BENJAMIN LAMPÉRIER

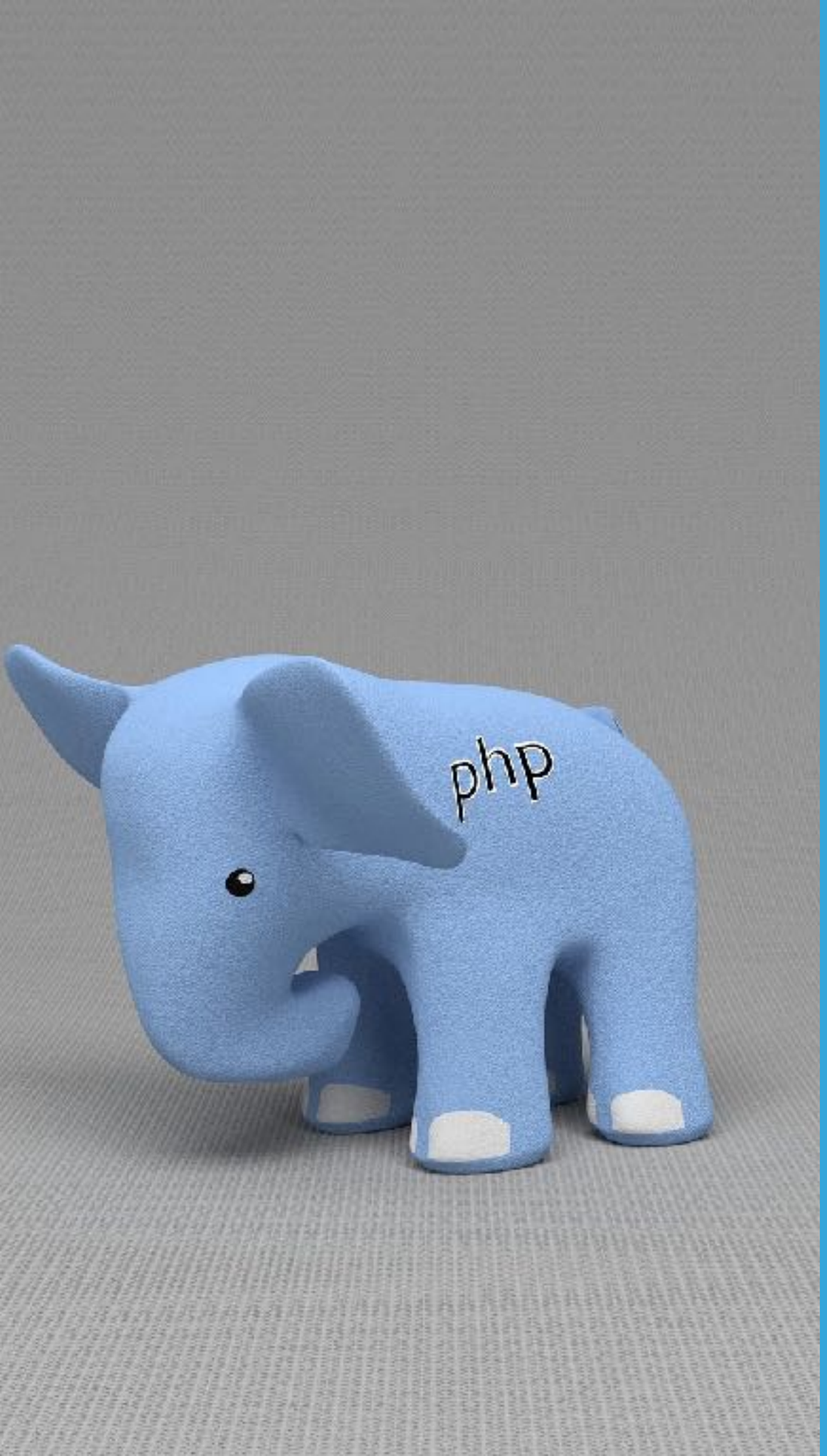
# TD WEB SERVEUR

ITERACODE

# BENJAMIN LAMPÉRIER

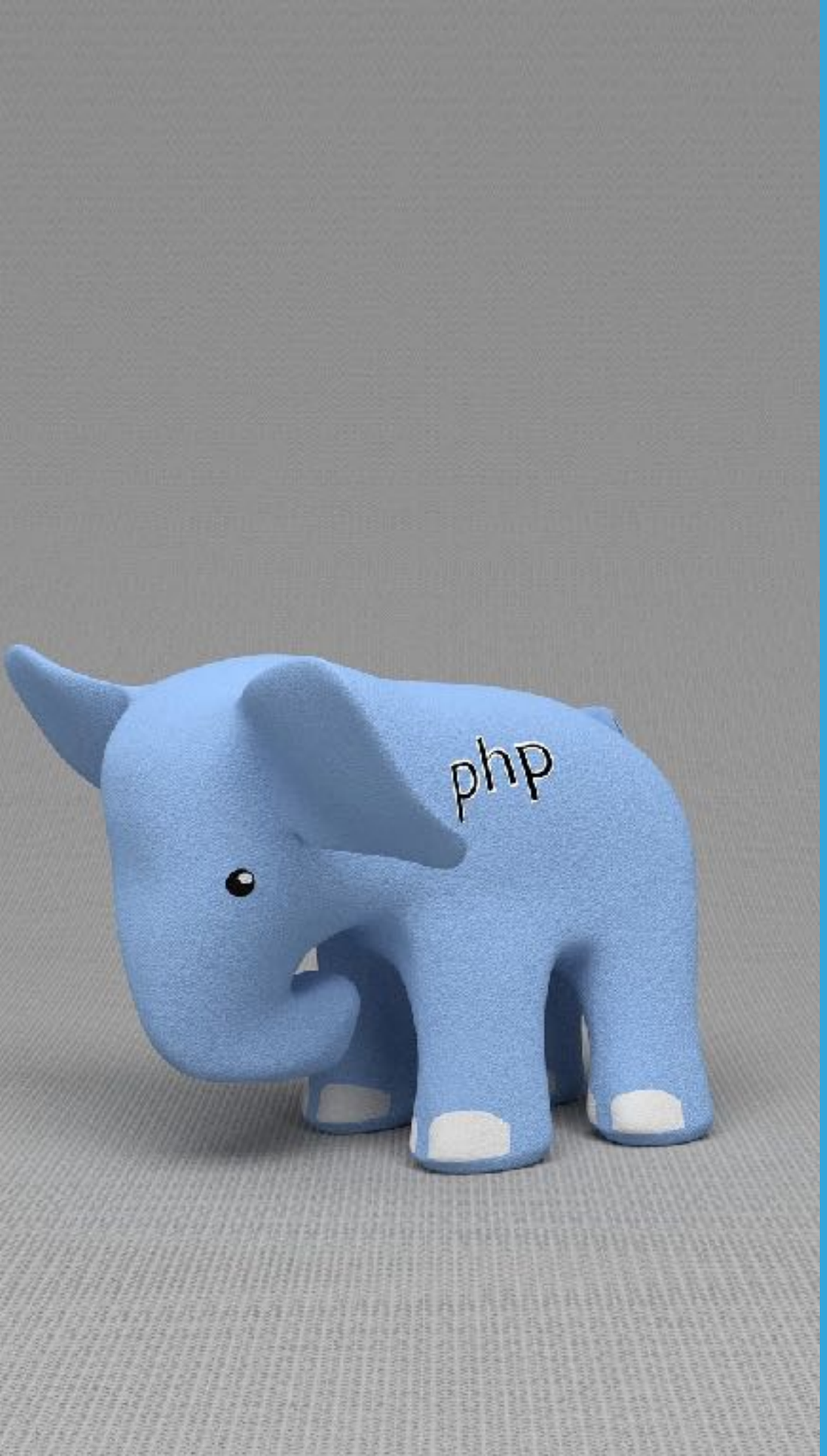
co-gérant / CTO

benjamin@iteracode.fr

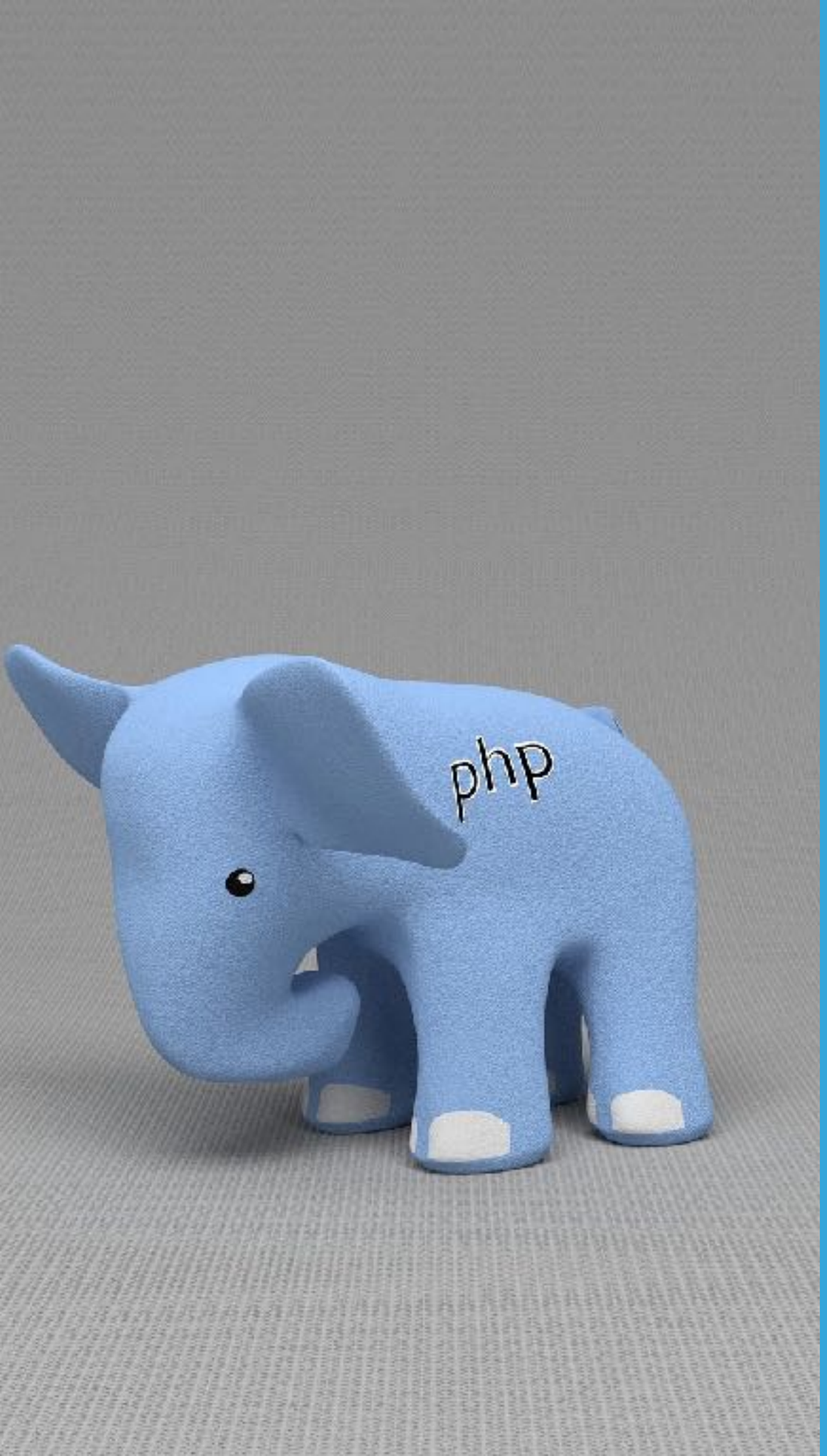



# DÉVELOPPEMENT

# WEB SERVEUR PHP




# DNDLB1QM423




# Linux
# Apache
# Mysql
# Php

# (GIT)

# TOOLS

# GITHUB
# CODEENVY

# RULES

# PRO PROPRE

HTTPS://GITHUB.COM/

