

Leçon 4 : Introduction à Bootstrap

*Cours_ UVCi
PRW2103-2*

DIABATE Nagbegna

Table des matières



I - Objectifs	3
II - Introduction	4
III - Section 1 - Présentation de Bootstrap	5
1. 1. Les frameworks clients Web	5
2. 2. Choisir son framework	5
3. 3. le Framework Bootstrap	7
4. Exercice	8
IV - Section 2 : Installation de Bootstrap 3	10
1. 1. Installation du serveur web	10
2. 2. Test du serveur Web	17
3. 3. Installation de Bootstrap 3	17
4. Exercice	21



Objectifs

À la fin de cette leçon, vous serez capable de :

- *Définir* un Framework et ses fonctionnalités ;
- *Comprendre* l'utilité et le fonctionnement de Bootstrap ;
- *Installer* un serveur web local ;
- *Installer* Bootstrap dans un projet Web ;
- *Lancer* sa première page Web avec Bootstrap.

Introduction



Bootstrap Twitter est composé d'un ensemble d'outils qui simplifie fortement la création de site web.

Ce framework est composé de code :

- HTML
- Javascript
- CSS

Vous y retrouverez un grand nombre d'éléments et composants Javascript permettant de facilement créer un carousel, des boîtes de dialogue etc.

Cependant, avant d'aborder cette leçon, *vous devez avoir compris et validé les leçons précédentes. Car cette leçon est l'exploitation pratique de toutes les notions abordées dans les leçons précédentes.*

Section 1 - Présentation de Bootstrap



1. 1. Les frameworks clients Web

Un *framework* est, comme son nom l'indique en anglais, un “*cadre de travail*”. L'objectif d'un *framework* est généralement de simplifier le travail des développeurs ou codeurs informatiques, en leur offrant une architecture “*prête à l'emploi*” et qui leur permette de ne pas repartir de zéro à chaque nouveau projet.

Les *frameworks* sont comparables aux patrons de couture. Les principaux avantages sont donc :

- la *réutilisation* des codes
- la *standardisation* de la programmation
- la *formalisation* d'une architecture adaptée aux besoins de chaque entreprise

À noter aussi que les *frameworks* sont toujours « enrichis » de l'expérience de tous les développements antérieurs.

En français, plusieurs termes et expressions permettent actuellement de désigner le *framework* : *cadriciel*, *canevas*, *socle d'applications*, *cadre d'applications*, *environnement de développement*, *boîte à outils*, etc.

Un *framework*, c'est une base d'outils et de fonctionnalités conçus pour s'agencer entre eux. En partant d'une grille, qui est la plupart du temps prévue pour être responsive, vous pouvez ajouter facilement des éléments qui ont été maintes et maintes fois testés par d'autres développeurs et dont la plupart des sites ont besoin (des comptes utilisateurs, des authentifications, des connexions à diverses sources de données, des moteurs de recherche interne, des boutons de partage sur les réseaux sociaux...). Le tout sans avoir besoin de les coder vous-même, ce qui représente un gain de temps considérable.

Par ailleurs, le *framework* est automatiquement mis à jour pour s'adapter aux nouvelles pratiques du net (comme ce fut le cas pour le passage aux sites responsive, notamment).

2. 2. Choisir son framework

Il existe toute une panoplie de frameworks, d'autres diront simplement qu'il existe des *frameworks pour tout* : *développer une application mobile*, *un jeu*, *un site web*, *un module d'extension*, des *CSS*... La plupart d'entre eux est spécifique à un langage de script ou de programmation. Contrairement aux apparences, les débutants ne sont pas nécessairement la cible première des frameworks : mieux vaut se frotter aux bases pour commencer avant de pouvoir tirer le maximum de bénéfices de l'utilisation d'un framework pour gagner en productivité. Par exemple, un débutant qui voudrait utiliser *Django*, un framework de développement Python pour le web, aurait sans doute beaucoup de mal à en appréhender les bénéfices sans avoir une connaissance et une expérience approfondie en programmation avec Python.

Bien que les frameworks ne répondent pas souvent aux mêmes besoins, plusieurs d'entre eux peuvent être combinés dans diverses situations permettant de les regrouper en plusieurs catégories :

- les frameworks applicatifs web
- les frameworks de développement d'applications de bureau

- les frameworks de logging
- ou encore les frameworks de présentation de contenu web.
- etc...

On distingue au moins deux types de frameworks : *front-end* et *back-end*.

a) le *Front-end*

Le terme « front-end » désigne les éléments d'un site que l'on voit à l'écran et avec lesquels on peut interagir depuis un navigateur. En effet, tout ce qu'on voit sur un site internet par exemple, est une combinaison de HTML, CSS et JavaScript. Ces langages de programmation utilisés par le développeur Front End sont interprétés par le navigateur de votre ordinateur pour afficher un résultat « visuel ». Il s'agit notamment de polices, de menus déroulants, de boutons, de transitions, de curseurs, de formulaires de contact, etc.

Le Frontend se compose généralement :

- D'un design créé par un Web designer qui réalise des maquettes graphiques via des outils de création comme Photoshop ou Fireworks.
- De code HTML, CSS, JavaScript et jQuery mis en place par un développeur Frontend.

Maintenant, pour que le travail du développeur Front End prenne vie et que le site soit fonctionnel, il manque encore quelques éléments. C'est là qu'entre en scène le développeur Back End !

Source : <https://www.nemesis-studio.com>

b) Le *back-end*

Le « back-end » est un peu comme la partie immergée d'un iceberg. On ne la voit pas en tant que simple Internaute mais elle représente une très grande partie d'un projet web. C'est dans le back-end que travaillent les administrateurs du site web.

Le Backend se compose généralement de trois éléments :

- Un serveur (hébergement web)
- Une application (site web, administration)
- Une base de données (sorte de feuille de calcul pour organiser les données)

Prenons un exemple pour comprendre le fonctionnement du Back End : Imaginons que vous deviez réserver un vol en ligne pour vos futures vacances. Vous vous rendez sur le site de la compagnie aérienne et recherchez le vol qui vous convient. Une fois le vol sélectionné, vous renseignez vos informations personnelles et validez votre réservation. Vos informations sont alors enregistrées dans une base de données stockée sur un serveur.

Toutes ces informations restent sur le serveur, alors quand vient l'heure des vacances (2 mois plus tard) vous vous connectez à l'espace client (application) pour imprimer vos billets d'avion et toutes les informations que vous aviez renseignées lors de votre réservation sont disponibles sur votre compte.

