



Extrait du Les Ateliers Pierrot

<http://spip.ateliers-pierrot.fr/nos-activites/nos-projets/carte-blanche>

Carte Blanche

- Nos Activités - Nos projets -

Date de mise en ligne : mercredi 13 mars 2013

Les Ateliers Pierrot

Le des Ateliers ; simple, léger et facile à utiliser pour tout développement .

Plus d'infos : <http://github.com/php-carteblanche/...>

[sommaire]

Présentation

CarteBlanche est un framework PHP construit comme une application MVC prête à tourner. Il est basé sur un noyau simple de bibliothèque de classes suffisantes pour le faire tourner et permet de développer rapidement et simplement des entités externes.

Il prend en charge des **bundles**, entités de code prenant en charge une action précise, qui peuvent être exportés après développement en tant qu'entité autonome si besoin ; il propose également un panel d'outils, les **tools**, sous forme de classes PHP simples, une par outil, pour aider à construire des vues (notamment).

Le rendu final est intégré dans un modèles de vues "prêt à l'emploi" (mais simpliste), codé en HTML5.

Installation & Usage

Installation

Le paquet à installer est [carteblanche/carteblanche](#).

Lorsque vous avez téléchargé ou cloné les sources du paquet, l'installation est assez simple. Elle requière [Composer](#) pour installer les dépendances. Voyez la section "Dépendances" ci-dessous pour plus d'information sur Composer, ce qu'il est et ce qu'il fait.

Installation pas-à-pas :

1. déposez le répertoire de CarteBlanche dans le système de fichiers de votre serveur ; vous pouvez télécharger une archive à l'adresse <https://github.com/php-carteblanche...> ou cloner le dépôt GIT complet en exécutant les commandes suivantes :

Carte Blanche

```
~$ git clone --recursive https://github.com/php-carteblanche/carteblanche path/to/CarteBlanche
// ou, si vous avez une vieille version de GIT sans l'option "recursive":
~$ git clone https://github.com/php-carteblanche/carteblanche path/to/CarteBlanche
~$ cd path/to/CarteBlanche
~$ git submodule update --init
```

2. allez dans le répertoire de CarteBlanche en ligne de commande :

```
~$ cd path/to/CarteBlanche
```

3. installez les dépendances en exécutant le script `bin/build.sh` :

```
~$ ./bin/build.sh install
// ou, si vous devez définir certaines options comme les chemins vers Composer ou PHP:
~$ ./bin/build.sh -h
```

4. CarteBlanche est prêt !

Utilisation

Comme beaucoup d'autres frameworks, le code PHP et les exécutables sont séparés des "assets" (les médias utiles aux vues web). Globalement, tout les fichiers qui doivent être accessibles par le navigateur sont stockés dans le répertoire `www/`. Tous les autres fichiers ne doivent PAS être accessibles par un navigateur. Les sources PHP sont principalement rangées dans le répertoire `src/`.

Pour accéder au contrôleur du framework depuis un navigateur, chargez simplement le fichier `index.php` depuis le répertoire `www/` de CarteBlanche. Une version de développement est pré-défini et accessible en chargeant le fichier `dev.php` au lieu de `index.php`.

Un contrôleur en ligne de commande est également prévu, auquel vous pouvez accéder en exécutant :

```
~$ php bin/console
```

Documentation

Voyez les contenus du répertoire `doc/` pour une documentation sur la configuration, la structure et l'utilisation.

Règles & Dépendances

Règles de codage

Afin de construire un code compréhensible et maintenable, nous essayons de suivre les règles et normes de codage

et de nommage les plus couramment utilisées :

- les [standards des paquets PEAR](#)
- les [standards du PHP Framework Interoperability Group](#).

Sachant cela, les classes du framework sont nommées et organisées dans en architecture autorisant l'utilisation du [chargeur automatique standard "SplClassLoader"](#).

Dépendances

Comme expliqué dans la section "Installation" ci-dessus, ce paquet utilise des paquets tiers afin d'intégrer les standards communs du développement PHP actuel. En plus de [Composer](#), requis pour l'installation classique des dépendances, CarteBlanche propose un panel d'autres paquets externes utilisables pour le développement et des paquets d'"assets" utilisés pour le rendu HTML5.

Pour installer les dépendances de développement, exécutez simplement :

```
~$ ./bin/build.sh --set=dev install
// ou, en utilisant Composer directement
~$ composer install --dev
```

Une documentation PHP est disponible, générée depuis le code source grâce à l'outil [Sami](#). Pour la générer ou la re-générer, vous aurez besoin des dépendances de développement (*comme expliqué plus haut*) et d'exécuter :

```
~$ ./bin/build.sh render-doc
// ou, en utilisant Sami directement
~$ php src/vendor/sami/sami/sami.php render config/vendor/sami.config.php
```