



Bài 21

MVC Model

Module: Advanced Programming with PHP

- Trình bày được kiến trúc phân tầng
- Trình bày được kiến trúc MVC
- Trình bày được các lợi ích của kiến trúc phân tầng và kiến trúc MVC
- Triển khai được các ứng dụng theo kiến trúc MVC



Thảo luận

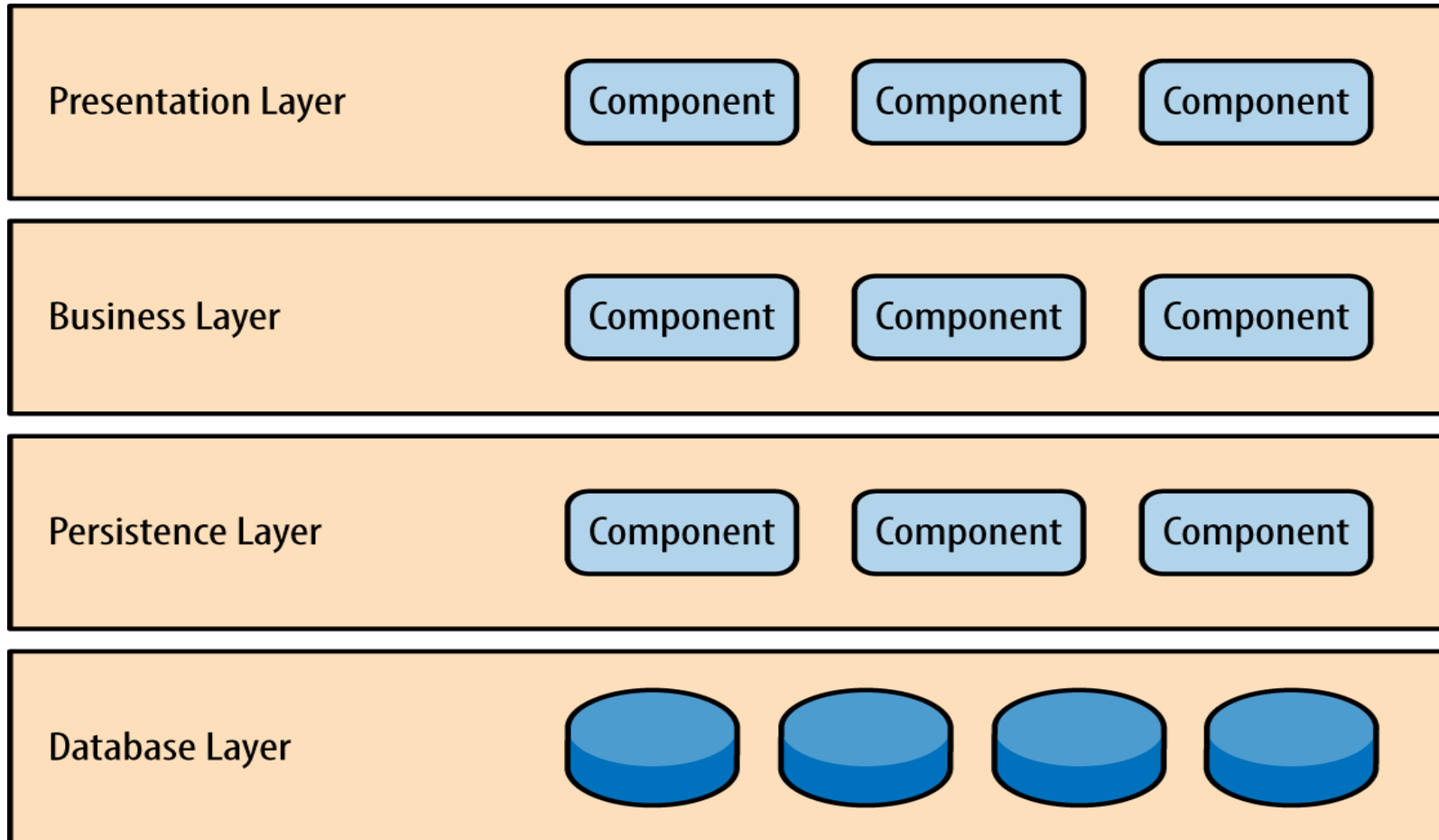
Kiến trúc MVC

Kiến trúc phân tầng



- Kiến trúc phân tầng (multilayered/multitier/n-tier/architecture) là một kiến trúc phần mềm trong đó các thành phần của hệ thống được tổ chức theo các tầng theo chiều ngang, mỗi tầng thực hiện một nhóm nhiệm vụ cụ thể
- Chẳng hạn: Tầng giao diện, tầng điều khiển, tầng dịch vụ, tầng lưu trữ dữ liệu...
- Dạng kiến trúc n-tier phổ biến là 3 tầng (three-tier)

