Bài tập thực hành 06

**MSSV: B2203469**

**Họ và tên sinh viên: Trương Hữu Thái**

Tạo CRUD với Laravel và CakePHP

**Mục tiêu cần đạt**:

Sau buổi này, các bạn cần nắm thao tác tạo các trang cơ bản CRUD (Create-Read-Update-Delete) với Laravel và CakePHP:

* Cài đặt project CakePHP với composer
* Thiết lập các lệnh sinh code tự động với Cakephp, nắm các quy tắc đặt tên
* Cài đặt gói sinh code tự động Laravel, nắm các lệnh và quy trình để sinh code với Laravel

Các bạn thực hiện viết code như hướng dẫn để làm quen, sau đó trả lời các **Yêu cầu** ở dưới bài hướng dẫn.

**Gợi ý Tham khảo:**

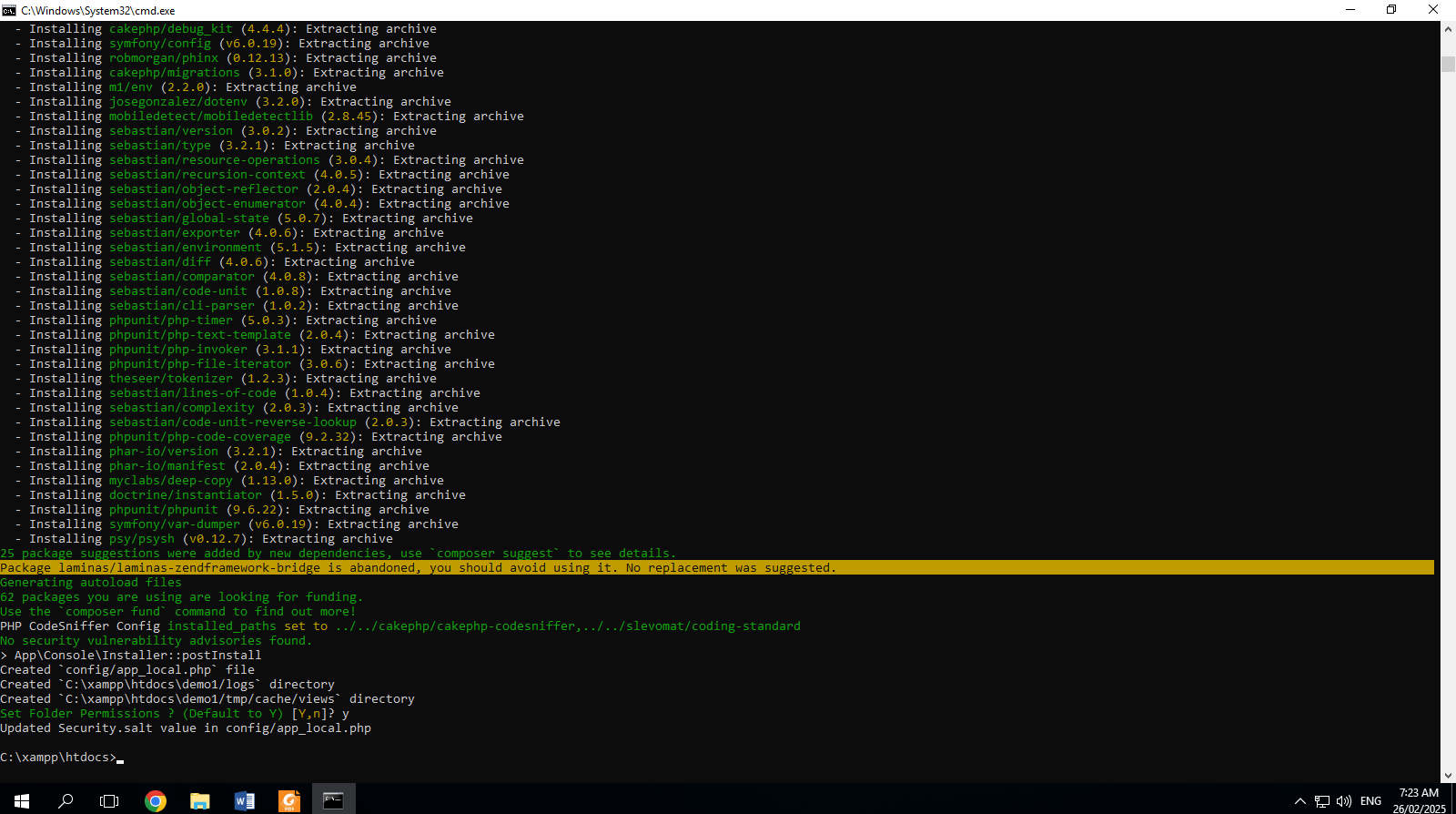
* <https://viblo.asia/p/gioi-thieu-package-laravel-generator-Eb85oMRkZ2G>
* [https://github.com/CrestApps/laravel-code-generato](https://github.com/CrestApps/laravel-code-generator)
* <https://book.cakephp.org/3/en/installation.html>: Cài đặt bản v3
* <https://book.cakephp.org/4/en/installation.html>: Cài đặt bản v4
* <https://book.cakephp.org/4/en/console-commands.html>: Các lệnh console cho cakephp
* <https://book.cakephp.org/4/en/orm/retrieving-data-and-resultsets.html#using-finders-to-load-data> và các nguồn khác mà sinh viên tìm được

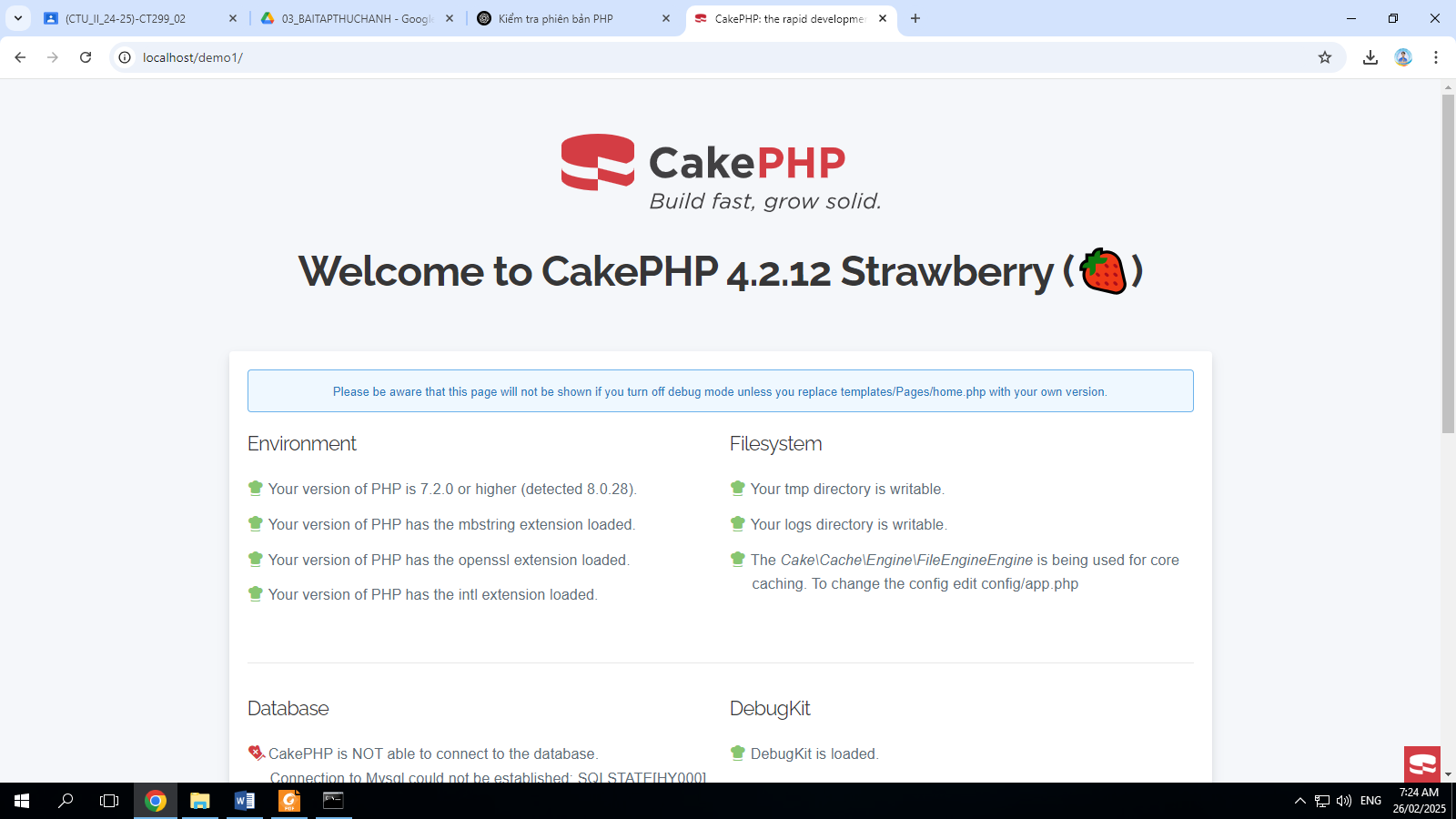
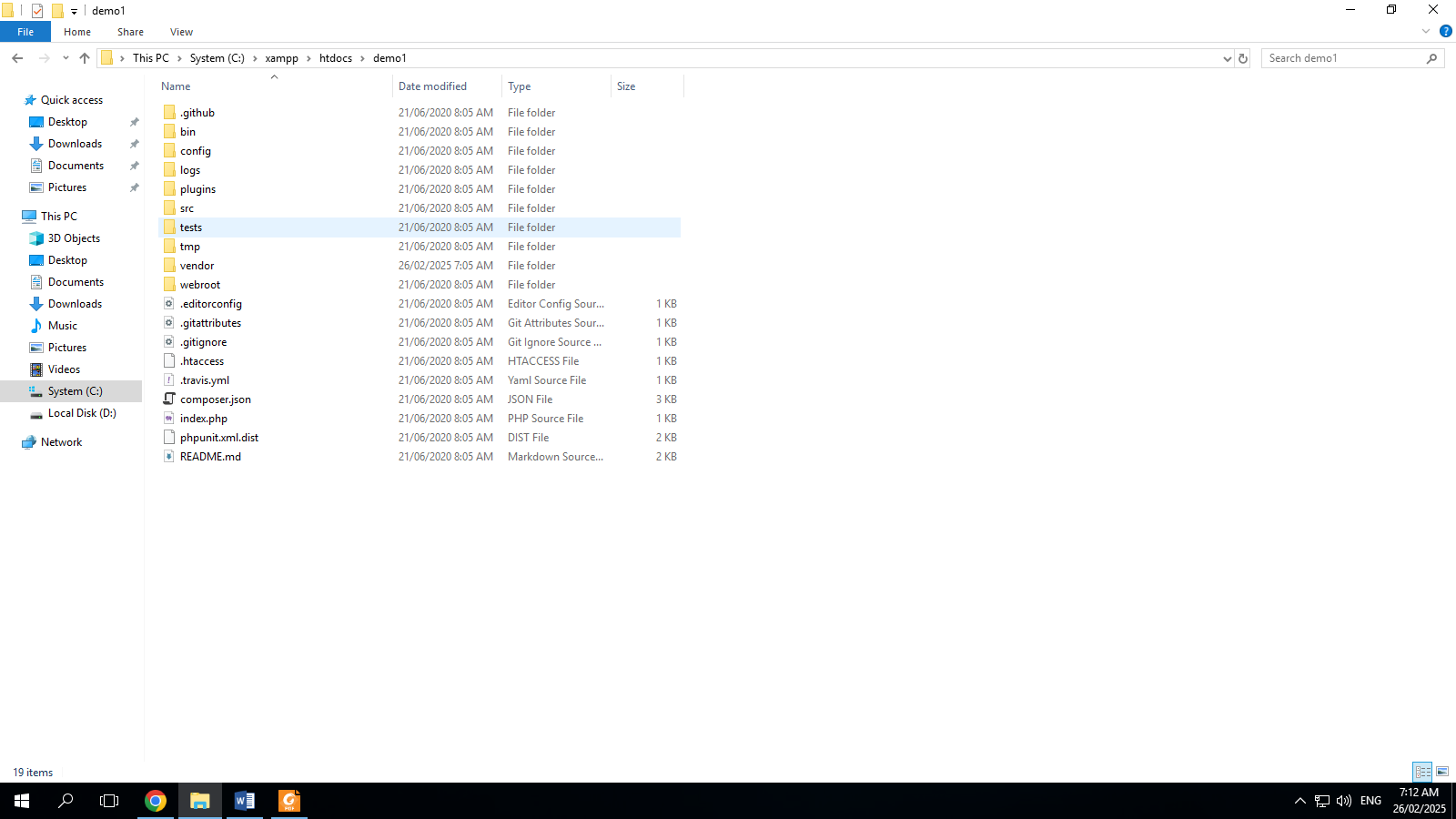
# Yêu cầu bài thực hành:

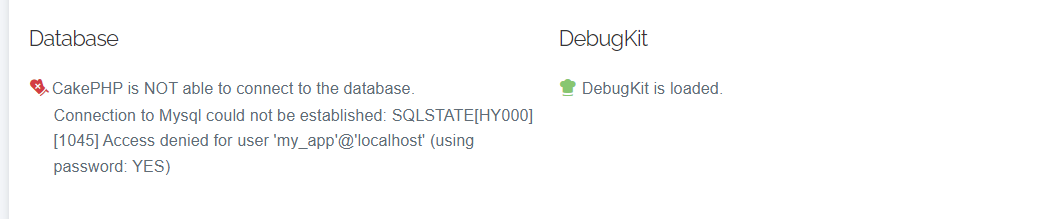
1. Bạn hãy chạy tất cả các lệnh hướng dẫn ở trên và chụp lại màn hình kết quả.

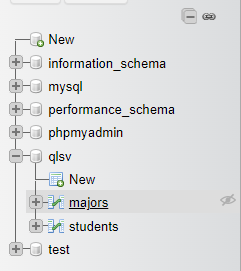
Tạo project với Cakephp

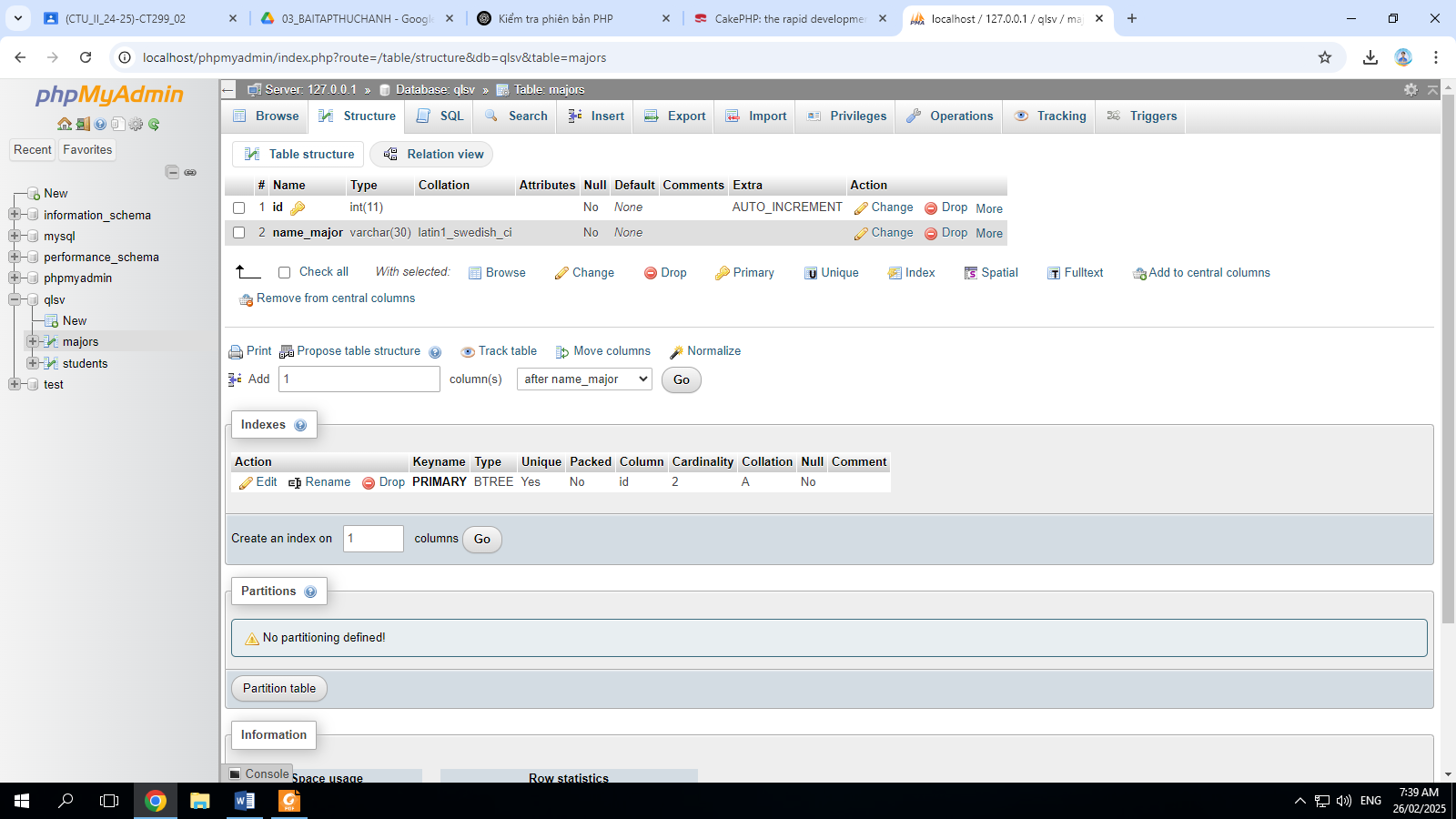
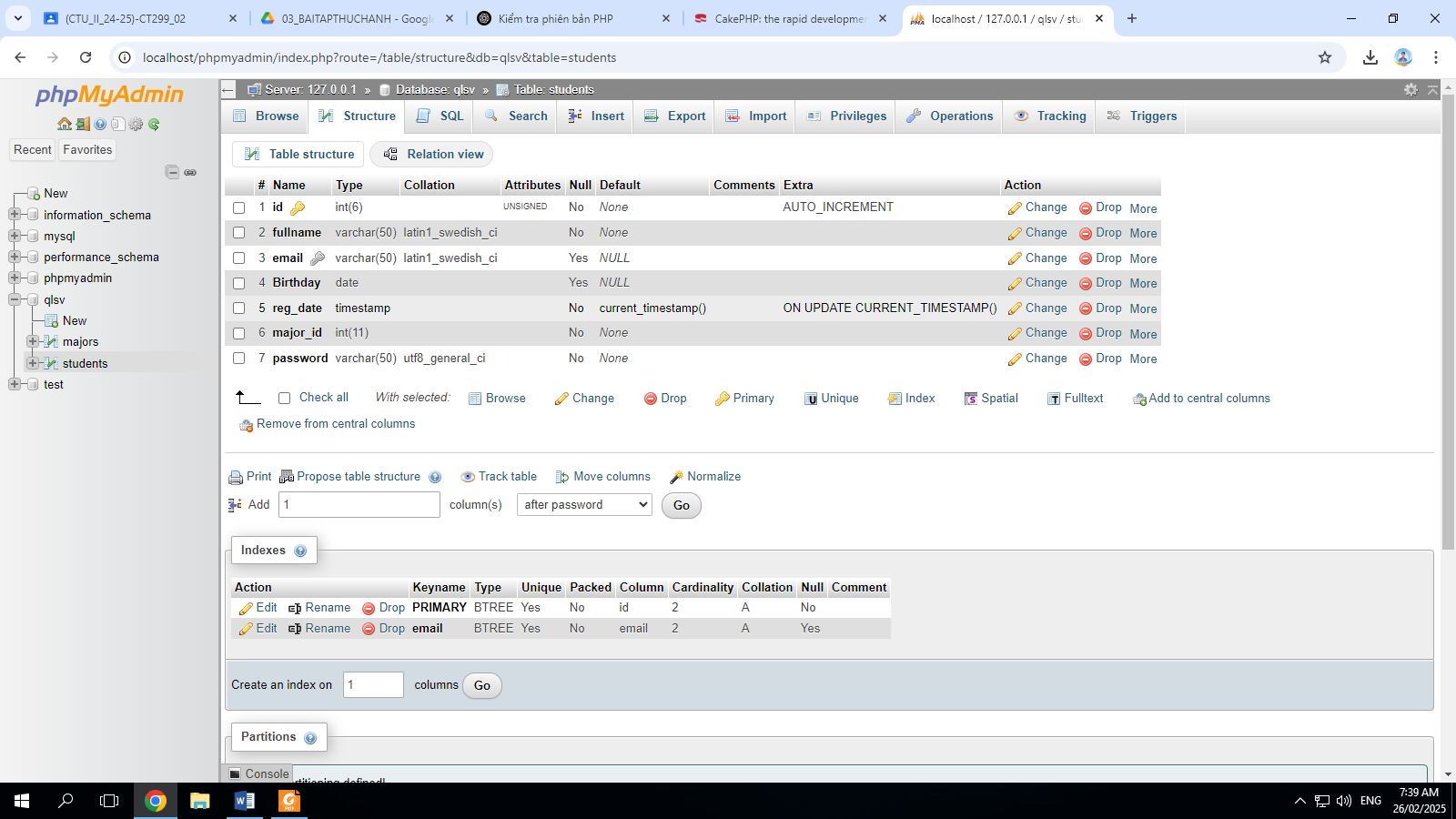
Các bạn cd vào xampp\htdocs sau đó dùng composer để tạo project Cakephp tên **demo1 bản** **v3.9**:  
composer create-project --prefer-dist cakephp/app:^3.9 demo1  
Quá trình tạo project demo1 sẽ mất vài phút



