

Développeur
Web

Architecture & Développement Web

Date



SOMMAIRE

- 1 Tour d'histoire
- 2 Composition du Web
 - Les RFC
 - Exemples de briques logicielles
- 3 Composition d'une application Web
 - Structure globale
 - Architecture d'un projet
 - Cycle de vie d'un projet
- 4 Les outils du développeur
 - Développement
 - Industrialisation



Histoire du web

Quelques dates

- 1969 : Arpanet
- 1972 : Premier courriel
- 1973 : TCP/IP
- 1982 : Internet
- 1989 : World Wide Web (Time Bernes Lee)
- 1990 : 1^{er} serveur, 1^{er} site, 1^{er} navigateur : WorldWideWeb
- 1993 : Mise dans le domaine public du WWW
- 1995 : W3C, Yahoo, Netscape

Le web toujours en mouvement

➤ 90's : Web = Partage d'information & eCommerce

- 1995 : Altavista, ebay, Amazon
- 1998 : Google, Paypal

➤ 2000's : Avènement du « Web Social »

- Wikipedia (2001), LinkedIn (2002), Skype, iTunes, Myspace (2003), Facebook (2004), Youtube (2005), Twitter, Leboncoin (2006), Instagram, Pinterest (2010)
- 2006 : 19 millions de site internet
- Changement de consommation : iPhone (2007) & smartphone android (2008), 1ères tablettes (2010)

➤ 2010's : Le Web devient intelligent

- 2014 : +1Md de sites internet
- 2015 : +50 % recherche google depuis smartphone/tablette
- Premiers objets connectés (2011), IoT (2014), TV connectées (2015)
- Big Data (2013)
- Siri (2011), Cortana, Alexa (2014), Google Assistant (2016)
- Blockchain (2017) ...

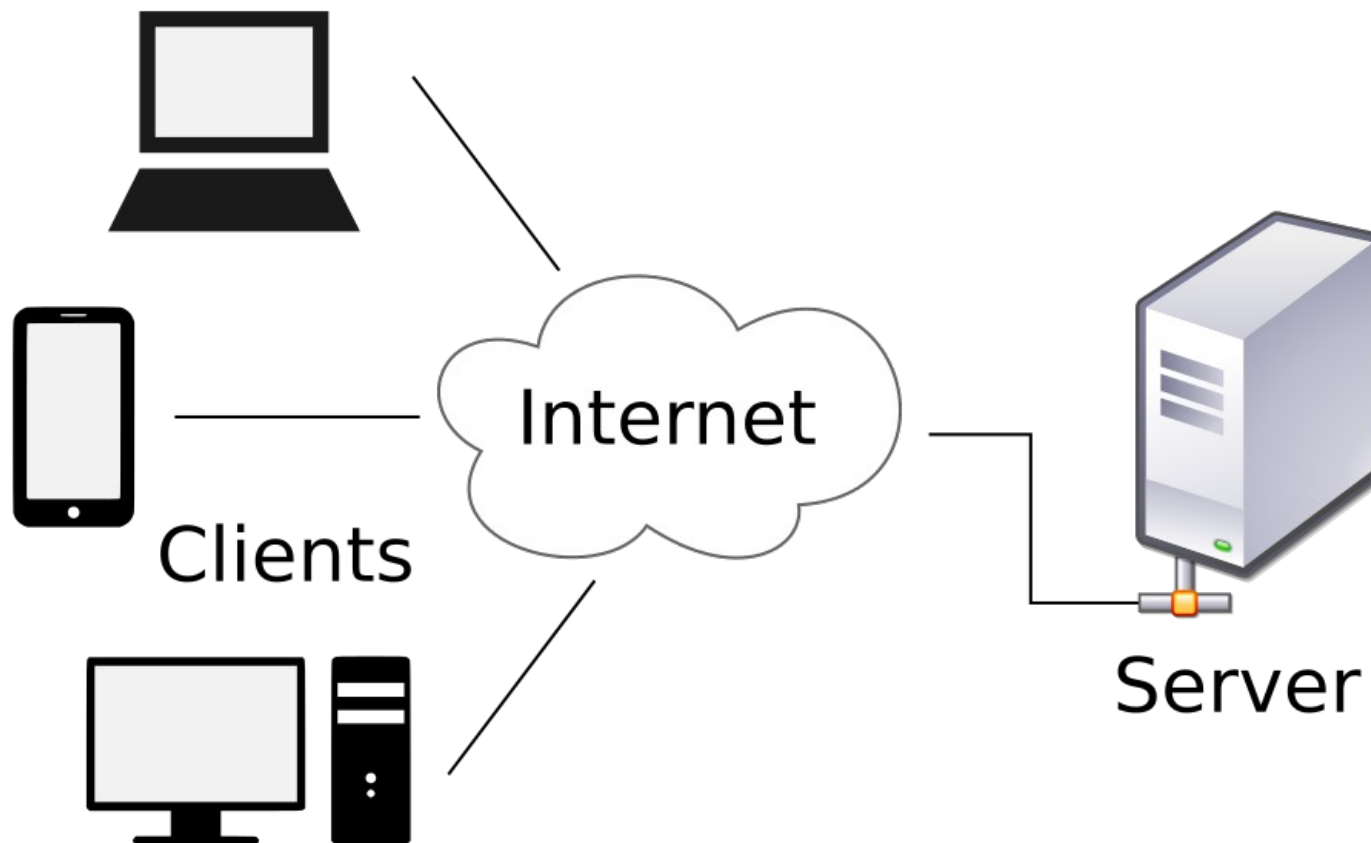


Composantes du Web

RFC : Request For Comments

- RFC 114, 765, 959, ... : FTP
- RFC 791 : Internet Protocol (IP)
- RFC 793 : TCP
- RFC 821, 2821, 5321 : Simple Mail Transfer Protocol
- 1738 : Uniform Resource Locators
- 1866 : HTML 2.0 (en 1995)
- 2616 : HTTP 1.1

Structure du Web...