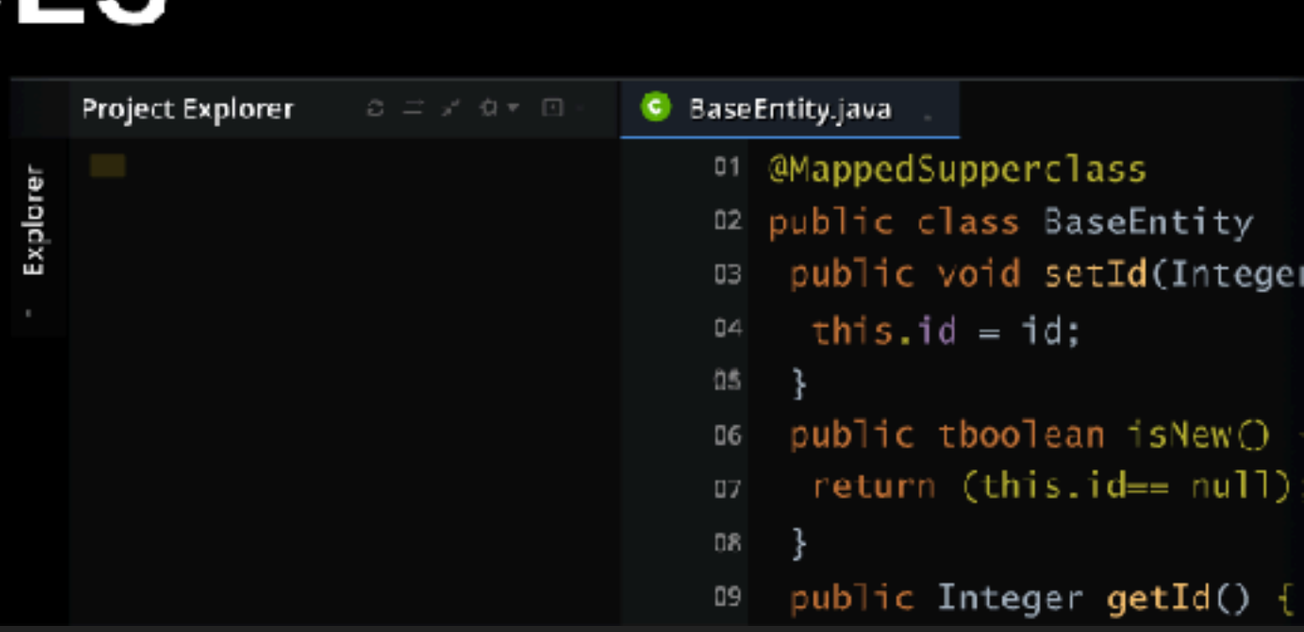
CODENVY

PRODUCT PACKAGES PRICING GET STARTED

# CLOUD WORKSPACES FOR DEV TEAMS

Develop code with portable Docker runtimes. Workspaces "dev mode" production adding intellisense, debuggers and desktop sync. Create teams for sharing at codenvy.io or in a private install.