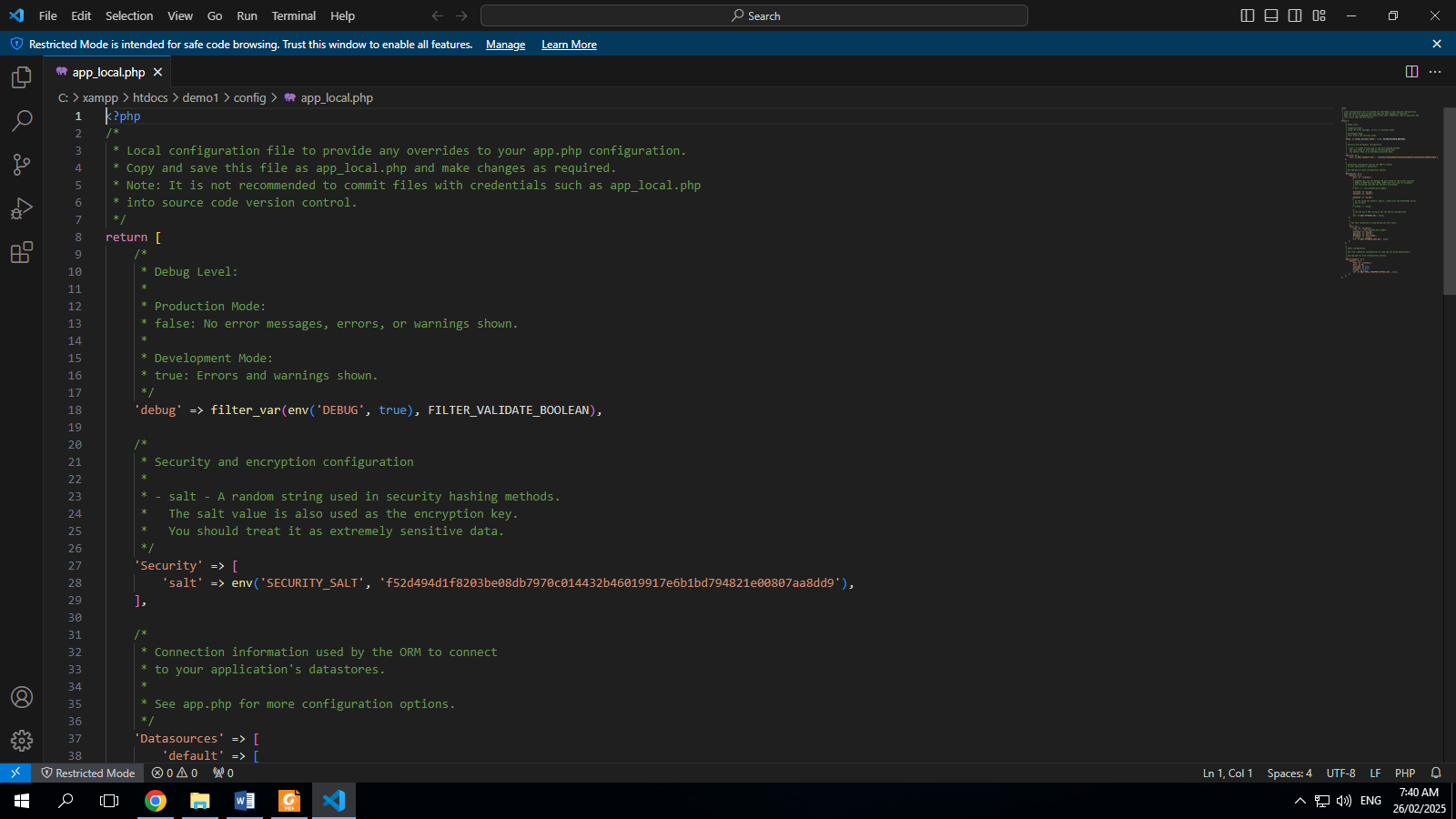
Chú ý là lúc này chúng ta chưa thiết lập kết nối với csdl, nên khi kéo xuống dưới ta sẽ thấy



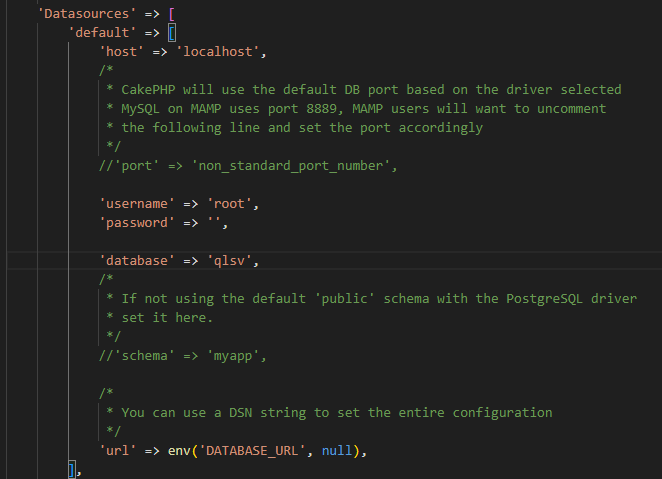
Tạo CSDL để thực hiện kết nối với project cakephp vừa tạo



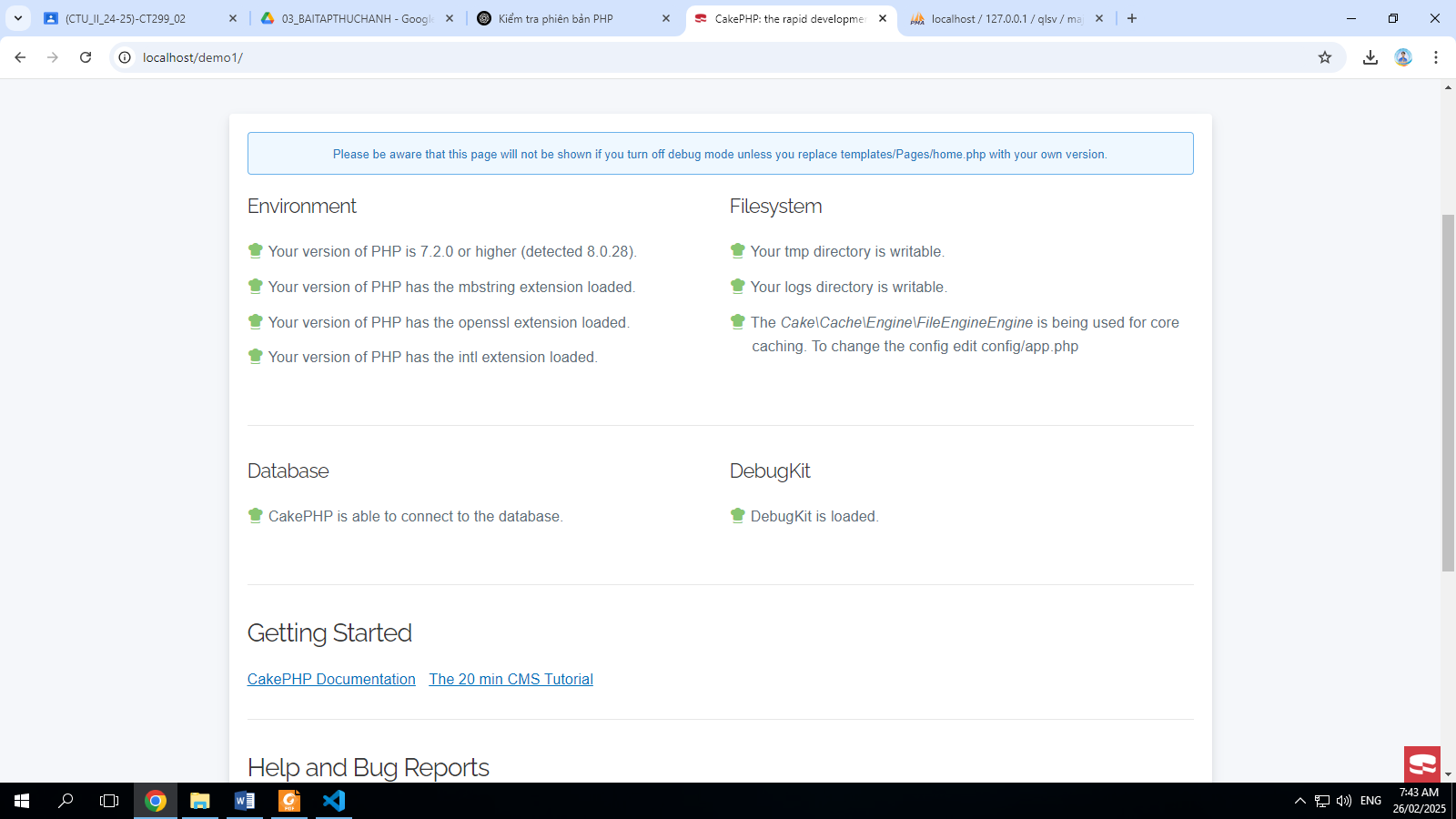
Truy cập đến C:\xampp\htdocs\demo1\config, mở file cấu hình **app\_local.php**



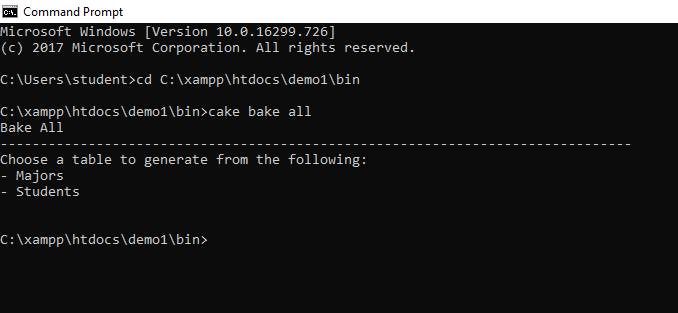
Chỉnh sửa lại username và password thành các tài khoản kết nối mysql trên phpmyadmin hợp  
lý (thường mặc định là root, password rỗng), chúng ta tạo csdl tên là qlsv nên khi chỉnh sửa sẽ  
là:



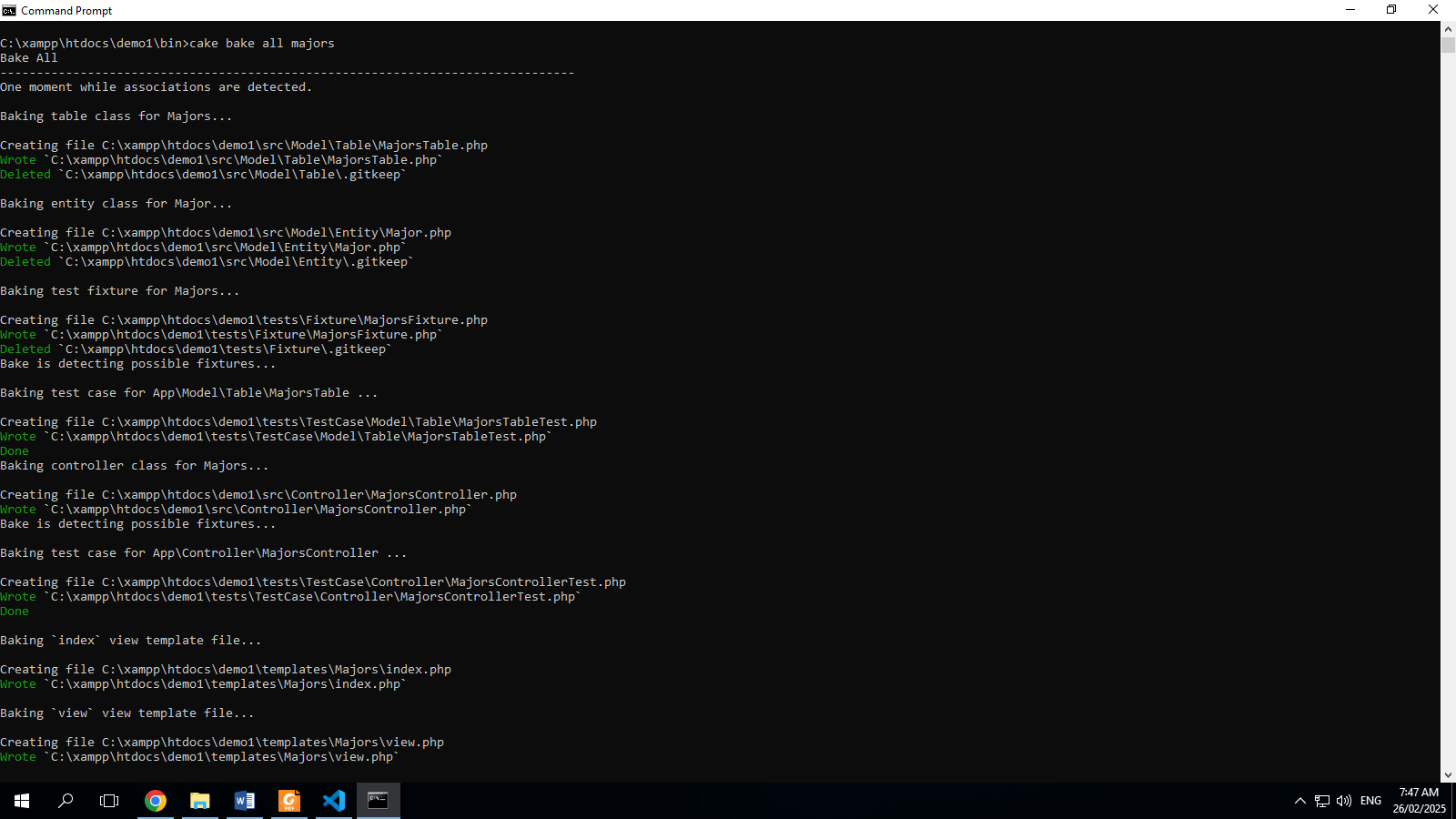
Bấm f5 lại đường link: http://localhost/demo1/, ta sẽ thấy:



Lúc này chúng ta đã kết nối đến CSDL thành công sau khi sửa file cấu hình ở app\_local.php.  
Sinh code tự động với CakePHP  
Sau khi chúng ta tạo CSDL qlsv có 2 bảng students và majors, các bảng này chúng ta đã đặt  
theo quy cách đặt tên của Cakephp và chúng ta có thể dùng lệnh hỗ trợ trong framework  
cakephp để sinh code tự động.  
Chúng ta trở lại giao diện dòng lệnh cmd ban nãy và gõ

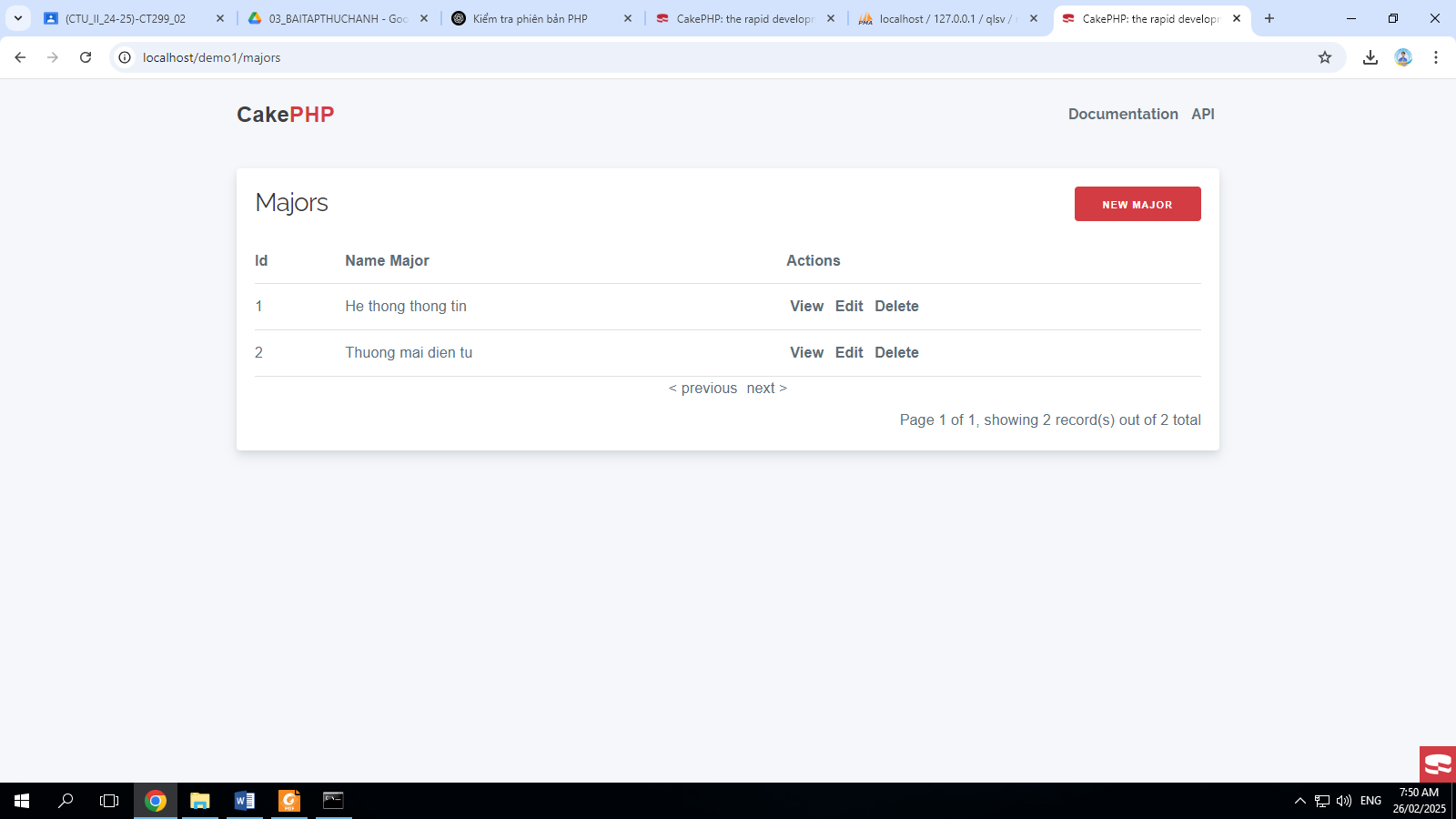
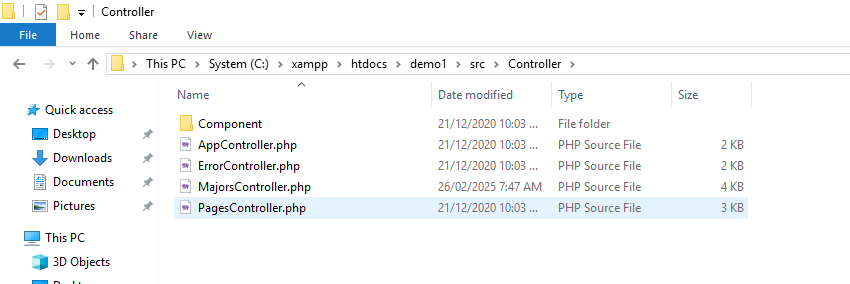


