

Initiation au développement Web

Ressource R209

David Gatel - david.gatel@univ-rennes1.fr

Michel Guilleméau - michel.guilleméau@univ-rennes1.fr

#1 Présentation

Ressource R209

Objectifs de la ressource R209

RT1-Administrer Niveau 1	RT2-Connecter Niveau 1	RT3-Programmer Niveau 1
<p>AC0112 Comprendre l'architecture des systèmes numériques et les principes du codage de l'information</p> <p>AC0114 Maîtriser les rôles et les principes fondamentaux des systèmes d'exploitation afin d'interagir avec ceux-ci pour la configuration et administration des réseaux et services fournis</p>		<p>AC0311 Utiliser un système informatique et ses outils</p> <p>AC0312 Lire, exécuter, corriger et modifier un programme</p> <p>AC0313 Traduire un algorithme, dans un langage et pour un environnement donné</p> <p>AC0314 Connaître l'architecture et les technologies d'un site Web</p> <p>AC0315 Choisir les mécanismes de gestion de données adaptés au développement de l'outil</p> <p>AC0316 S'intégrer dans un environnement propice au développement et au travail collaboratif</p>

Objectifs de la ressource R209

Prérequis	R107 Fondamentaux de la programmation R109 Introduction aux technologies Web R207 Sources de données R208 Analyse et traitement de données structurées
Descriptif	<p>Le professionnel R&T peut être amené à développer, pour ses besoins personnels ou pour ses collaborateurs, un site Web, par exemple pour fournir une interface de présentation du réseau informatique : compétence «Créer des outils et applications informatiques pour les R&T» (RT3).</p> <p>Il doit en appréhender tous les éléments : il doit aussi bien connaître les protocoles de communication du Web que veiller à la sécurité de ceux-ci. Il doit également pouvoir accéder, traiter et afficher des informations provenant de différentes sources de données telles que des SGBD, des API ou des fichiers structurés. La présente ressource contribue aux apprentissages critiques mentionnés précédemment.</p>
Contenus	<ul style="list-style-type: none"> • Introduction au protocole HTTP. • Mise en forme de pages Web : <ul style="list-style-type: none"> ◦ balises HTML avancées ; ◦ structure d'une page avec son DOM ; ◦ CSS avancé ou Framework ; ◦ initiation au dynamisme côté client (JavaScript, bibliothèques comme jQuery) • Scripts côté serveur. • Eléments d'interaction client-serveur (requête HTTP, URL, formulaire). • Interrogation d'un SGBD ou d'une API. • Sensibilisation à la sécurisation de sites : failles XSS, XSS stockée, injections SQL. <p>L'utilisation de l'anglais est préconisée dans la documentation du code.</p>
Mots-clés	Web, Développement, Algorithmes, SGBD, API, Sécurité, Client-serveur.

Organisation de la ressource R209

1 CM

7 TD

- En salle info cours/TD

2 TP

- 2 mini-projets à développer en 4h

1 examen

- sur papier
- avec un document autorisé (équivalent d'une feuille A4 recto-verso)

Technologies utilisées

HTML



CSS et framework Bootstrap



JavaScript



Apache



PHP



JSON



Plan de la ressource R209

1. Présentation du PHP
2. Les bases du langage PHP
3. Données en PHP
4. PHP et sécurité
5. PHP et autres langages

Web statique, web dynamique

Web statique

Page écrite en HTML et CSS

Ne tient pas compte de
l'utilisateur

Suffisant pour une page de
présentation d'un contenu sans
interaction avec l'utilisateur.

Web dynamique

Page écrite avec un langage de
programmation évolué, exécuté
côté serveur

Tient compte de l'utilisateur à
travers ses requêtes ou action.

Architecture Web Client / Serveur

Le **client**, c'est le **navigateur** de l'utilisateur qui est installé sur un ordinateur, un smartphone.

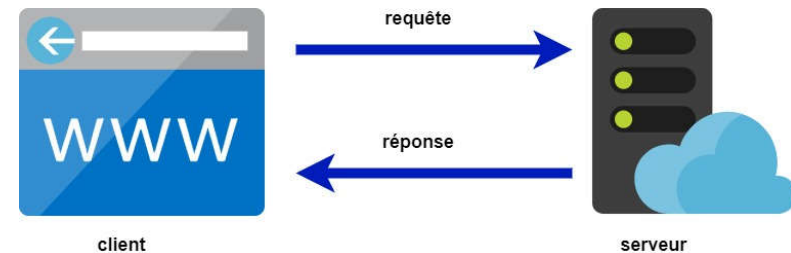
Le **serveur HTTP** est installé sur une machine physique ou virtuelle, localement ou bien à distance.