GET STARTED




HTTPS://CODENVY.COM/

# CD

# LS

# MKDIR

# RM

# CHMOD / CHOWN

# APT-GET

# SERVICE APACHE2 RESTART

# <DIRECTORY>

# PHP

## The Right Way

LAST UPDATED: 2017-10-04 17:27:30 +0000

SHARE ON TWITTER


### Welcome

Translations

How to Contribute

Spread the Word!

### Getting Started

Use the Current Stable Version (7.1)

Built-in Web Server

Mac Setup

Windows Setup

### Code Style Guide

### Dependency Injection

Basic Concept

Complex Problem

Containers

Further Reading

### Databases

MySQL Extension

PDO Extension

Interacting with Databases

Abstraction Layers

### Templating

### Servers and Deployment

Platform as a Service (PaaS)

Virtual or Dedicated Servers

Shared Servers

Building Your Application

### Virtualization

Vagrant

Docker

### Caching

Opcode Cache

Object Caching


HTTP://WWW.PHPTHERIGHTWAY.COM/

# CODE STYLE : PSR 1 ET PSR 2

# THE BASICS

```
($a === 5); ($a == ‘5’);

if ($a) return true;

function ($a) { return true; }

‘ my value is ‘ . $a // « my value is $a »

($a == $b) ? true : false
```

# DATE AND TIME

# => DATETIME CLASS

```php
<?php
$raw = '22. 11. 1968';
$start = DateTime::createFromFormat('d. m. Y', $raw);

echo 'Start date: ' . $start->format('Y-m-d') . "\n";
```

# POO

```php
<?php
class Foo {
    public $aMemberVar = 'aMemberVar Member Variable';
    public $aFuncName = 'aMemberFunc';


    function aMemberFunc() {
        print 'Inside `aMemberFunc()`';
    }
}

$foo = new Foo;
?>
```

