











Les sites et outils de référence pour développer

Sommaire

Introduction.....	3
I. Les sites de référence et d'entraide	4
A. Les sites officiels de référence	4
B. Les sites d'entraide	5
II. Les outils utiles pour développer	6
A. Les éditeurs de code	7
B. Les générateurs de code.....	8
C. Les CMS (Content Management System).....	9

Crédits des illustrations :
©DR, Fotolia.

Les repères de lecture

 Retour au chapitre	 Définition	 Objectif(s)	 Espace Élèves	 Vidéo / Audio
 Point important / À retenir	 Remarque(s)	 Pour aller plus loin	 Normes et lois	 Quiz



Introduction

Voici quelques outils et moyens indispensables pour vous aider à trouver ou à développer la solution web la plus adaptée à votre projet.

Nous allons vous présenter des ressources mais aussi des pratiques qui permettent d'obtenir des informations sur un langage ou des éléments de code déjà conçus.

I. Les sites de référence et d'entraide

La méthodologie d'un travail de développement passe nécessairement par une recherche permanente d'information sur l'actualité des langages, les scripts associés, ou encore de nouvelles solutions.

Le Web est à dimension mondiale, il faut s'habituer à ce que certaines ressources ne soient disponibles qu'en anglais.

Avoir des notions d'anglais est donc un plus lorsque l'on souhaite faire de la programmation.

A. Les sites officiels de référence

Les sites officiels sont des sites créés et maintenus par les éditeurs de ces solutions ou par des organismes de confiance reconnus.

Les éditeurs de langages fournissent rarement, si ce n'est jamais, de documentation papier de leurs langages. De par leur nature, il est plus simple, plus ergonomique et moins coûteux de rendre ces documentations accessibles via Internet.

Les sites officiels des langages sont les références les plus sûres quant à la qualité de l'information diffusée. Vous y trouverez facilement la liste des instructions disponibles.

En règle générale, ces informations sont bien organisées et accessibles. Cependant, les sites officiels manquent parfois d'explication, et surtout, de retour d'expérience.

Vous trouverez ci-dessous une **liste des documentations officielles** les plus importantes sur les langages que nous avons abordés dans ce cours :

– Le HTML :

<http://www.w3.org/TR/html401/>. Cette documentation est éditée par le W3C qui est un organisme référent de confiance dans le domaine de la programmation web.

– Le JavaScript :

Le cas de ce langage est particulier. Il existe une documentation officielle, mais comme les navigateurs intègrent bien souvent leurs propres moteurs d'interprétation du JavaScript, il arrive que certaines instructions soient légèrement différentes d'un navigateur à l'autre. Il est donc conseillé de se tourner directement vers les documentations produites par les éditeurs de ces navigateurs.

En ce qui concerne Firefox, visitez https://developer.mozilla.org/fr/Firefox/Developer_Edition

Si vous devez développer pour le grand public, vous serez amené à consulter ces documentations, voire plus, afin de rendre vos scripts compatibles avec le plus de navigateurs possibles.

– **Le CSS :**

<http://www.w3.org/TR/CSS2>

Cette documentation est éditée par le W3C qui est un organisme référent de confiance dans le domaine de la programmation web.

– **Le PHP :**

<http://www.php.net/manual/fr>

Ce site est particulièrement bien réalisé. Par expérience, c'est le site de documentation le plus pratique et le plus efficace. Il intègre une référence du langage et un forum d'utilisateurs particulièrement bien organisés.

– **Le Java :**

<http://download.oracle.com/javase/6/docs/api/>

Cette documentation est aussi très bien réalisée grâce à une ergonomie très efficace.

– **Le C# :**

[https://msdn.microsoft.com/fr-fr/library/618ayhy6 \(v = vs.140\).aspx](https://msdn.microsoft.com/fr-fr/library/618ayhy6 (v = vs.140).aspx)

Cette documentation est très complète, cependant il est parfois difficile de s'y repérer.

– **Le SQL :**

Tout comme pour le JavaScript, les instructions de ce langage peuvent varier en fonction du moteur de base de données utilisé (Oracle, MySQL, ou autre...).