La personne qui administre toute cette technologie est le développeur back end. Les technologies Backend se composent généralement de langages comme PHP, Ruby, Python, etc. Pour les rendre encore plus faciles à utiliser, ils sont généralement améliorés par des Framework comme Ruby on Rails, Cake PHP, Symfony et Code Igniter qui rendent le développement plus rapide et plus sécurisé.

Source : <https://www.nemesis-studio.com>

Dans ce module, nous allons nous intéresser aux frameworks front-end.

Si le fonctionnement des différents frameworks présents sur internet est relativement similaire, les disparités sont suffisamment fortes pour nécessiter de choisir correctement son framework de départ.

La sélection d'un framework se décide souvent en fonction de sa facilité d'apprentissage, comme de sa bibliothèque de contenus (certains, plus connus, sont bien implantés dans le paysage des développeurs et bénéficient de nombreux cours, tutoriels et articles de blog les concernant).

Ainsi, faire une sélection de framework est une tâche ardue. Il y a ceux dont tout le monde parle, d'autres qui sont moins connus, d'autres encore qui ne sont plus mis à jour:

c) Quelques frameworks web front-end

- *Bootstrap* compte parmi les plus populaires. Open Source, il présente un nombre important de modules et d'extensions. La documentation disponible est tout aussi impressionnante. Il fonctionne sur un système de grille en douze colonnes avec un affichage adapté pour différentes tailles d'écran et il est possible de personnaliser les fichiers CSS.
- *Foundation* dispose d'une grande flexibilité. Il est proposé avec un fichier démo utile pour les débutants et dispose de son propre centre de formation. Il est cependant légèrement plus complexe d'utilisation.
- *Semantic UI* se veut plus simple et intuitif. Son principe repose sur l'utilisation d'un langage naturel, accessible aux néophytes, qui permet d'obtenir un code HTML facile à lire et à comprendre.
- *KNACSS* est un modèle de framework relativement léger mais complet. Il est conçu en France, ce qui rend la documentation plus accessible.
- *Materialize* s'appuie sur les recommandations design de Google et est fourni avec de nombreux composants prêts à l'emploi.
- *Material UI* est également basé sur les principes du Material Design de Google, mais il sera d'un abord plus complexe, d'autant qu'il nécessite de comprendre au préalable React.js pour pouvoir l'utiliser.

3. 3. le Framework Bootstrap

Il est le plus connu et le plus utilisé des frameworks front-end web . Certainement boosté par la notoriété de Twitter qui a conçu son propre framework et a choisi de le mettre à disposition du public. Bootstrap est ainsi né fin 2011, depuis sa notoriété n'a cessé de grandir, en raison de sa simplicité d'implémentation et de la diversité de ses composants.

Bootstrap est fourni avec des fichiers CSS, JavaScript et une police. Il est également possible de télécharger des fichiers HTML d'exemple afin de découvrir ce dont est capable le framework.



Bootstrap propose :

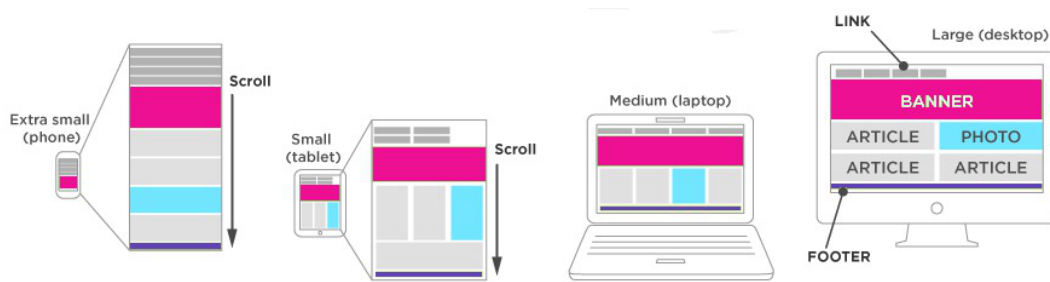
- une grille responsive, des styles CSS pour tous les éléments HTML standards (formulaire, bouton, texte, image ...)
- des composants JS comme des listes déroulantes dynamiques, des navigations interactives ou un carousel
- une police qui est en fait un set d'icônes. L'avantage est que les icônes n'étant pas des images mais des polices, elles sont vectorisées et donc toujours nettes sur tout type d'appareils (on pense notamment aux appareils mac disposant d'un écran Retina).

- Les intérêts de Bootstrap

- Grâce à de nombreux composants pré-configurés, *Bootstrap* permet aux développeurs de gagner du temps dans leurs développements et donc dans leur productivité.
- *Bootstrap* garantit la compatibilité de votre site avec tous les navigateurs récents et toutes les tailles d'écran.
- Il est gratuit ! Et comme il est largement utilisé, *Bootstrap* est très bien documenté (c'est très pratique si vous rencontrez des problèmes) et évolue constamment.
- *Bootstrap* est pensé mobile-first avec un système de grille avec 12 colonnes, ce qui permet facilement de construire votre site pour les mobiles puis de l'adapter pour les plus grands écrans.
- Aucun site *Bootstrap* ne se ressemble, vous pouvez personnaliser votre site comme vous le souhaitez. Il existe également une large galerie de *templates Bootstrap* pour trouver le design qui vous convient le mieux

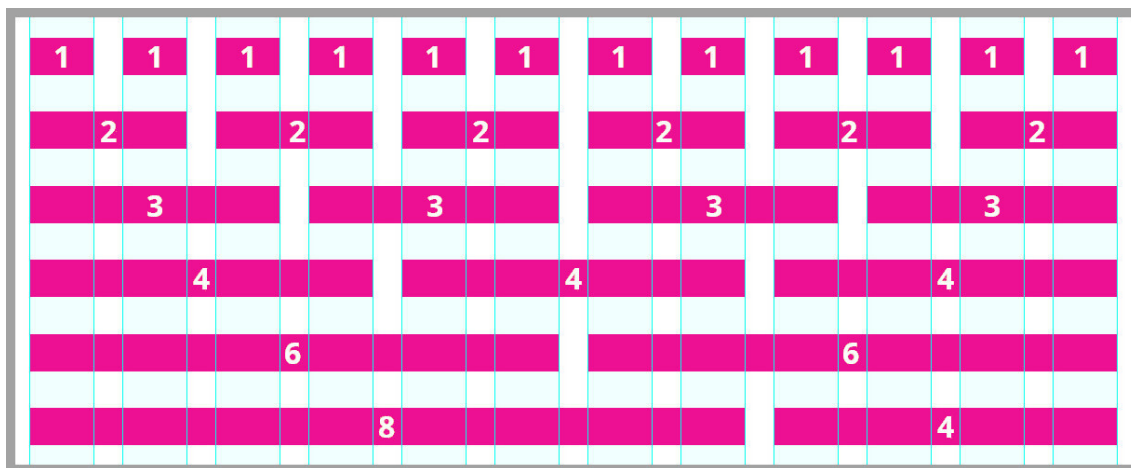
- le principe de Bootstrap

Selon un article du site *YesYouWeb*, *La méthode de base de Bootstrap* est de "designer" votre site pour le plus petit écran puis augmenter son échelle pour de plus grands écrans le rendra compatible avec n'importe quel appareil. Bien pensées, vos colonnes s'adapteront afin d'optimiser votre contenu selon la taille de l'écran.