# NEW

# GET / SET / __CONSTRUCT()

# GIT INIT

# GIT STATUS

# GIT BRANCH

# GIT CHECKOUT

# GIT ADD

# GIT COMMIT

# GIT PULL

# GIT PUSH

# COMPOSER INSTALL

# COMPOSER UPDATE

# COMPOSER REQUIRE

## HTTPS://PACKAGIST.ORG/

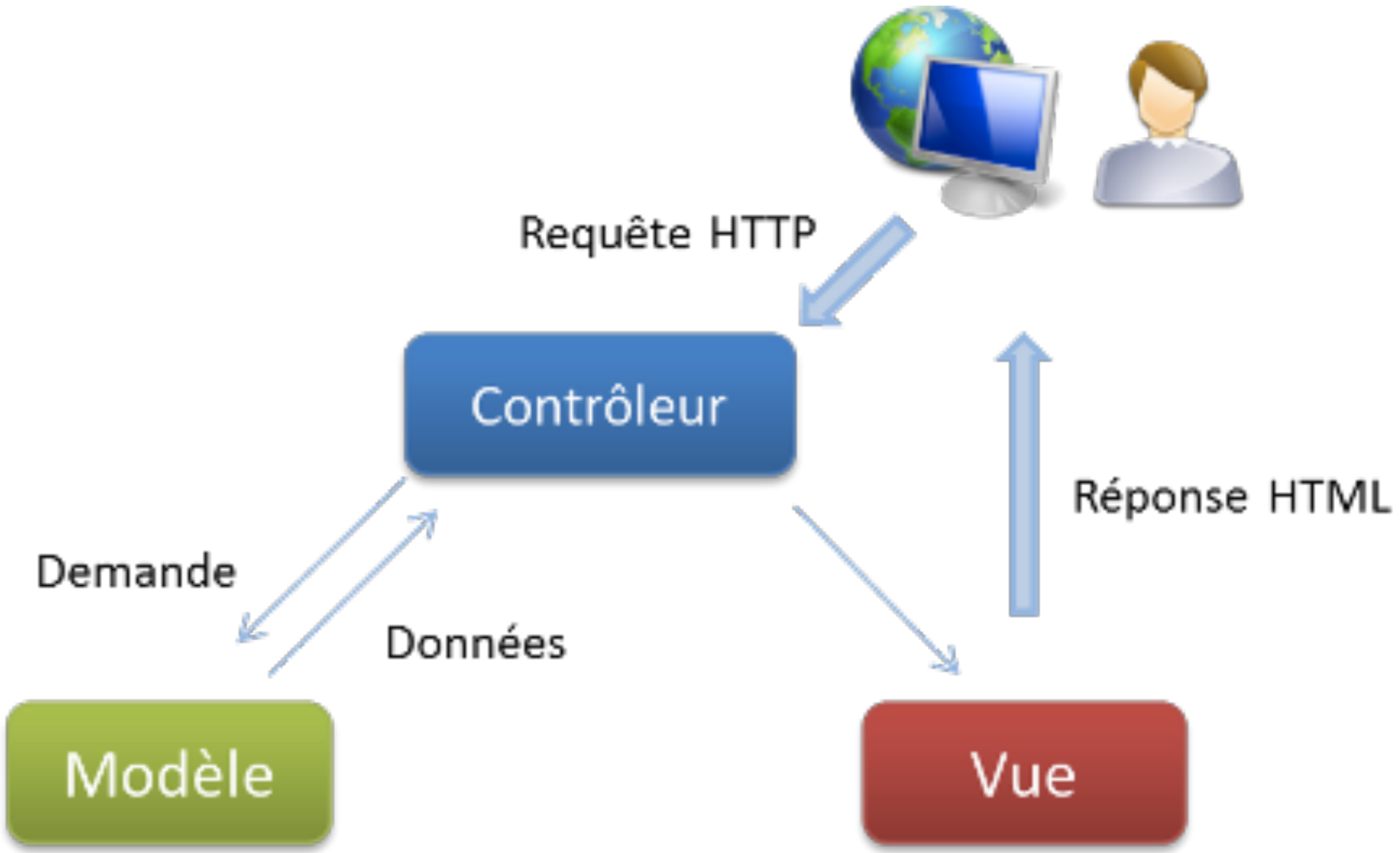
# CAKEPHP 3

# FRAMEWORK MVC

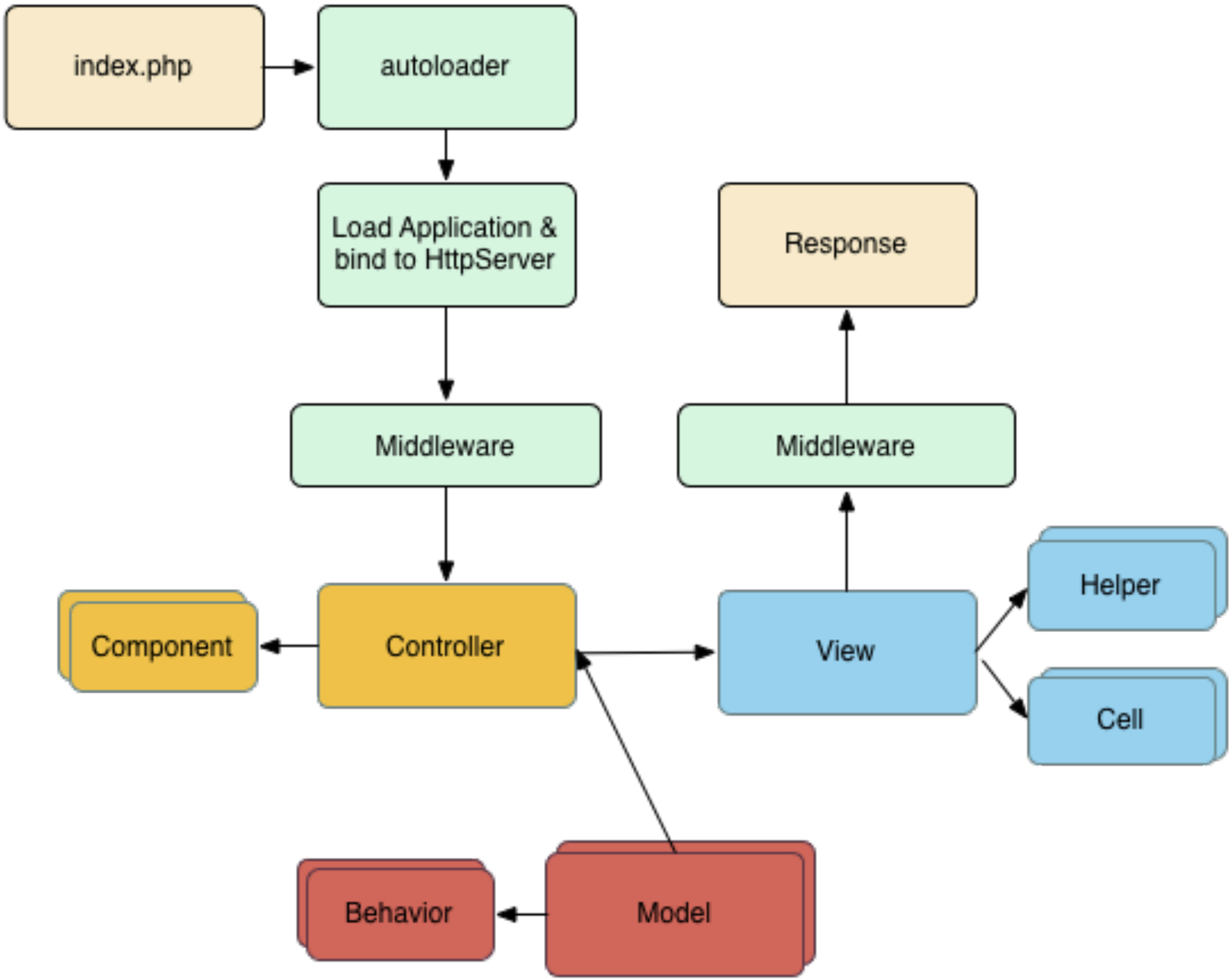
CakePHP 3

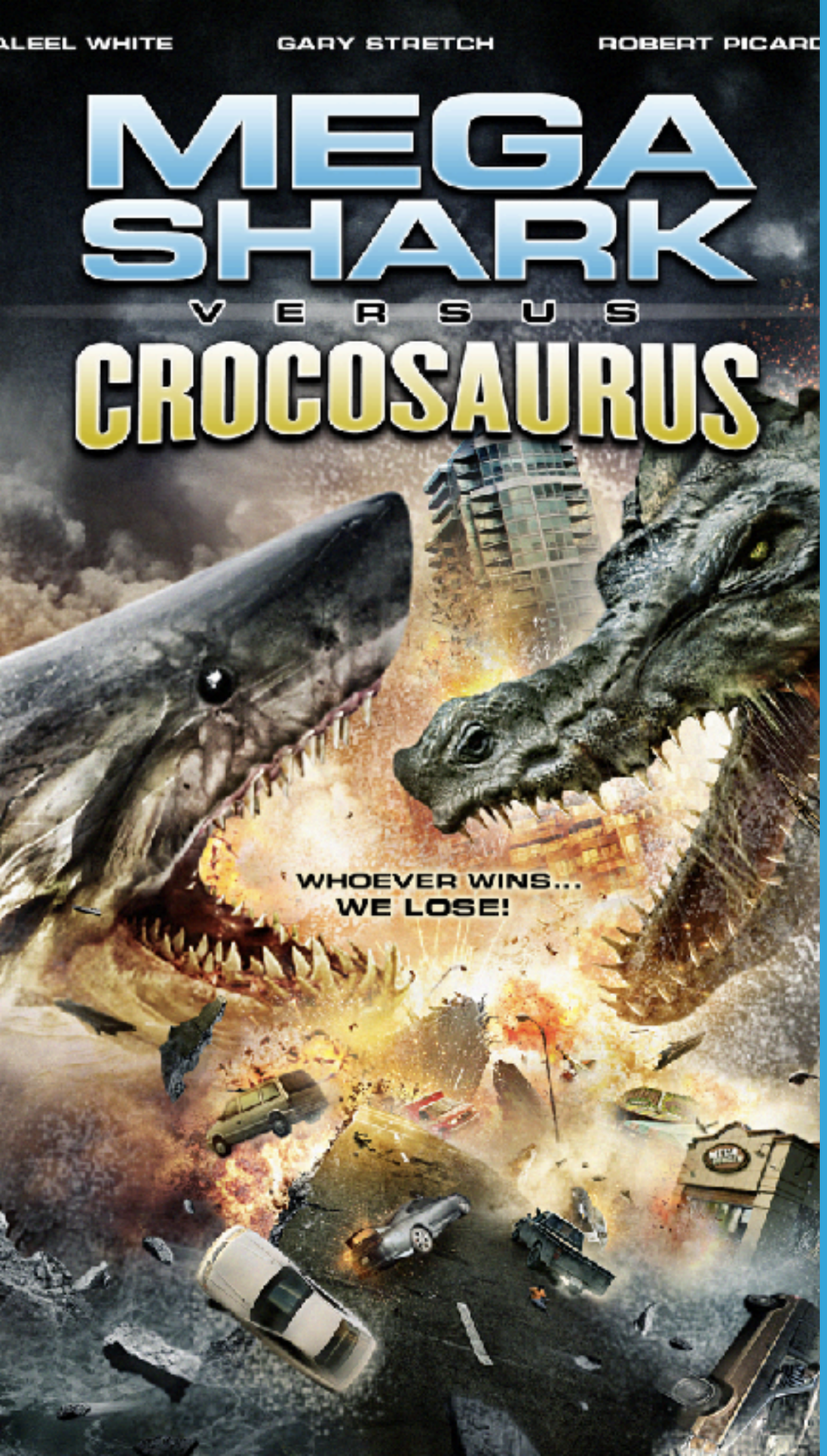