Chúng ta thấy cakephp nhận dạng được các bảng trong CSDL đã kết nối.  
Để tạo CRUD cho bảng majors (là bảng cha), ta thực hiện lệnh

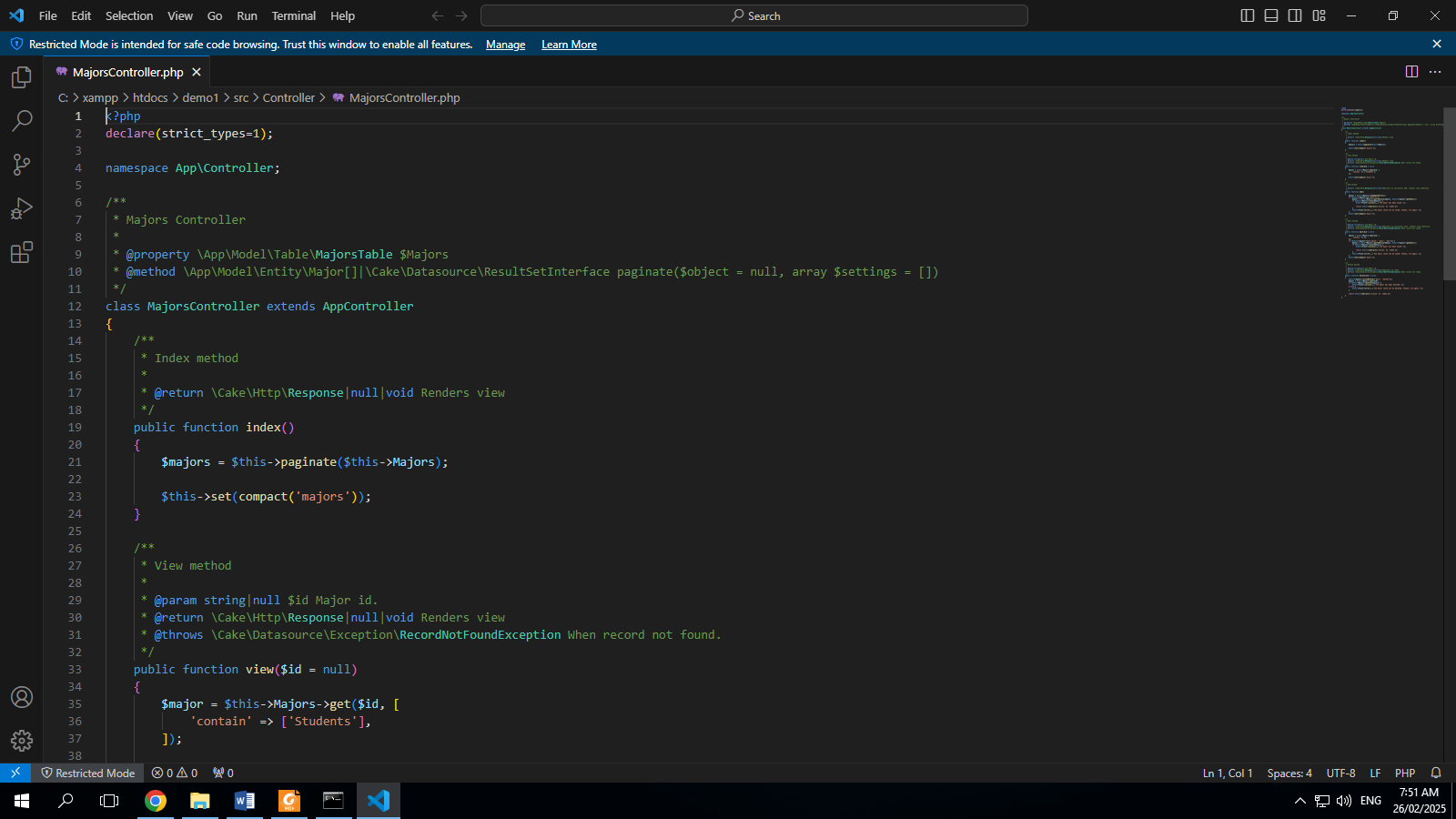
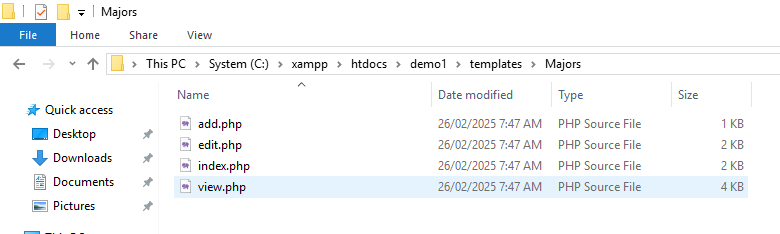


Một loạt tập tin sẽ được tạo, các tập tin này hình thành từ kiến trúc M-V-C (Model-ViewController) với các giao diện index, thêm, sửa, hiển thị chi tiết của bảng majors nằm trong các  
thư mục Model, View (hoặc trong Template), Controller

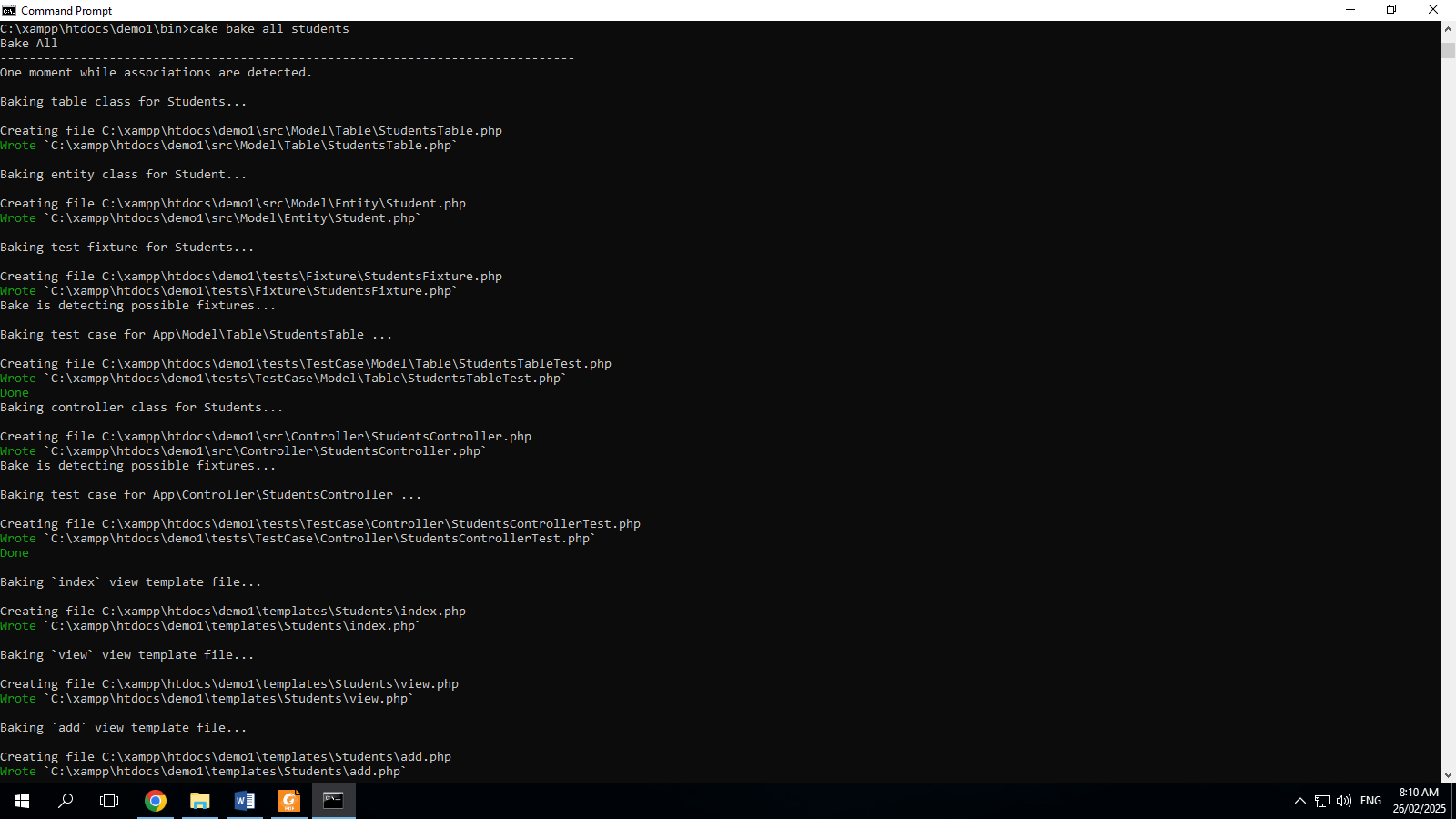
Giờ lên trình duyệt ta gõ: <http://localhost/demo1/majors>

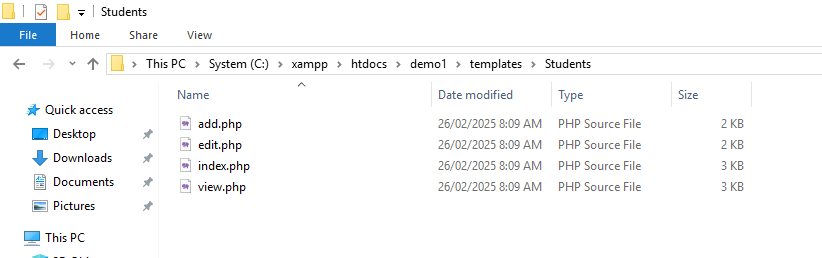
Các phương thức này đã được cakephp định nghĩa trong Controller: majors:

Mở tập tin MajorsController.php, chúng ta sẽ thấy các phương thức đã được định nghĩa trong  
tập tin này:

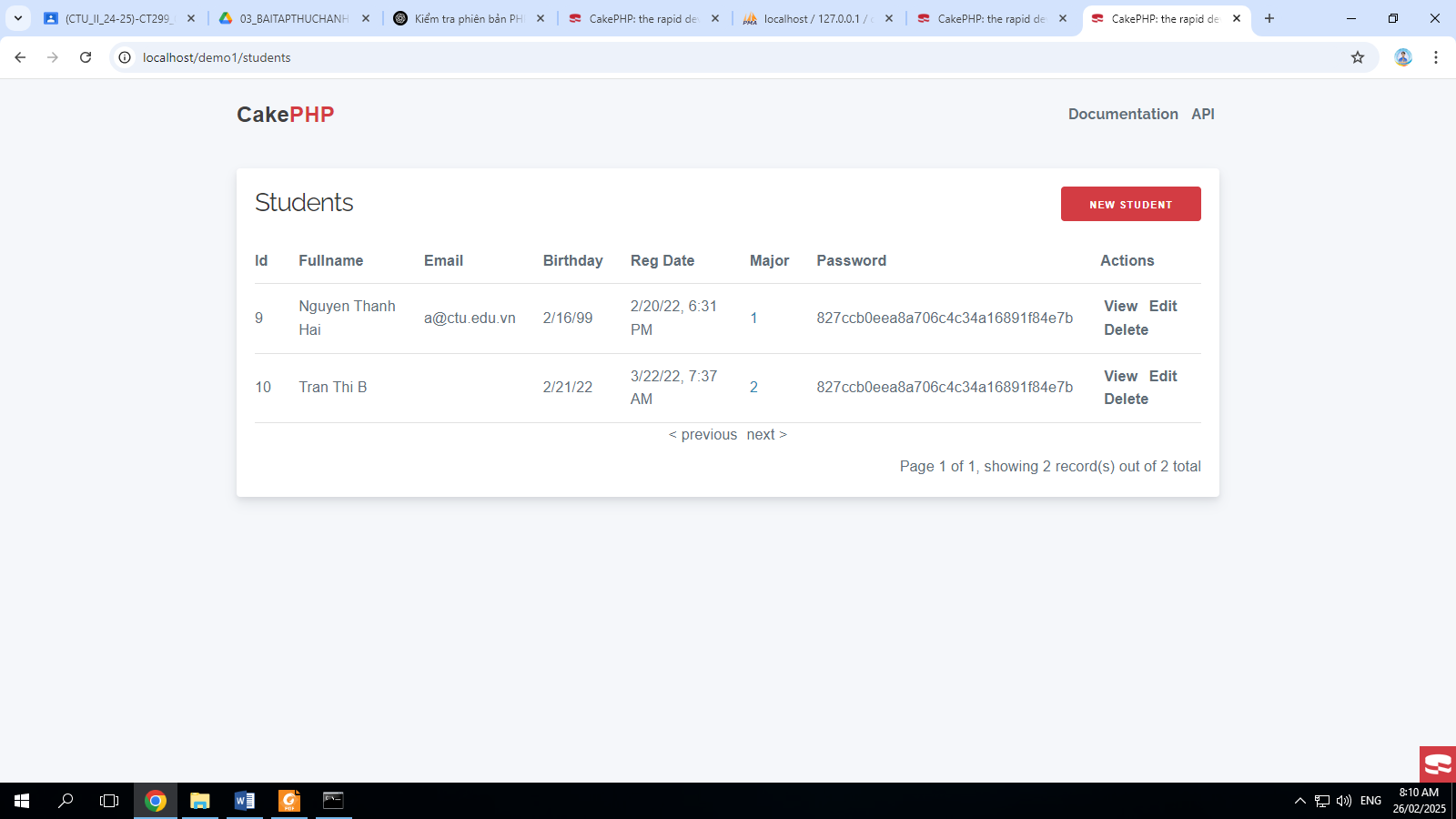
Các phương thức định nghĩa trong Controller sẽ xử lý những hành động trong các tập tin ở  
phần hiển thị nằm trong thư mục Template/Majors

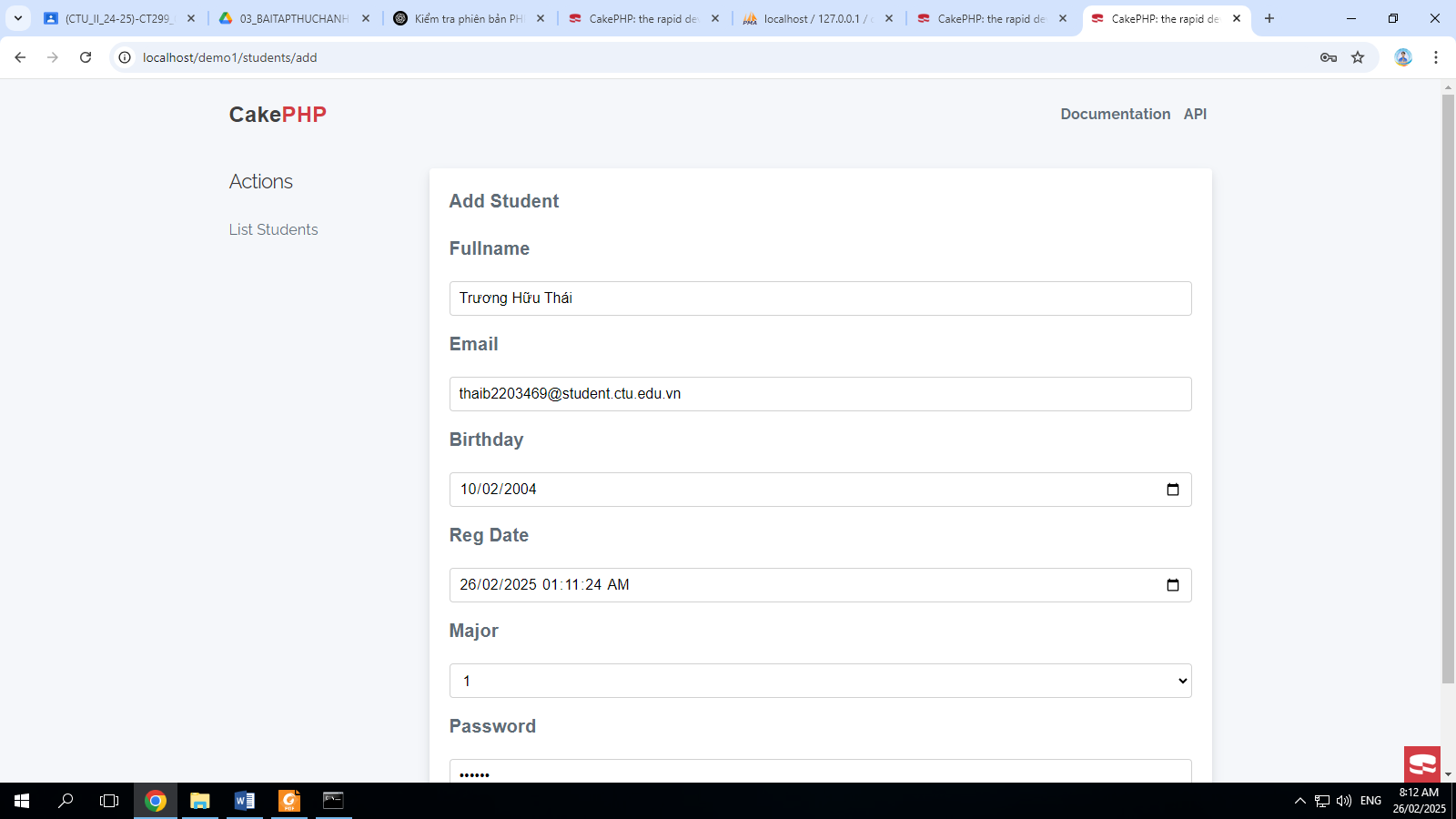
Giờ lặp lại tương tự đối với bảng students