Source : <https://yesyouweb.com/developpez-bootstrap-mobile-first/>

Bootstrap est basé sur un système de grille avec 12 colonnes. Vous pouvez ensuite répartir vos contenus avec autant de flexibilité que vous le souhaitez : 12 colonnes, 6 colonnes, 2/3 – 1/3 soit 1 colonne de 8 et 1 de 4, 1 seule colonne... Toutes les combinaisons sont possibles. Si vous avez besoin de plus de 12 colonnes ou de moins, il est toujours possible de changer la configuration.



Source : <https://yesyouweb.com/developpez-bootstrap-mobile-first/>

Le système de grille Bootstrap contient 4 classes :

- xs – pour les smartphones
- sm – pour les tablettes
- md – pour les ordinateurs classiques
- lg – pour les ordinateurs les plus grands

4. Exercice

Exercice

1) Quel terme désigne les éléments d'un site que l'on voit à l'écran et avec lesquels on peut interagir depuis un navigateur ?

- ☐ front-end
- ☐ back-end

Exercice

2) *Quel(s) langage(s) de programmation est (sont) utilisé(s) dans le framework Bootstrap ?*

- ☐ HTML
- ☐ PHP
- ☐ CSS
- ☐ JavaScript
- ☐ Java
- ☐ Python

Exercice

3) *Quels sont parmi ces affirmations, les intérêts de l'utilisation d'un framework ?*

- ☐ Réutilisation des codes
- ☐ Pas besoin de savoir programmer pour utiliser les frameworks
- ☐ Standardisation de la programmation
- ☐ Formalisation d'une architecture adaptée aux besoins de chaque entité

Exercice

4) *Laquelle des affirmations suivante est juste ?*

- ☐ Bootstrap est basé sur un système de grille avec 12 colonnes et 12 lignes
- ☐ Bootstrap est basé sur un système de grille avec 6 lignes et 6 colonnes
- ☐ Bootstrap est basé sur un système de grille avec 12 colonnes
- ☐ Aucune de ses affirmations n'est correcte !

Exercice

5) *Quelle classe Bootstrap correspond aux tablettes ?*

- ☐ md-
- ☐ xs-
- ☐ lg-
- ☐ ml-
- ☐ sm-

Section 2 : Installation de Bootstrap 3



1. 1. Installation du serveur web

Nous avons jusque-là écrit du code HTML et CSS pour des pages web statique faisant appel à aucune ressource extérieure, nous écrivions donc du code qui pouvait s'exécuter par double clic sur le fichier et qui pouvait être mis à n'importe endroit sur l'ordinateur.

Dans le cadre d'exploitation de *Bootstrap*, nous allons utiliser des ressources extérieurs qui pourraient nécessiter les services d'un serveur web pour s'exécuter.

Nous avons donc besoin d'environnement de travail et de développement qui nous permettent de développer et mettre au point nos sites, en réelle condition client / serveur. Nous allons donc dans cette section, installer un serveur web local, qui sera désormais notre espace de travail en développement Web.

Nous allons utiliser *XAMPP* pour installer notre serveur.

XAMPP est synonyme de multi-plateforme (X), Apache (A), MySQL (M), PHP (P) et Perl (P). C'est une distribution Apache simple et légère qui permet aux développeurs de créer facilement un serveur web local à des fins de test.

XAMPP est également multi-plateforme, ce qui signifie qu'il fonctionne aussi bien sur Linux, Mac et Windows. Comme la plupart des déploiements de serveurs Web actuels utilisent les mêmes composants que *XAMPP*, la transition d'un serveur de test local à un serveur VPS ou dédié est également extrêmement simple.



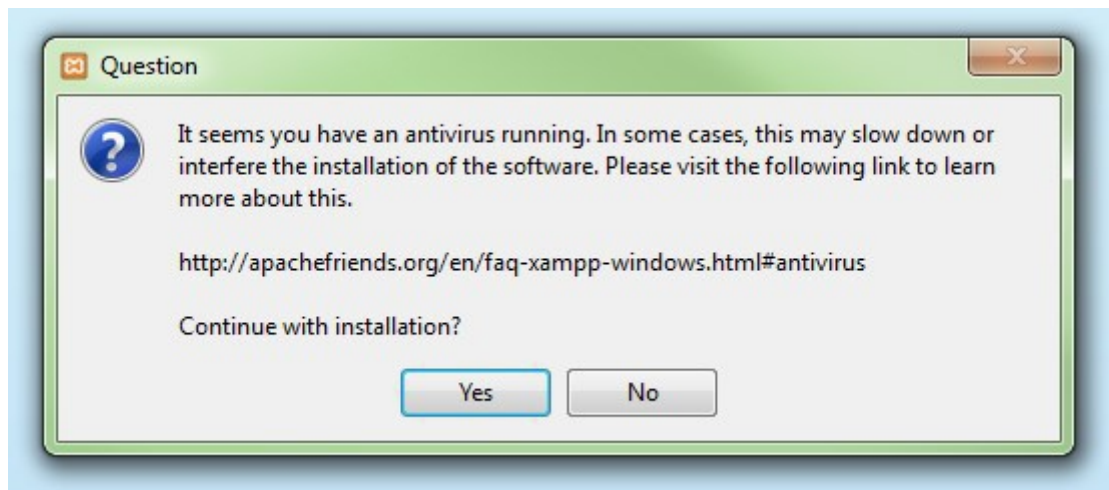
La suite de cette section explique le processus d'installation et d'utilisation de *XAMPP*.

- Téléchargement et lancement de l'installation de *XAMPP*

XAMPP est mis à disposition par le projet à but non lucratif Apache Friends. Les versions avec PHP5.5, 5.6 ou 7 sont téléchargeables gratuitement sous www.apachefriends.org/fr/download.html.