# CAKEPHP 3

# CONVENTIONS OVER CONFIGURATIONS


Requête HTTP
Contrôleur
Réponse HTML
Demande
Données
Modèle
Vue

# CAKEPHP / PRINCIPE DE FONCTIONNEMENT

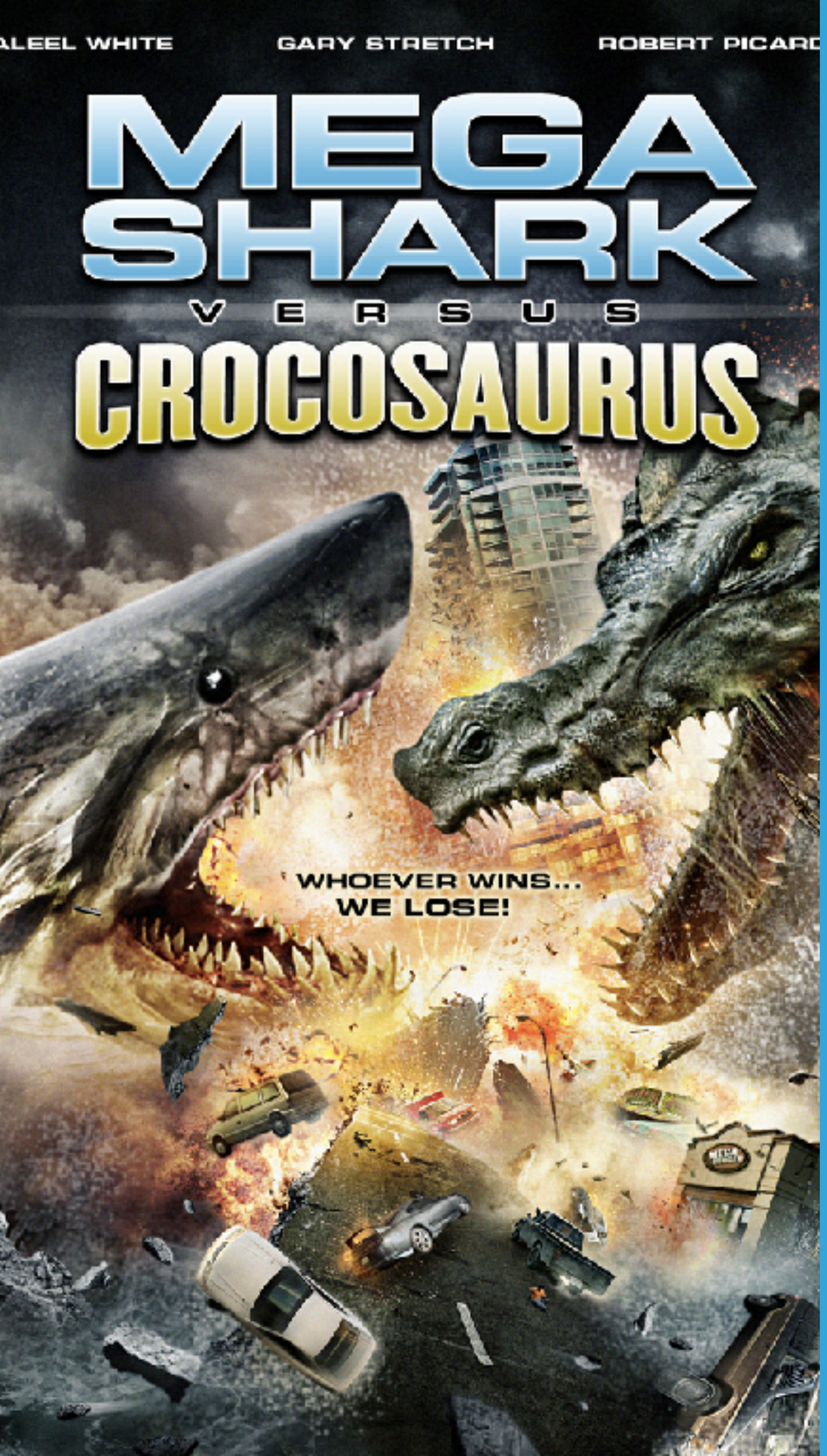





# PROJET CINÉ MEGASHARK

## GROUPE DE 2

## NOTÉ /20

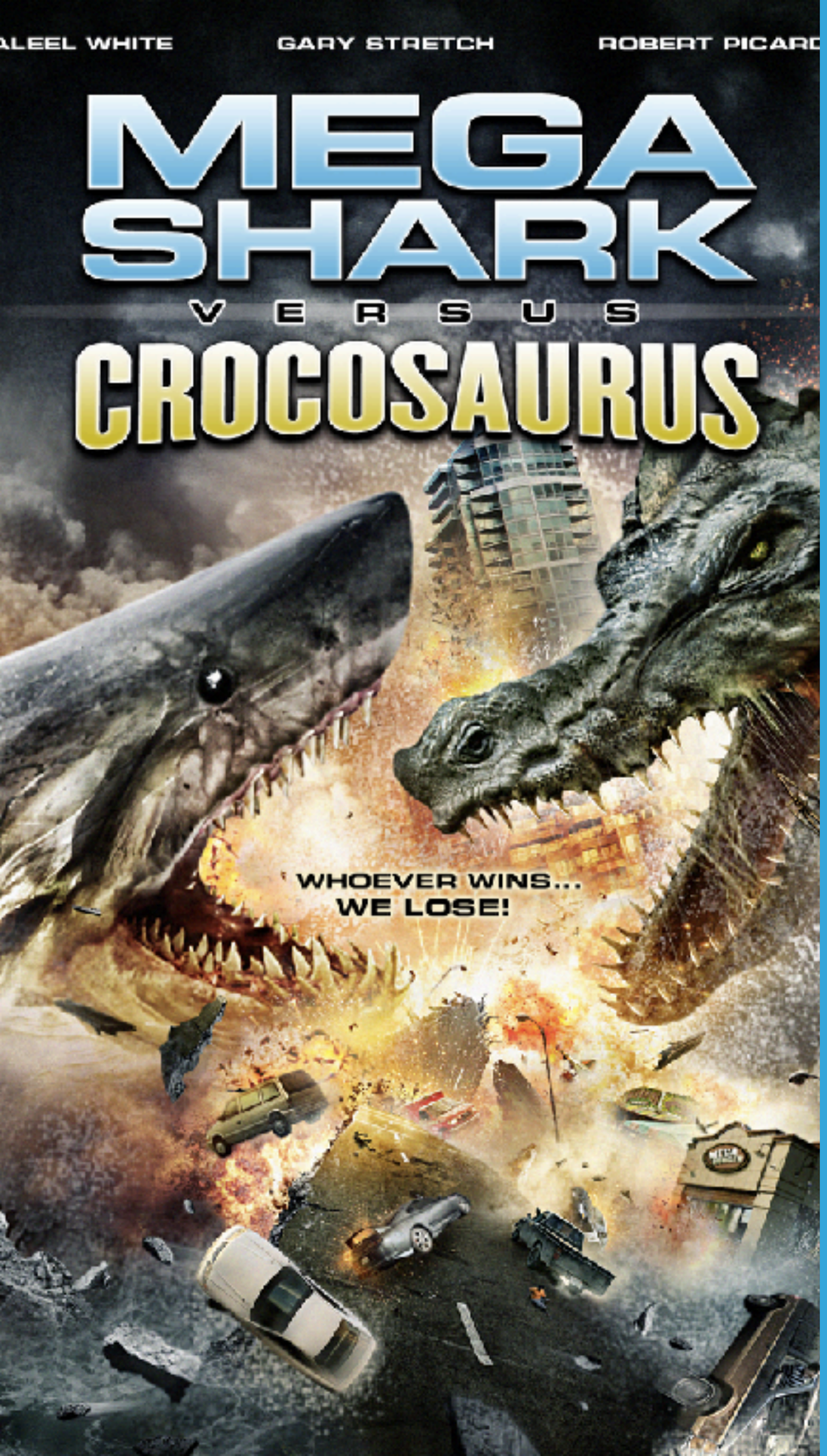



# VOTRE CLIENT :

Le cinéma MegaShark, petit cinéma indépendant, diffuse exclusivement des films de la société de production http://www.theasylum.cc/.

