09/09/2022

Mohammed LAMNAOUR

ISMO Tetouan

Introduction au développement Front End

Table des matières

[I. Comment fonctionne un site web ? 2](#_Toc113540499)

[II. Site web statique 2](#_Toc113540500)

[III. Site web dynamique 2](#_Toc113540501)

[IV. Expérience Utilisateur (UX) 3](#_Toc113540502)

[a. Définition de l'expérience utilisateur (UX) 3](#_Toc113540503)

[b. Définition de design UX 3](#_Toc113540504)

[c. Points pris en compte dans un design UX 3](#_Toc113540505)

[V. Différence entre Framework et Bibliothèque 4](#_Toc113540506)

[VI. Développement Front End 4](#_Toc113540507)

[a. Définition 4](#_Toc113540508)

[b. Les langages front-end 5](#_Toc113540509)

[c. Les librairies et Framework front-end 6](#_Toc113540510)

[d. Développeur front-end 6](#_Toc113540511)

[VII. Développement Back End 6](#_Toc113540512)

[a. Définition 6](#_Toc113540513)

[b. Les Framework back-end 7](#_Toc113540514)

[VIII. Application web mono-page (SPA) 7](#_Toc113540515)

[a. Définition 7](#_Toc113540516)

[b. Fonctionnement de SPA 8](#_Toc113540517)

[c. Les avantages du SPA 8](#_Toc113540518)

[d. Les inconvénients du SPA 9](#_Toc113540519)

[IX. Client side Rendering 10](#_Toc113540520)

[X. Server side Rendering 10](#_Toc113540521)

# Comment fonctionne un site web ?

Le **Web** est un système d’échange entre un **client** et un **serveur** :

**Le client est notre navigateur Web**, c’est lui qui va nous permettre de voir le Web depuis notre ordinateur, smartphone ou tablette. C’est celui que vous connaissez sans doute sous le nom de Chrome, Firefox, Safari, Internet Explorer, etc.

**Le serveur** est un ordinateur puissant qui stocke et héberge des sites Web. C’est sur cet ordinateur que se trouvent les pages Web, c’est à dire tous les fichiers du site internet auquel on veut accéder.

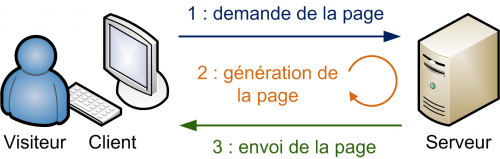


Figure 1: Architecture client-serveur

# Site web statique

Un site statique, comme son nom l’indique, est composé de **pages statiques**, visibles telles qu’elles ont été conçues. Cela ne signifie pas qu’elles sont sans mouvement ou sans animation. On peut très bien y trouver des animations flash, des images, ou même des musiques et des vidéos. La seule chose que cela signifie, c’est qu’elles seront toujours identiques, vous ne risquez pas de voir des images qui changent au cours de vos visites.

**Question :** Quelles sont les langages de programmation que vous connaissez pour développer un site web statique ?

# Site web dynamique

Un site dynamique est quant à lui, tout le contraire du site statique. Les informations seront présentées de façon différente selon l’interaction avec le visiteur. C’est en général le cas des sites vitrines qui présentent des vitrines classées selon un critère. Le travail du webmaster est alors de concevoir une programmation qui permettra de présenter telle ou telle information selon la requête de l’internaute.

**Question :** Quelles sont les langages de programmation que vous connaissez pour développer un site web dynamique ?

# Expérience Utilisateur (UX)

## Définition de l'expérience utilisateur (UX)

Le terme UX est de plus en plus utilisé dans le jargon du marketing. Signifiant expérience utilisateur en français, l’acronyme UX désigne l’expérience vécue et ressentie par un utilisateur dans la globalité de son interaction avec un produit ou une interface numérique. Une bonne expérience utilisateur implique donc que les besoins de l’utilisateur sont anticipés et ont été répondus le plus efficacement possible.

## Définition de design UX

Un design UX représente l’ensemble des moyens mis en œuvre pour concevoir une interface qui offre **une bonne expérience utilisateur**. Son objectif premier est donc de fournir la meilleure expérience utilisateur possible.

Du démarrage d’un projet jusqu’au contrôle de qualité final, [un expert UX intervient pour s’assurer que les interfaces sont agréables pour les utilisateurs](https://leonardagenceweb.com/expertise/strategie/design-experience-utilisateur/). Il évalue plusieurs éléments d’un design pour que l’utilisateur se sente en confiance, et puisse trouver les informations qu’il cherche tout en ayant envie d’interagir avec l’interface.