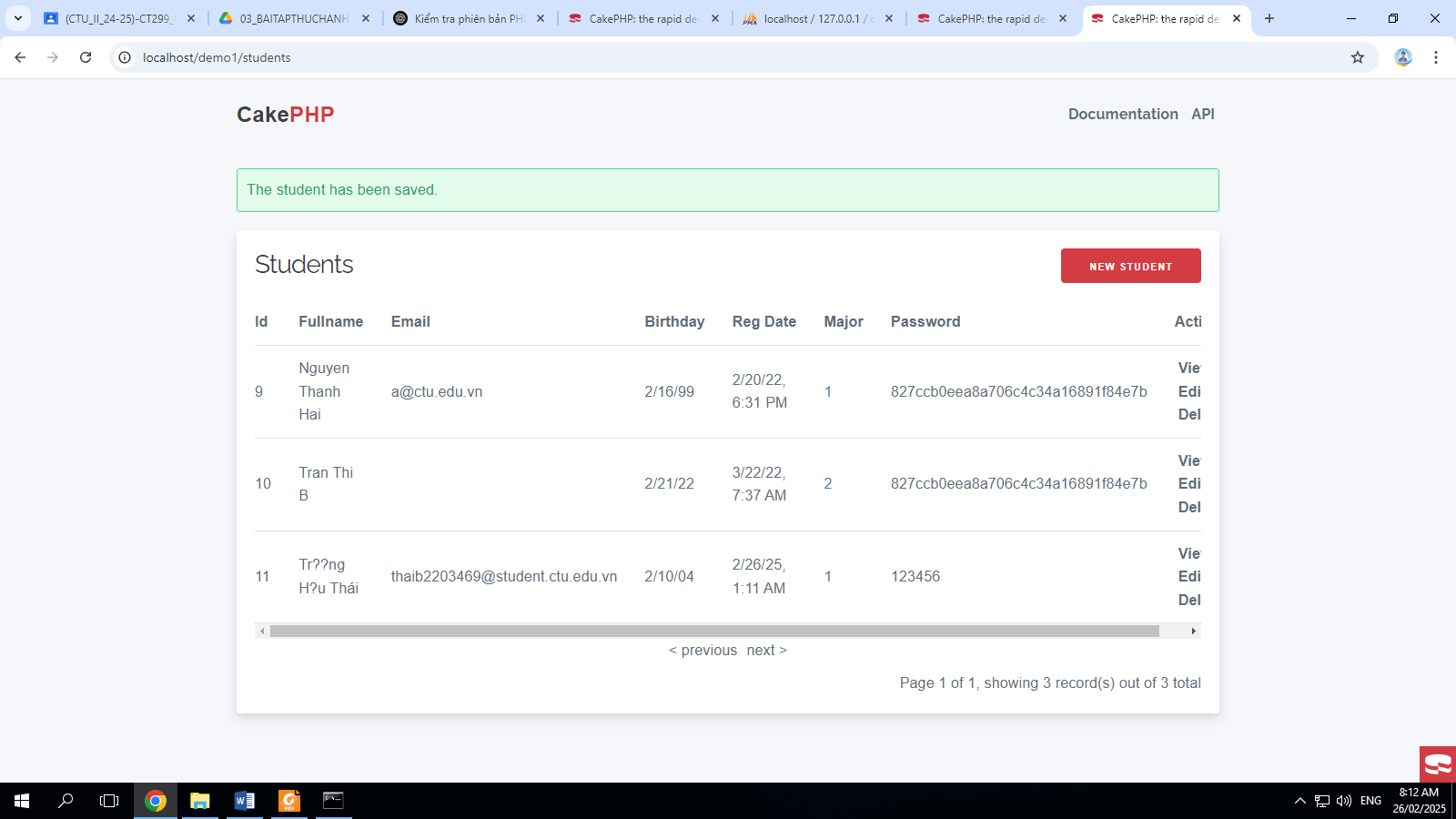


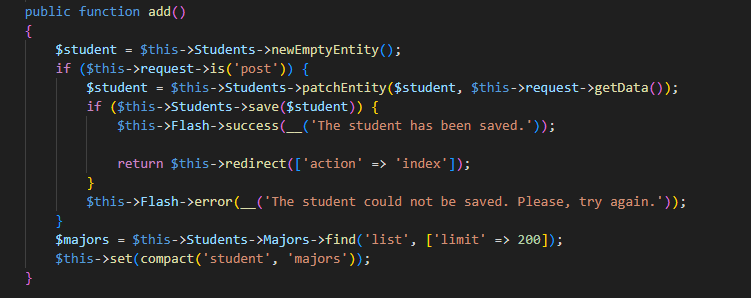


Truy cập thử: <http://localhost/demo1/students>

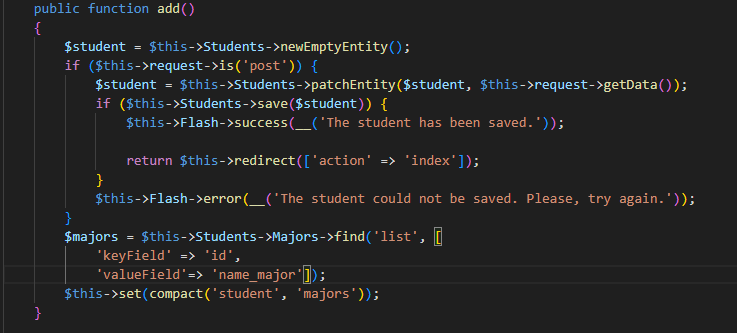
\Click vào “New Student” để đến form tạo sinh viên mới



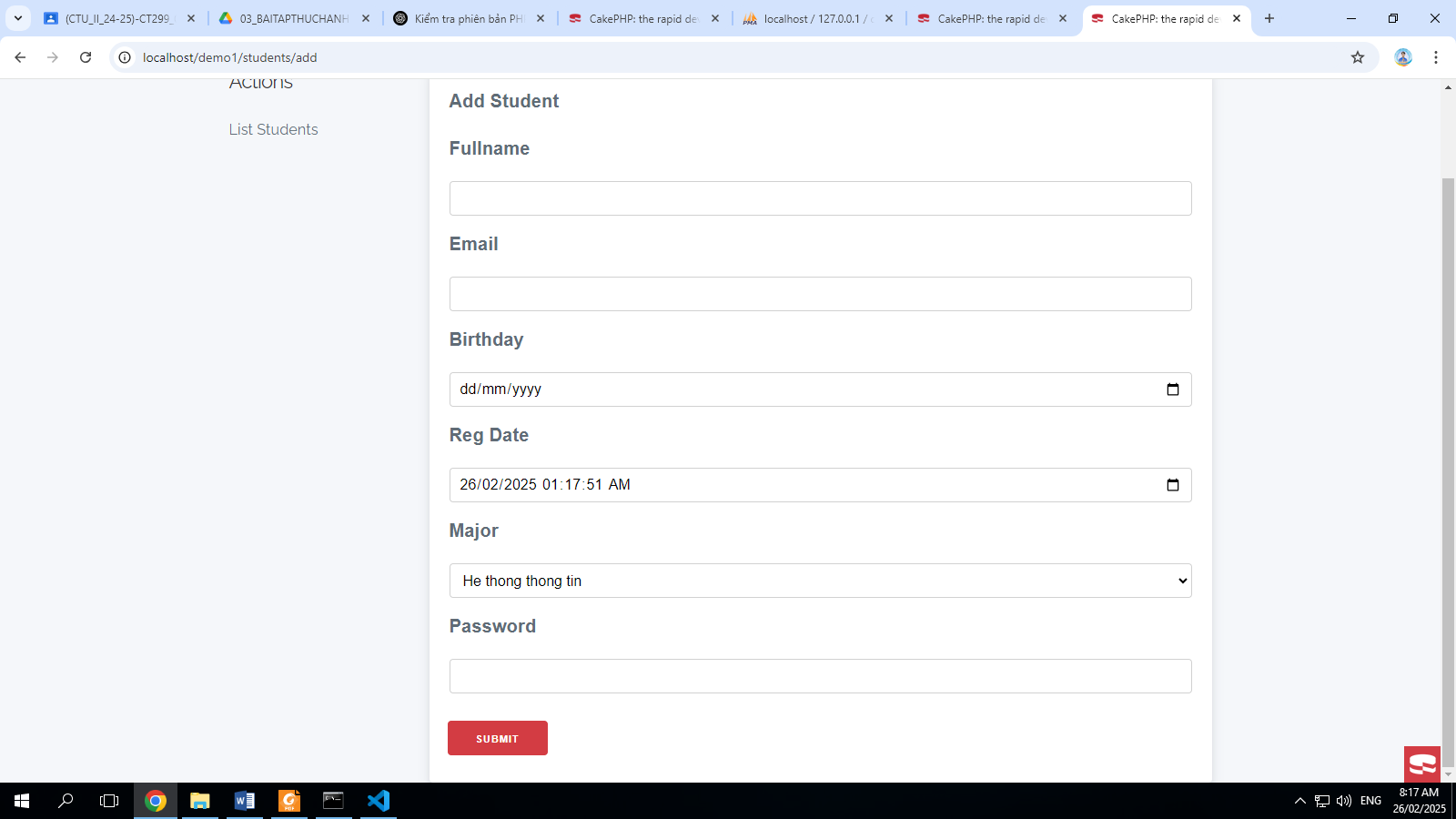


Ta sẽ thấy được cakephp tự động tạo cho chúng ta các ô nhập tùy theo kiểu dữ liệu mà chúng  
ta đã tùy chỉnh trong MySQL, ngay cả khóa ngoại cũng sẽ là dạng combobox (Major)  
Quan sát trong phương thức add trong Controller của Student ta sẽ thấy biến này được thấy từ  
bảng Majors đã được kết nối tự động với bảng Student

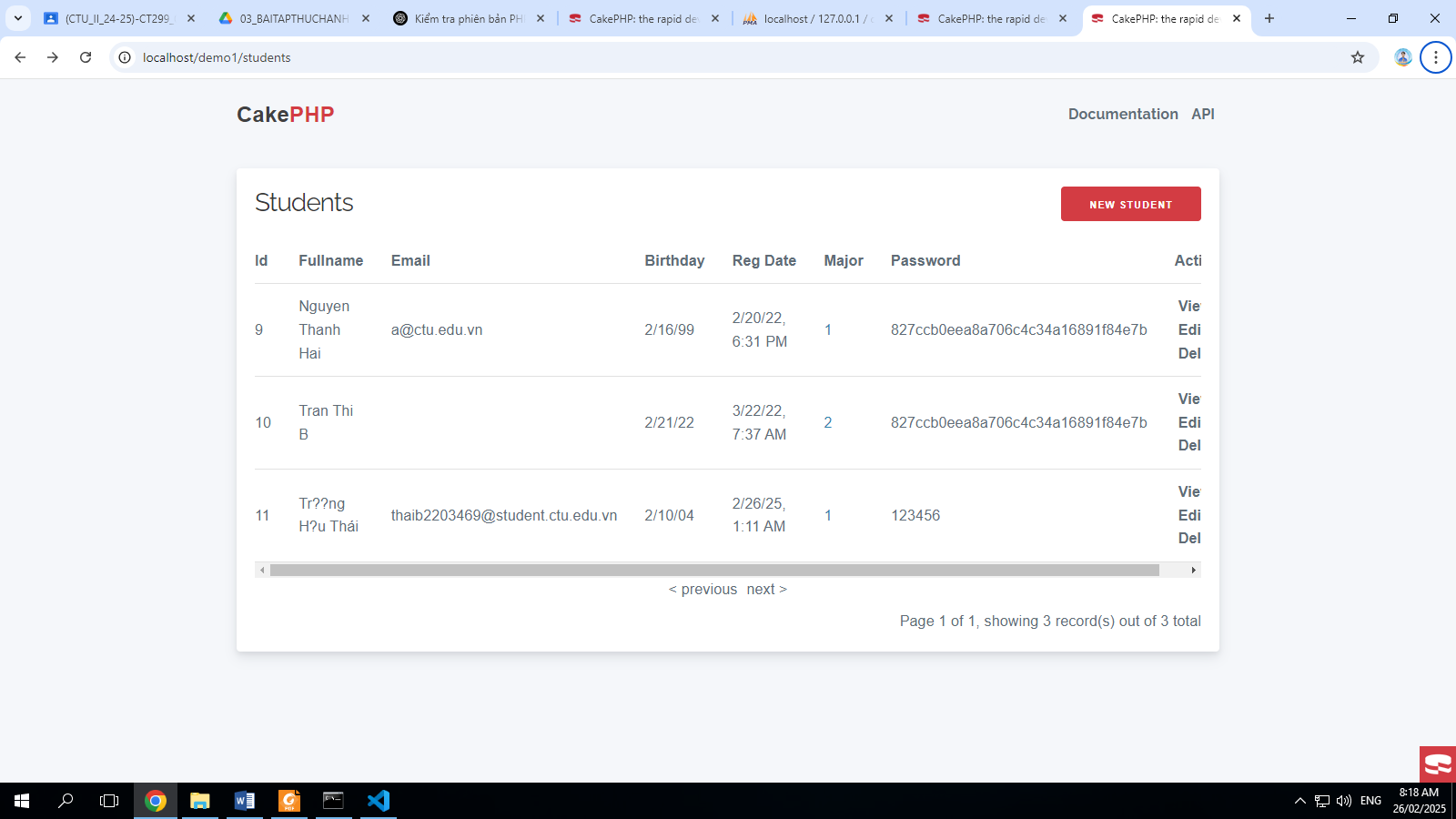
Giờ chúng ta chỉnh sửa lại để combobox chọn major sẽ hiển thị tên chuyên ngành thay vì mã chuyên ngành1



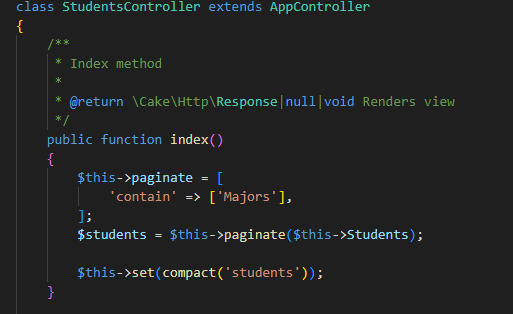
Quan sát kết quả:



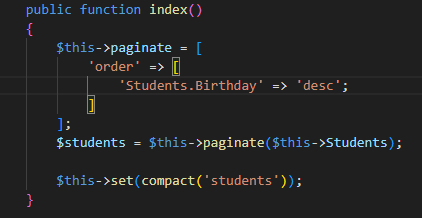
Để sắp xếp dữ liệu hiển thị trong trang index, ví dụ ds ban đầu  
trong Students



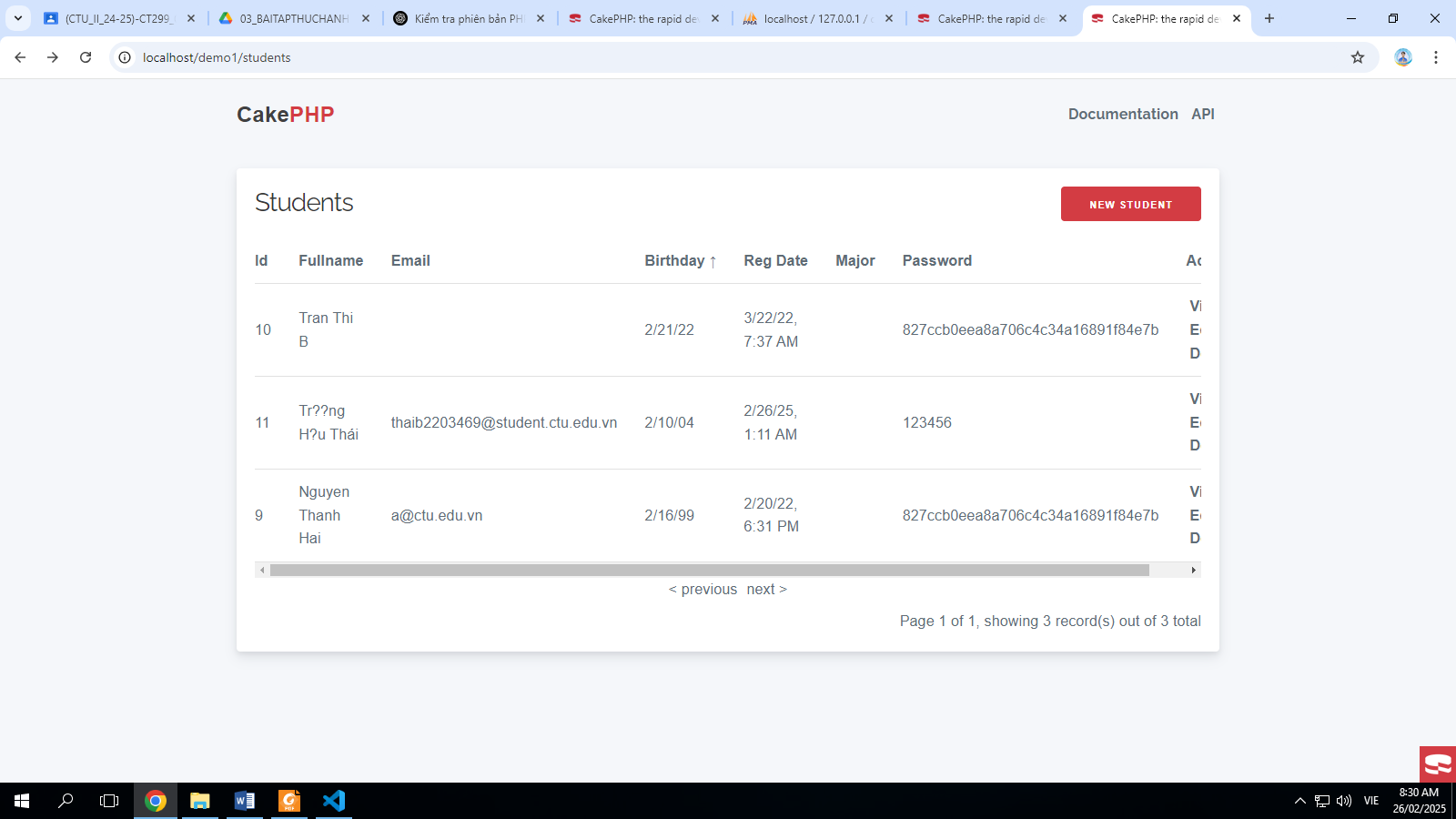
Ta muốn sắp xếp dữ liệu với ngày sinh lớn hơn sẽ hiển thị trước, chúng ta sẽ vào Controller của Students



Chỉnh sửa nội dung phương thức index() tùy chỉnh sắp xếp giảm dần theo ngày sinh

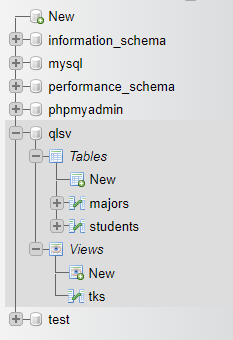


Quan sát kết quả:

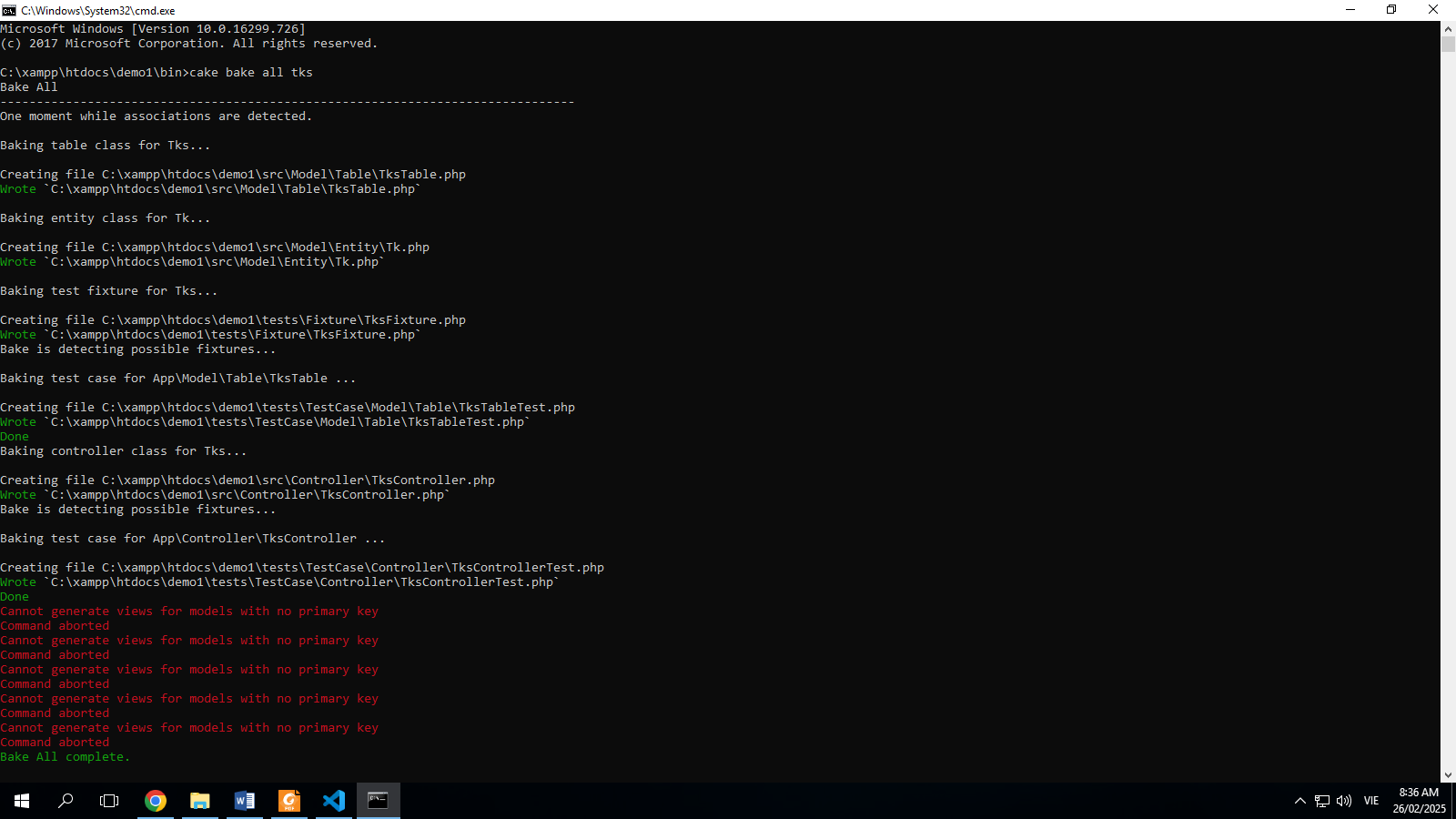


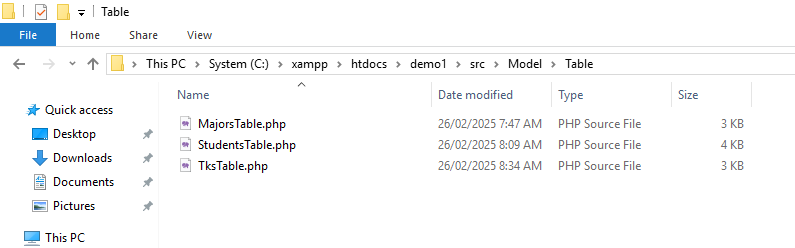
Hướng dẫn tạo CRUD từ view CSDL với CakePHP:

Giả sử chúng ta muốn hiển thị danh sách các chuyên ngành cùng với số thành viên trong  
chuyên ngành đó, chúng ta có thể dùng view CSDL để tạo query trả về kết quả mong muốn và  
rồi dùng Cakephp để tạo ra giao diện web hiển thị kết quả từ query này.  
Trước tiên chúng ta viết lệnh tạo view (vd tên là tks) để thống kê số lượng sinh viên của mỗi  
chuyên ngành

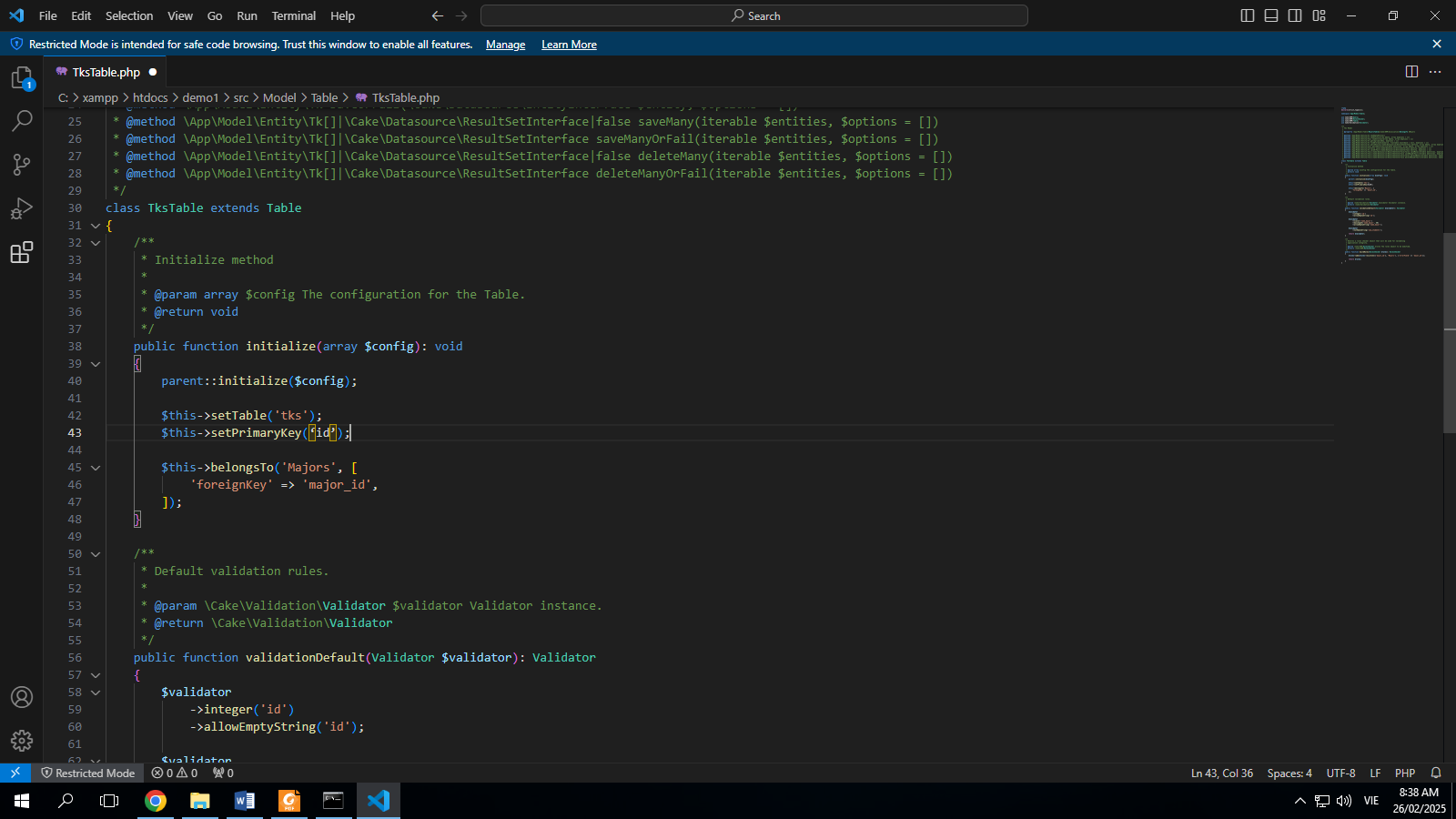


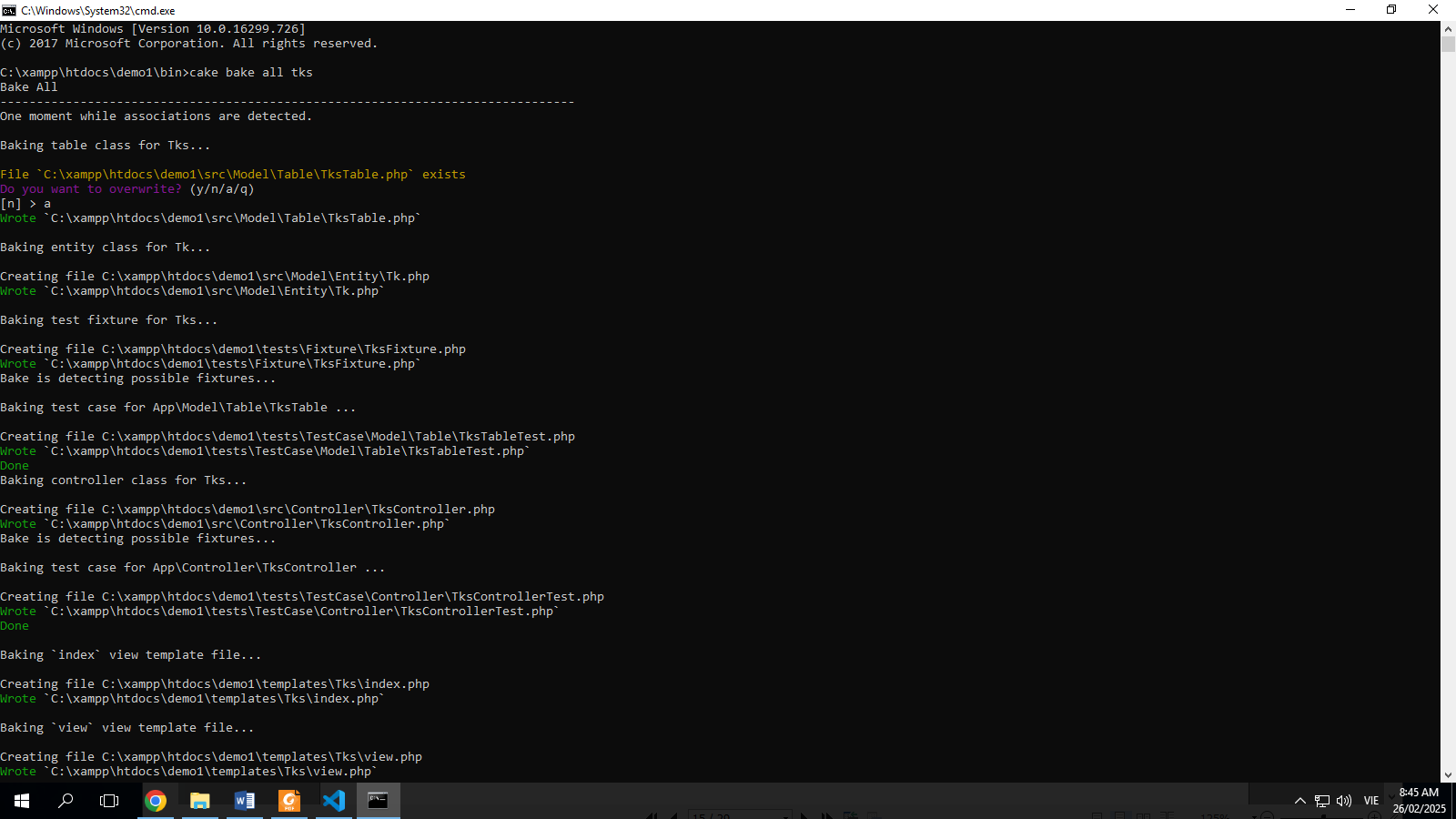
Giờ ta tiến hành tạo CRUD cho tks



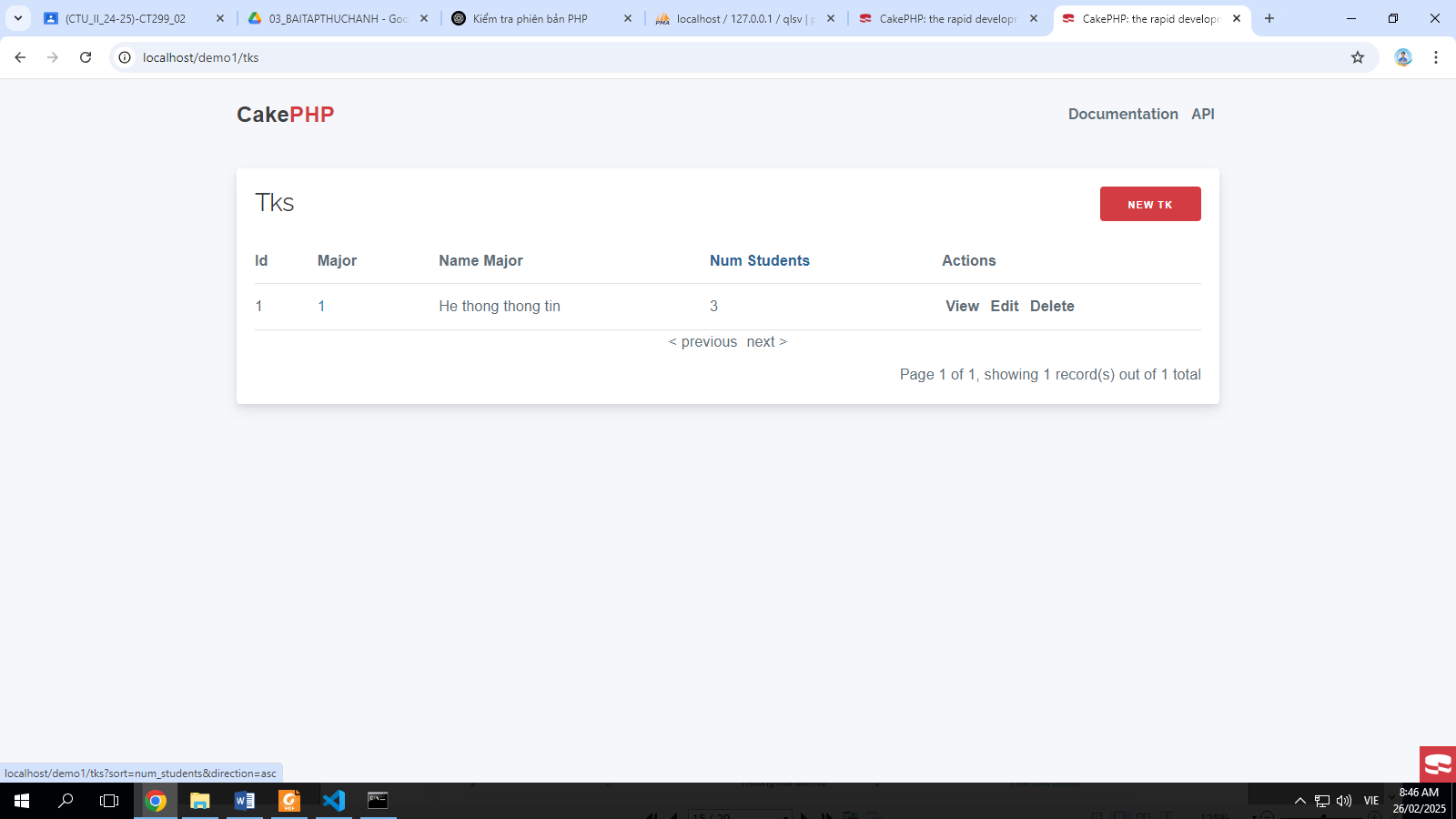


Trong phương thức *initialize*  
Sau lệnh $this->setTable(“tks”);  
Các bạn thêm lệnh sau:  
*$this->setPrimaryKey(‘id’);*  
Lệnh này để Cakephp nhận biết id là khóa trong tks để có thể tiến hành tự động CRUD khi ta  
thực hiện lệnh cake bake





Bây giờ chạy localhost/demo1/tks, ta sẽ được giao diện lấy dữ liệu từ view CSDL đã tạo:

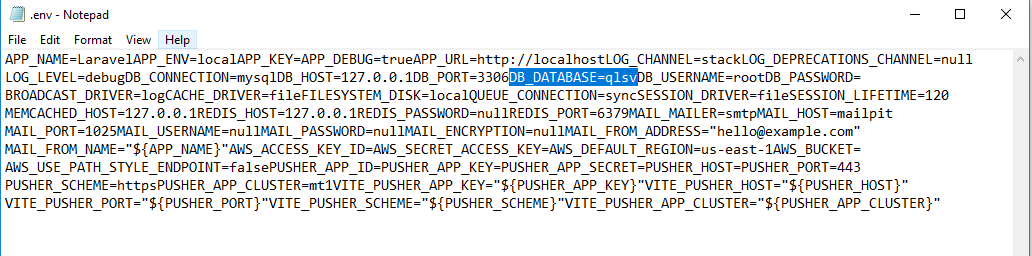


Tạo project Laravel và kết nối CSDL

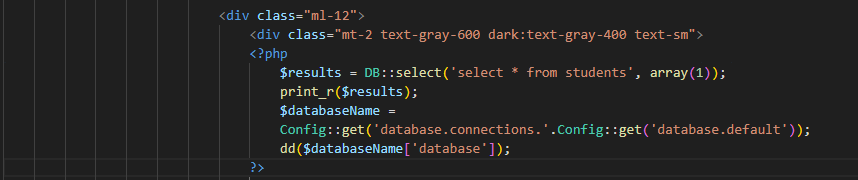
Trước tiên bạn cần tạo một project mới với lệnh composer, vd: chúng ta tạo 1 project buoi5  
composer create-project laravel/laravel buoi5