Kiến trúc phân tầng: Ví dụ

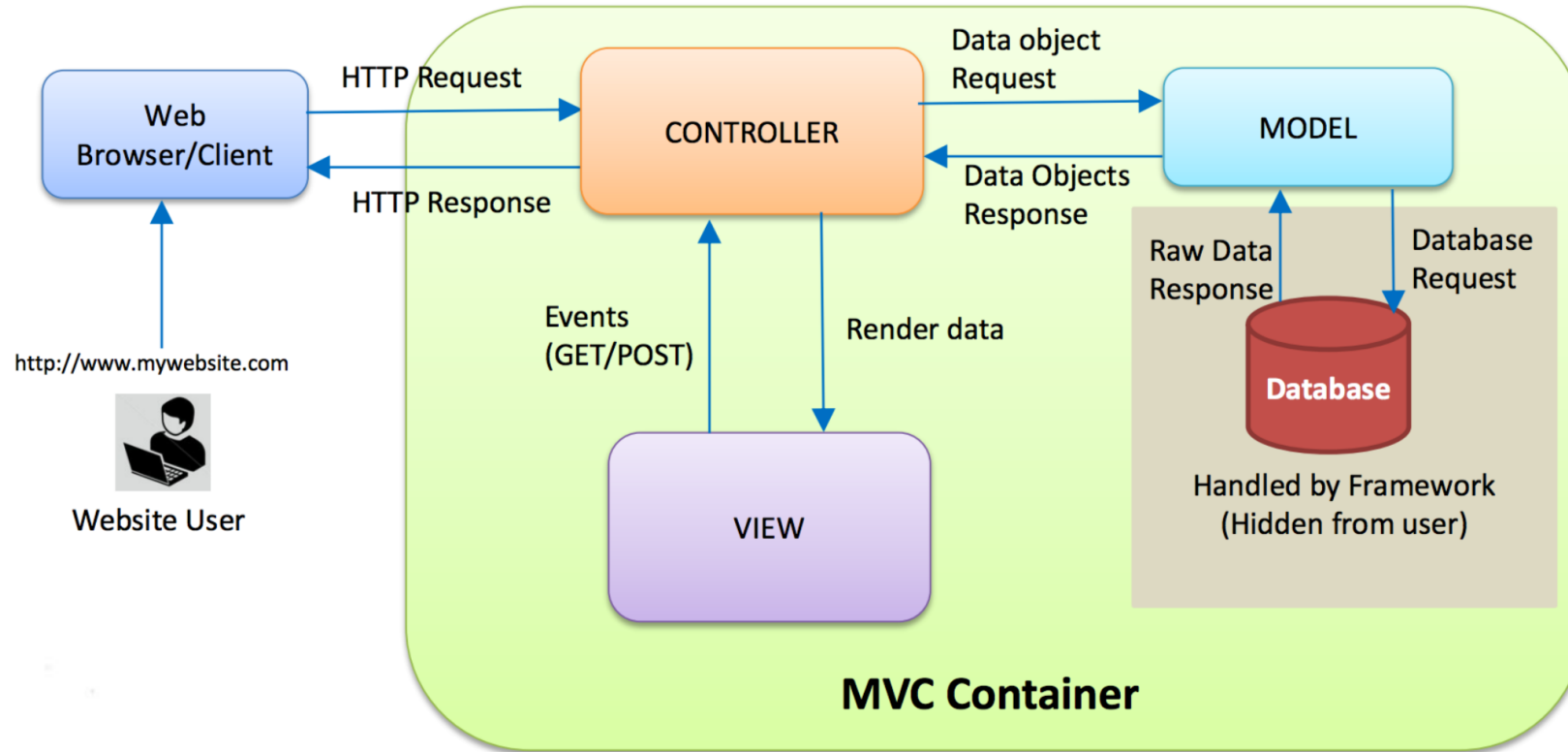


Kiến trúc MVC



- MVC là viết tắt của Model-View-Controller
- Kiến trúc MVC tổ chức các thành phần của hệ thống vào 3 tầng riêng biệt và có kết nối đến nhau:
 - Tầng Model: Biểu diễn dữ liệu và các logic nghiệp vụ
 - Tầng View: Hiển thị dữ liệu và là giao diện tương tác với người dùng
 - Tầng Controller: Xử lý các thao tác từ người dùng, kết nối giữa Model và View

MVC là gì