## Points pris en compte dans un design UX

**L’efficacité** : L’efficacité mesure la vitesse et la rapidité avec laquelle les utilisateurs peuvent accomplir les tâches ou trouver les réponses à leurs questions.

**L’utilité des fonctionnalités disponibles :** Chaque fonctionnalité doit offrir de la valeur et être utile pour les utilisateurs. L’expert en UX effectue des recherches et des analyses approfondies pour comprendre ce qui est utile dans l’expérience de l’utilisateur.

**La crédibilité :** La crédibilité est liée à la capacité de l’utilisateur à faire confiance à votre marque et à vos produits. Le design UX doit rassurer et convaincre les internautes.

**L’accessibilité :** L’accessibilité se préoccupe de savoir si tous les utilisateurs sont en mesure d’accéder à une expérience équivalente.

# Différence entre Framework et Bibliothèque

En français, on traduit le mot Framework par “**cadre applicatif**”, car c'est son but, il donne un cadre, **une organisation**, un squelette, **une méthode de travail**. Tandis que la Bibliothèque, elle, n'offre que **des fonctionnalités souvent non lie les unes des autres.**

**“Le développeur utilise une bibliothèque en appelant le code de cette dernière, tandis que le rôle du Framework est d’exécuter le code du développeur”**

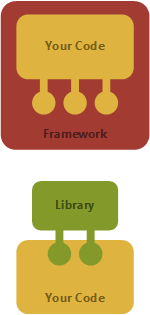


Figure 2 : Bibliotheque vs Framework

# Développement Front End

# Définition

Le **front-end** c’est la partie du code qui est reçue par le client (**navigateur Web)**. Il s’agit des éléments du site web que l’on aperçoit à l’écran et avec lesquels les utilisateurs interagissent.

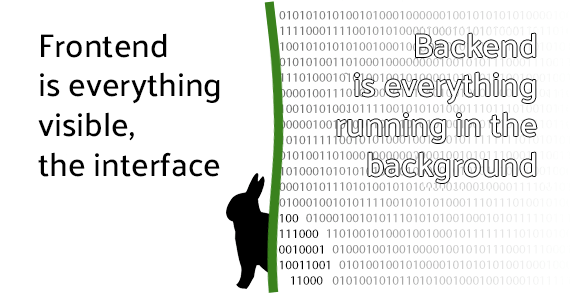


Figure 3 : Front End vs Back End

# Les langages front-end

* **HTML** : est un langage composé de tags, balises en Français. Il va nous permettre de représenter la structure, le **squelette de nos pages Web.**
* **CSS** : est un langage qui va **mettre en forme nos pages Web** et les décorer. Il va désigner nos **éléments HTML** à l’aide des sélecteurs et va leur appliquer un **style CSS**. C’est ce **langage CSS** qui est responsable des couleurs, des tailles, de la mise en page, etc.
* **JAVASCRIPT** : Ce langage de programmation permet d’implémenter des éléments complexes sur une page web. Il met à jour le contenu de la page web d’une manière dynamique en contrôlant le multimédia, les images…

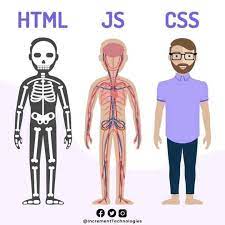


Figure 4 : HTML vs CSS vs JS

# Les librairies et Framework front-end

* **JQuery** est une bibliothèque JavaScript qui permet de coder plus vite et plus simplement.
* **Bootstrap** est un Framework développé par l'équipe du réseau social Twitter. Proposé en open source, ce Framework utilisant les langages HTML, CSS et JavaScript fournit aux développeurs des outils pour créer un site facilement**.**
* **React** est une bibliothèque JavaScript développé par Facebook, l’une des plus populaires du moment.
* **Vue** ou Vue.js est un Framework JavaScript qui est réputé pour être simple à prendre en main. Il vient se placer en concurrent direct de React.
* **Angular** est aussi un Framework JavaScript initié par Google mais qui a perdu en popularité ces dernières années.

