**BÀI KIỂM TRA GIỮA KỲ LẦN 2**

CodeIgniter 4 là một framework PHP nhẹ và mạnh mẽ, thích hợp cho việc phát triển các ứng dụng web. Dưới đây là các bước cài đặt và cấu hình CodeIgniter 4 để chạy dự án của bạn.

**1. Các Vấn Đề Cài Đặt để Chạy CodeIgniter 4**

**Các bước cài đặt:**

1. **Cài đặt Composer:**

Truy cập Composer và tải về bản cài đặt phù hợp với hệ điều hành của bạn.

Sau khi cài đặt, kiểm tra bằng lệnh “Command prompt”: composer –version

1. **Tải CodeIgniter 4:**

Mở C:\xampp\htdocs và mở Command prompt chạy lệnh sau để tải CodeIgniter 4:

A black background with white text

Description automatically generated

1. **Cấu hình môi trường:**

* Trong thư mục dự án, sao chép file .env.example thành .env:

A black background with white text

Description automatically generated

* Mở file .env và cấu hình các thông số cần thiết như môi trường hoạt động (development, production), cơ sở dữ liệu, v.v.

1. **Thiết lập quyền cho thư mục writable:**

Đảm bảo rằng thư mục writable có quyền ghi phù hợp:

A black background with white text

Description automatically generated

1. **Chạy dự án:**

Sử dụng built-in PHP server để chạy dự án:

A black background with white text

Description automatically generated

1. **Fix các lỗi khi gặp phải**

* Fixed: the zip extension and unzip/7z commands are both missing xampp windows
* Mở **php.ini** tập tin. Bạn có thể định vị nó trong thư mục cài đặt PHP của mình. Ví dụ: nếu bạn đã cài đặt PHP ở định dạng C:\**php.ini**, thì **php.ini** tệp phải ở định dạng **C:\php\php.ini**.
* bỏ dấu chấm phẩy “ ;extension=zip ” thành “ extension=zip ”
* Fixed : Failed to download codeigniter4/framework from dist: curl error 28 while downloading
* Xóa cache của Composer trược khi tải codeigniter :