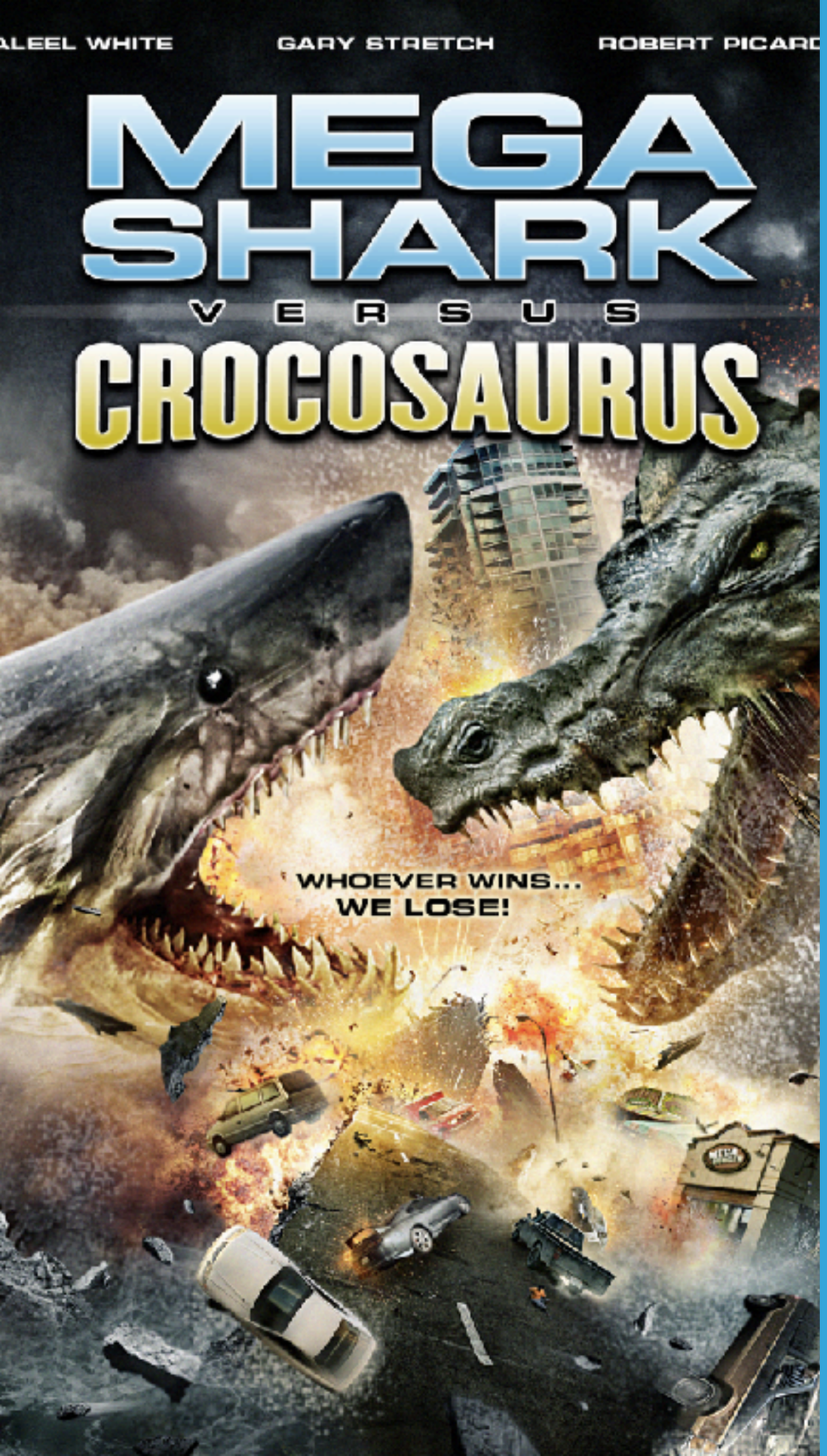
Il souhaite mettre en place un outil de gestion pour faciliter la planification des séances de cinéma.

Pour réaliser ce projet vous utiliserez CakePHP.




# FONCTIONNALITÉS :

- gérer la liste des films
- gérer la liste des salles
- planifier des séances




Un film :

- un nom
- une description
- durée
- année de sortie

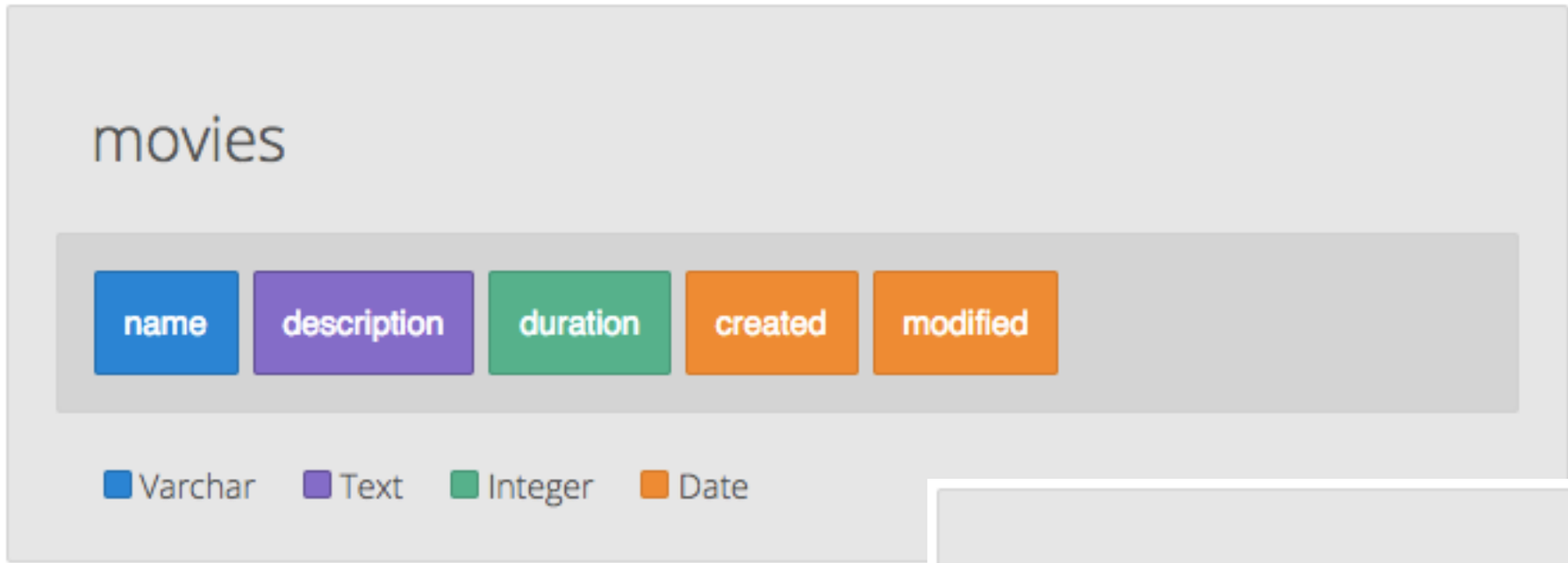
Une salle :