Le serveur héberge un service qui stocke les différents fichiers qui composent le site web (.html, .css, .php, .js, ...) et les rend « accessibles » à partir des requêtes.

Le **client émet** des **requêtes** vers le serveur.

Le **serveur HTTP répond** aux requêtes du client.

Le **client affiche** la réponse.



Architecture Web

Client / Serveur

Le **serveur** HTTP va **recevoir** la **requête**.

Il va **lire, interpréter** et **exécuter** le code.

Puis, il **renvoie** le résultat, généralement sous la forme de code HTML, au navigateur.

Le client **interprète** la réponse du serveur qui est composé de :

HTML pour la structure et le contenu de la page web

CSS pour la forme de la page web

JavaScript pour modifier dynamiquement le contenu ou le format de la page en fonction des interactions de l'utilisateur

Langages du web dynamique

PERL

interprété,

polyvalent,

adapté au traitement et à la manipulation de fichiers texte, notamment du fait de l'intégration des expressions régulières dans la syntaxe même du langage, puissant et, même si le PHP et Java l'ont supplanté dans le développement Web, qui reste une solution tout à fait pertinente pour la programmation de scripts.

JavaScript

est un langage de programmation de scripts principalement utilisé dans les pages web interactives

généralement côté client : du code javascript est exécuté par le navigateur du client.

JS est un langage orienté objet à prototype

Langages du web dynamique

PHP

langage de scripts libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP,

mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale, en exécutant les programmes en ligne de commande.

ASP

est un langage interprété proposé par Microsoft.

Les applications ASP sont des programmes conçus à partir de plusieurs sources :

Visual Basic Script (VBScript), ou JScript (le javascript de Microsoft)

composants et objets ASP (fonctionnalités propres au serveur IIS (Internet Information Server))

Les scripts ASP peuvent effectuer de nombreuses tâches comme accéder à des bases de données, générer un contenu HTML, gérer des dossiers et autres fichiers directement sur le serveur hôte, etc.

Langages du web dynamique

Ruby

est un langage interprété entièrement orienté objet, et multi plate-forme.

est très intuitif.

peut servir de langage de script au même titre que PERL ou d'autres, mais son essor récent est surtout dû à l'apparition de Ruby on Rails, un framework web extrêmement puissant basé sur Ruby.

Python

En tant que puissant langage de scripting adapté à la fois au prototypage rapide et à de plus gros projets, Python est largement utilisé dans le développement d'applications web.

Langages du web dynamique

© W3Techs.com	usage	change since 1 February 2020
1. PHP	78.8%	-0.2%
2. ASP.NET	10.5%	-0.1%
3. Java	3.5%	+0.1%
4. Ruby	3.3%	+0.2%
5. static files	1.7%	-0.1%

percentages of sites

© W3Techs.com	usage	change since 1 August 2021
1. PHP	78.7%	-0.4%
2. ASP.NET	8.2%	-0.2%
3. Ruby	5.4%	+0.4%
4. Java	3.7%	+0.1%
5. Scala	2.1%	+0.1%

percentages of sites

© W3Techs.com	usage	change since 1 January 2022
1. PHP	78.0%	-0.1%
2. ASP.NET	7.9%	-0.1%
3. Ruby	6.1%	+0.1%
4. Java	3.7%	
5. Scala	2.3%	