– Pour Oracle, consultez http://download.oracle.com/docs/cd/B28359_01/server.111/b28286/toc.htm.

– Pour MYSQL, consultez <https://dev.mysql.com/doc/>.

N'hésitez pas à rechercher sur Internet pour retrouver les documentations des autres technologies abordées dans ce cours. Ce type de ressources se trouve en général facilement.

B. Les sites d'entraide

On entend ici par site d'entraide les sites de documentation non officiels qui proposent en général des documentations orientées vers une utilisation particulière, dans un domaine précis.

Ces derniers sont très nombreux pour les langages du Web, et dispensent en général des informations pertinentes.

Par ailleurs, n'hésitez pas à rechercher des informations sur les sites des facultés et écoles, elles proposent parfois des documentations et des tutoriels très utiles et pédagogiques.

Ces sites d'entraide sont pratiques et accessibles, cependant, il faut être vigilant en les utilisant, car rien ne garantit la véracité des informations diffusées.

Ils sont parfois écrits par des personnes dont le niveau technique est difficilement vérifiable.

Les forums sont, quant à eux, très utiles lorsque vous avez des problèmes pour implémenter un comportement.

Les sites permettent d'échanger avec d'autres internautes, parfois professionnels, voire même avec des programmeurs ayant participé à l'élaboration de langages.

Cela dit, il est en général difficile d'identifier le niveau des participants, mais comme les forums sont des échanges, il devient alors plus simple de se rendre compte du niveau des personnes avec qui vous serez en contact.

Vous trouverez ci-dessous quelques sites qui peuvent s'avérer très utiles :

– <http://fr.wikipedia.org/>

Ce site est un wiki, c'est-à-dire un site dont les pages sont modifiables par les internautes.

Ce qui veut dire que les informations que vous y trouverez ne sont pas forcément vérifiables. En général, elles sont de bonne qualité, mais il faut les vérifier.

– <http://www.developpez.com/>

Ce site est très bien réalisé et propose des informations très pertinentes. Il propose des tutoriels de bonne qualité ainsi que des forums sur lesquels vous pourrez échanger sur les problèmes les plus souvent rencontrés.

Il existe bien d'autres sites, n'hésitez donc pas à rechercher d'autres ressources par vous-même sur le Web.

Règles d'or :

- Utiliser les diverses sources d'information sur les langages à disposition sur le Web est indispensable pour créer un site techniquement actuel ;
- Utiliser les documentations officielles d'un langage est toujours préférable, elles constituent une référence à l'international.

II. Les outils utiles pour développer

Certains de ces outils vous accompagneront durant les phases de développement et de test de vos sites. D'autres vous permettront même de directement générer du code. Nous vous présenterons dans cette partie, les avantages et inconvénients de ces solutions.

A. Les éditeurs de code

Ces outils sont incontournables puisqu'ils permettent de réaliser, de coder, des programmes.

Le plus populaire à ce jour est « **Notepad ++** », un éditeur de texte léger. Ce dernier vous permettra de créer aisément les sources de vos sites.

Cependant, les fichiers que vous aurez à créer peuvent rapidement devenir d'une taille conséquente. Ils devront aussi respecter une syntaxe stricte, ce qui rendra vos sources difficilement compréhensibles, surtout si vous en écrivez beaucoup.

Par exemple, il peut devenir problématique d'analyser un code que vous n'avez pas écrit ou alors qui a été écrit plusieurs mois auparavant.

Nous avons donc choisi de vous présenter quelques fonctionnalités les plus utilisées par les développeurs, ainsi que certains des logiciels qui les proposent.

1. Coloration de la syntaxe

La plus courante des aides au développement est la coloration de syntaxe. **Elle colore certaines instructions du langage utilisé, ce qui permet de retrouver rapidement des parties identifiées d'un code que vous êtes en train d'écrire.**

De plus, **elle permet d'identifier facilement la plupart des erreurs de syntaxe** que vous pourriez commettre en temps réel. L'utilisation de cette fonctionnalité ergonomique représente un gain de temps et un confort considérable lorsque l'on programme.

