

# Guia de configuração (vers. 3.3)

## Servidor Apache com PHP, *userdir* e CGI

### Instalação em sistemas com SO Linux (Ubuntu 16.04, 18.04 e 20.04)

Comece pela instalação do servidor apache fazendo

```
sudo apt-get install apache2
```

Após esta instalação o servidor Apache pode ser iniciado com os comandos

```
sudo /etc/init.d/apache2 start
```

ou

```
sudo service apache2 start
```

após a execução desse comando podemos testar o funcionamento do servidor colocando no browser o endereço `http://localhost` ou `http:// 127.0.0.1` e a página seguinte página deverá ser apresentada:



Na página de apresentação podemos encontrar informação diversa sobre a configuração do servidor apache2.

Para que possa trabalhar na área de utilizador, isto é, na área de documentos onde cada utilizador tem todos os privilégios, devemos criar o diretório `~/public_html` com as permissões 755, atribuir à diretoria raiz do utilizador (`~`) as permissões 711.

Na raiz da conta do utilizador (`~`) crie o diretório `public_html` com o comando

```
mkdir ~/public_html
```

Use os comandos seguintes para efetuar a alteração das permissões

```
chmod 755 ~/public_html  
chmod 711 ~
```

Para que o servidor apache que instalámos possa aceder às pastas `public_html` dos utilizadores e apresentar as páginas que lá se encontram, deve ativar-se no servidor Apache o módulo o `userdir` com a instrução

```
sudo a2enmod userdir
```

a partir dessa altura passaram a existir dois novos ficheiros no diretório de configuração dos módulos ativos do servidor apache em `/etc/apache2/mods-enabled/`; no ficheiro `userdir.conf` são configuradas as definições de acesso aos diretórios `public_html` que poderão existir na área de cada utilizador.

Instale agora o interpretador de scripts php

```
sudo apt-get install php
sudo apt-get install libapache2-mod-php
```

de modo a aferir versão do interpretado PHP instalada execute o comando

```
php -version
```

```
PHP X.Y.12...
Copyright (c) The PHP Group
```

e ative esse modulo executando o comando referente à versão X.Y

```
sudo a2enmod phpX.Y
```

Após a instalação deste módulo surge dois novos ficheiros no diretório de configuração dos módulos ativos do servidor apache em `/etc/apache2/mods-enabled/`; no ficheiro `phpX.Y.conf` são configuradas as regras de funcionamento do interpretador PHP do servidor. Para darmos autorização ao Apache para interpretar as scripts em php que forem colocadas no diretório `~/public_html` temos que editar o ficheiro `phpX.Y.conf` como administrador com os comandos

```
sudo gedit /etc/apache2/mods-enabled/phpX.Y.conf
```

e comentar as últimas 5 linhas desse ficheiro, de acordo com a descrição incluída no ficheiro:

```
#<IfModule mod_userdir.c>
#
#
#</IfModule>
```

Para que as recentes alterações na configuração do Apache sejam ser consideradas, deve reiniciar-se o servidor o executando o comando

```
sudo service apache2 restart
```

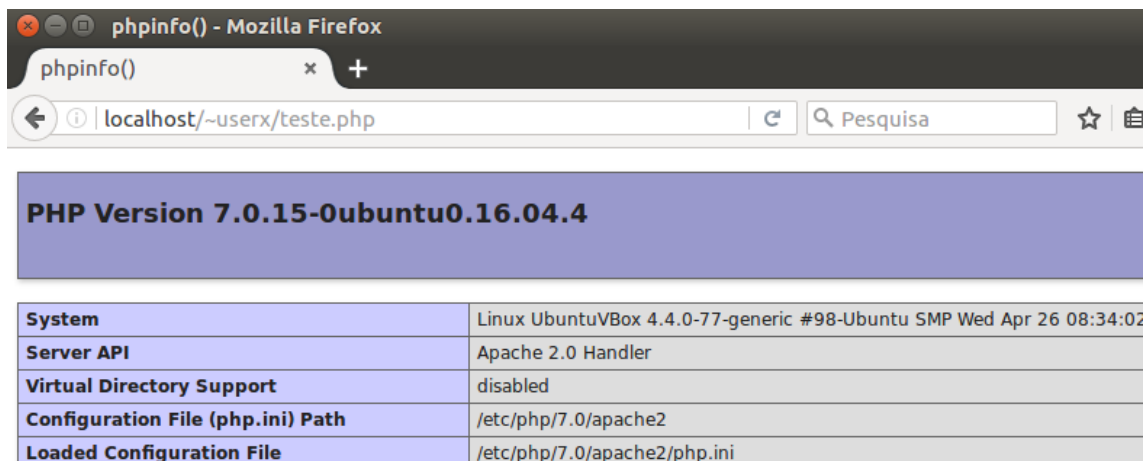
Para confirmar a instalação do interpretador PHP do servidor podemos construir um ficheiro teste.php de teste executando a seguinte instrução na linha de comando

```
echo "<?php phpinfo(); ?>" > teste.php
```

