

Autoloading & Composer

Qu'est-ce que l'autoloading ?

Require

En utilisant la Programmation orientée objet, on se retrouve à utiliser des classes et devoir utiliser des instructions require pour chacun des fichiers contenant des classes.

Structure de dossiers

La structure des dossiers se faisant de plus en plus complexe on se retrouve avec beaucoup de fichiers imbriqués.



```
<?php
```

```
declare(strict_types=1);
```

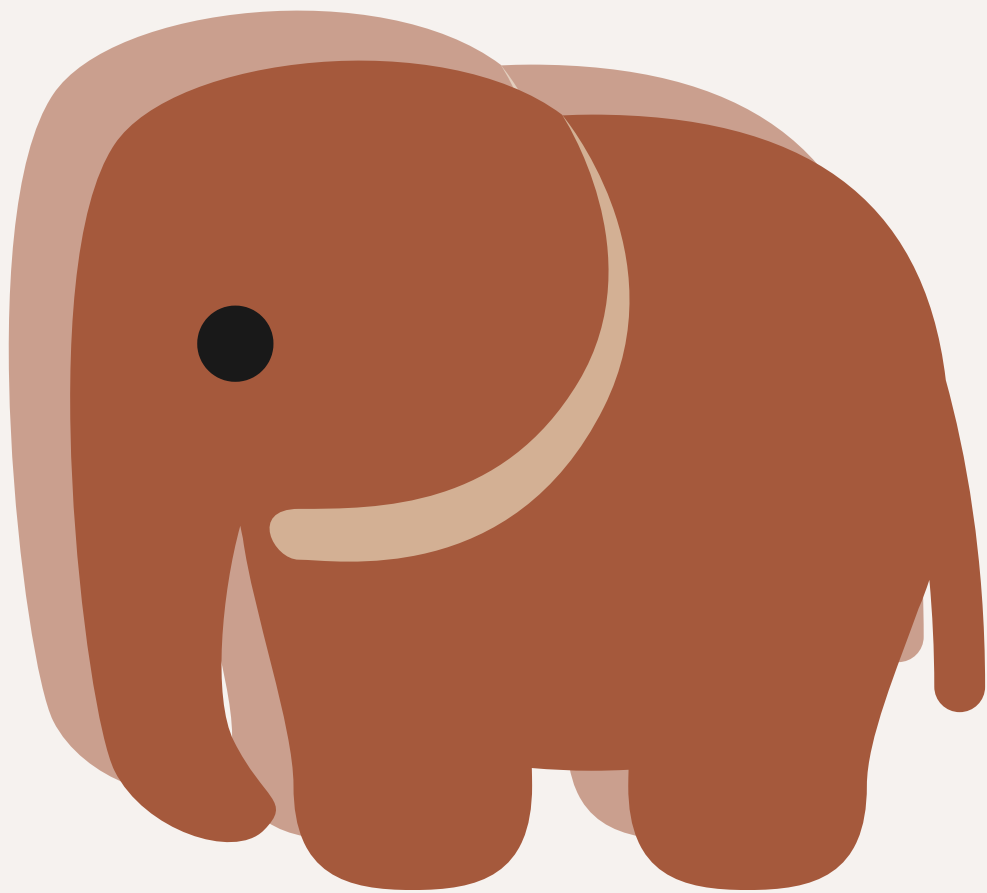
```
require '../PaymentGateway/Paypal/Transaction.php';
```

```
use App\PaymentGateway\Paypal\Transaction;
```

```
$transaction = new Transaction();
```

Schéma classique

On retrouve require, use et l'appel
à la classe.



PHP Standard Recommendations (PSR)

On va s'appuyer sur le PSR-4 : Autoloading



```
<?php

declare(strict_types=1);

spl_autoload_register(function ($class) {

    // project-specific namespace prefix
    $prefix = 'Foo\\Bar\\';

    // base directory for the namespace prefix
    $base_dir = __DIR__ . '/src/';

    // does the class use the namespace prefix?
    $len = strlen($prefix);

    // get the relative class name
    $relative_class = substr($class, $len);

    // replace the namespace prefix with the base directory, replace namespace
    // separators with directory separators in the relative class name, append
    // with .php
    $file = $base_dir . str_replace('\\\\', '/', $relative_class) . '.php';

    // if the file exists, require it
    if (file_exists($file)) {
        require $file;
    }
});
```

Schéma d'un autoload selon PSR-4

On utilise la fonction `spl_autoload_register` qui permet de lancer une fonction php qui s'exécute à chaque appel d'une classe.



Package Manager de PHP

Gestionnaire de package



A quoi ça sert ?



Outil de gestion de dépendances en PHP



Permet d'installer les librairies localement par déclaration



Les librairies s'appellent des packages en PHP



Packagist : <https://packagist.org/>



Autoloading des packages importés, mais aussi des classes existantes.

Commandes basiques sur Composer

Initialisation Composer

`composer init`

Installation Package

`composer require UnPackage`

Génération de l'autoloader

`composer dump autoload`

Génération de l'autoload

Code qui permet d'inclure les classes utilisateurs dans l'autoloader



```
{  
    "autoload": {  
        "psr-4": {"Chemin\\du\\dossier\\classes": "src/" }  
    }  
}
```

Importer les classes

Désormais on n'a plus besoin de require chacune des classes à la main.



```
require '../vendor/autoload.php';
```