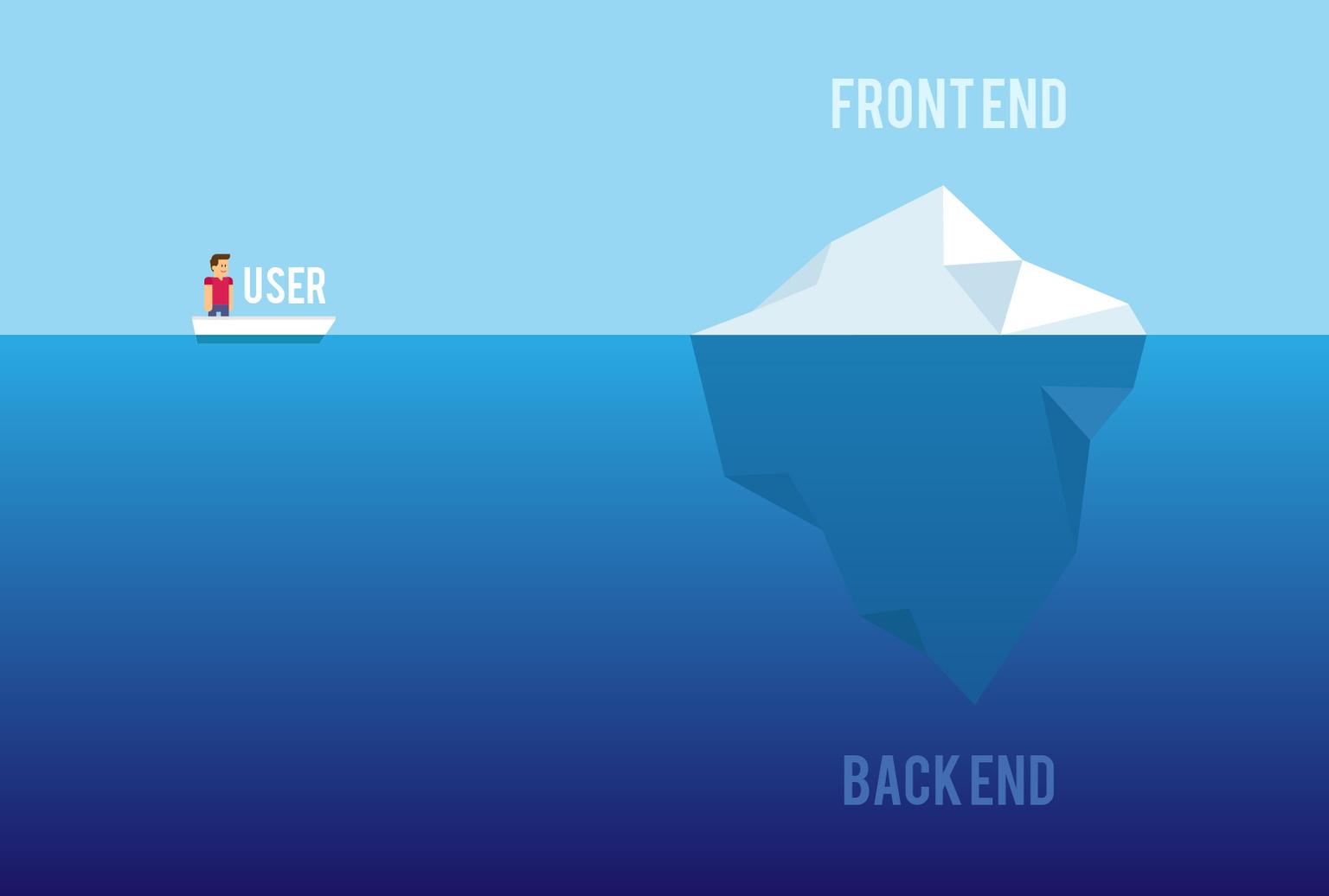
# Développeur front-end

Un développeur front-end est un programmeur dédié au design. Il travaille à la création de l’interface utilisateur d’un site ou d’une application, d’après les maquettes graphiques élaborées par les webdesigners.

# Développement Back End

# Définition

Le développement Back-end (ou Backend) a pour objectif de permettre à un site ou à une application de fonctionner. Il s’agit donc de mettre en place la **programmation au niveau du serveur**, pour exécuter les requêtes qui sont réalisées sur le site web par les internautes.



# Les Framework back-end

* [**Django**](https://github.com/django/django) : est un Framework backend open-source de premier plan basé sur le langage de programmation Python. Il suit le modèle du contrôleur de vue de modèle (MVC).
* [**Lavarel**](https://laravel.com/) est un Framework web PHP open-source pour développer des applications web basées sur Symfony qui suivent l’architecture modèle-vue-contrôleur (MVC).
* [**Express.js**](https://expressjs.com/) : également connu sous le nom d’Express. C’est un Framework d’application web Node.js et un logiciel open-source disponible sous la licence du MIT (open source).
* **Question :** Qu'est-ce qu'un développeur full stack ?

