ergonomie du web

Voici les principaux aspects de l'ergonomie du web :

- **Feedback utilisateur** : Il est important que le site fournisse des retours immédiats aux actions de l'utilisateur, comme les confirmations lors de l'envoi d'un formulaire ou les messages d'erreur en cas de problème. Ce feedback améliore la clarté et réduit la frustration.
- **Minimalisme et focus sur le contenu** : Une interface épurée, sans distractions inutiles, aide les utilisateurs à se concentrer sur l'essentiel, c'est-à-dire le contenu et les fonctionnalités principales du site.

## En résumé

L'ergonomie du web vise à rendre l'expérience utilisateur intuitive, agréable et efficace. En suivant ces principes, un site web devient plus convivial, augmentant ainsi la satisfaction et la fidélité des utilisateurs.

# Définition

Les langages de programmation sont utilisés pour créer des applications web dynamiques, interactives et performantes. Ils permettent de concevoir des sites web, des services en ligne, ainsi que des logiciels.

# Langages côté client

Les langages côté client sont des langages de programmation qui s'exécutent directement dans le navigateur de l'utilisateur, sans avoir besoin de communication avec un serveur. Ils sont principalement utilisés pour rendre les pages web interactives, dynamiques, et visuellement attrayantes. Les principaux langages côté client sont :

- **HTML** pour structurer le contenu.
- **CSS** pour définir l'apparence et la mise en page.
- **JavaScript** pour ajouter des fonctionnalités interactives et modifier le contenu sans recharger la page.

Ces langages permettent de créer des interfaces utilisateur rapides et réactives.

# Langages côté serveur

Les langages côté serveur sont des langages de programmation qui s'exécutent sur un serveur web avant que les pages ne soient envoyées au navigateur de l'utilisateur. Ils sont utilisés pour gérer des tâches complexes comme l'accès à une base de données, la gestion des utilisateurs, ou la génération dynamique de contenu en fonction des interactions de l'utilisateur. Les principaux langages côté serveur sont :

- **PHP** pour générer des pages web dynamiques.
- **Python** utilisé avec des frameworks comme Django ou Flask.
- **Java** via JSP (Java Server Pages) pour les applications web.

Ces langages permettent de créer des interfaces utilisateur rapides et réactives.

## Exercice de résumé et de compréhension

**Veillez définir les termes suivants en quelques lignes :  
Application Web, Application mobile, Navigateur Web  
et Site Web.**

Voici la définition de chaque terme en quelques lignes :

## Correction

**Application Web :** Une application accessible via un navigateur web, hébergée sur un serveur distant, et qui utilise des technologies comme HTML, CSS, et JavaScript.

Contrairement aux applications installées localement, elle fonctionne entièrement en ligne et est accessible depuis n'importe quel appareil connecté à Internet.

**Application mobile :** Un logiciel conçu pour fonctionner sur des appareils mobiles (smartphones, tablettes) et optimisé pour leurs écrans et capacités. Elle peut être téléchargée et installée via des magasins d'applications comme Google Play ou l'App Store, et fonctionne en ligne ou hors ligne.



Voici la définition de chaque terme en quelques lignes :

## Correction

**Navigateur Web** : Un logiciel qui permet d'accéder et de naviguer sur le Web. Il interprète les fichiers HTML pour afficher des pages web. Les navigateurs les plus connus incluent Google Chrome, Firefox, Safari et Microsoft Edge.

**Site Web** : Un ensemble de pages web interconnectées et hébergées sur un serveur, accessibles via un navigateur web en utilisant une URL (Uniform Resource Locator). Chaque site web est conçu pour fournir des informations ou des services spécifiques, et peut contenir du texte, des images, des vidéos et d'autres éléments multimédias.