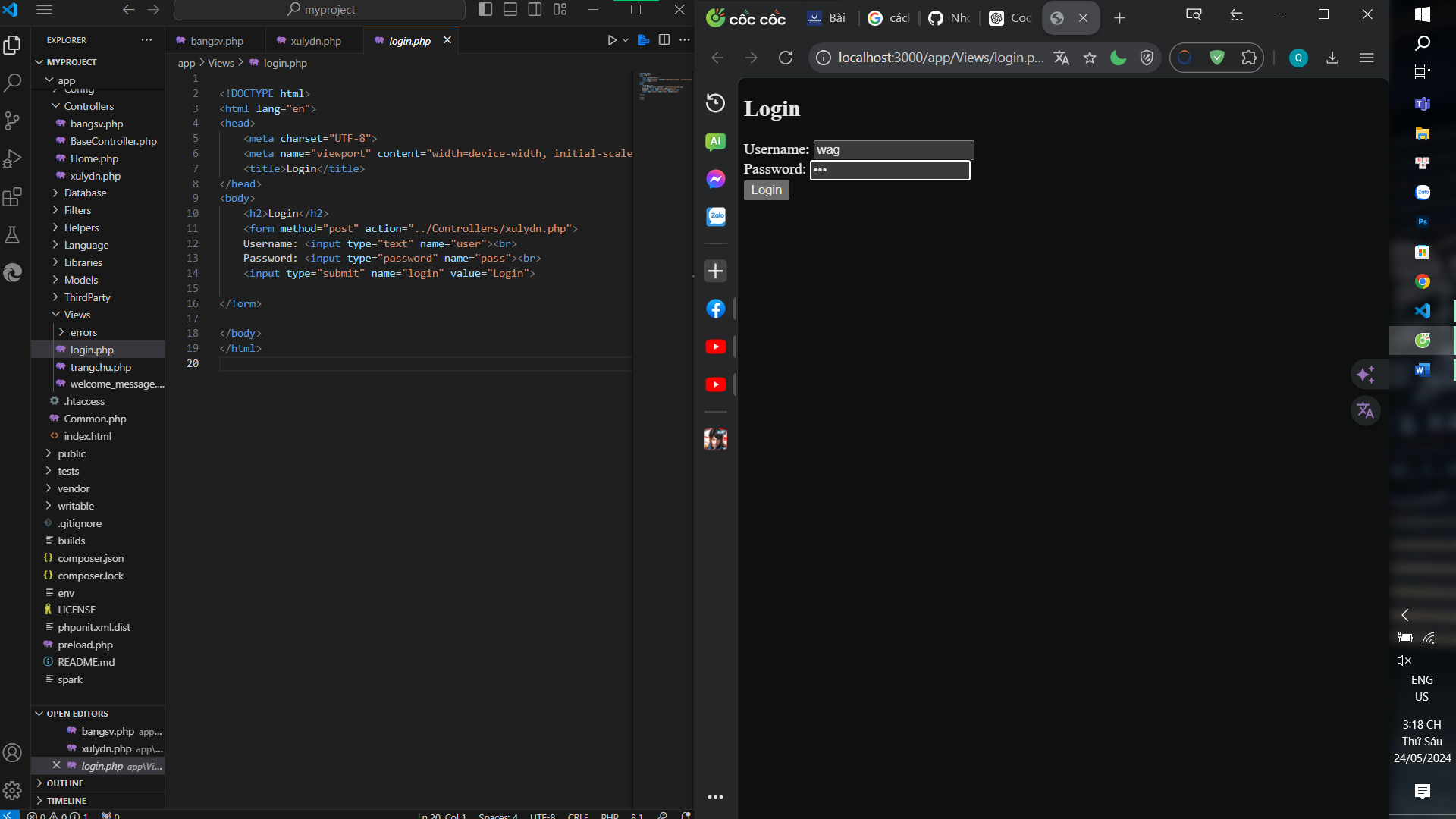
A black background with white text

Description automatically generated

* Fixer : tiện ích mở rộng ‘ intl ’ chưa được cài đặt
* Mở file php.ini: Tìm file php.ini. Vị trí thường là trong thư mục cài đặt PHP, ví dụ: C:\xampp\php\php.ini hoặc C:\wamp64\bin\php\php.ini.
* Tìm dòng ;extension=intl và bỏ dấu chấm phẩy (;) ở đầu dòng để kích hoạt tiện ích mở rộng này.

2. Screenshot màn hình chạy Project

Trang login.php/views -> xulydn.php/Controllers -> trangchu.php/views

****

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer screen

Description automatically generated**

**Tìm hiểu về codeignite 4**

CodeIgniter 4 là một framework PHP nhẹ và mạnh mẽ, nó thiết kế cho các nhà phát triển cần một bộ công cụ đơn giản và hiệu quả để tạo ra các ứng dụng web đầy đủ tính năng.

Với các lợi ích:

1. **Kích thước nhỏ và hiệu suất cao**:
   * CodeIgniter 4 có kích thước nhỏ, chỉ với 1.1MB, giúp tối ưu hóa hiệu suất và chạy nhanh hơn nhiều framework PHP khác​ ([codeigniter](https://www.codeigniter.com/" \t "_blank))​​ ([codeigniter](https://codeigniter.com/download" \t "_blank))​.
2. **Dễ sử dụng và học tập**:
   * Framework này được đánh giá cao về tính đơn giản và dễ sử dụng, phù hợp ngay cả với những người mới bắt đầu. Tài liệu chi tiết và dễ hiểu giúp các nhà phát triển dễ dàng bắt đầu nhanh chóng​ ([codeigniter](https://www.codeigniter.com/" \t "_blank))​​ ([htmlpanda](https://www.htmlpanda.com/blog/7-reasons-why-codeigniter-is-the-best-php-framework/" \t "_blank))​.
3. **Linh hoạt**:
   * CodeIgniter không bắt buộc phải tuân theo các quy ước lập trình nghiêm ngặt. Các nhà phát triển có thể lựa chọn phong cách mã hóa và phương pháp phát triển ưa thích của mình, điều này có thể có lợi cho các yêu cầu dự án khác nhau​ ([htmlpanda](https://www.htmlpanda.com/blog/7-reasons-why-codeigniter-is-the-best-php-framework/" \t "_blank))​.
4. **Kiến trúc MVC**:
   * Framework này tuân theo kiến trúc Model-View-Controller (MVC), giúp tách biệt logic với giao diện, làm cho mã nguồn trở nên có tổ chức hơn và dễ bảo trì hơn​ ([htmlpanda](https://www.htmlpanda.com/blog/7-reasons-why-codeigniter-is-the-best-php-framework/" \t "_blank))​​ ([trustradius](https://www.trustradius.com/products/codeigniter/reviews?qs=pros-and-cons" \t "_blank))​.

mặc dù là một framework phát triển web mạnh mẽ và linh hoạt, nhưng cũng có một số nhược điểm cần xem xét:

1. **Tài liệu chưa đầy đủ:** Mặc dù CodeIgniter 4 đã ra mắt từ một thời gian, nhưng tài liệu vẫn chưa hoàn chỉnh hoặc chi tiết đầy đủ, đặc biệt là so với các framework khác như Laravel. Điều này có thể làm tăng độ khó khi học và sử dụng framework cho người mới.
2. **Ít tính năng so với các framework khác:** So với các framework phát triển web khác như Laravel hoặc Symfony, CodeIgniter 4 cung cấp ít tính năng tích hợp sẵn hơn. Điều này có thể đồng nghĩa với việc bạn phải tự thêm các tính năng hoặc thư viện bên ngoài vào dự án của mình, làm tăng thời gian phát triển và khả năng phụ thuộc vào các bên thứ ba.