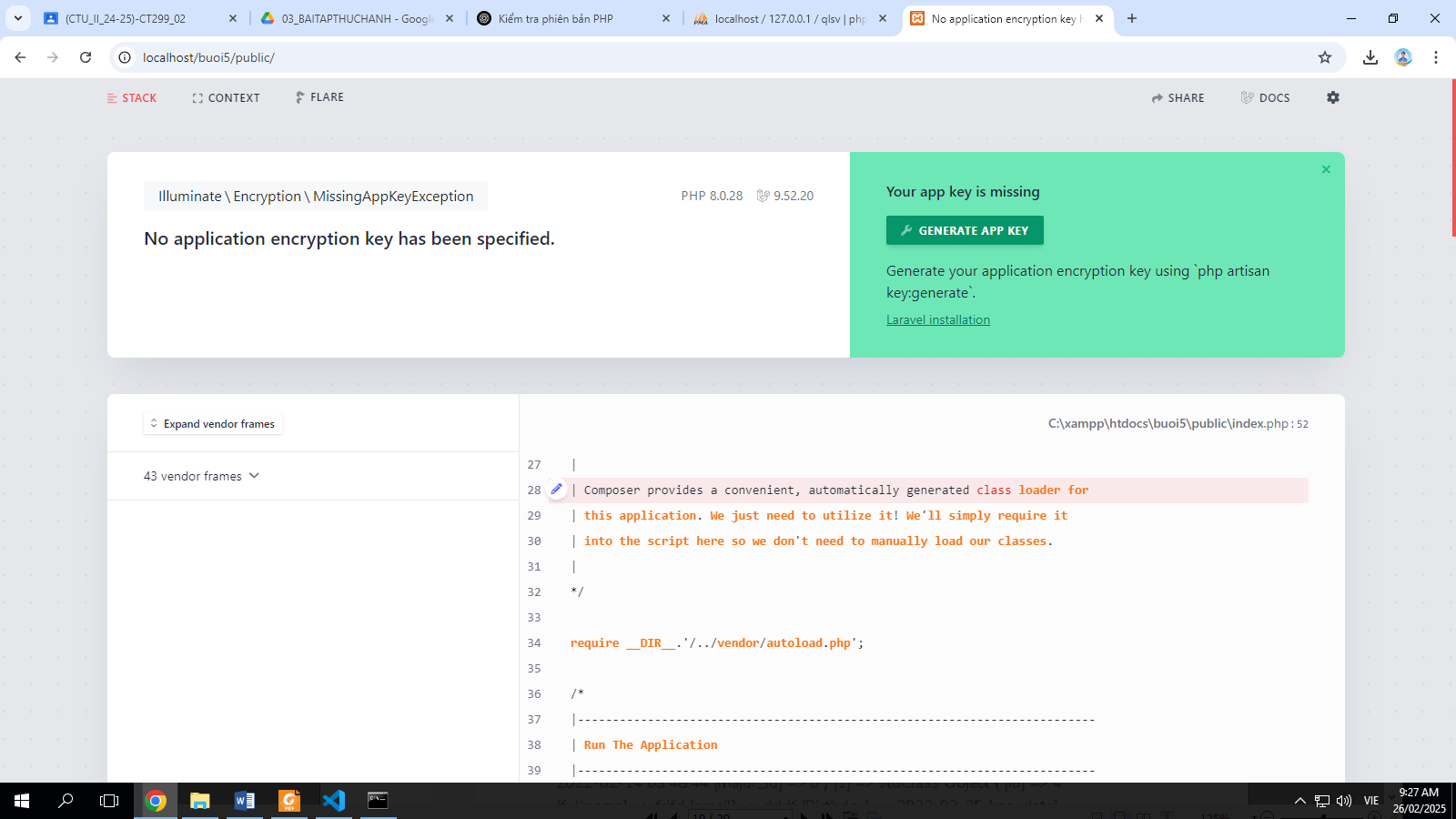




Chú ý là chúng ta sẽ cần đặt tên csdl trong .env

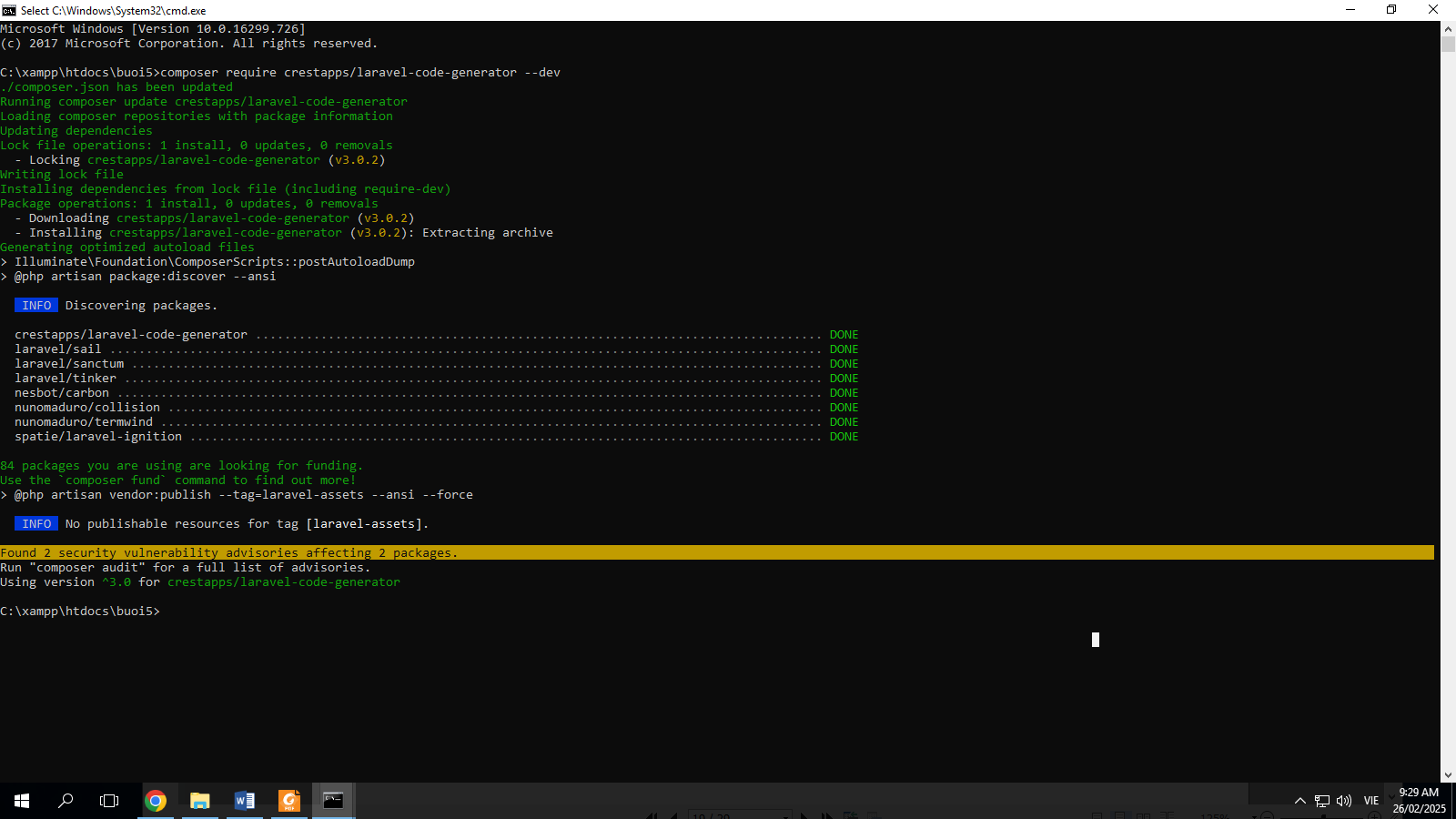
Chèn đoạn code sau để kiểm tra việc kết nối CSDL trong C:\xampp\htdocs\buoi5\resources\views\welcome.blade.php



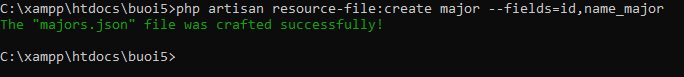


Cài đặt CrestApps/laravel-code-generator

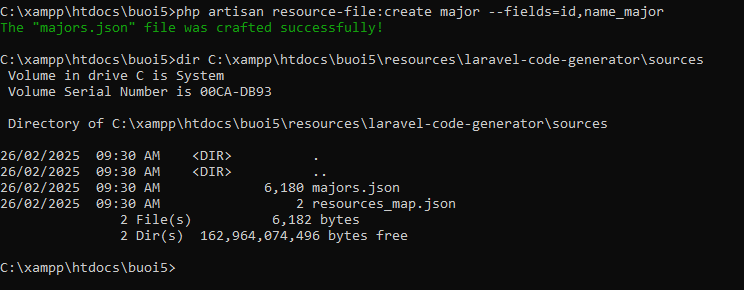
chúng ta thực hiện lệnh sau  
composer require crestapps/laravel-code-generator --dev  
Để đưa gói này vào project vừa tạo



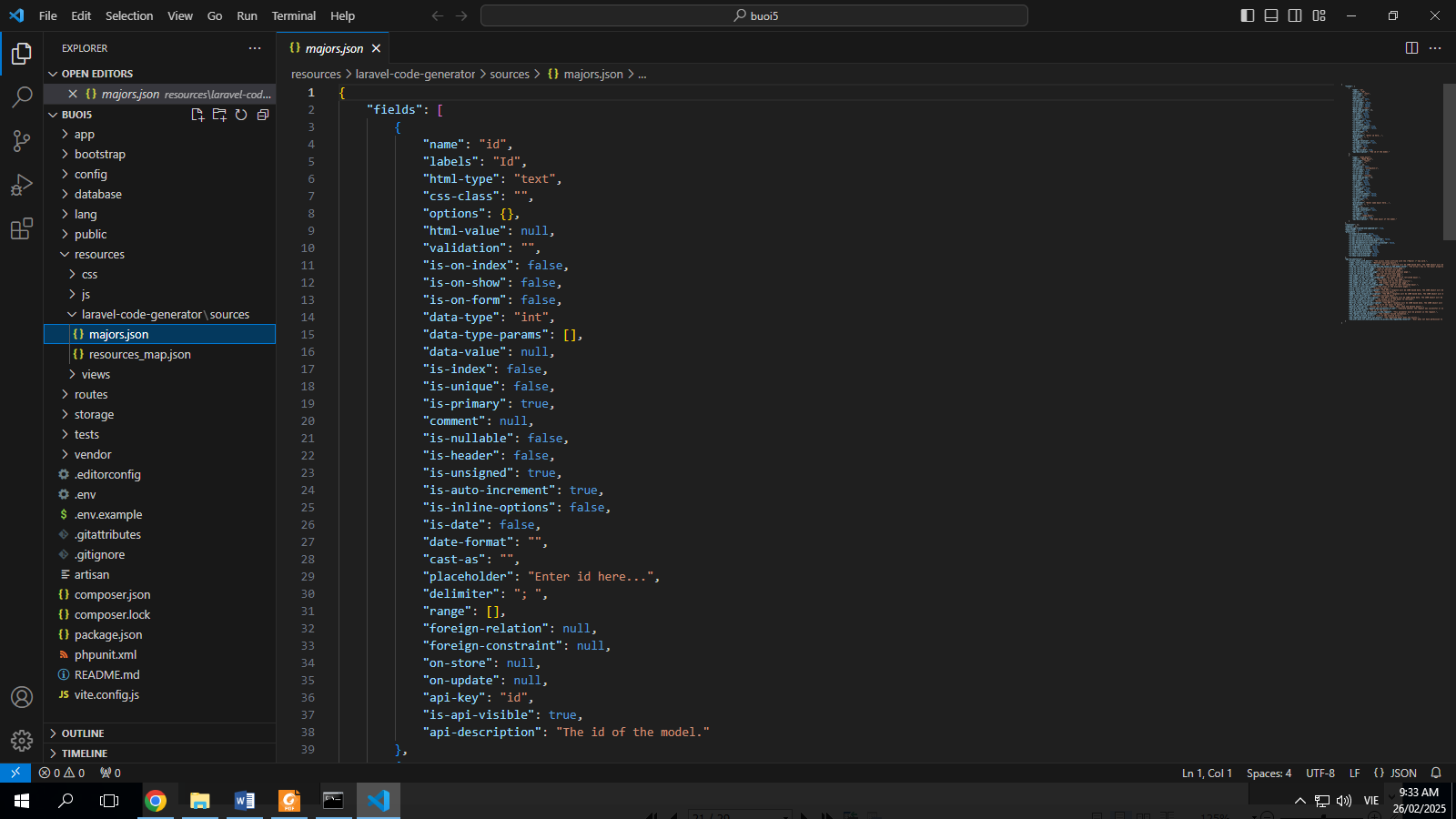
Tạo các script mô tả để sinh ra code:



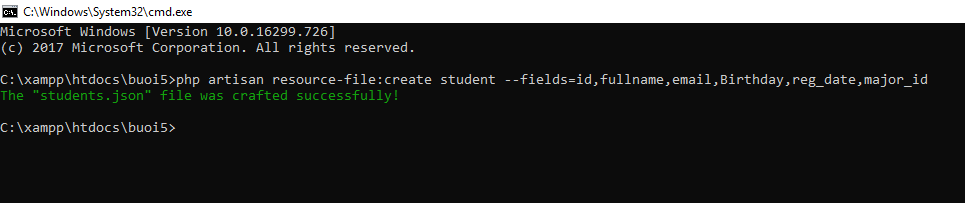
Ta sẽ thấy tập tin majors.json được tạo ra ở  
dir C:\xampp\htdocs\buoi5\resources\laravel-code-generator\

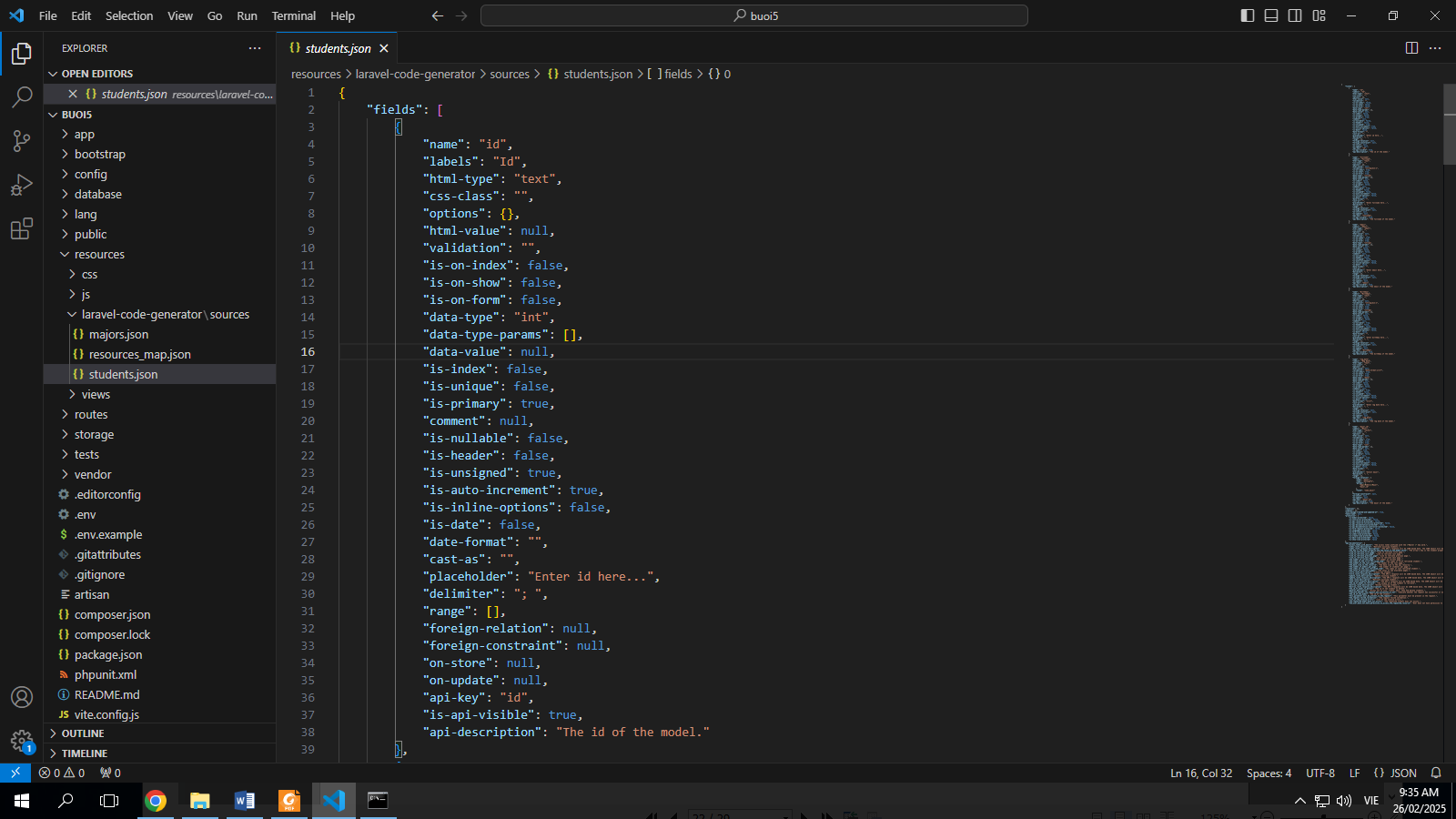


Nội dung trong tập tin là



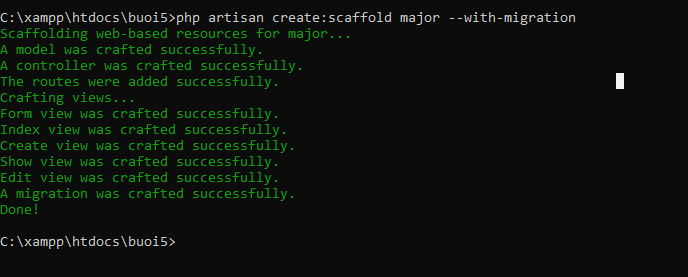
php artisan resource-file:create student --fields=id,**fullname**,**email**,**Birthday,reg\_date,major\_id**

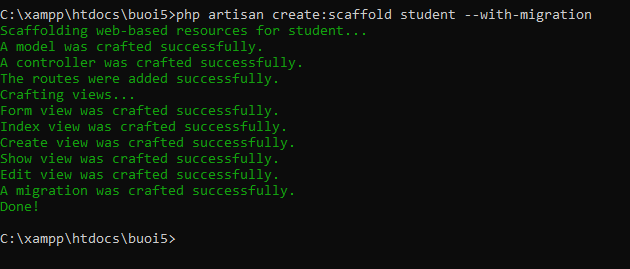




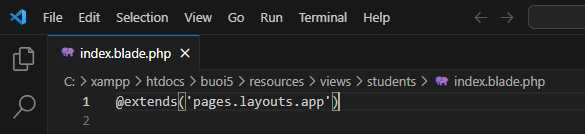
Thực hiện tạo CRUD cho bảng major

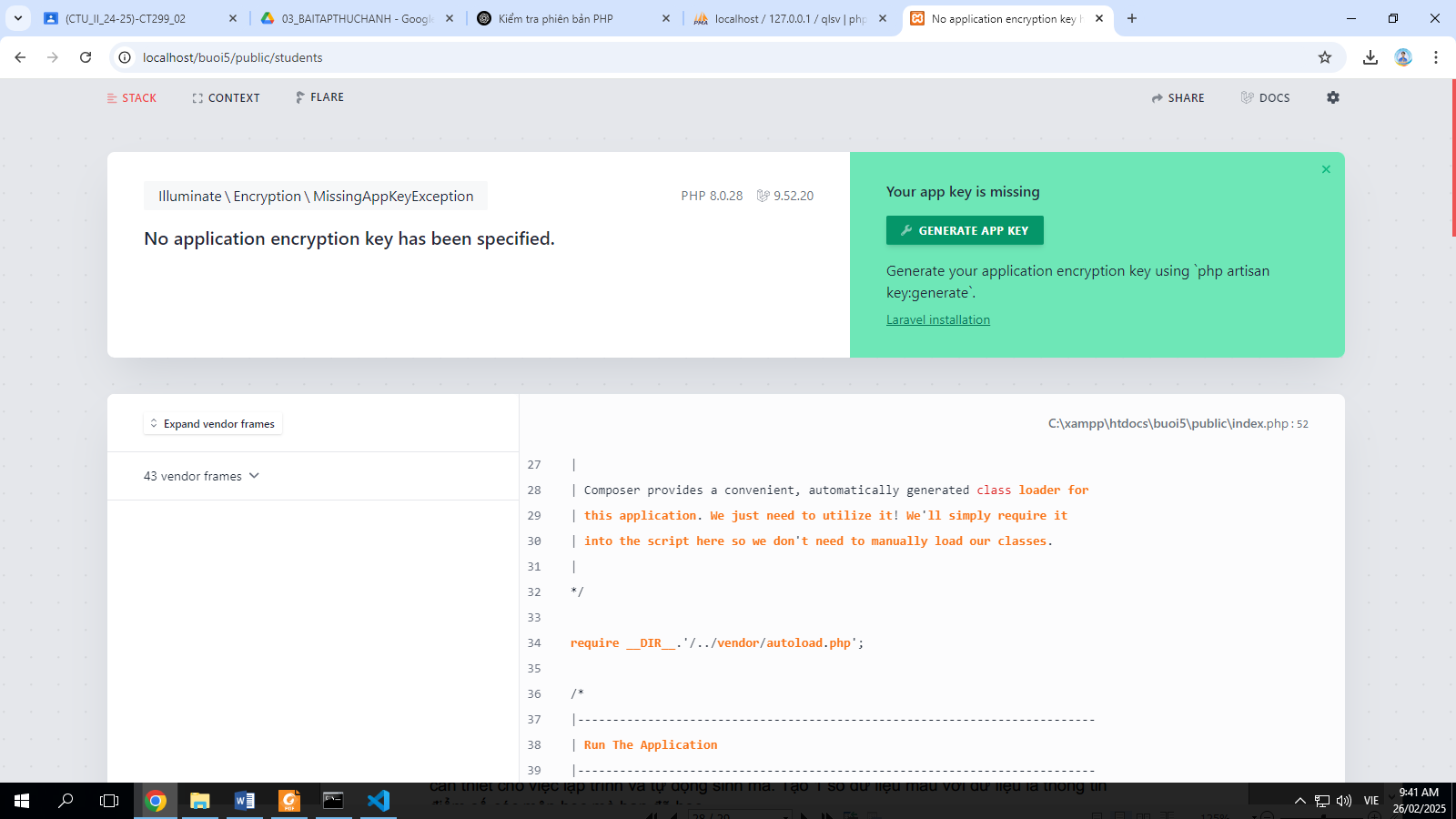
php artisan create:scaffold major --with-migration

Đối với bảng Student



Sửa dòng  
@extends('layouts.app') như hình:





1. Đọc và tìm hiểu về Cakephp tại đây: [[1]](#footnote-0), bạn hãy cho biết Cakephp là gì? Hãy mô tả các quy tắc trong các Layer Model, View, Controller của Cakephp. Tìm hiểu thêm trên Internet, các bạn trình bày Cakephp ra đời năm nào? Tác giả, lịch sử phát triển các phiên bản.