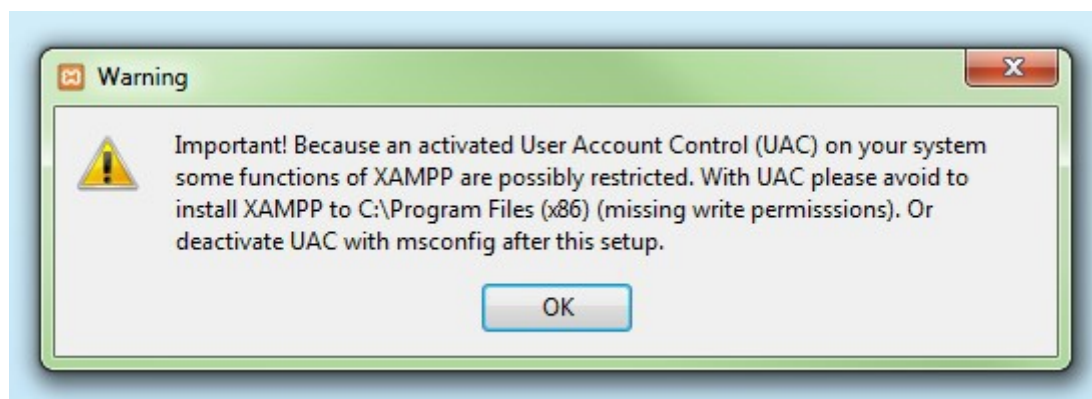
Selon la configuration de votre système, vous avez plusieurs fichiers au choix à télécharger. Téléchargez la version qui convient à votre système. Une fois le logiciel téléchargé, lancez l'installation avec un double-clic sur le fichier avec la terminaison *.exe*.

« Étant donné qu'un programme antivirus actif peut nuire au processus d'installation, il est conseillé de désactiver temporairement ces applications, jusqu'à ce que tous les composants *XAMPP* soient installés avec succès. »



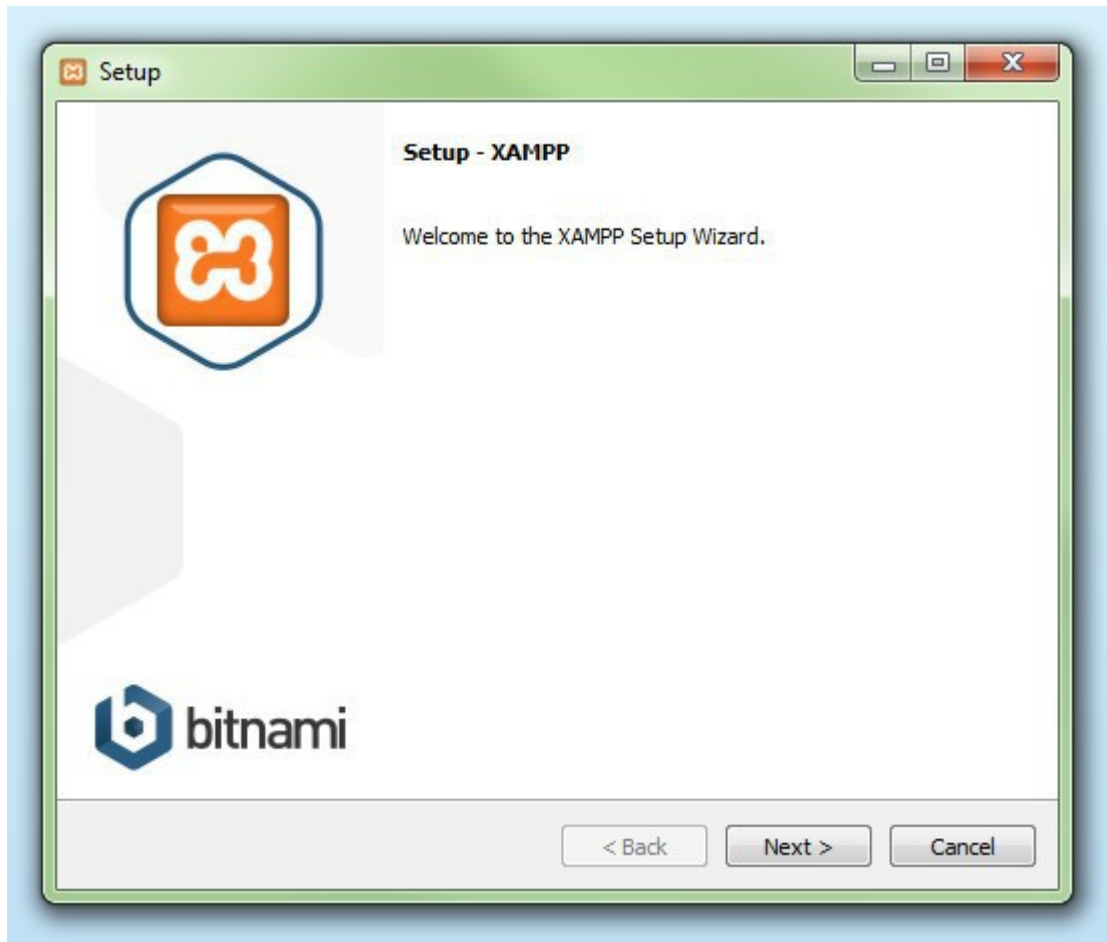
Il est recommandé de désactiver temporairement le programme antivirus avant l'installation XAMPP

Le contrôle du compte utilisateur (UAC pour User Account Control) peut aussi gêner l'installation de XAMPP car les droits d'écriture du lecteur C : sont limités. Le site de l'assistance Windows vous indique comment procéder.