**Ví Dụ:** Laravel có Eloquent ORM và Symfony có Doctrine ORM, cả hai đều cung cấp một cách tiện lợi và mạnh mẽ để tương tác với cơ sở dữ liệu. giúp giảm thiểu việc viết mã SQL thủ công và tăng khả năng tái sử dụng mã. Mà codeigniter ko có

1. **Cơ sở dữ liệu không linh hoạt:** CodeIgniter giới hạn trong việc tương tác với cơ sở dữ liệu so với các framework khác. Cấu hình và tính năng của nó không cung cấp sự linh hoạt hoặc tiện ích mạnh mẽ như Laravel Eloquent hoặc Doctrine trong Symfony.
2. **Quản lý phụ thuộc thủ công:** CodeIgniter 4 yêu cầu quản lý các phụ thuộc thư viện và tính năng thủ công, thay vì sử dụng công cụ quản lý phụ thuộc như Composer như các framework khác.
3. **Cộng đồng nhỏ hơn:** So với các framework khác như Laravel, CodeIgniter có một cộng đồng nhỏ hơn, điều này có thể làm giảm khả năng tìm kiếm giải pháp cho các vấn đề cụ thể hoặc nhận được hỗ trợ từ cộng đồng

Dưới đây là công dụng của từng thành phần chính trong CodeIgniter 4:

Với Thư mục:

**Config**

Thư mục `Config` chứa các tệp cấu hình cho ứng dụng. Những tệp này cho phép thiết lập các cài đặt chung, như cấu hình cơ sở dữ liệu, email, bộ nhớ cache, và các thiết lập khác. Ví dụ:

- `App.php`: Cấu hình chung của ứng dụng.

- `Database.php`: Cấu hình kết nối cơ sở dữ liệu.

- `Routes.php`: Định tuyến URL của ứng dụng.

**Controllers**

Controllers là phần của mô hình MVC (Model-View-Controller) trong CodeIgniter. Controllers chịu trách nhiệm xử lý các yêu cầu HTTP, tương tác với các Models để lấy dữ liệu và trả về các Views để hiển thị kết quả cho người dùng.

- \*\*Ví dụ\*\*: Một controller có thể nhận yêu cầu từ URL, gọi model để lấy dữ liệu từ cơ sở dữ liệu và trả về một view hiển thị dữ liệu đó.

**Database**

Thư mục `Database` chứa các lớp và các tệp liên quan đến tương tác cơ sở dữ liệu.

**Filters**

Filters cho phép thực hiện các tác vụ trước và sau khi một yêu cầu được xử lý bởi controller. Chúng thường được sử dụng để kiểm tra quyền truy cập, xác thực người dùng, và lọc dữ liệu.

- \*\*Ví dụ\*\*: Một filter có thể kiểm tra xem người dùng có đăng nhập hay không trước khi cho phép truy cập vào một trang nhất định.

**Helpers**

Helpers là các hàm tiện ích giúp thực hiện các tác vụ thường gặp trong phát triển ứng dụng. Các helpers không phải là một phần của MVC nhưng cung cấp các chức năng bổ trợ.

- \*\*Ví dụ\*\*: URL Helper để tạo URL, Form Helper để xây dựng các biểu mẫu HTML.

**Language**

Thư mục `Language` chứa các tệp ngôn ngữ cho ứng dụng, cho phép bạn quốc tế hóa ứng dụng của mình bằng cách cung cấp bản dịch cho nhiều ngôn ngữ.

- \*\*Ví dụ\*\*: Bạn có thể tạo các tệp ngôn ngữ cho tiếng Anh, tiếng Việt, và các ngôn ngữ khác.

**Libraries**

Libraries là các lớp cung cấp các chức năng nâng cao và có thể được sử dụng lại trong toàn bộ ứng dụng. CodeIgniter có nhiều thư viện tích hợp sẵn như thư viện gửi email, thư viện thao tác tệp, và nhiều hơn nữa.

- \*\*Ví dụ\*\*: Email Library để gửi email, Session Library để quản lý phiên làm việc của người dùng.