como administrador copie o ficheiro para o diretório de documentos do servidor apache em `/var/www/html` e invoque-o num browser, e deverá ser apresentada a página seguinte com diversos dados do interpretador PHP e do servidor Apache.



Copie também o ficheiro de teste para o directório ~/public\_html do utilizador userx, também essa página será interpretada e apresentada



### Activação do cgi no directório public\_html/cgi-bin

Crie o directório ~/public\_html/cgi-bin com as permissões 755. Para activar o cgi no apache devemos activar esse módulo com o comando

```
sudo a2enmod cgi
```

No ficheiro /etc/apache2/mods-enabled/userdir.conf devemos acrescentar as seguintes linhas a seguir a </Directory>

```
<Directory /home/*/public_html/cgi-bin>
    Options +ExecCGI
    SetHandler cgi-script
</Directory>
```

Reinicie de novo o servidor para que estas configurações sejam carregadas

```
sudo service apache2 restart
```

A partir deste momento os ficheiros executáveis com permissões 755, colocados em `public_html/cgi-bin` deverão ser invocados pelo servidor apache para processamento de informação enviada nos formulários.

Ativação da interpretação PHP no `~/public_html` pela linha de comandos

No ficheiro `/etc/php/X.Y/apache2/php.ini` devemos alterar a definição da `userdir` para

```
userdir="~/public_html"
```

## Instalação em sistemas com SO Linux (Fedora24 a 29)

As seguintes instruções assumem que está a usar a conta de um utilizador normal, sem privilégios especiais.

Comece pela instalação do servidor apache fazendo

```
sudo dnf install httpd
```

Verifique qual a versão do servidor apache que instalou com o comando seguinte

Instale agora o interpretador de scripts php e a biblioteca de extensão php-mysql que inclui as funções que permitem às páginas PHP aceder a bases de dados de sistemas MySQL.

```
sudo dnf install php
sudo dnf install php-mysql
```

Após esta instalação o servidor apache com suporte para interpretação de php pode ser iniciado com os comandos

