

# PATRONES BLOG

Carlos Eduardo Jara Rodríguez  
Código 102509  
Cj102509@ingenieria.san martin.edu.co

Cuando construimos aplicaciones con muchos formularios u otras secciones con fragmentos repetitivos, a veces utilizamos componentes, como puede ser un cuadro desplegable (combo box u option select, llámale como quieras) para elegir una ciudad. Ya que las páginas se generan en el servidor (PHP, Java, Ruby...), se han inventado diversos artificios para reutilizar estos componentes: simples funciones o variables que devuelven el código HTML deseado, mini-plantillas (Smarty), elements de CakePHP, etc.

## EJEMPLOS DE PATRONES

### 1. CFWheels.gitignore

```
# unpacked plugin folders
plugins/**/*

# files directory where uploads go
files

# DBMigrate plugin: generated SQL
db/sql

# AssetBundler plugin: generated bundles
javascripts/bundles
stylesheets/bundles
```

### 2. CodeIgniter.gitignore

Es un código abierto rápido desarrollo framework de aplicaciones web , para su uso en la construcción de sitios web dinámicos con PHP . "Su objetivo es

---

permitir a los desarrolladores [] para desarrollar proyectos mucho más rápido que escribir código desde cero, proporcionando un rico conjunto de librerías para tareas comunmente necesarias, así como una interfaz sencilla y la estructura lógica para acceder a estas bibliotecas

### Ejemplos

```
*/config/development
*/logs/log-*.php
*/logs/index.html
*/cache/*
*/cache/index.html
*/cache/!htaccess
```

### 3. CommonLisp.gitignore

Es un lenguaje multi paradigma de propósitos generales. Soporta una combinación de paradigmas de programación como procedimental (imperativo), funcional, y orientada al objeto. Como un lenguaje de programación dinámica, facilita el desarrollo de software de una manera evolutiva e incremental, con la compilación iterativa en programas eficientes en tiempo de ejecución.

```
*.FASL
*.fasl
*.lisp-temp
```

### Composer.gitignore

```
composer.phar
vendor/

# Commit your application's lock file
```

<http://getcomposer.org/doc/01-basic-usage.md#composer-lock-the-lock-file>

```
# You may choose to ignore a library lock file
http://getcomposer.org/doc/02-libraries.md#lock-file
# composer.lock
```

#### 4. Concrete5.gitignore

Es un código abierto sistema de gestión de contenidos (CMS) para publicar contenidos en la World Wide Web y la intranet.

Fue diseñado para facilitar su uso, para los usuarios con un mínimo de conocimientos técnicos. Permite a los usuarios editar el contenido del sitio directamente desde la página. Proporciona gestión de versiones para cada página, similar a la wiki software, otro tipo de software de desarrollo de sitios web.

Permite a los usuarios editar imágenes a través de un editor incrustado en la página

```
config/site.php
files/cache/*
files/tmp/*
```

#### 5. Dart.gitignore

es un lenguaje de programación de código abierto, desarrollado por Google. Fue revelado en la conferencia goto; en Aarhus, Dinamarca el 10 octubre de 2011 El objetivo de Dart no es reemplazar JavaScript como el principal lenguaje de programación web en los navegadores web, sino ofrecer una alternativa más moderna. El espíritu del lenguaje puede verse reflejado en las declaraciones de Lars Bak, ingeniero de software de Google, que define a Dart como un “lenguaje estructurado pero flexible para programación Web”

```
// Don't commit the following files and
directories created by pub and dart2js
packages/
*.js_
```

```
*.js.deps
*.js.map
```

```
// Include when developing application packages
pubspec.lock
```

```
// Avoid committing generated JavaScript files
*.dart.js
```

#### 6. Delphi.gitignore

Delphi es un entorno de desarrollo de software diseñado para la programación de propósito general con énfasis en la programación visual. En Delphi se utiliza como lenguaje de programación una versión moderna de Pascal llamada Object Pascal

```
*.dcu
*.*
*.local
*.drc
*.map
*.exe
*.dll
*.bpl
*.dcp
*.so
*.apk
```

#### 7. Drupal.gitignore

```
# Ignore configuration files that may contain
sensitive information.
sites/*/settings*.php
```

```
# Ignore paths that contain generated content.
cache/
files/
sites/*/files
sites/*/private
```

```
# Ignore default text files
robots.txt
```

```
/CHANGELOG.txt
/COPYRIGHT.txt
/INSTALL*.txt
/LICENSE.txt
/MAINTAINERS.txt
/UPGRADE.txt
/README.txt
sites/all/README.txt
sites/all/modules/README.txt
sites/all/themes/README.txt

# Ignore everything but the "sites" folder ( for non
core developer )
.htaccess
web.config
authorize.php
cron.php
index.php
install.php
update.php
xmlrpc.php
/includes
/misc
/modules
/profiles
/scripts
/themes
```

#### 8. EPiServer.gitignore

```
#####
## EPiServer Files
#####
*License.config
```

#### 9. Eagle.gitignore

```
# Ignore list for Eagle, a PCB layout tool

# Backup files
*.s#?
*.b#?
```

```
*.l#?

# Eagle project file
# It contains a serial number and references to the
file structure
# on your computer.
# comment the following line if you want to have
your project file included.
eagle.epf
```

#### # CAM files

```
*.$$$
*.cmp
*.ly2
*.l15
*.sol
*.plc
*.stc
*.sts
*.crc
*.crs
```

```
*.dri
*.drl
*.gpi
*.pls
```

```
*.drd
*.drd.*
```

```
*.info
```

```
*.eps
```

#### 10. Emacs.gitignore

```
# Compiled
*.elc
```

#### 11. Erlang.gitignore

\*.EXE

.eunit  
deps  
\*.o  
\*.beam  
\*.plt  
erl\_crash.dump

## REFERENCIAS

### 12. ExpressionEngine.gitignore

.DS\_Store

# Images  
images/avatars/  
images/captchas/  
images/smileys/  
images/member\_photos/  
images/signature\_attachments/  
images/pm\_attachments/

# For security do not publish the following files  
system/expressionengine/config/database.php  
system/expressionengine/config/config.php

# Caches  
sized/  
thumbs/  
\_thumbs/  
\*/expressionengine/cache/\*

### 13. Tasm.gitignore

# Object Files  
\*.OBJ

# List Files  
\*.LST

# Map files  
\*.MAP

# Executable files

<http://www.getskeleton.com/>  
<http://foundation.zurb.com/>  
<http://israelviana.es/patrones-de-desarrollo-web-componentes-html-reutilizables-2/>