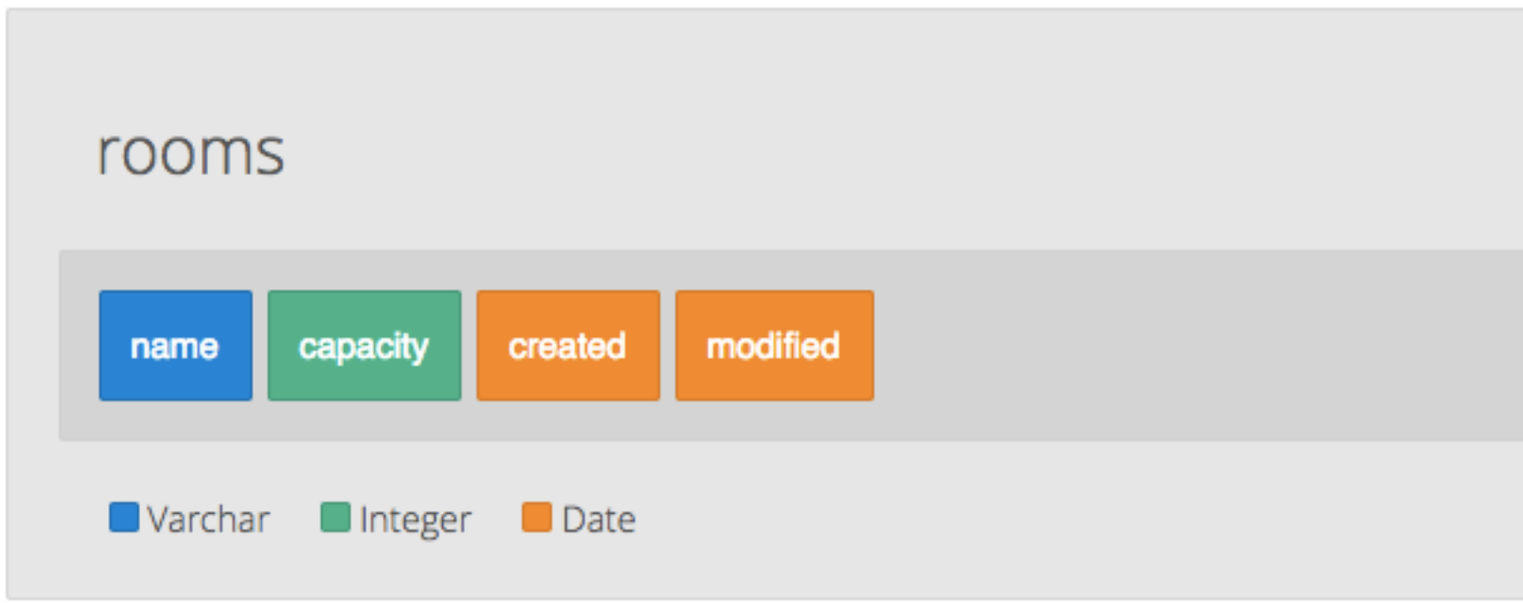
- nom
- capacité

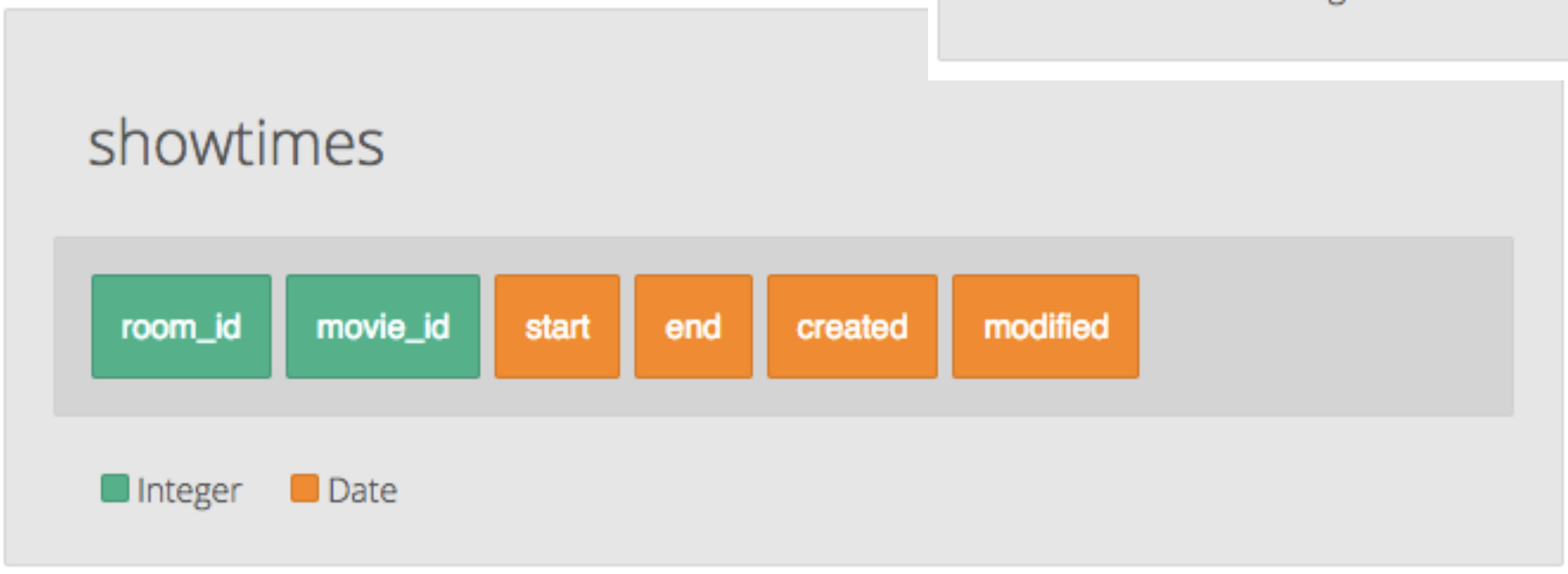
Une séance :