L'User Account Control peut mettre à mal l'installation de XAMPP

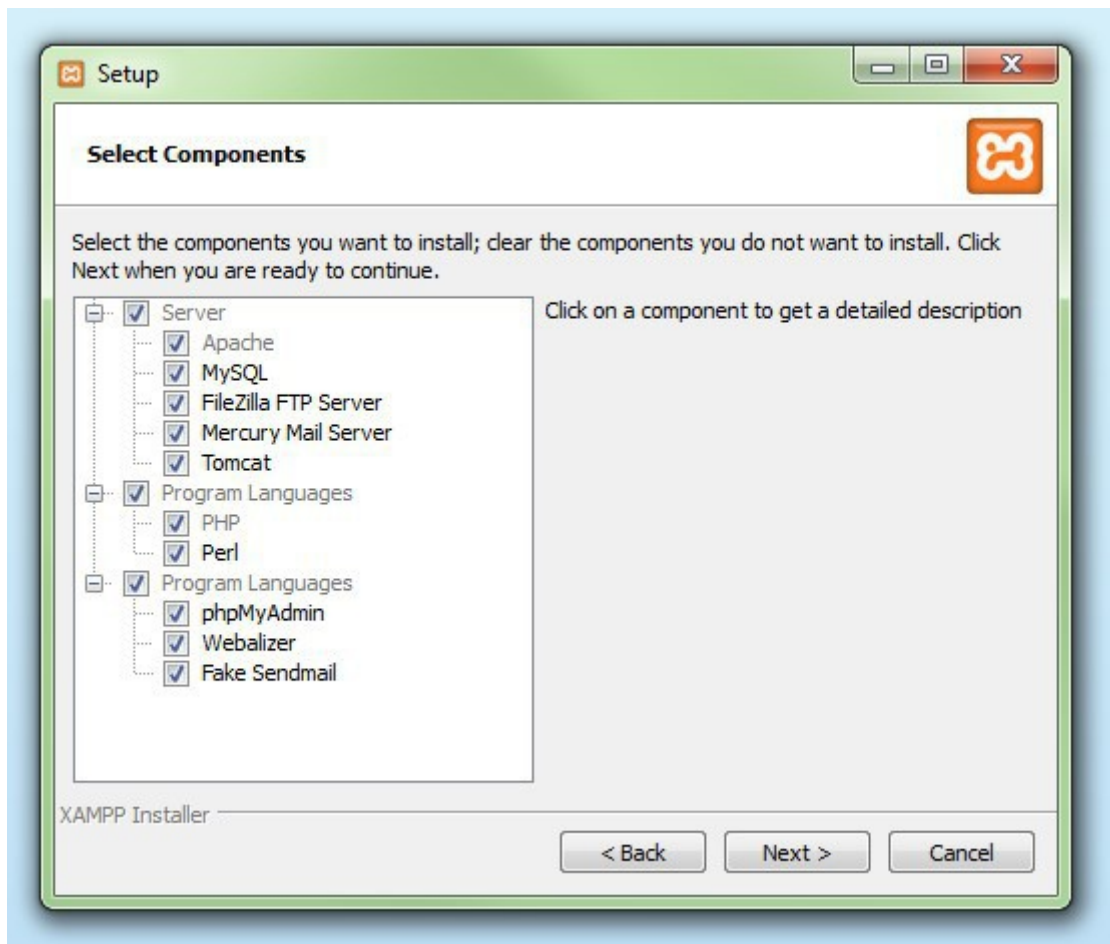
Une fois le fichier .exe exécuté et les programmes de sécurité gênant l'installation écartés, l'écran de démarrage du *Setup-Wizard XAMPP* apparaît. Cliquez sur suivant pour configurer les paramètres d'installation.



L'installation peut commencer depuis l'écran de démarrage de l'assistant , Cliquez sur "Next" pour continuer le processus d'installation.

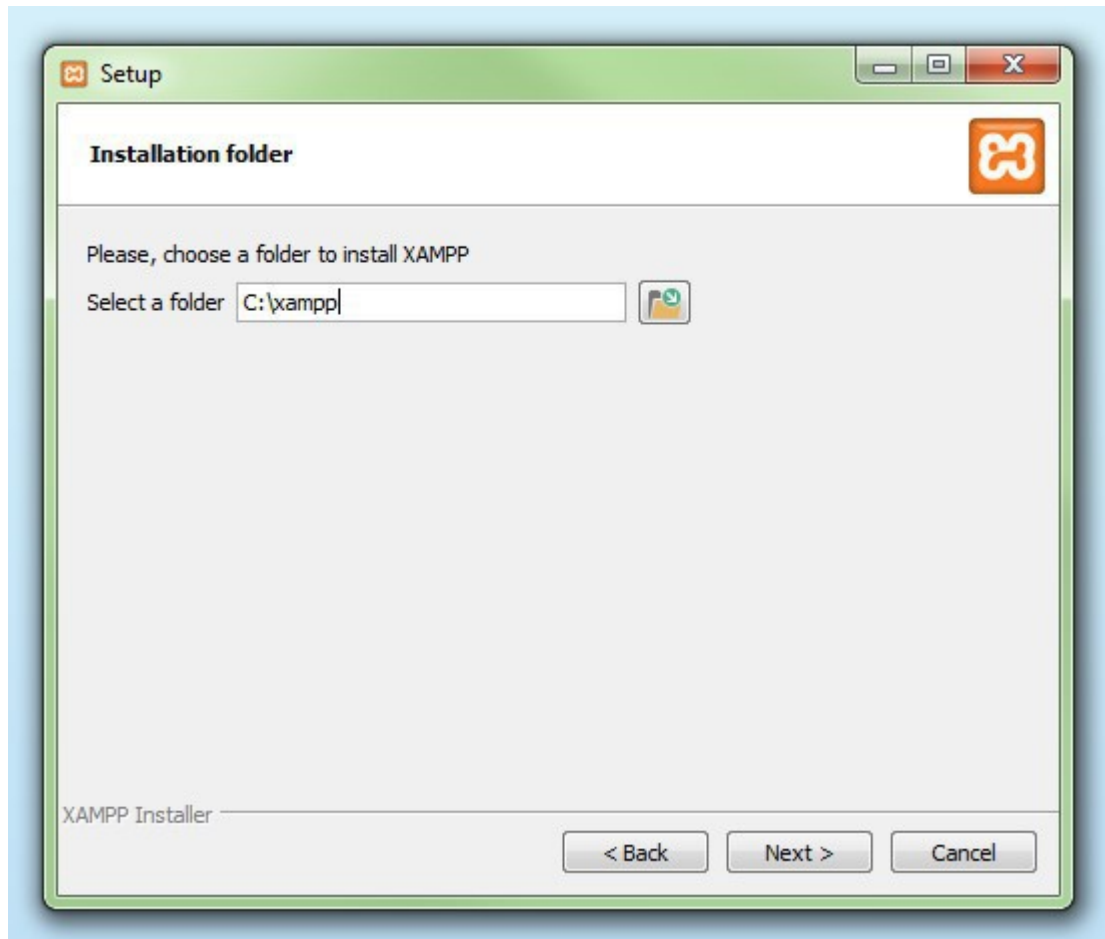
- Choix des composants à installer

Vous avez la possibilité de sélectionner certains composants de l'ensemble XAMPP lors de l'installation sous « choisir les composants ». Dans le cas d'un serveur de test local nous vous recommandons d'opter pour les paramètres par défauts qui entraînent l'installation de tous les composants disponibles. Confirmez votre choix en cliquant sur « *Next* ».