Il existe plusieurs éditeurs de texte, simples d'approche, proposant cette fonctionnalité : Notepad ++, Sublime Text, UltraEdit sont des solutions bien réalisées.

2. Complétion de code



Lorsque l'on tape du code, l'éditeur pourvu de cette fonctionnalité suggère des mots-clés, des instructions, voire même des paramètres.

Certains éditeurs intègrent, ou vous permettent de configurer, un compilateur de code. En fonction de l'éditeur utilisé, ces suggestions peuvent être plus ou moins pertinentes. Cela dit, elles s'avèrent en général d'une aide précieuse, car cela vous permet d'avoir en temps réel des propositions d'instructions à utiliser.

Malheureusement, peu d'outils proposent la complétion de code.

L'éditeur « Notepad ++ », par exemple, propose cette fonctionnalité, même si elle n'est pas intégrée par défaut.

D'autres éditeurs, « Dreamweaver », « Éclipse » et « Visual Studio » l'offrent parmi d'autres fonctionnalités.

Cette fonctionnalité est adaptée aux langages ayant besoin d'être compilés pour pouvoir être exécutés : les langages Java et C#, par exemple.

Il existe pléthore de ces environnements pour ces deux langages, cependant, deux logiciels restent des références incontournables :

- **Pour le Java, « Éclipse »** est une solution très complète et gratuite. Ce logiciel peut paraître compliqué de prime abord, cependant il existe une communauté très importante et de nombreux tutoriels peuvent être trouvés sur le Net ;
- **En ce qui concerne le C#, le meilleur éditeur reste aujourd'hui « Visual Studio », une solution Microsoft** qui a le mérite d'intégrer parfaitement les spécificités de ce langage. Malheureusement cet environnement est payant.

3. Le débogage

Une fois un site réalisé, la plupart du temps vous découvrirez des bugs. Le débogage fait partie intégrante du cycle de vie d'un programme en général et donc d'un site en particulier.

Les langages compilés sont en général plus complexes, et donc, les bugs générés sont plus difficiles à identifier. Pour vous aider dans cette tâche, les éditeurs les plus complets proposent des fonctionnalités de débogage.

Nous pouvons citer « Visual Studio » et « Éclipse » comme outils de débogage couramment utilisés.

Cependant, le cas des langages client est légèrement différent. En effet, comme ils s'exécutent sur le navigateur plutôt que sur le serveur, le rôle du débogueur incombe naturellement aux navigateurs.

Pour Firefox, il existe plusieurs modules très efficaces, le plus répandu étant « Web Developer ».

B. Les générateurs de code

L'une des fonctionnalités de ces générateurs, la plus convoitée, est le **WYSIWYG : What You See Is What You Get** ; comprenez : « ce que vous voyez est ce que vous obtenez ». Cette fonctionnalité est particulièrement adaptée à la réalisation de pages HTML.

Il s'agit de modules qui permettent, grâce à des interfaces graphiques, de définir la présentation de votre programme sans avoir à saisir de code.

Exemple de générateur offrant cette fonctionnalité : l'outil le plus connu est « Dreamweaver ». Cependant il s'agit d'une solution payante.

Microsoft propose aussi une suite logicielle dédiée au développement Web : « Expression Web ».

Elle est très orientée vers les produits Microsoft, c'est-à-dire qu'elle est très efficace lorsqu'il s'agit de développer un site pour Internet Explorer/ Microsoft Edge. Elle s'intègre aussi très bien avec « Visual Studio ».

Mais, elle s'avère plus complexe lorsqu'il s'agit de produire des sites compatibles avec tous les navigateurs.

Ces générateurs sont très séduisants, mais ils doivent être utilisés avec parcimonie, car le code généré n'est pas forcément optimisé pour le projet développé, ce qui peut parfois entraîner des failles de sécurité.

Beaucoup d'éditeurs offrent des fonctionnalités de génération totale de code.

Ces outils, après vous avoir posé un certain nombre de questions, vous produiront du code réalisant les fonctionnalités demandées.