- un film
- une salle
- une heure de début
- une heure de fin

# BDD


movies
name
description
duration
created
modified
Varchar
Text
Integer
Date


rooms
name
capacity
created
modified
Varchar
Integer
Date


showtimes
room_id
movie_id
start
end
created
modified
Integer
Date

# FORK REPOSITORY (GITHUB : HTTPS://GITHUB.COM/BENJAMINLAMPERIER/MEGASHARK)

# INSTALL CAKEPHP (WITH COMPOSER)

# PHP-INTL (APT-GET)

# CONFIG APACHE

# START MYSQL

# CREATE DATABASE MEGASHARK

# CREATE DATABASE TESTS

# CREATE MIGRATION FOR TABLE MOVIES

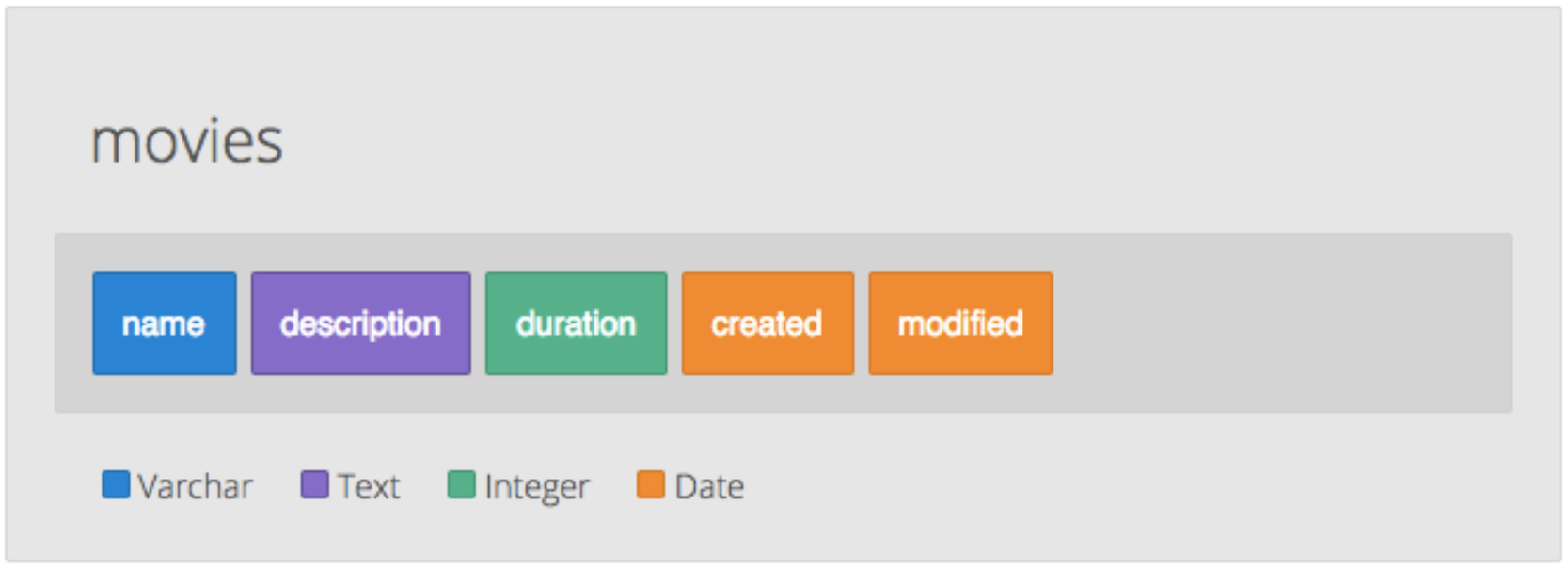
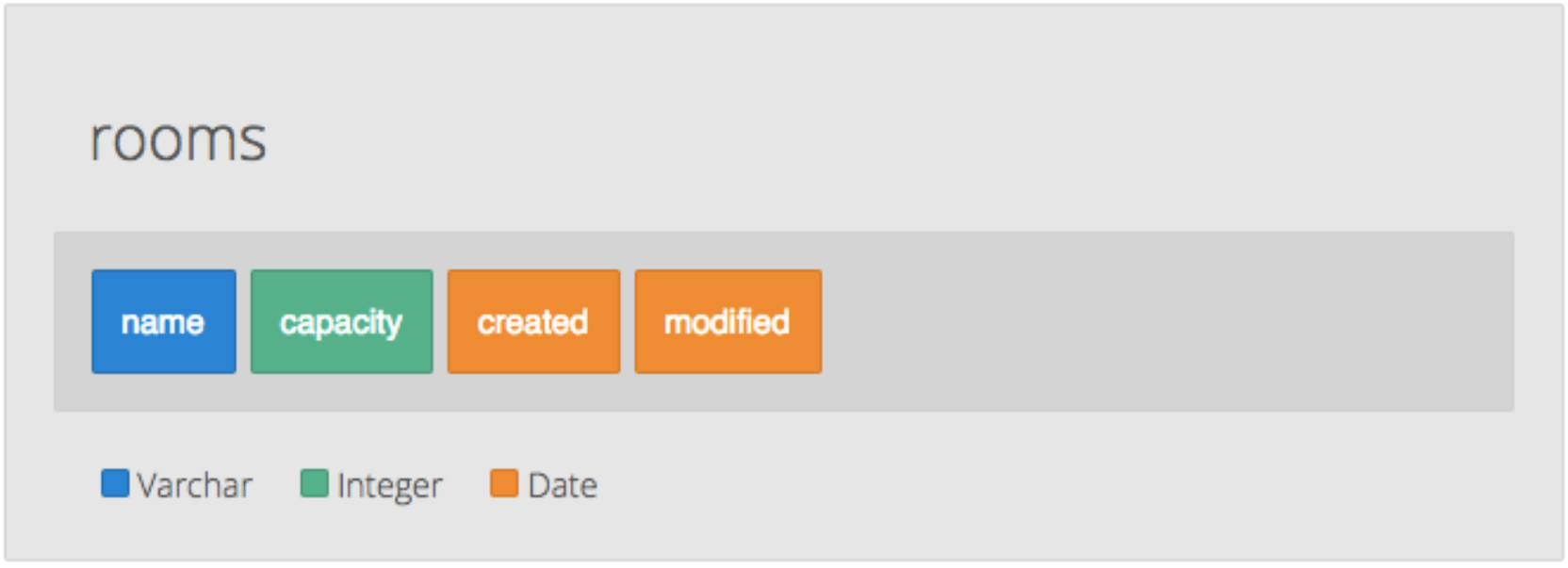
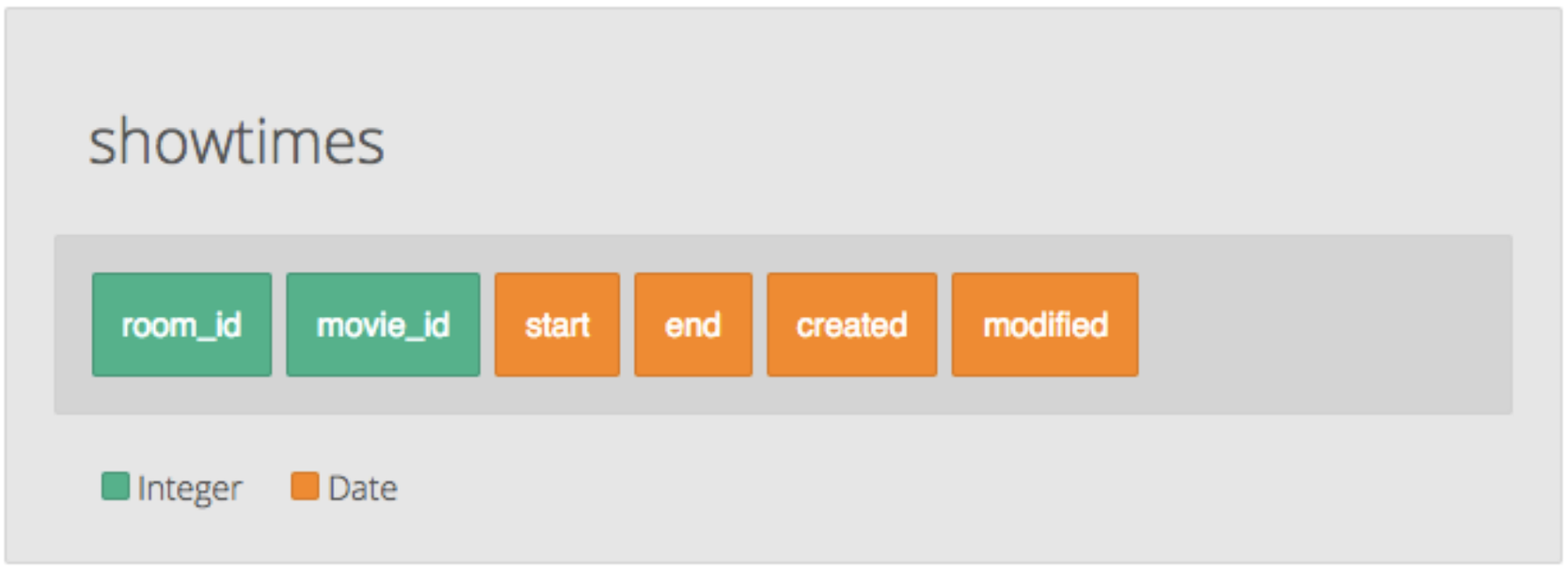
# MIGRATE

# BAKE

# ENJOY (COMPOSER RUN CHECK)

# LOOK THE CODE

# COMMIT



# STEP 3 - ROOMS AND SHOWTIMES