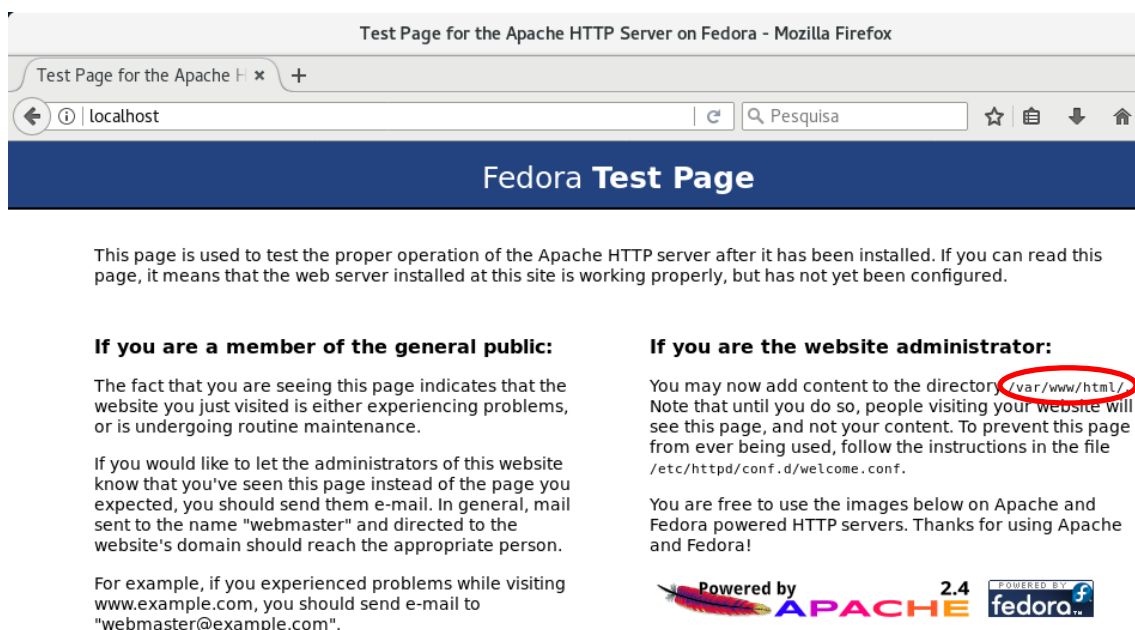
```
sudo systemctl start httpd
```

e para que arranque automaticamente assim que se inicia o sistema devemos executar

```
sudo systemctl enable httpd
```

após a execução desse comando podemos testar o funcionamento do servidor colocando no browser o endereço `http://localhost` ou `http:// 127.0.0.1`

A página de confirmação da configuração e do funcionamento do servidor Apache no Fedora deverá ser apresentada.

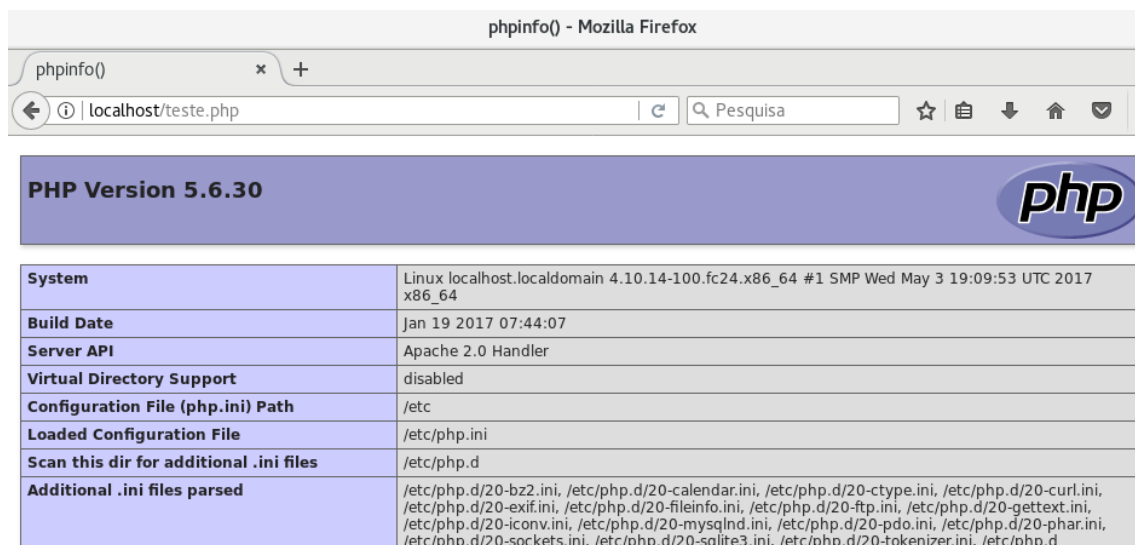


Como é referido nessa página, o administrador do servidor agora instalado deverá colocar os documentos, isto é, as páginas html e php, numa área de acesso reservado apenas ao administrador, em /var/www/html. Por ser uma área de acesso restrito só o poderemos fazer como administradores.

Para testar a instalação do interpretador PHP do servidor podemos contruir o ficheiro **teste.php** de teste executando a seguinte instrução na linha de comando

```
echo "<?php phpinfo(); ?>" > teste.php
```

como administrador copie o ficheiro para o diretório de documentos do servidor apache em /var/www/html e invoque-o num browser, e deverá ser apresentada seguinte com diversos dados do interpretador PHP e do servidor Apache.



<b>System</b>	Linux localhost.localdomain 4.10.14-100.fc24.x86_64 #1 SMP Wed May 3 19:09:53 UTC 2017 x86_64
<b>Build Date</b>	Jan 19 2017 07:44:07
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc
<b>Loaded Configuration File</b>	/etc/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php.d
<b>Additional .ini files parsed</b>	/etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d

Para que possa trabalhar na área de utilizador, isto é, na área de documentos onde cada utilizador tem todos os privilégios, devemos criar o diretório ~/public\_html com as permissões 755, atribuir à diretoria raiz do utilizador (~) as permissões 711.