- *Choix du répertoire d'installation*

L'étape suivante vous offre la possibilité de choisir le dossier où installer la suite de logiciels. Ce sera notre espace de travail désormais. Par défaut, un dossier du nom de XAMPP est créé sous C:/, comme l'indique la figure suivante. Conservez donc cette proposition et cliquez sur *Next* ou *Suivant*.



Une fois les réglages par défaut effectués, vous commencez l'installation. Le *Setup-Wizard* décompresse maintenant les composants sélectionnés et les place dans le répertoire choisi. Ce processus peut prendre quelques minutes. L'avancement de l'installation est indiqué via une barre verte.

En général, un message apparaît pour vous indiquer que *votre pare-feu bloque le processus de chargement*. Cochez les cases correspondantes pour autoriser la communication du *serveur Apache avec votre réseau privé ou professionnel*. Tenez compte du fait qu'il n'est pas recommandé de désactiver de telles protections pour un serveur XAMPP dans le cas d'un réseau public.

- Terminer l'installation

Une fois tous les composants décompressés et installés, fermez le setup—wizard en cliquant sur « terminer ». Pour ouvrir XAMPP control panel à la suite de l'installation, cochez la case correspondante. En cliquant sur „*Finish*“, vous fermez l'assistant d'installation.



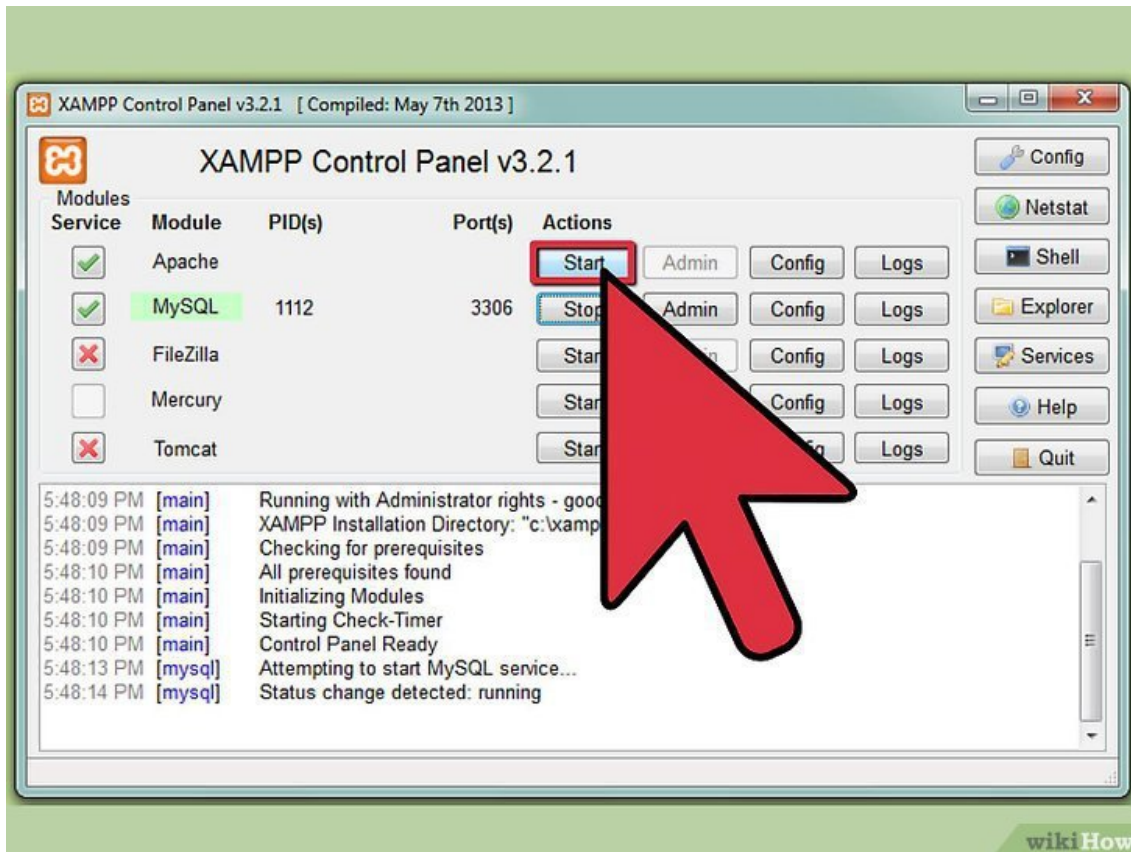
- *XAMPP Control Panel*

Ouvrez XAMPP Control Panel. Double-cliquez sur l'icône *XAMPP Control Panel* depuis le bureau pour ouvrir la fenêtre de contrôle de XAMPP.

XAMPP Control Panel vous permet de contrôler les composants de votre serveur de test. L'interface claire recueille toutes les actions et vous permet de démarrer ou d'arrêter certains modules en un clic. Cette interface propose des actions diverses :

- *Config* : permet la configuration de votre *XAMPP* ainsi que des composants partiels
- *Netstat*: affiche les processus courants sur l'ordinateur local
- *Shell*: ouvre une cellule *UNIX*
- *Explorer*: ouvre le dossier *XAMPP* dans Windows Explorer
- *Services* : affiche les services fonctionnant en arrière-plan
- *Aide* : propose des liens vers des forums pour utilisateurs
- *Terminer* : ferme le *XAMPP-Control-Panel*

Lancez *Apache* et *MySQL*. Dans le panneau de contrôle de *XAMPP*, appuyez sur *Start* pour *Apache* et *MySQL* pour les activer. Vous pouvez aussi lancer les autres éléments.