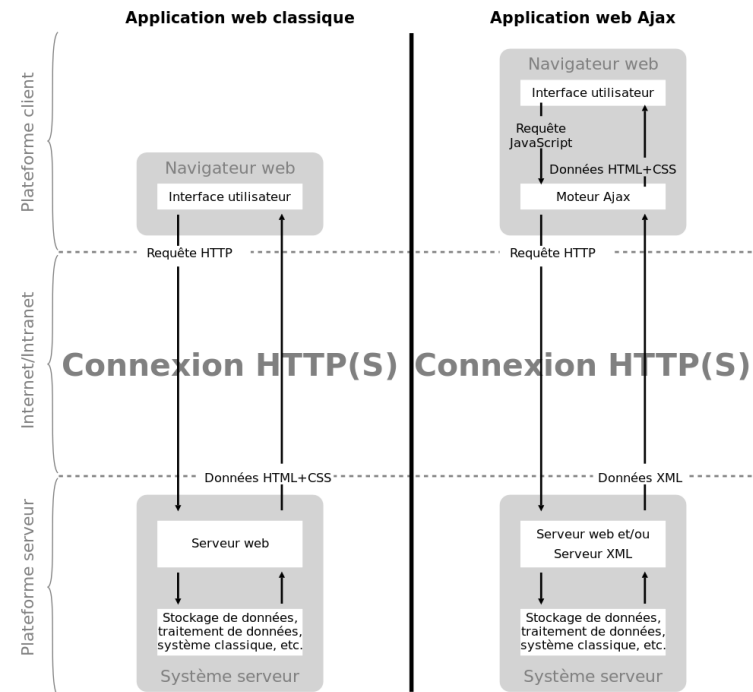
percentages of sites

<https://w3techs.com/>

Langages du web dynamique

Ajax

est une architecture informatique qui permet de construire des applications Web et des sites web dynamiques interactifs sur le poste client en se servant de différentes technologies ajoutées aux navigateurs web (JavaScript et XML)



Langage PHP

PHP permet plusieurs choix de programmation de scripts:

- programmation classique dite procédurale, ou
- programmation objet.

C'est un langage interprété :

- ne nécessite pas d'être compilé (comme en C).
- présence d'un interpréteur côté serveur.

PHP

- est facile d'accès pour le web.
- intégré au code HTML. S'imbrique dans le code HTML en étant délimité comme tel.
- ne nécessite pas de répertoires spéciaux comme le fameux "/cgi-bin" pour les CGI.
- s'inspire de divers langages comme C, Perl, C++ et Java. Facilite donc son apprentissage.
- au même titre que Perl, Python ou TCL, il est capable de lancer des scripts interactifs ou non.
- peut aussi servir à créer des interfaces graphiques (extension GTK).
- interface simple avec beaucoup de fonctionnalités au SGBD (Système de Gestion de Base de données).

Interpréteur PHP

page **bonjour.php** demandée par le navigateur au **serveur**

Le serveur envoie le résultat de l'interprétation de la page **bonjour.php** au **client** navigateur.

```
<html>
<head>
<title>Test PHP</title>
</head>
<body>
<?php
echo '<p>Bonjour le monde</p>';
?>
</body>
</html>
```

```
<html>
<head>
<title>Test PHP</title>
</head>
<body>
<p>Bonjour le monde</p>
</body>
</html>
```



Caractéristiques et configuration du PHP

Le fichier de configuration (php.ini) est lu par PHP au démarrage. Si vous avez compilé PHP en module, le fichier n'est lu qu'une seule fois, au lancement du serveur web.

La liste des directives du fichier php.ini vous permet de définir votre installation de PHP

Où une directive de configuration peut-elle être modifiée ?

Certaines directives peuvent être modifiées dans un script PHP avec la fonction `ini_set()`, alors que d'autres ont besoin d'être modifiées dans les fichiers `php.ini` ou `httpd.conf`.

Historique du PHP

PHP/FI (Personal Home Page/Forms Interpreter) conçu en **1995** par Rasmus Lerdorf.

PHP/FI 2.0 sort en **1997**, développé par R. Lerdorf. 50000 sites (1% des noms de domaines) disaient l'utiliser ou le supporter.

- gestion de formulaires et l'accès à la base de données mSQL permet au langage de se développer rapidement.

PHP 3.0 en **1998**, réalisé par R. Lerdorf, A. Gutmans et Z. Suraski.

- est modulaire et extensible ; il est possible de lui greffer des API supplémentaires, avec le support de nombreuses bases de données, de formats et de protocoles. Il dispose d'une syntaxe plus cohérente et d'un support basique de l'objet. Il a été installé sur 10% du parc mondial des serveurs web avec des dizaines de milliers de développeurs et des centaines de milliers de sites.

PHP 4.0 en **2000**, l'analyseur PHP est migré sur le moteur d'analyse Zend afin d'offrir de meilleures performances et de supporter un plus grand nombre d'extensions. Un nouveau moteur a vu le jour: Zend Engine (contraction de Zeev et Andi).

- apporte des performances accrues, le support de plusieurs serveurs web, les sessions, une sécurité accrue.