# Application web mono-page (SPA)

## Définition

Le concept de l'application web mono-page (en anglais Single Page Application (SPA)) commence à émerger au début de l'année 2000.

Une application à page unique est une application qui interagit avec les utilisateurs en réécrivant les pages Web existantes avec de nouvelles données provenant du serveur Web, au lieu d’utiliser la technique par défaut du navigateur qui exécute une toute nouvelle page.

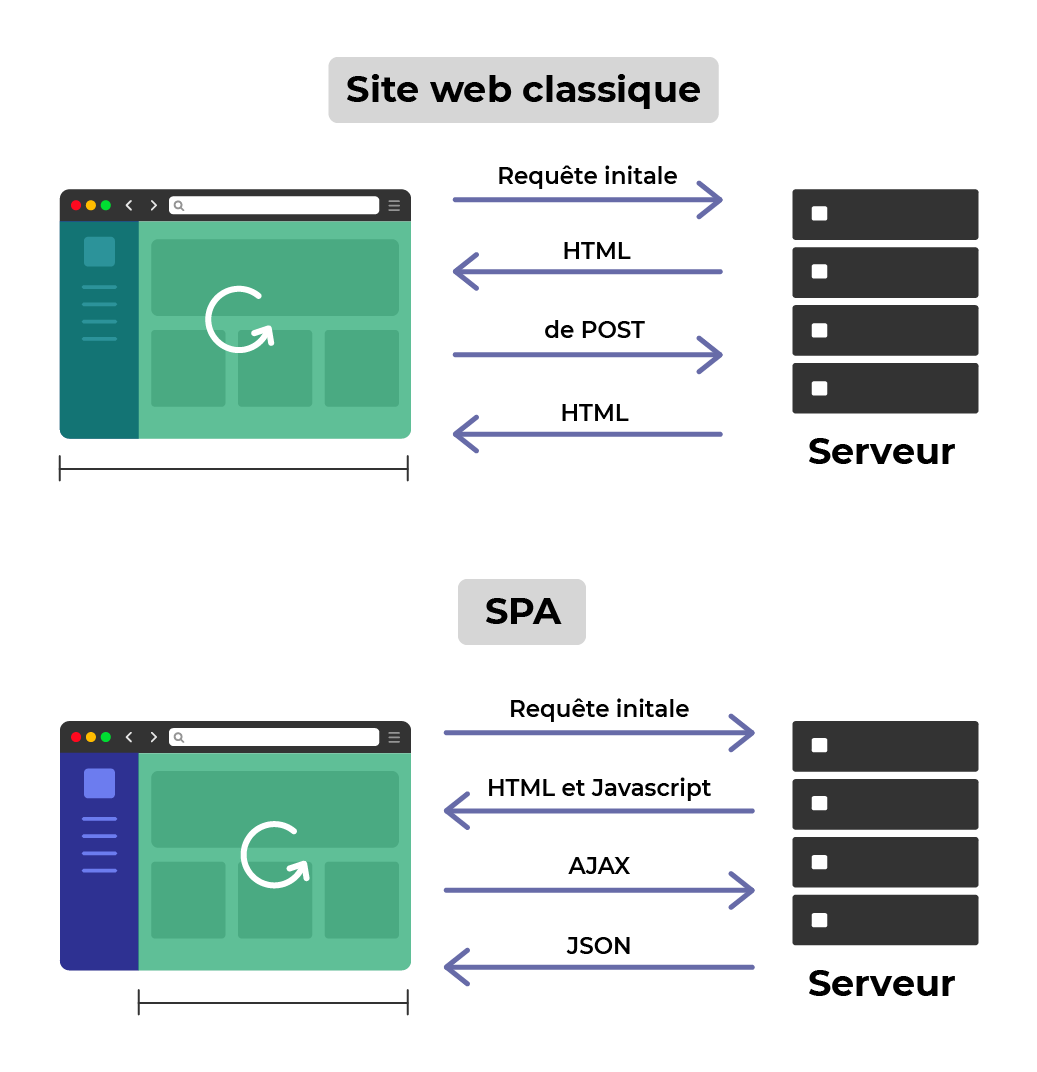
## Fonctionnement de SPA

Le cœur d’un SPA est basé sur Ajax, un ensemble de techniques de développement qui permet au client d’envoyer et de récupérer des données du serveur de manière asynchrone (en arrière-plan) sans interférer avec l’affichage et le comportement de la page web. Ajax permet aux pages web et, par extension, aux applications web, de modifier le contenu de manière dynamique sans avoir à recharger la page entière.