Le code produit est, en général, très lourd, difficilement modifiable et, bien souvent, intègre des failles de sécurité évidentes.

Il existe, dans le monde de la programmation, diverses solutions très efficaces pour générer du code, mais leur mise en place demande une architecture imposante et donc beaucoup de travail, ce qui s'avère disproportionné si vous souhaitez créer un site vitrine. Elles deviennent intéressantes dans le cas de sites de gestion d'activités qui sont amenés à évoluer régulièrement.

C. Les CMS (Content Management System)

Un **CMS (Content Management System)**, comprenez: « **système de gestion de contenu** », est un logiciel qui permet de créer et de gérer tout le contenu d'un site web.

Il permet de modifier les pages créées, d'en supprimer certaines, voire toutes, ou bien de les rendre non visibles provisoirement ou définitivement.

La mise en page générale de votre site web passe par le choix d'un thème (ou template en anglais) qui définit la forme du site, ce choix est modifiable à tout moment. Il est possible de changer l'aspect du site en améliorant le design ou bien en le changeant complètement tout en gardant le contenu déjà créé ou publié.

L'utilisation du CMS permet de structurer le contenu : blogs, FAQ, forum de discussion, etc.

Un site web utilisant un CMS est donc un site destiné à s'enrichir, par son contenu et par ses capacités fonctionnelles.

Cette solution est adaptée au **travail collaboratif**, plusieurs administrateurs peuvent y intervenir.

Tous les CMS requièrent du PHP pour leur fonctionnement.

Les CMS ont de nombreux avantages, ils permettent notamment de :

- gérer des contenus simplement;
- faire des mises en ligne automatiques;
- déléguer des publications;
- gérer les droits des différents intervenants.

Un CMS possède **un système de workflow** (chaîne de publication) permettant aux différents intervenants de travailler en même temps sur le contenu du site avec validation par un responsable.

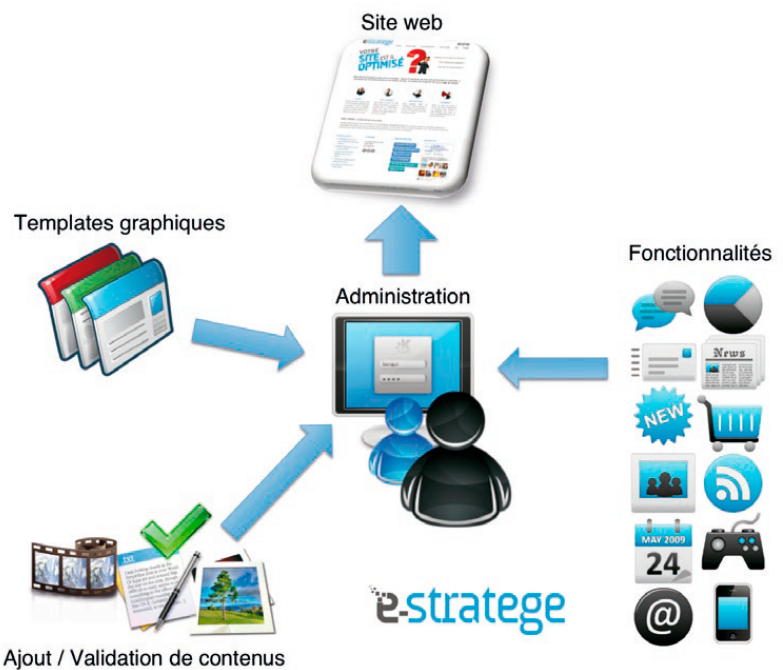


Fig.1 CMS. Principe du fonctionnement. Source : e-strategie.com

Les CMS les plus connus sont : Wordpress, Drupal, Joomla, PrestaShop.

Il peut être très intéressant de développer soit même son CMS, en fonction des objectifs de son site, afin d'en faire un outil ultra adapté aux particularités de son site web, pour garder la main sur son évolution quotidienne par exemple, ou la création de nouvelles fonctionnalités.