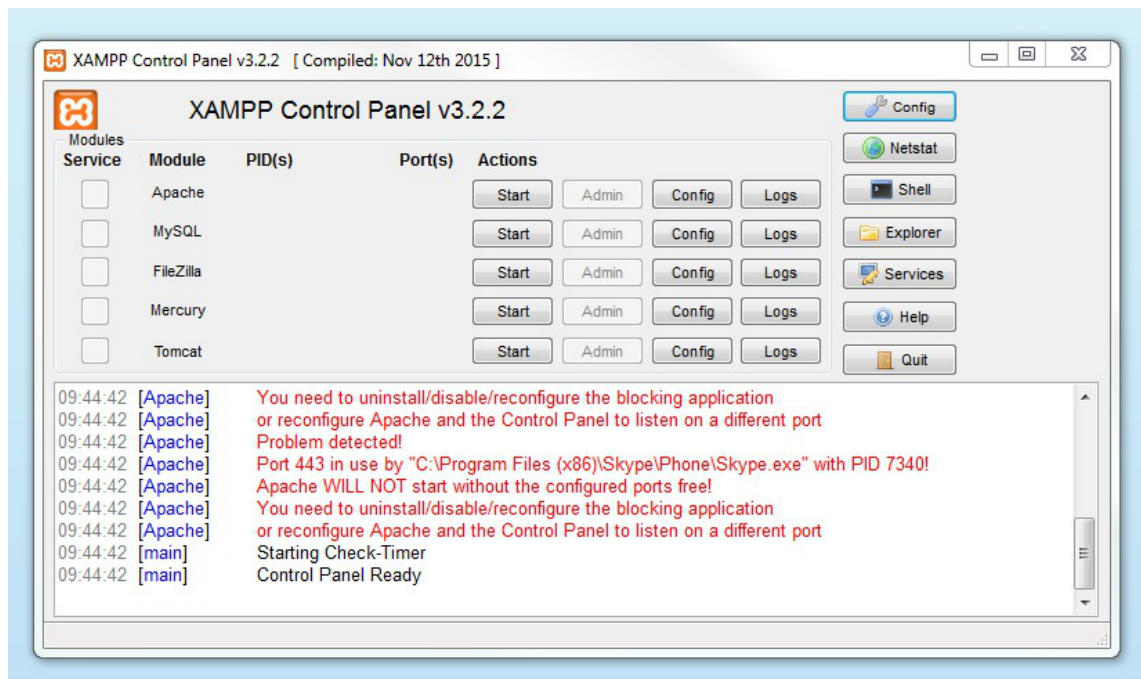


Si un module ne peut pas être lancé en raison d'une erreur, il apparaît surligné en rouge sur le control panel XAMPP. Un rapport d'erreurs détaillé vous aide à identifier les causes de tels problèmes.

Conseil : Aménager XAMPP

Une source d'erreur courante en relation avec Apache est le blocage d'un port. XAMPP renvoie dans ce cas par défaut au port principal 80 et au port SSL 443. Ce dernier est surtout bloqué par d'autres programmes en général. Sur la capture d'écran ci-dessus, on constate un conflit avec le service de messagerie skype (port SSL 443), empêchant le démarrage du serveur Web. Voici trois approches pour résoudre ce type de problème :

- *Modifier le port pour Skype* : pour changer les réglages du port dans Skype, lancez le programme et rendez-vous dans le menu « connexion » puis cliquez sur « actions », « options » puis « avancé ». Décochez la case des ports 80 et 443 pour l'utilisation de connexions supplémentaires entrantes.
- *Modifier les réglages du port Apache* : cliquez sur le bouton de configuration pour le module Apache et ouvrez les fichiers httpd.conf et httpd-ssl.conf. Remplacez les ports 80 dans httpd.conf et 443 dans httpd-ssl.conf par des ports libres quelconques. Enregistrez les fichiers. Utilisez le bouton de configuration général sur le côté droit et choisissez « adapter les services et ports ». Adaptez les ports pour le serveur Apache avec les modifications correspondantes dans le fichier conf.
- *Quitter Skype* : la méthode la plus simple pour contourner des conflits de port avec Skype consiste à quitter le programme puis le relancer si le serveur Apache fonctionne.



2. 2. Test du serveur Web

Maintenant que notre serveur est installé et lancé, nous allons le tester pour voir s'il fonctionne bien. Pour cela :

- Rendez-vous dans le dossier `C : \xampp\htdocs`
- créez un dossier et nommez-le "test"
- Dans ce dossier, créez une page html que vous allez nommer `index.html` avec le code ci-dessous

```

1 <doctype html>
2 <html lang="fr">
3   <head>
4     <meta charset="utf-8">
5     <title> Test du serveur apache web </title>
6   </head>
7   <body>
8     <h1> Mon premier site sur le serveur web local! </h1>
9     <p>Si vous voyez cette page, c'est que le serveur fonctionne</p>
10  </body>
11 </html>

```

Enregistrez votre page `index.html`.

Dans le navigateur Web, saisissez l'adresse suivante `localhost/test` ou encore `127.0.0.1/test`

Vous devez voir votre page `index.html` s'afficher. C'est donc le processus pour définir un site web local sous Xampp :

1. Créer un répertoire du site dans `C : \Xampp\htdocs`
2. Créer une page `index` dans ce répertoire
3. le site est désormais accessible sous `localhost/<nom_du_repertoire>` ou encore `127.0.0.1/<nom_du_repertoire>`

3. 3. Installation de Bootstrap 3

Bootstrap est actuellement à la version 3. Cependant, dans ce tutoriel, nous allons installer la version 3 de ce framework pour des questions de formalisme et de disponibilité de ressource et de

documentation, toutefois que vous serez familiarisés à la version 3, vous pourrez passer à la version 4 de vous mêmes.

- Téléchargement de Bootstrap

Vous pouvez télécharger le framework ici: [Télécharger bootstrap twitter](#)

Ou sur le site officiel: [Bootstrap twitter](#) sur lequel vous aurez l'écran suivant

Download

Bootstrap (currently v3.3.7) has a few easy ways to quickly get started, each one appealing to a different skill level and use case. Read through to see what suits your particular needs.

Bootstrap

Compiled and minified CSS, JavaScript, and fonts. No docs or original source files are included.

Download Bootstrap

Source code

Source Less, JavaScript, and font files, along with our docs. **Requires a Less compiler and some setup.**

Download source

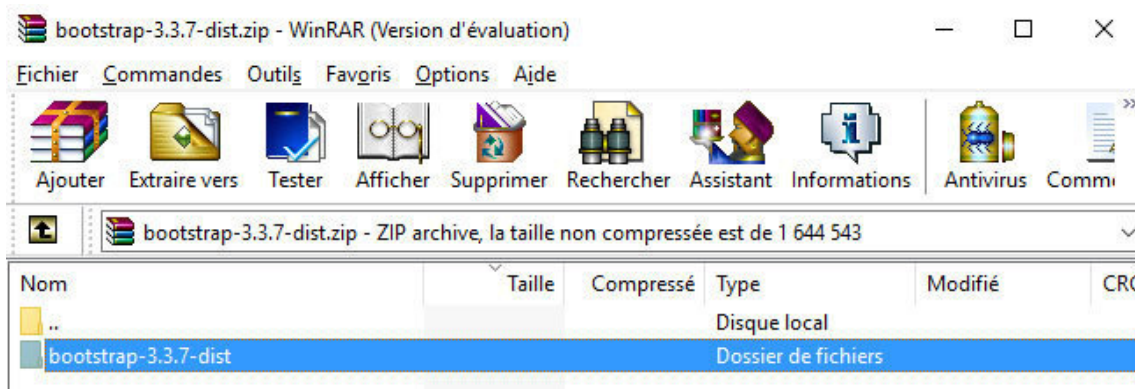
Sass

Bootstrap ported from Less to Sass for easy inclusion in Rails, Compass, or Sass-only projects.