Na raiz da conta do utilizador (~) crie o diretório public\_html com o comando

```
mkdir ~/public_html
```

Use os comandos seguintes para efetuar a alteração das permissões

```
chmod 755 ~/public_html
chmod 711 ~
```

Por fim, temos que configurar o servidor para possa aceder e apresentar as paginas colocadas pelo utilizador no diretório ~/public\_html.

Para que o servidor apache que instalámos possa aceder às pastas public\_html dos utilizadores e apresentar as páginas que lá se encontram, devemos alterar a configuração do servidor httpd que se encontra no ficheiro /etc/httpd/conf.d/userdir.conf. Edite esse ficheiro com o comando

```
sudo gedit /etc/httpd/conf.d/userdir.conf
```

e altere as linhas 17 e 24, respetivamente, para

```
UserDir enabled
e
UserDir public_html
```

### Activação do CGI no diretório ~public\_html/cgi-bin

Edite de novo o ficheiro `/etc/httpd/conf.f/userdir.conf` e acrescente as seguintes linhas a seguir a `</Directory>`

```
<Directory /home/*/public_html/cgi-bin>
    Options +ExecCGI
    SetHandler cgi-script
</Directory>
```

Grave essas alterações.

Active as permissões de acesso do Apache aos diretórios dos utilizadores no SELinux com o comando

```
sudo setsebool -P httpd_enable_homedirs on

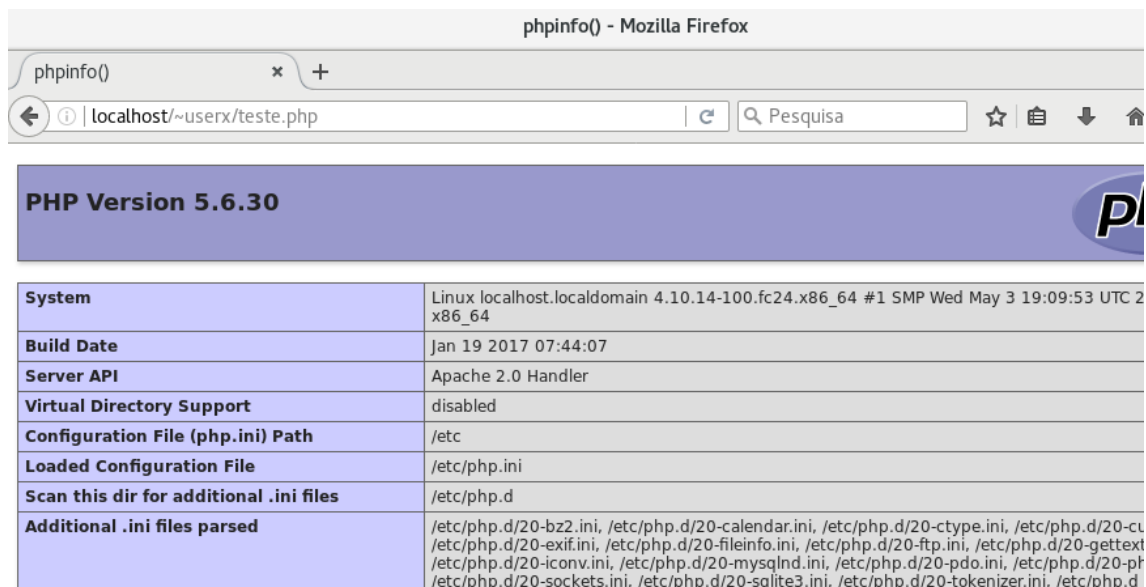
sudo chcon -R -t httpd_sys_content_t /home/*/public_html

sudo setsebool -P httpd_unified on
```

Reinicie o servidor httpd para que as novas configurações entrem em funcionamento usando o comando

```
sudo systemctl restart httpd
```

Após estas alterações deverá abrir as páginas html e php colocada no diretório `~/public_html`. Colocando o ficheiro teste.php no diretório `~/public_html` de um utilizador userx, o servidor Apache irá interpretar o php e apresentar a mesma informação.



The screenshot shows a Mozilla Firefox browser window titled "phpinfo() - Mozilla Firefox". The address bar shows "localhost/~userx/teste.php". The page content displays "PHP Version 5.6.30" and a table of system information.