Lợi ích của kiến trúc MVC

- Dễ tái sử dụng
- Dễ bảo trì và mở rộng
- Tách phần view và phần nghiệp vụ riêng biệt
- Cho phép các Lập trình viên làm việc trên các thành phần khác nhau trong cùng một thời điểm

DEMO MVC

Ví dụ demo Model-View-Controller

Bài toán quản lý danh sách người dùng



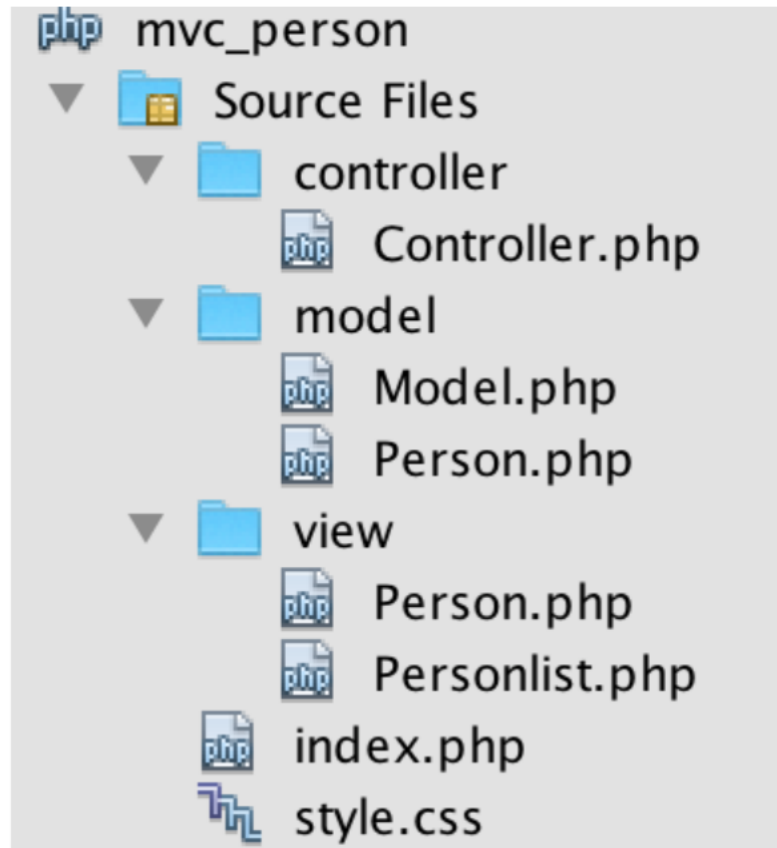
Title	Author	Description
Peter Pan	19XX	Một nhân vật hư cấu
Moonwalker	01/01/1900	I don't know
Trung	1985	He is very crazy