Quelques exemples de logiciels

- Serveurs Web : Apache, Internet Information Services
- Navigateurs (Firefox, Chromium, IE, Opera...)
- Messagerie courriel (Thunderbird, Outlook, K9-Mail, ...)
- Site web
- Applications smartphone



Application Web ?

Structuration globale

- Application lourde VS Application « Légère »
- Architecture logicielle trois-tiers : trois couches logicielles
 - Présentation
 - Traitement
 - Données
- Multiple interactions :
 - Humaine (le client)
 - Non humaine (Base de données, serveur de fichier, d'autres applications ...)

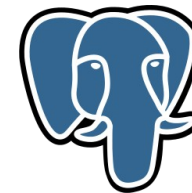
Back-end : face immergée

- Partie « serveur » d'un logiciel
- Traitement des informations et opérations pour fournir un résultat (application)
- Gestion des ressources (base de données, fichiers)
- Mutualisation de ressources, Centralisation de l'information (hébergement)

Quelques exemples de technologies backend



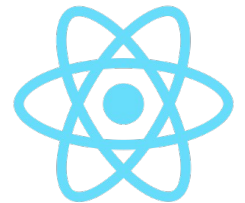
C#



Front-end, premier contact !

- Côté « client »
- Présentation des informations/données
- Souvent présentée par le navigateur
- Navigateur = interpréteur

Exemples de technologies Front-End



Structure un projet : La roue existe déjà

- Un langage vient avec sa **grammaire** de base, mais souvent aussi avec des **fonctionnalités basiques** (ex : sommer un ensemble, trier ...)
- Il existe des « **bibliothèques/librairies** » ajoutant des fonctionnalités particulières
- Des **frameworks** permettent également de simplifier la mise en place de fonctionnement

Petit dictionnaire d'un cycle de projet

- Cahier des charges
- Analyse fonctionnelle
- Prototypage
- alpha, beta, RC, SNAPSHOT / Release
- Tests unitaires / Tests fonctionnels
- Mock
- Recette / Test
- (Pré)Production
- Tickets / Reporting / Bugs



Outils de développeur Web

Écrire du code

➤ Avec un IDE :



- Avec un éditeur de texte simple (sublime text, notepad)
- « Outils de développement » du navigateur (Ctrl+Maj+I) pour manipuler et tester un peu de code front

Construire le projet

➤ Automatisation de construction de projet

maven



gradle



➤ Analyse de code

PMD

checkstyle

sonarqube

JS Hint

➤ Communication :

- Courriel
- Liste de diffusion du projet
- Messagerie instantannée

➤ Lister les fonctionnalités/taches/bugs



Gérer le code source

➤ Partage du code avec un gestionnaire de sources



➤ Visualisation des contributions (commit), des patches avec des forges logicielles



➤ Note : ces forges permettent également de gérer la vie du projet côté organisation (tickets)



Et côté humain ?

- Maitrise d'Oeuvre : réalisateur du projet
 - Développeur
 - Architecte
 - Chef de projet
 - Directeur de projet

- Maitrise d'ouvrage (MOA) : client du projet, pas forcément utilisateur final
 - Product Owner (PO)
 - Équipe de recette

- Peut faire partie de la MOE, ou être en périphérie :
 - Administrateur système (Admin sys / AdSys)
 - Administrateur Base de Données (DBA)

Les types de projets

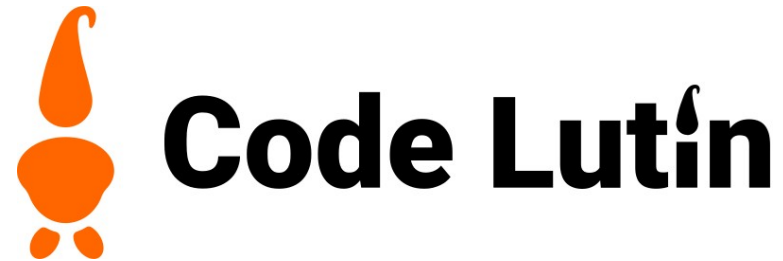
- ESN : anciennement SSII/SSLL
- Éditeur
- Forfait VS Régie
- DSI

Communautés

- Communauté des projets libres (développeurs, traducteurs, testeurs, ...)
- Stack Overflow : les réponses à vos questions
- Meetup : JUG, NantesJS, Software Crafter, ...
- Conférences : DefFest, Devoxx, BreizhCamp, Web2Day
- <https://dev.tube/>



Vos questions ?



02 • 40 • 50 • 29 • 28

www.codelutin.com