PHP Version 5.6.30	
<b>System</b>	Linux localhost.localdomain 4.10.14-100.fc24.x86_64 #1 SMP Wed May 3 19:09:53 UTC 2017 x86_64
<b>Build Date</b>	Jan 19 2017 07:44:07
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc
<b>Loaded Configuration File</b>	/etc/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php.d
<b>Additional .ini files parsed</b>	/etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlrpc.ini

Também a partir deste momento os ficheiros executáveis com permissões 755, colocados em `~/public_html/cgi-bin` poderão ser invocados pelo servidor apache para processamento de informação enviada nos formulários.



## Instalação XAMPP em sistemas com SO Windows

Para os PCs que têm instalados os sistemas Windows e para os alunos que preferam desenvolver esta parte dos trabalhos neste ambiente, recomenda-se a instalação do pacote de aplicações XAMPP. Este pacote integra diferentes aplicações que podem ser seleccionadas ou não, para instalação.

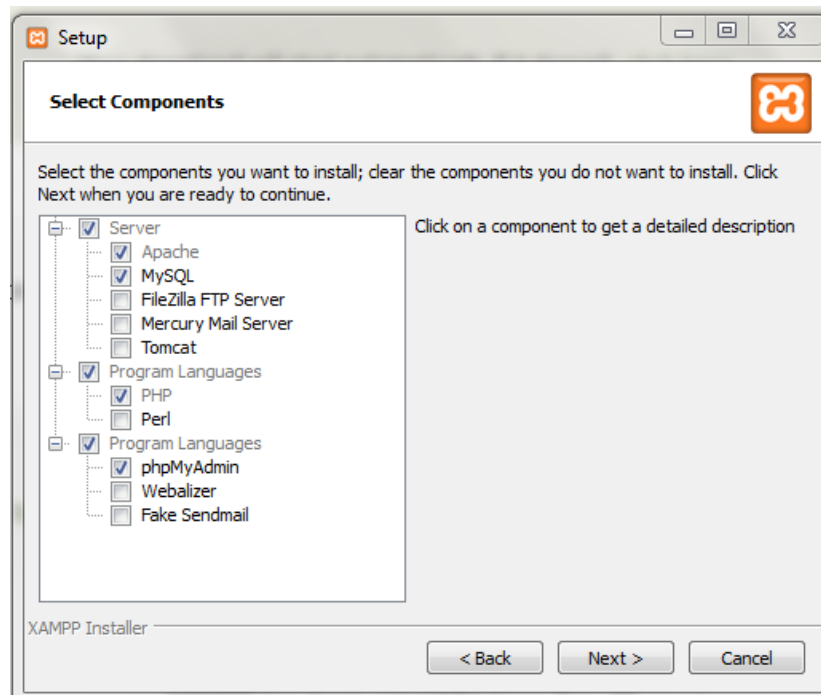
O pacote de aplicações XAMPP pode ser obtido em vários websites, por exemplo em

[https://www.apachefriends.org/pt\\_br/index.html](https://www.apachefriends.org/pt_br/index.html)

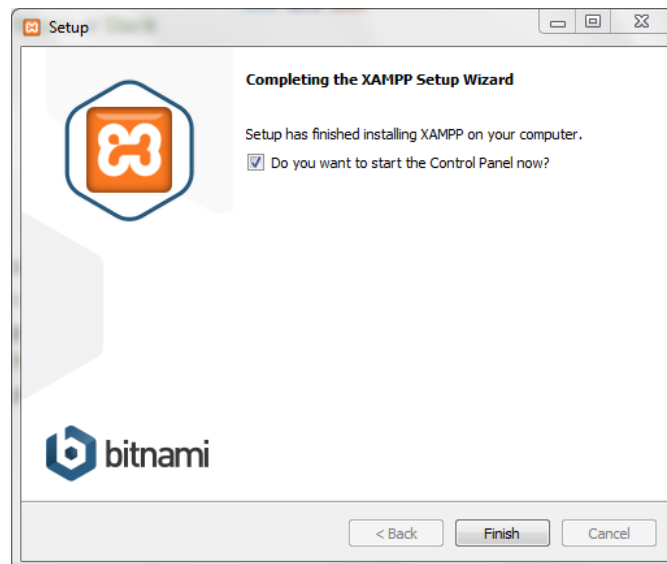
Fazer download do ficheiro de instalação

xampp-win32-1.8.3-4-VC11-installer (ou o mais recente)