**Models**

Models là phần của mô hình MVC, chịu trách nhiệm tương tác với cơ sở dữ liệu. Chúng lấy, lưu trữ và xử lý dữ liệu, sau đó trả dữ liệu đó cho controller.

- \*\*Ví dụ\*\*: Một model có thể đại diện cho một bảng trong cơ sở dữ liệu và chứa các phương thức để truy vấn dữ liệu từ bảng đó.

**ThirdParty**

Thư mục `ThirdParty` chứa các thư viện của bên thứ ba mà bạn có thể thêm vào ứng dụng của mình. Đây là nơi để lưu trữ các gói hoặc thư viện không phải là một phần của CodeIgniter nhưng bạn muốn sử dụng trong dự án của mình.

- \*\*Ví dụ\*\*: Một thư viện của bên thứ ba để xử lý các yêu cầu API bên ngoài.

**Views**

Views là phần của mô hình MVC, chịu trách nhiệm hiển thị dữ liệu cho người dùng. Chúng là các tệp HTML hoặc PHP chứa mã để hiển thị giao diện người dùng.

- \*\*Ví dụ\*\*: Một view có thể hiển thị danh sách các bài viết từ cơ sở dữ liệu, bao gồm cả HTML và PHP để lặp qua các bài viết và hiển thị chúng.

**Floder Main khác**

**Public**

thư mục "public" dùng để chứa các tài nguyên công khai như hình ảnh, tập tin CSS, JavaScript và các tài nguyên tĩnh khác.

**system**

Với thư mục "system" đây là nơi chứa mã nguồn của CodeIgniter, bao gồm các thành phần cơ bản như các lớp Controller, Model, View, thư viện hỗ trợ, và các tập tin cấu hình.

**TEst**

Thư mục "test" sử dụng để chứa các tập tin liên quan đến việc kiểm thử (testing) trong dự án phần mềm. Trong một dự án CodeIgniter, thư mục "test" có thể được sử dụng để lưu trữ các tập tin kiểm thử tự động (automated tests) hoặc các tập tin kiểm thử thủ công (manual tests).

**writable**

thư mục "writable" sử dụng để chứa các tập tin hoặc thư mục có thể ghi được .Các tập tin trong thư mục để lưu trữ dữ liệu động như các file log, cache, session hoặc các liệu tạm thời nào mà ứng dụng cần lưu trữ và truy cập một cách linh hoạt.

* **Logs**: ghi lại các thông tin quan trọng như lỗi, cảnh báo hoặc thông tin debug.
* **Caches**: được sử dụng để lưu trữ dữ liệu tạm thời nhằm tăng tốc độ truy cập dữ liệu. Các tệp cache này có thể là các bản sao của dữ liệu từ cơ sở dữ liệu hoặc kết quả tính toán trước đó.
* **Sessions**: Thư mục này thường chứa các file hoặc thư mục dùng để lưu trữ thông tin làm việc của người dùng. Thông tin phiên làm việc này có thể bao gồm thông tin đăng nhập, cài đặt ngôn ngữ, và các dữ liệu phiên khác.
* **Uploads**: Thư mục này thường được sử dụng để lưu trữ các tệp tin được tải lên từ người dùng. Điều này bao gồm tất cả các tệp tin được gửi từ các biểu mẫu tải lên trong ứng dụng web của bạn.

**Cấu hình Routes**

Và việc cấu hình Routes (định tuyến) trong CodeIgniter là một phần quan trọng của framework, nó sẽ giúp điều hướng các yêu cầu HTTP đến đúng controller và phương thức để xử lý.

**Xóa dấu chấm phẩy (;):** Trong file **php.ini**, dấu chấm phẩy ở đầu dòng đánh dấu rằng dòng đó là một comment và không được PHP đọc

**extension=intl**

mô-đun PHP Intl cung cấp các chức năng để:

1. **Xử lý văn bản đa ngôn ngữ**: Intl cung cấp các công cụ để xử lý văn bản trong các ngôn ngữ và kịch bản khác nhau. Điều này bao gồm việc sắp xếp chuỗi, định dạng ngày tháng và số, và các chức năng văn bản phong phú khác.
2. **Xử lý vùng và ngôn ngữ**: Cung cấp các chức năng để làm việc với các ngôn ngữ và vùng lãnh thổ khác nhau, bao gồm các hàm để chuyển đổi mã quốc gia, định dạng tiền tệ, và thực hiện các chức năng liên quan đến các khía cạnh văn hóa khác nhau.
3. **Xử lý văn bản Unicode**: Intl cung cấp hỗ trợ cho các chức năng xử lý văn bản Unicode, bao gồm các hàm để phân tích, so sánh và chuyển đổi các ký tự và chuỗi Unicode.