Title	Author	Description
Trung	1985	He is very crazy

[Back to list](#)

Cấu trúc project



Tạo file Model/Person.php



```
<?php
```

```
class Person {
```

```
    // Khai bao properties
```

```
    public $name;
```

```
    public $birthday;
```

```
    public $description;
```

```
    // Ham khoi tao
```

```
    public function __construct($name, $birthday, $description) {
```

```
        $this->name = $name;
```

```
        $this->birthday = $birthday;
```

```
        $this->description = $description;
```

```
    }
```

```
}
```

Tạo file Model/Model.php



```
include_once("model/Person.php");

class Model {

    /**...4 lines */
    public function getPersonList() {
        // here goes some hardcoded values to simulate the database
        return array(
            "Peter Pan" => new Person("Peter Pan", "19XX", "Một nhân vật hư cấu"),
            "Moonwalker" => new Person("Moonwalker", "01/01/1900", "I don't know"),
            "Trung" => new Person("Trung", "1985", "He is very crazy")
        );
    }

    /**...5 lines */
    public function getPerson($name) {
        $allPerson = $this->getPersonList();
        return $allPerson[$name];
    }

}
```

Tạo View/Person.php



```
<html>
  <head>
    <link rel="stylesheet" href="style.css" type="text/css" />
    <title>welcome <?php print($person->name);?> </title>
  </head>

  <body>

    <table>
      <tr>
        <th>Title</th>
        <th>Author</th>
        <th>Description</th>
      </tr>
      <?php
        echo '<tr><td>'. $person->name. '</td><td>'. $person->birthday
          . '</td><td>'. $person->description. '</td></tr>';
      ?>
    </table>
    <a href="index.php"><h1>Back to list</h1></a>

  </body>
</html>
```

Tạo View/Personlist.php



```
<html>
  <head>
    <link rel="stylesheet" href="style.css" type="text/css" />
    <title>welcome Demo </title>
  </head>

  <body>

    <table>
      <tr>
        <th>Title</th>
        <th>Author</th>
        <th>Description</th>
      </tr>
      <?php
        foreach ($persons as $name => $person) {
          echo '<tr><td><a href="index.php?person=' . $person->name . '">'
            . $person->name . '</a></td><td>' . $person->birthday
            . '</td><td>' . $person->description . '</td></tr>';
        }
      ?>
    </table>

  </body>
</html>
```

Tạo Controller/Controller.php



```
<?php
include_once("model/Model.php");

class Controller {
    public $model;

    public function __construct() {
        $this->model = new Model();
    }

    public function invoke() {
        if (!isset($_GET['person'])) {
            // no special book is requested, we'll show a list of all available books
            $persons = $this->model->getPersonList();
            include 'view/Personlist.php';
        } else {
            // show the requested book
            $person = $this->model->getPerson($_GET['person']);
            include 'view/Person.php';
        }
    }
}
```


Tạo Model/Model.php



```
include_once("model/Person.php");

class Model {

    /**...4 lines */
    public function getPersonList() {
        // here goes some hardcoded values to simulate the database
        return array(
            "Peter Pan" => new Person("Peter Pan", "19XX", "Một nhân vật hư cấu"),
            "Moonwalker" => new Person("Moonwalker", "01/01/1900", "I don't know"),
            "Trung" => new Person("Trung", "1985", "He is very crazy")
        );
    }

    /**...5 lines */
    public function getPerson($name) {
        $allPerson = $this->getPersonList();
        return $allPerson[$name];
    }
}
```

Tạo trang index.php



```
<?php
```

```
/*  
 * To change this license header, choose License Headers in Project Properties.  
 * To change this template file, choose Tools | Templates  
 * and open the template in the editor.  
 */  
include_once("controller/controller.php");  
  
$controller = new Controller();  
$controller->invoke();
```

Tóm tắt bài học



Hướng dẫn

- Hướng dẫn làm bài thực hành và bài tập
- Chuẩn bị bài tiếp: ***Tổng quan Laravel Framework***