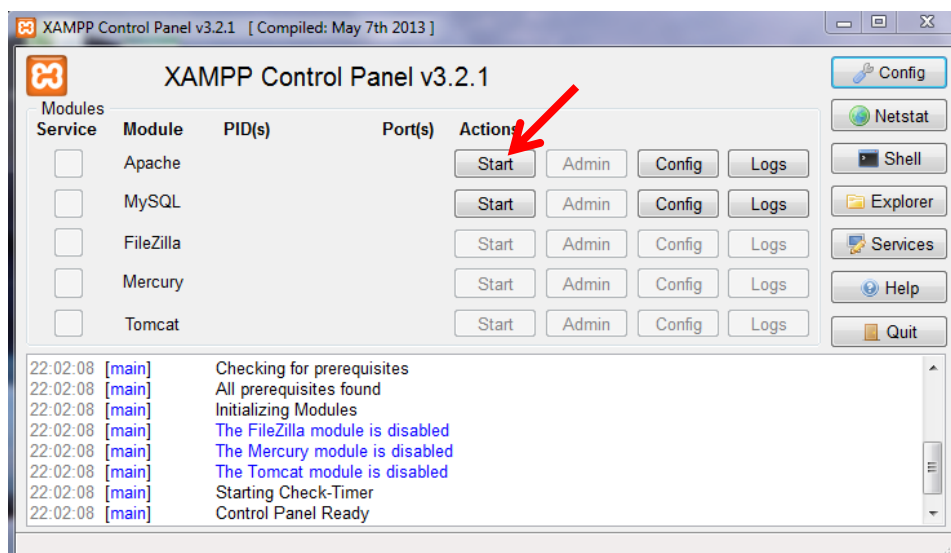
Durante o processo de instalação poderão ser seleccionada as funcionalidades e aplicações que pretendemos instalar. Para os trabalhos desenvolvidos no âmbito de DEAPC as seguintes opções são suficientes



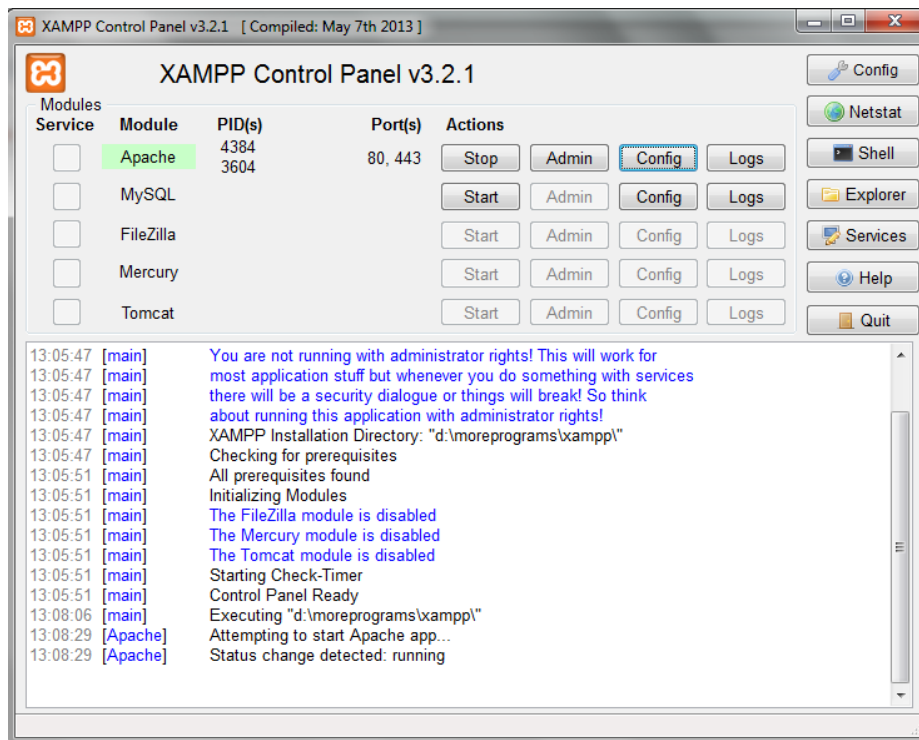
Após a instalação a aplicação de controlo (XAMPP Control Panel) pode ser iniciada automaticamente com a selecção da opção na janela seguinte



Na aplicação de controlo, o XAMPP Control Panel, podemos configurar, iniciar e parar os diversos servidores, como são disso exemplo o Apache e o Mysql. Para arrancarmos o servidor Apache basta pressionar sobre o Start



Nesta altura o servidor apache que instalou com o XAMPP já está em funcionamento, como se ilustra na figura seguinte.