**Un SPA fonctionne comme suit :**

Le client se connecte d’abord au serveur et obtient le contenu de la page, qui correspond principalement au code HTML, au CSS et à un bundle JavaScript, contenant tous les JavaScripts nécessaires pour exécuter la logique de l’application.

 L’action d’un utilisateur déclenche l’exécution de JavaScript associés, qui à leur tour demandent des données au serveur par le biais d’appels Ajax. Les données sont généralement fournies dans un format JSON et ne déclenchent pas de rafraîchissement complet de la page Web.



# Les avantages du SPA

**Performance et vitesse de chargement des pages**

La vitesse doit être l’un des plus grands avantages d’opter pour le développement d’une application à page unique. Elles sont beaucoup plus rapides que les applications web traditionnelles car elles peuvent charger de nouvelles informations dans une seule page à la demande du client au lieu de relier plusieurs pages HTML à l’architecture du site.

 La plupart des utilisateurs sont souvent déroutés et irrités de devoir cliquer sur de nombreux liens dans une application traditionnelle pour arriver là où ils veulent. Les applications à page unique éliminent ce problème en fournissant tout le contenu nécessaire dans une seule page déroulante. Cela fait également de la SPA un choix idéal pour les entreprises dont un grand nombre d’utilisateurs accèdent à leurs applications via des navigateurs mobiles.

**Capacités de mise en cache :**

Une application à page unique est capable de mettre en cache les données locales plus efficacement. Comme l’application à page unique n’envoie qu’une seule requête au serveur et stocke toutes les données qu’elle reçoit, elle peut également utiliser ces données pour travailler hors ligne.

**Débogage avec Chrome :**

Le débogage des SPA avec Chrome est beaucoup plus facile que le débogage de n’importe quelle application à pages multiples. De plus, comme les SPA sont construites avec des Framework tels que Angular et React, ces Framework ont leurs propres outils de débogage Chrome, ce qui facilite grandement leur débogage.

## Les inconvénients du SPA

**Historique du navigateur :**

L’un des inconvénients d’une expérience utilisateur transparente et de l’absence de chargement de page avec les applications à page unique est que les SPA n’enregistrent pas les sauts des utilisateurs entre les différents états. Cela signifie que lorsque l’utilisateur appuie sur le bouton « retour », il ne peut pas revenir en arrière. Le bouton retour ne ramène l’utilisateur qu’à la page précédente et non à l’état précédent dans l’application.

**Optimisation du référencement :**

L’un des plus grands signaux d’alarme qui empêchent le développeur d’explorer les applications à page unique est la difficulté d’optimisation du référencement.

# Client side Rendering

Le client-side rendering (ou le rendu cote client) consiste à **laisser le navigateur interpréter** le JavaScript pour générer le contenu HTML de votre site. Ainsi, lorsque le navigateur fait une requête au serveur, au lieu d’obtenir tout le contenu du document HTML lui-même, vous obtenez un document HTML simple, avec un fichier JavaScript qui permettra au navigateur de générer le reste du site.

# Server side Rendering

Le rendu côté serveur, ou *server side rendering* (SSR) correspond au fonctionnement des sites web classiques : le navigateur envoie une **requête au serveur**, celui-ci traite l’information et renvoie une requête http contenant le HTML complet au navigateur, qui peut ensuite facilement effectuer le rendu.

**Travail à faire :**

**1- Quels avantages que les Framework offrent aux développeurs ?**

**2- Quel est l'utilité de l'Ajax ?**

**3- Quel est la différence entre une application monopage (SPA) et une application multi-page (MPA) ?**

**4- Remplir le tableau suivant sur les avantages et les inconvénients d'une application SPA et une application MPA ?**

|  |  |  |
| --- | --- | --- |
| Type d'application | Avantages | Inconvénients |
| SPA |  |  |
| MPA |  |  |

**5- Remplir le tableau de comparaison entre les Framework front end les plus répandus :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Framework | Année d'apparition | Propriétaire | La dernière version | Les avantages | Les inconvénients |
| Angular |  |  |  |  |  |
| React Js |  |  |  |  |  |
| Vue JS |  |  |  |  |  |

**6- Citez les avantages et les inconvénients entre Server side Rendering et Client side Rendering ?**

|  |  |  |
| --- | --- | --- |
| Type de rendu | Avantages | Inconvénients |
| SSR |  |  |
| CSR |  |  |