Download Sass

Vous aurez alors à cliquer sur "Download Bootstrap" pour télécharger Bootstrap. Vous pouvez l'enregistrer n'importe où sur votre ordinateur.

Le téléchargement terminé, il vous faudra tout simple « dézipper » le dossier. Rien de plus simple avec 7-Zip ou avec WinRAR.



Que contient le dossier ? 3 autres dossiers qu'il va falloir placer intelligemment pour que tout puisse fonctionner correctement.

- Un dossier *Js*
- Un dossier *css*
- Un dossier *fonts*

Une fois que *bootstrap-3.3.7* a été décompressé, vous allez devoir créer un autre dossier *C : \xampp \htdocs * que nous allons appeler "*monsiebootstrap*", dans lequel vous mettrez les 3 dossiers vu précédemment (après la décompression).

Vous devriez voir quelque chose de semblable à ceci :

monsiebootstrap/

css/

| bootstrap.css

| bootstrap.min.css

| bootstrap-theme.css

```
| bootstrap-theme.min.css
|
| js/
| bootstrap.js
| bootstrap.min.js
|
| fonts/
| glyphsicons-halflings-regular.eot
| glyphsicons-halflings-regular.svg
| glyphsicons-halflings-regular.ttf
| glyphsicons-halflings-regular.woff
```

Il s'agit de Bootstrap sous sa forme la plus simple : des fichiers pré-compilés pour une utilisation rapide dans pratiquement n'importe quel projet Web. Nous proposons du *CSS* et du *JS* compliés (*bootstrap.**), ainsi que du *CSS* et du *JS* compilés et minimisés (*bootstrap.min.**). Les polices de *Glyphicons* sont incluses, tout comme le thème Bootstrap optionnel. La figure suivante explique le contenu de ces dossiers

Dossier	Fichier	Description
css	bootstrap.css	feuille de style de Bootstrap.
	bootstrap.min.css	version minimisée de la feuille de style bootstrap.css.
	bootstrap-theme.css	feuille de style additionnelle contenant des éléments provenant de Bootstrap 2.
	bootstrap-theme.min.css	version minimisée de la feuille de style bootstrap-theme.css.
js	bootstrap.js	fichier JavaScript de Bootstrap.
	bootstrap.min.js	version minimisée du fichier JavaScript bootstrap.js.
font	glyphicons-halflings-regular.eot	Fichiers de police contenant la bibliothèque d'icônes Glyphicon
	glyphicons-halflings-regular.svg	
	glyphicons-halflings-regular.ttf	
	glyphicons-halflings-regular.woff	

Source : <http://www.opentuto.com/installation-de-bootstrap-3/>

Qu'est ce qu'un fichier css/js minimisé?

Un fichier *css/js* minimisé est un fichier qui reprend le même contenu du fichier original en supprimant tous les sauts de ligne, commentaires et espaces inutiles.

Pour des raisons de performances, on opte souvent pour les fichiers minimisés dans les environnements de production et les fichiers originaux pendant la phase de développement.

Avantages et inconvénients d'un fichier css/js minimisé:

Lorsqu'on utilise les fichiers minimisés dans nos projets, les pages web qui l'utilisent, bénéficient d'un temps de chargement beaucoup plus amélioré par rapport aux pages utilisant les fichiers originaux, cela est dû, à la taille des fichiers après leur minimisation (suppression de tous les sauts de ligne, commentaires et espaces inutiles).

Cependant, un fichier minimisé est difficile à lire et à modifier.

- Lancement de notre première page avec Bootstrap

Créez une page `index.html` dans le dossier *monsiebootstrap* et ajoutez-y le code suivant

```

1 <!doctype html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">
7     <!-- liaison avec le CSS téléchargé de Bootstrap -->
8     <link href="css/bootstrap.min.css" rel="stylesheet">
9     <!-- liaison avec JQuery (CDN Google) -->
10    <script src=
11      "https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
12      <!-- Appel du JavaScript téléchargé de Bootstrap -->
13    <script src="js/bootstrap.min.js"></script>
14    <title>Hello world avec Bootstrap 3 </title>
15  </head>
16  <body>
17    <h1>Hello world!</h1>
18    <p> Bienvenue sur ma page web avec Bootstrap !</p>
19  </body>
20 </html>

```

Il s'agit d'une page Web classique avec quelques éléments spécifiques que vous avez déjà vu :

une balise `<link>` vers la feuille de style Bootstrap ;

une balise `<meta name="viewport">` définissant le *viewport*, c'est-à-dire la surface de la fenêtre du navigateur qui affiche la page. Cela va permettre à la page de s'adapter à la dimension du terminal client ;

une balise `<meta>` permettant un affichage correct sur le navigateur Internet Explorer ;

des balises `<script>` qui activent les plugins jQuery de Bootstrap. Ces plugins sont nécessaires pour animer les composants Bootstrap.

Remarque

jQuery est ici inclus en ligne via le *CDN Google*. Il est également possible de le télécharger et de l'installer localement.

Ouvrez cette page dans votre navigateur Web favori. Vous obtenez le résultat ci-dessous.

Lancez alors le site dans votre navigateur avec son adresse locale : `localhost/monsiebootstrap` et vous verrez que ça fonctionne !

4. Exercice

Exercice

1) Le logiciel XAMPP inclu des outils qui seront dans le

- ☐ front-end
- ☐ back-end

Exercice

2) Quel module de Xampp dois-je démarrer obligatoirement pour accéder à mes sites en local avec l'adresse localhost ?

- ☐ Apache
- ☐ Tomcat
- ☐ Filezilla
- ☐ http

Exercice

3) Quels sont les répertoires présents dans l'archive de bootstrap 3 quand vous la téléchargez ?

- ☐ Bootstrap
- ☐ JS
- ☐ Fonts
- ☐ JQuery
- ☐ CSS
- ☐ Htdocs

Exercice

4) Dans quel répertoire du dossier de xampp doit on créer les dossiers de nos sites web en local ?

Exercice

5) La structure générale d'une page avec Bootstrap est-elle différente de la structure d'une page HTML en général ?

- ☐ Oui , évidemment puisque c'est Bootstrap
- ☐ Non, elles se basent toutes sur la même structure générale d'un document web !