Teste o seu servidor colocando o seguinte no URL no browser <http://localhost> ou <http://127.0.0.1>, que são equivalentes. Deverão surgir as páginas seguintes:



English / Deutsch / Français / Nederlands / Polski / Italiano / Norwegian / Español / 中文 / Português (Brasil) / 日本語



**XAMPP**  
1.9.3  
[PHP: 5.5.11]

Bem Vindo

Status

Segurança

Documentação

Componentes

Applications

**PHP**

phpinfo()

Coleção de CD

Biblioteca

Arte Instantânea

Agenda de Telefones

**Java**

Info

Tomcat examples

**Tools**

phpMyAdmin

FileZilla FTP

Mail

© 2014  
XAMPP  
FR

**Bem vindo ao XAMPP para Windows!**

**Congratulações:**  
Você instalou corretamente o XAMPP em seu sistema!

Você pode agora iniciar a utilização do Apache e outros aplicativos. Primeiramente tente verificar o »Status« no menu lateral para ter certeza que tudo está funcionando corretamente.

Para suporte em OpenSSL por favor utilize o certificado teste em <https://127.0.0.1> ou <https://localhost>

Boa sorte, Kay Vogelgesang + Kai 'Oswald' Seidler

**Install applications on XAMPP using BitNami**

Apache Friends and BitNami are cooperating to make dozens of open source applications available on XAMPP, for free. BitNami-packaged applications include Wordpress, Drupal, Joomla! and dozens of others and can be deployed with one-click installers. Visit the [BitNami XAMPP page](#) for details on the currently available apps.

Nota: recomenda-se a exploração dos vários itens localizados na barra à direita, em particular o que permite avaliar as condições de segurança do Xampp. Se prescindir de rectificar as condições de segurança sugere-se que desligue o computador da rede enquanto trabalha localmente.

Em `phpinfo()` pode verificar que o interpretador PHP já se encontra em funcionamento se surgir a seguinte página.

**XAMPP for Windows** English / Deutsch / Français / Nederlands / Polski / Italiano / Norwegian / Español / 中文 / Português (Brasil) / 日本語

**XAMPP 1.8.3**  
[PHP: 5.5.11]

Bem Vindo  
Status  
Segurança  
Documentação  
Componentes  
Applications

**Php**  
phpinfo()  
Coleção de CD  
Bioritmo  
Arte Instantânea  
Agenda de Telefones

**J2ee**  
Info  
Tomcat examples

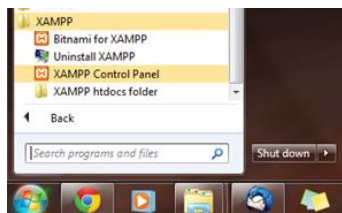
**Tools**  
phpMyAdmin  
FileZilla FTP  
Mail

©2002-2014  
...**APACHE**  
**FRIENDS**...

**PHP Version 5.5.11**

System	Windows NT JMA2001GB-PC 6.1 build 7601 (Windows 7 Business Edition Service Pack 1) i586
Build Date	Apr 8 2014 15:01:59
Compiler	MSVC11 (Visual C++ 2012)
Architecture	x86
Configure Command	csconfig/molog configure.js "--enable-snapshot-build" "--disable-isapi" "--enable-debug-pack" "--without-mssql" "--without-pdo-mssql" "--without-p3web" "--with-pdo-oci=C:\php-sd\oracle\v86\instantclient10sdk\shared" "--with-oci8=C:\php-sd\oracle\v86\instantclient10sdk\shared" "--with-oci8-11g=C:\php-sd\oracle\v86\instantclient11sdk\shared" "--enable-object-out-dir=.obj/" "--enable-com-dotnet=shared" "--with-mcrypt=static" "--disable-static-analyze" "--with-pgsql"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	D:\MorePrograms\Xampp\php\php.ini
Scan this dir for additional .ini files	(none)

Se pretendermos aceder ao XAMPP Control Panel podemos fazê-lo através dos menus do Windows.



Nesta zona é também disponibilizada (em *XAMPP htdocs folder*) uma ligação directa para a raiz da área de páginas do servidor (Document root), onde se devem colocar as páginas html e php do servidor.

Os programas compilados e/ou as scripts cgi devem ser colocados no diretório cgi-bin existente nos diretórios de configuração do XAMPP.