CakePHP là một framework mã nguồn mở được viết bằng ngôn ngữ PHP, hỗ trợ phát triển ứng dụng web một cách nhanh chóng và hiệu quả. Nó tuân theo kiến trúc Model-View-Controller (MVC), giúp tách biệt logic ứng dụng, giao diện người dùng và quản lý dữ liệu. CakePHP được phát hành dưới giấy phép MIT, cho phép sử dụng và phân phối tự do.

Các quy tắc trong các Layer của CakePHP:

1. Model (Mô hình):
   * Chức năng: Đại diện cho dữ liệu và logic nghiệp vụ của ứng dụng. Model quản lý tương tác với cơ sở dữ liệu, xác thực dữ liệu và các quy tắc nghiệp vụ.
   * Quy tắc:
     + Mỗi bảng trong cơ sở dữ liệu tương ứng với một Model.
     + Sử dụng tính năng ORM (Object-Relational Mapping) để tương tác với cơ sở dữ liệu thông qua các đối tượng.
     + Hỗ trợ các quan hệ giữa các bảng như hasOne, hasMany, belongsTo, và belongsToMany.
2. View (Giao diện):
   * Chức năng: Chịu trách nhiệm hiển thị dữ liệu cho người dùng. View nhận dữ liệu từ Controller và render ra giao diện HTML.
   * Quy tắc:
     + Mỗi action trong Controller có một View tương ứng.
     + Sử dụng các helper để tạo và quản lý các thành phần giao diện như form, HTML, JavaScript, và AJAX.
     + Hỗ trợ layout để tạo cấu trúc chung cho toàn bộ ứng dụng.
3. Controller (Bộ điều khiển):
   * Chức năng: Xử lý các yêu cầu từ người dùng, tương tác với Model để lấy dữ liệu và chọn View phù hợp để hiển thị.
   * Quy tắc:
     + Mỗi Controller quản lý một nhóm các hành động (actions) liên quan đến một phần cụ thể của ứng dụng.
     + Sử dụng các component để chia sẻ logic chung giữa các Controller.
     + Xử lý điều hướng và chuyển hướng giữa các actions và controllers.

Lịch sử phát triển của CakePHP:

* Năm ra đời: CakePHP bắt đầu phát triển vào tháng 4 năm 2005 bởi lập trình viên người Ba Lan Michal Tatarynowicz. Ông đã viết một phiên bản tối thiểu của framework phát triển ứng dụng nhanh bằng PHP, đặt tên là Cake, và phát hành nó dưới giấy phép Công cộng. Sau đó, giấy phép này được chuyển sang giấy phép MIT.
* Lịch sử các phiên bản:
  + Tháng 5 năm 2006: Phiên bản 1.0 chính thức được phát hành.
  + Tháng 10 năm 2009: Quản lý dự án Garrett Woodworth và nhà phát triển Nate Abele rời dự án để tập trung vào framework Lithium, một dự án tách ra từ CakePHP.
  + Tháng 10 năm 2011: Phát hành phiên bản 2.0 với nhiều cải tiến và tính năng mới.
  + Tháng 3 năm 2012: Phát hành phiên bản 2.1.
  + Tháng 9 năm 2012: Phát hành phiên bản 2.2.2.
  + Tháng 1 năm 2013: Phát hành phiên bản 2.3.0.
  + Tháng 8 năm 2013: Phát hành phiên bản 2.4.0.
  + Tháng 5 năm 2014: Phát hành phiên bản 2.5.0.
  + Tháng 12 năm 2014: Phát hành phiên bản 2.6.0.

1. Dựa vào [[2]](#footnote-1) hãy cho biết các quy tắc đặt tên các bảng trong CSDL để Cakephp có thể nhận dạng được hỗ trợ cho việc sinh CRUD tự động.

Để CakePHP tự động nhận diện và hỗ trợ sinh CRUD (Create, Read, Update, Delete), bạn cần tuân theo các quy tắc đặt tên bảng trong cơ sở dữ liệu như sau:

1. Tên bảng ở dạng số nhiều và sử dụng gạch dưới (\_) để phân tách các từ:
   * Ví dụ: Nếu bạn có một model là Person, tên bảng tương ứng sẽ là people. Tương tự, model BigPerson sẽ có bảng là big\_people.
2. Khóa chính mặc định:
   * CakePHP mặc định sử dụng cột id làm khóa chính cho các bảng, với kiểu dữ liệu INT và tự động tăng (AUTO\_INCREMENT).
   * Nếu bạn muốn sử dụng một cột khác làm khóa chính, bạn có thể định nghĩa lại trong model bằng cách thiết lập thuộc tính $primaryKey.
3. Khóa ngoại trong quan hệ giữa các bảng:
   * Trong các quan hệ hasMany, belongsTo, hasOne, khóa ngoại được đặt tên bằng dạng số ít của tên bảng liên kết, kèm hậu tố \_id.
     + Ví dụ: Bảng news có thể có khóa ngoại news\_category\_id để liên kết với bảng news\_categories.
4. Bảng liên kết cho quan hệ nhiều-nhiều (hasAndBelongsToMany):
   * Tên bảng liên kết được đặt bằng cách kết hợp tên của hai bảng theo thứ tự bảng chữ cái, sử dụng gạch dưới (\_) để nối.
     + Ví dụ: Để liên kết giữa apples và zebras, tên bảng liên kết sẽ là apples\_zebras, không phải zebras\_apples.
5. Hãy tìm và trình bày các số liệu mới nhất về người sử dụng Cakephp và Laravel mà bạn tìm thấy được trên Internet.

Laravel:

* Theo thống kê, Laravel hiện đang được sử dụng trên 1.321.140 trang web, trong đó có 661.172 trang web đang hoạt động và 659.968 trang web đã sử dụng Laravel trong quá khứ. Tại Việt Nam, có 5.602 trang web sử dụng Laravel.
* Laravel được đánh giá cao nhờ cú pháp rõ ràng, dễ hiểu và hỗ trợ mạnh mẽ cho việc phát triển ứng dụng web theo mô hình MVC (Model-View-Controller).

CakePHP:

* Mặc dù không có số liệu cụ thể về số lượng trang web sử dụng CakePHP trong các nguồn hiện tại, nhưng CakePHP vẫn được biết đến như một framework PHP mạnh mẽ, hỗ trợ phát triển ứng dụng web nhanh chóng và hiệu quả.
* CakePHP cung cấp nhiều tính năng hữu ích như tích hợp ORM (Object-Relational Mapping), hệ thống xác thực mạnh mẽ và hỗ trợ tạo mã CRUD tự động, giúp lập trình viên tiết kiệm thời gian và công sức trong quá trình phát triển.

1. Hãy phân tích điểm mạnh và điểm yếu của Cakephp và Laravel, tiến hành so sánh 2 framework này.

Laravel:

*Điểm mạnh:*

* Cộng đồng hỗ trợ lớn: Laravel sở hữu một cộng đồng phát triển rộng lớn, cung cấp nhiều tài liệu, gói mở rộng và hỗ trợ cho lập trình viên.
* Khả năng mở rộng cao: Với hệ sinh thái phong phú, Laravel cho phép mở rộng và tích hợp dễ dàng các tính năng thông qua các package và module.
* Bảo mật tốt: Laravel tích hợp sẵn các biện pháp bảo mật như chống tấn công SQL injection, cross-site scripting và cross-site request forgery.
* Cú pháp rõ ràng, dễ hiểu: Laravel sử dụng cú pháp đơn giản, giúp lập trình viên dễ dàng tiếp cận và phát triển ứng dụng.

*Điểm yếu:*

* Phức tạp cho ứng dụng nhỏ: Với cấu trúc và tính năng phong phú, Laravel có thể trở nên quá phức tạp đối với các dự án nhỏ hoặc đơn giản.
* Hiệu suất: Do tích hợp nhiều tính năng, hiệu suất của Laravel có thể bị ảnh hưởng nếu không được tối ưu hóa cẩn thận.

CakePHP:

*Điểm mạnh:*

* Đơn giản và dễ sử dụng: CakePHP cung cấp cấu trúc rõ ràng và dễ tiếp cận, phù hợp cho cả người mới bắt đầu và lập trình viên có kinh nghiệm.
* Tích hợp sẵn nhiều tính năng: CakePHP đi kèm với nhiều tính năng tích hợp như xác thực, quản lý phiên, và bảo mật, giúp giảm thời gian phát triển.
* Thân thiện với SEO: Hỗ trợ tạo URL thân thiện với công cụ tìm kiếm, giúp cải thiện thứ hạng trang web.

*Điểm yếu:*

* Cộng đồng nhỏ hơn: So với Laravel, cộng đồng CakePHP nhỏ hơn, dẫn đến ít tài liệu và gói mở rộng hơn.
* Ít linh hoạt cho dự án lớn: Mặc dù mạnh mẽ, CakePHP có thể gặp hạn chế khi phát triển các ứng dụng quy mô lớn hoặc phức tạp.

So sánh tổng quan:

* Hiệu suất và quy mô dự án: Laravel thích hợp cho các dự án lớn với yêu cầu phức tạp, trong khi CakePHP phù hợp hơn cho các dự án nhỏ và vừa.
* Cộng đồng và tài liệu: Laravel có cộng đồng lớn và tài liệu phong phú hơn, cung cấp nhiều hỗ trợ cho lập trình viên.
* Độ phức tạp: Laravel có thể phức tạp hơn do tích hợp nhiều tính năng, trong khi CakePHP đơn giản và dễ tiếp cận hơn.

1. Bạn hãy tạo 1 CSDL đơn giản để lưu trữ dữ liệu với các mô tả bên dưới:

Môn học (mã môn, tên môn, số tín chỉ, có tính điểm tích lũy)

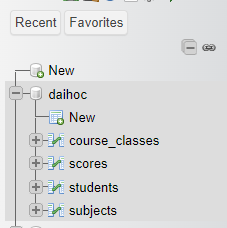
Sinh viên (mã sinh viên, họ tên sinh viên, ngày sinh, email, giới tính, số điện thoại, mật khẩu)

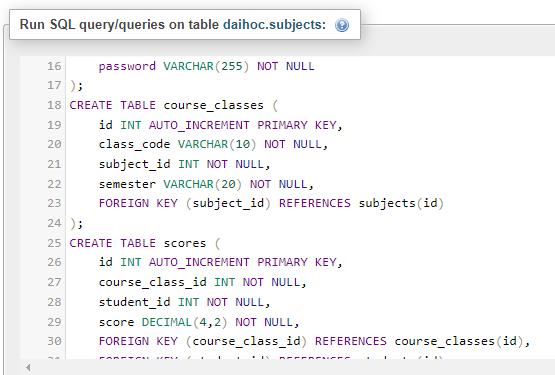
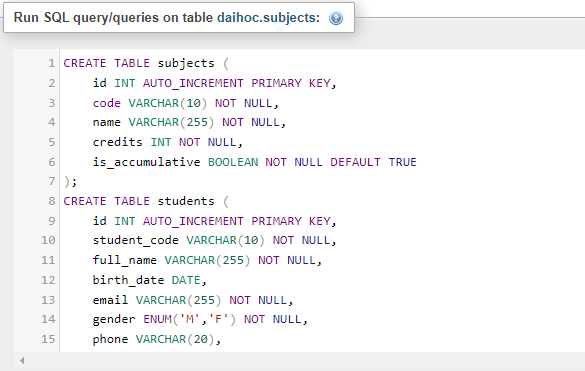
Lớp môn học (Mã lớp môn học, mã môn, học kỳ niên khóa)

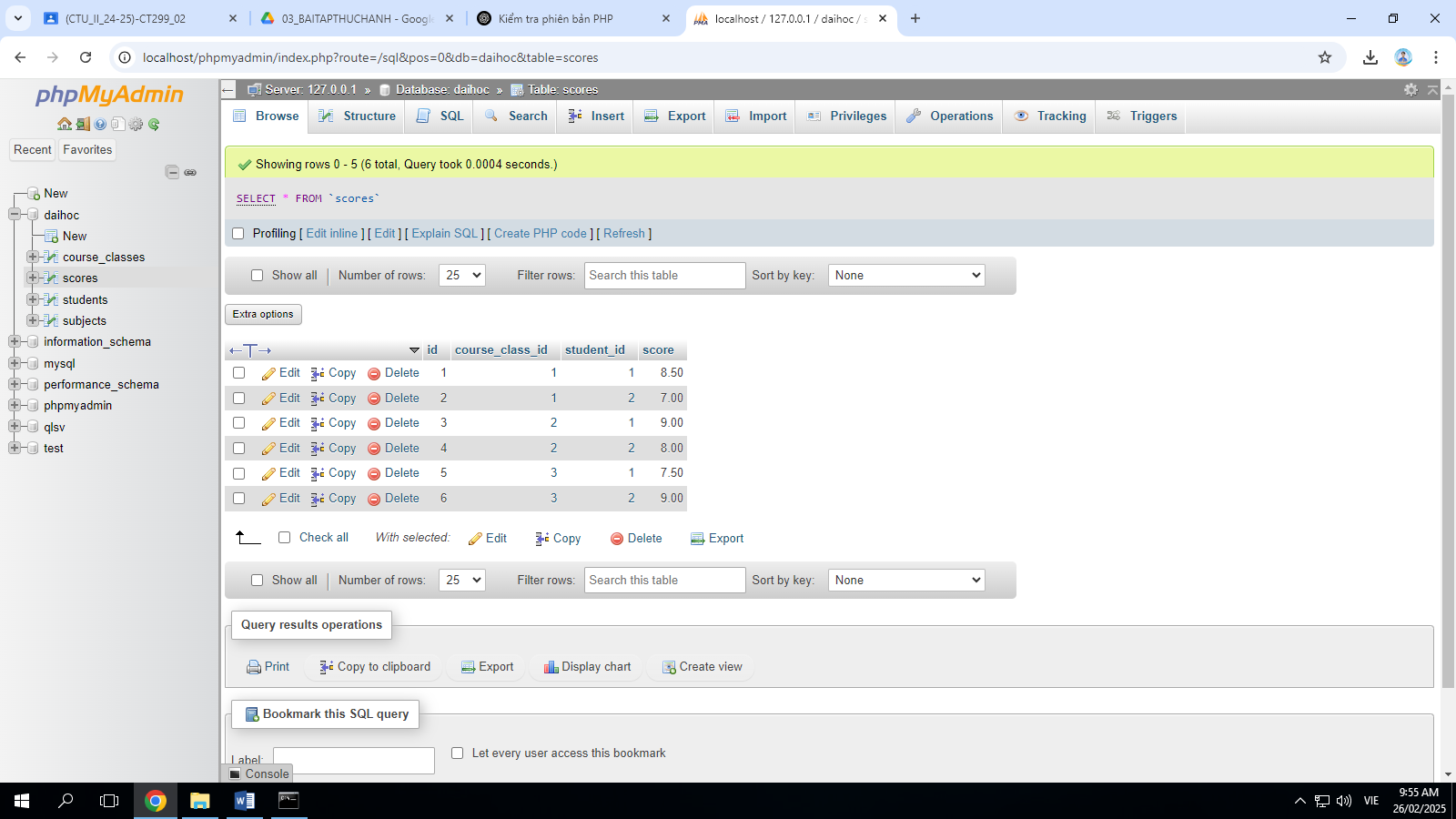
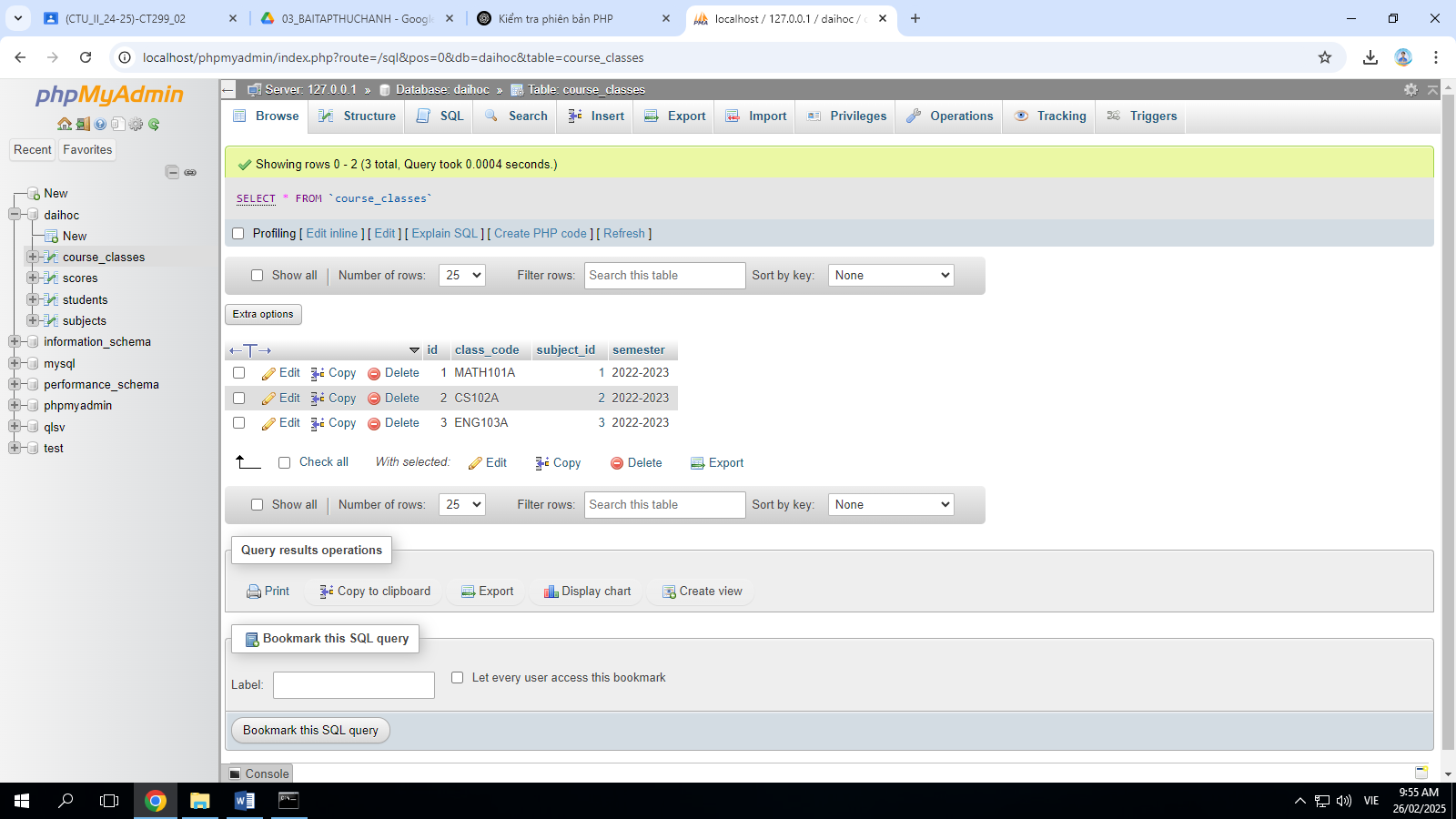
Điểm (mã lớp môn học, mã sinh viên, điểm)