PHP 5.0 en **2004**, intègre le développement orienté objet

PHP 7.0 le 3 décembre **2015** - PHP 7.4.28 le 17 février 2022

PHP 8.0.1 le 7 janvier **2021** - PHP 8.1.3 le 17 février 2022

Environnement de développement

Il faut

un serveur web HTTP

avec un interpréteur PHP

+ un serveur de base de données (facultatif)

Serveur HTTP

- Apache

Interpréteur PHP

- Version 7+
- Version 8+

Serveur de base de données :

- MySQL

© W3Techs.com	usage	change since 1 February 2020
1. Apache	40.4%	-0.8%
2. Nginx	31.8%	+0.2%
3. Cloudflare Server	13.2%	+0.4%
4. Microsoft-IIS	7.8%	-0.1%
5. LiteSpeed	6.1%	+0.3%

© W3Techs.com	usage	change since 1 August 2021
1. Nginx	34.0%	
2. Apache	31.0%	-1.1%
3. Cloudflare Server	20.1%	+0.9%
4. LiteSpeed	11.3%	+1.0%
5. Microsoft-IIS	6.5%	-0.2%

© W3Techs.com	usage	change since 1 January 2022
1. Nginx	33.1%	+0.1%
2. Apache	30.7%	-0.6%
3. Cloudflare Server	22.1%	+0.5%
4. LiteSpeed	12.4%	+0.5%
5. Microsoft-IIS	6.1%	-0.1%

<https://w3techs.com/> percentages of sites

Environnement de développement en TD et TP

Tout se fait sur une machine locale

Côté serveur (local) :

Serveur HTTP avec APACHE + PHP
+ MySQL

Accessible via

127.0.0.1 ou localhost

Fichiers stockés dans

/var/www/html

Côté client :

Navigateurs :

Firefox

Chrome

Opéra

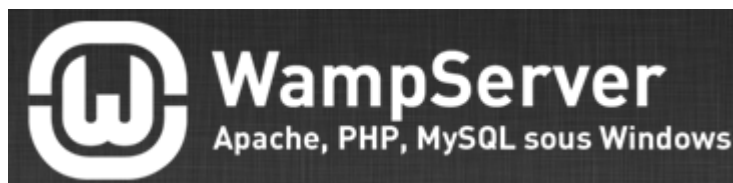
Environnement de développement en autonomie

Windows

Installer un serveur avec
WAMP ou EasyPHP

Linux

Installer un serveur LAMP :
Linux Apache MySQL PHP



Environnement de développement en autonomie

EasyPHP Administration

PHP | MySQL | Apache pour Windows

Nom de l'hôte : 127.0.0.1 Port : 80

Dossier EasyPHP : C:\Program Files (x86)\EasyPHP-5.3.8.1\

Bases MySQL : C:\Program Files (x86)\EasyPHP-5.3.8.1\mysql\

Nom utilisateur MySQL : root

Mot de passe MySQL :

Hôte MySQL : 127.0.0.1

VERSION 5.3.8.1

PhpMyAdmin

Administer vos bases de données

Paramètres

environnement PHP

extensions PHP

Options

+ ajouter un alias

+ ajouter un module

Options avancées

configuration PHP

configuration Apache

WEB LOCAL

Vous devez placer vos fichiers soit dans le dossier 'www' (C:\Program Files (x86)\EasyPHP-5.3.8.1\www) soit dans un dossier alias que vous avez créé (voir 'Ajouter un alias'), ainsi PHP peut interpréter vos fichiers PHP (*.php). Tous les dossiers créés dans 'www' apparaissent ci-dessous.

Root

ALIAS

exemples_PHP

D:\Enseignements\HTML_CSS_XML_PHP_MySQL\Exemples_Cours_HTML_CSS_SQL_JavaScript\Exemples_PHP\

supprimer

MODULES

Aucun module installé

ajouter un module

EasyPHP 5.3.8.1

- PHP 5.3.8 VC9
- Apache 2.2.21 VC9
- MySQL 5.5.16
- PhpMyAdmin 3.4.5
- Xdebug 2.1.2

EasyPHP est portable

Si vous voulez utiliser EasyPHP sur une clé USB, il suffit de copier l'intégralité du répertoire EasyPHP sur la clé. Placez tous vos scripts dans le répertoire 'www' et vérifiez que vos bases de données sont dans le répertoire 'mysql\data'.

Guide de migration vers PHP 5.3

La majorité des améliorations de PHP 5.3.x n'a pas d'impact sur le code existant. Cependant, il y a quelques incompatibilités et de nouvelles fonctionnalités qui doivent être prises en compte.

Donation

Soutenez ce projet

Environnement de développement

Éditeur de texte

- SublimText
- NotePad++
- Atom
- PhpStorm de JetBrains