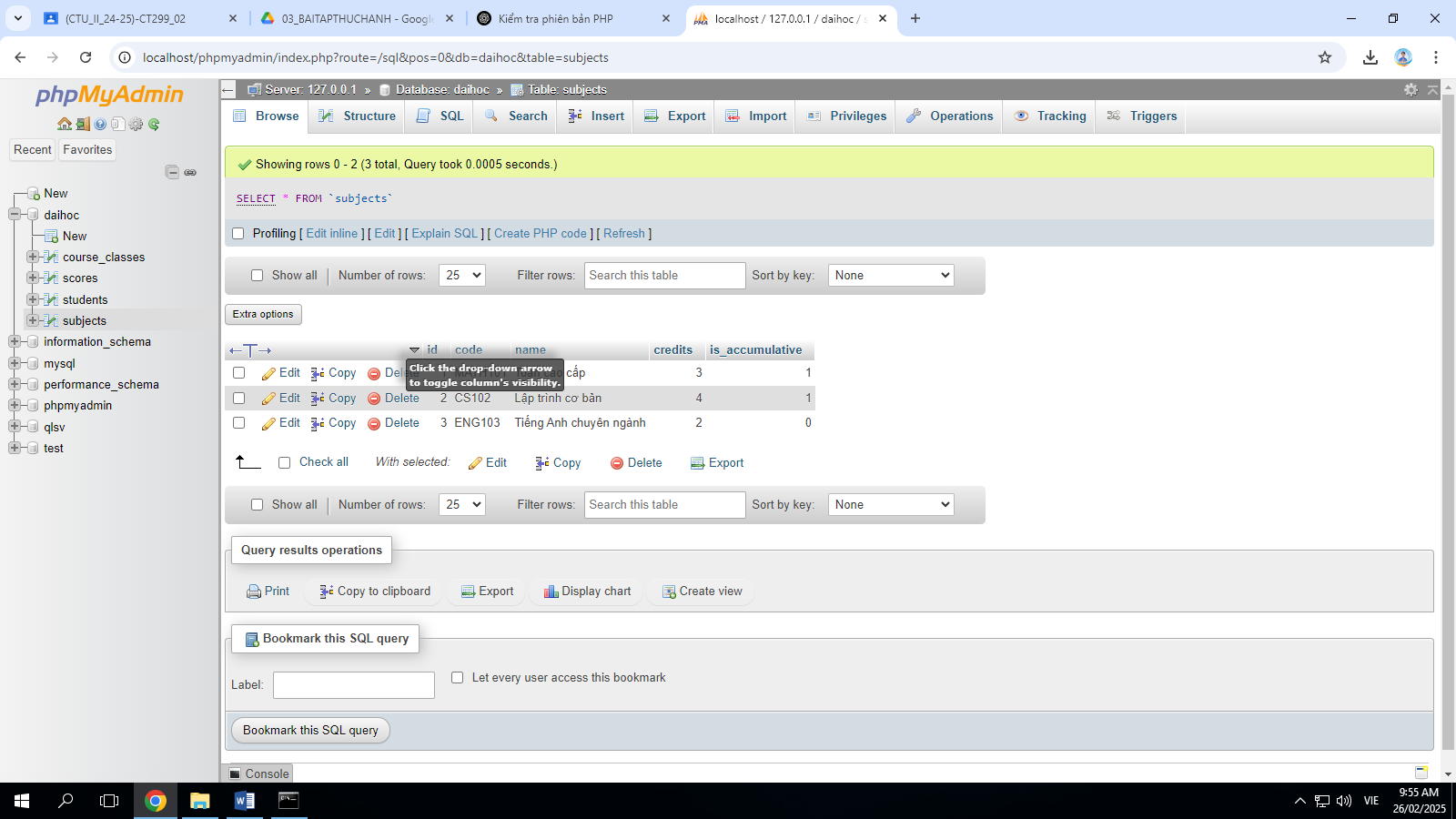
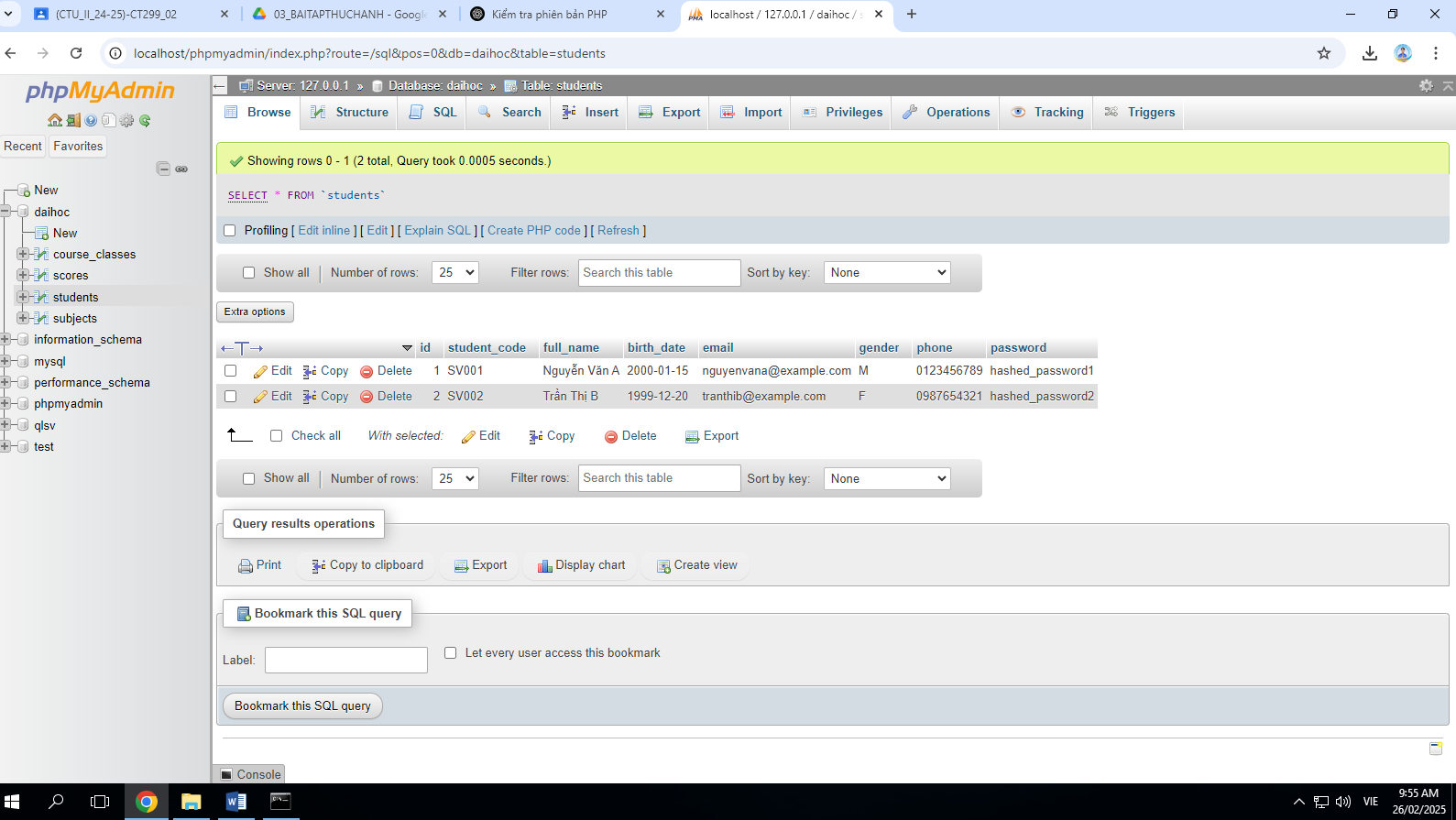
Tên bảng, tên cột, kiểu dữ liệu các bạn tự đặt sao cho thỏa các quy tắc để các framework nhận dạng được, vd dùng quy tắc với Cakephp:[[3]](#footnote-2), có thể thêm các cột khóa giả (ID) nếu thấy thực sự cần thiết cho việc lập trình và tự động sinh mã. Tạo 1 số dữ liệu mẫu với dữ liệu là thông tin điểm số các môn học mà bạn đã học.

Đưa các lệnh SQL tạo CSDL, tạo bảng,... vào bài làm

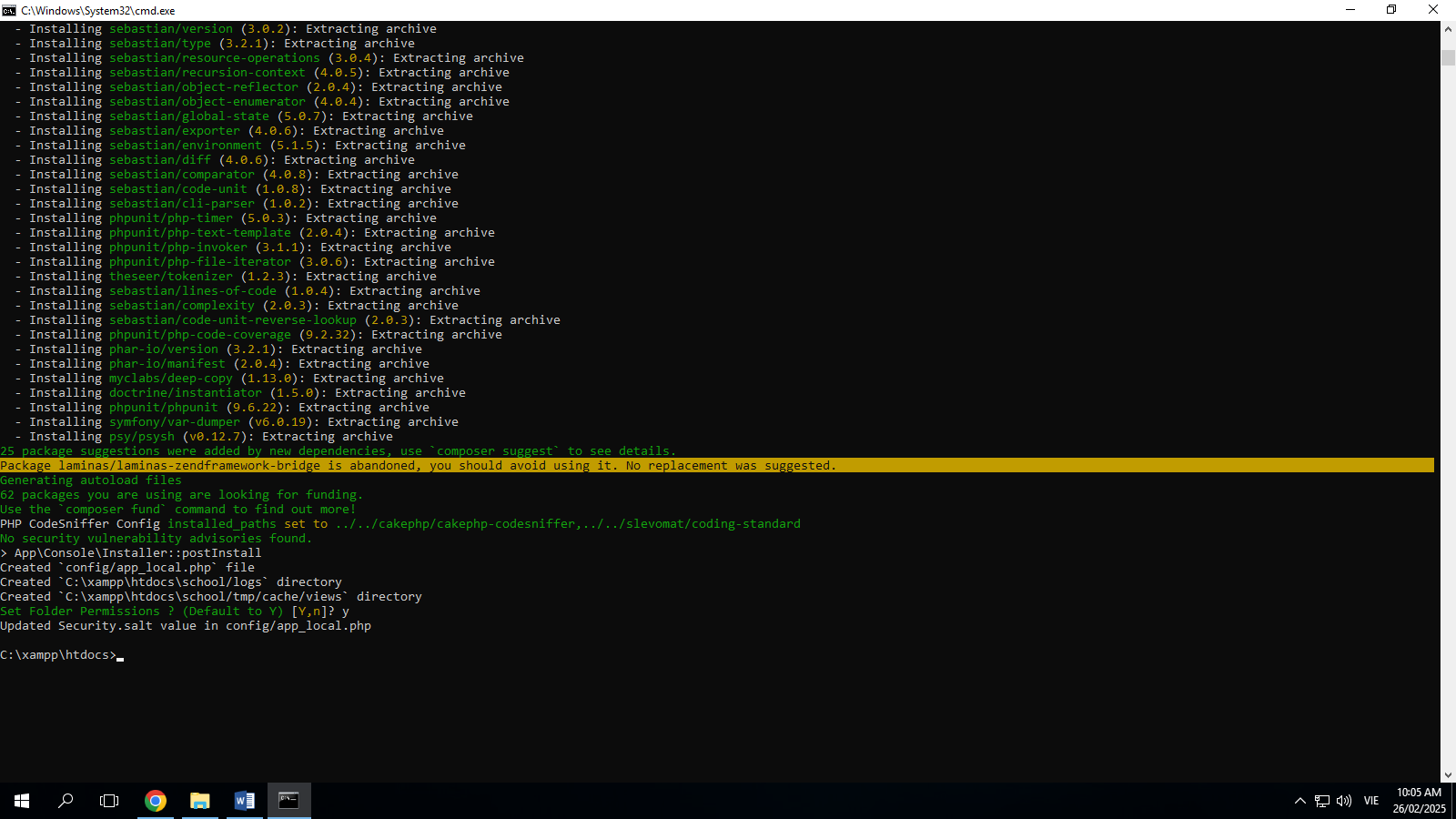
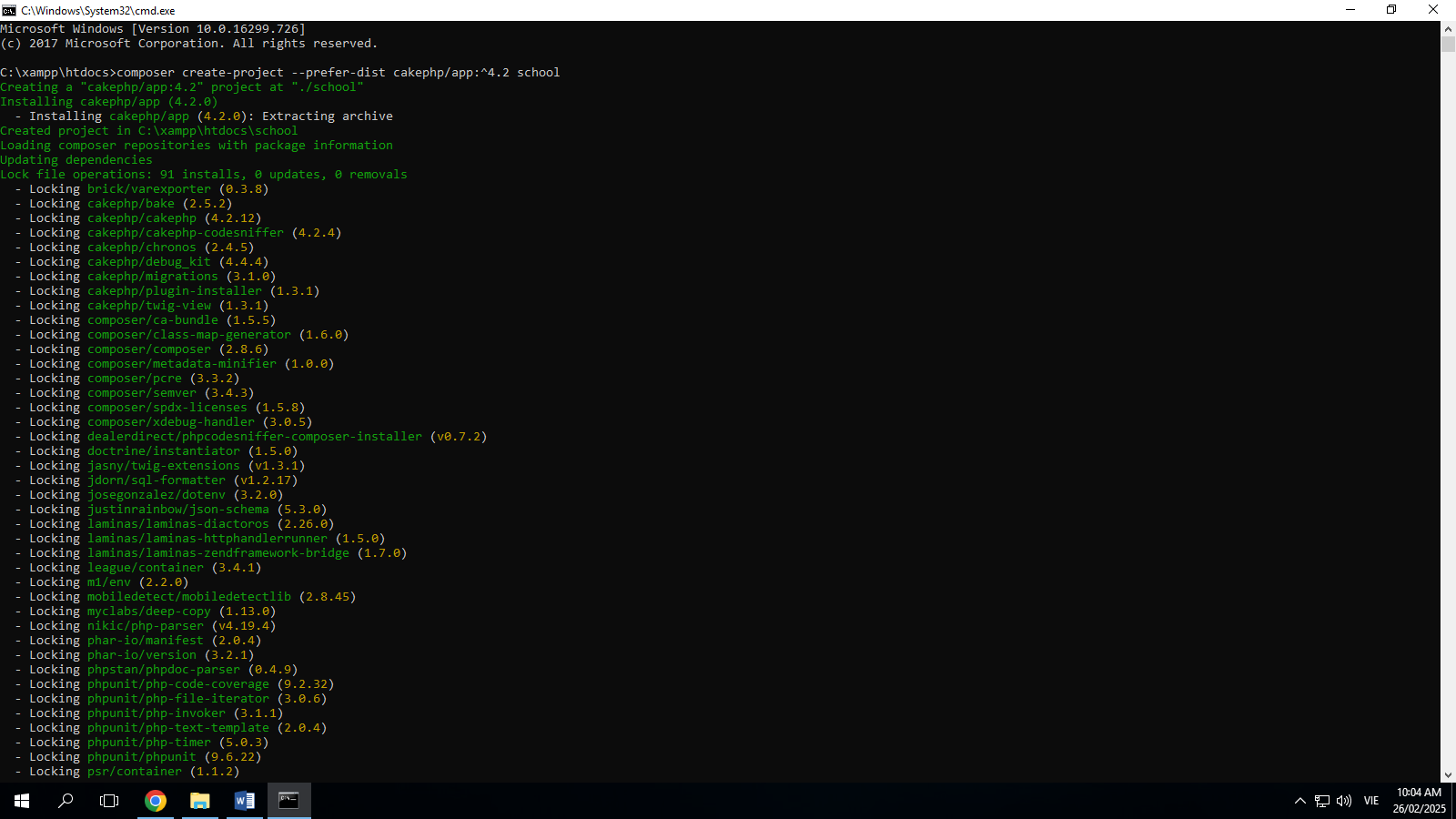


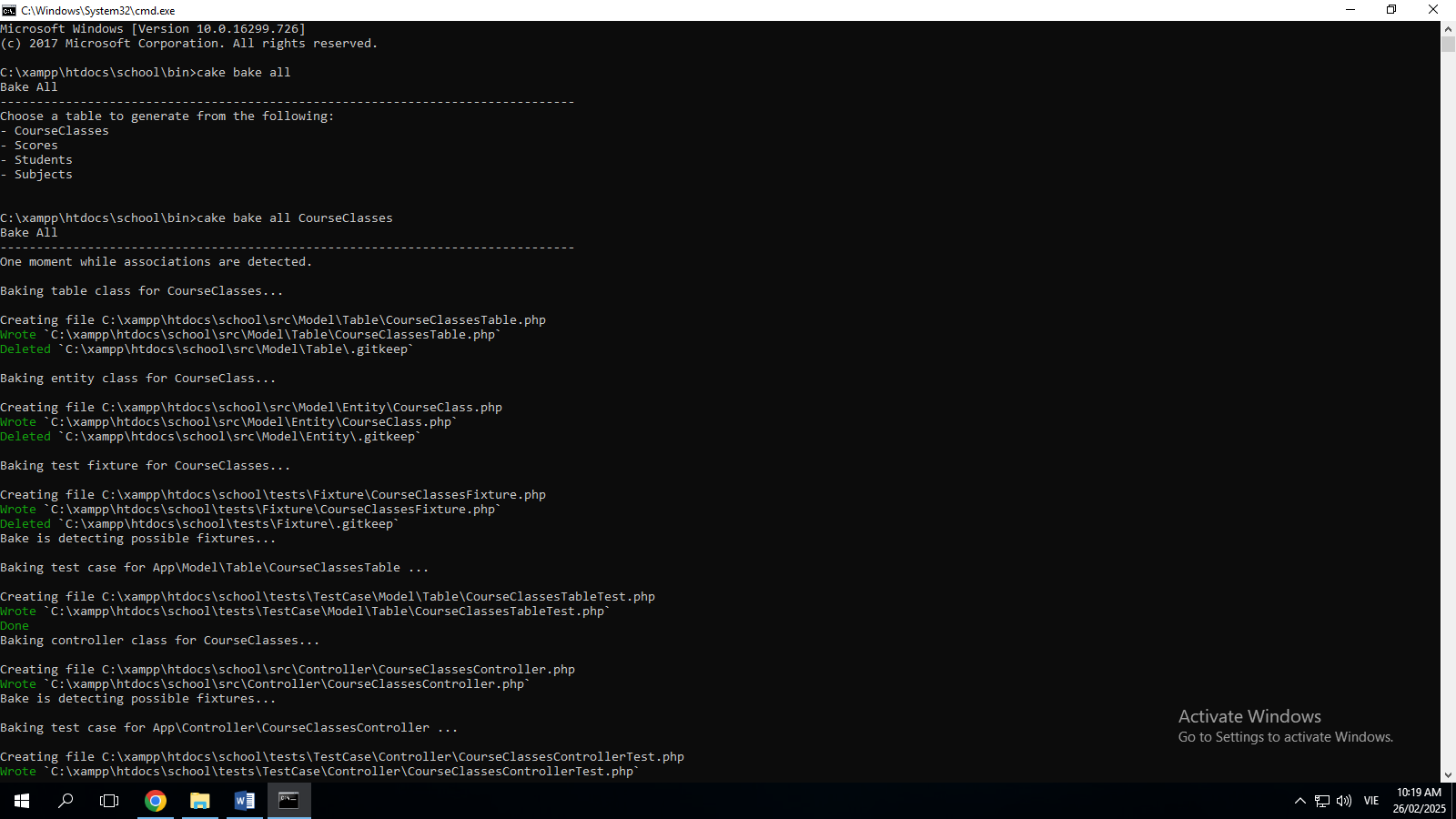
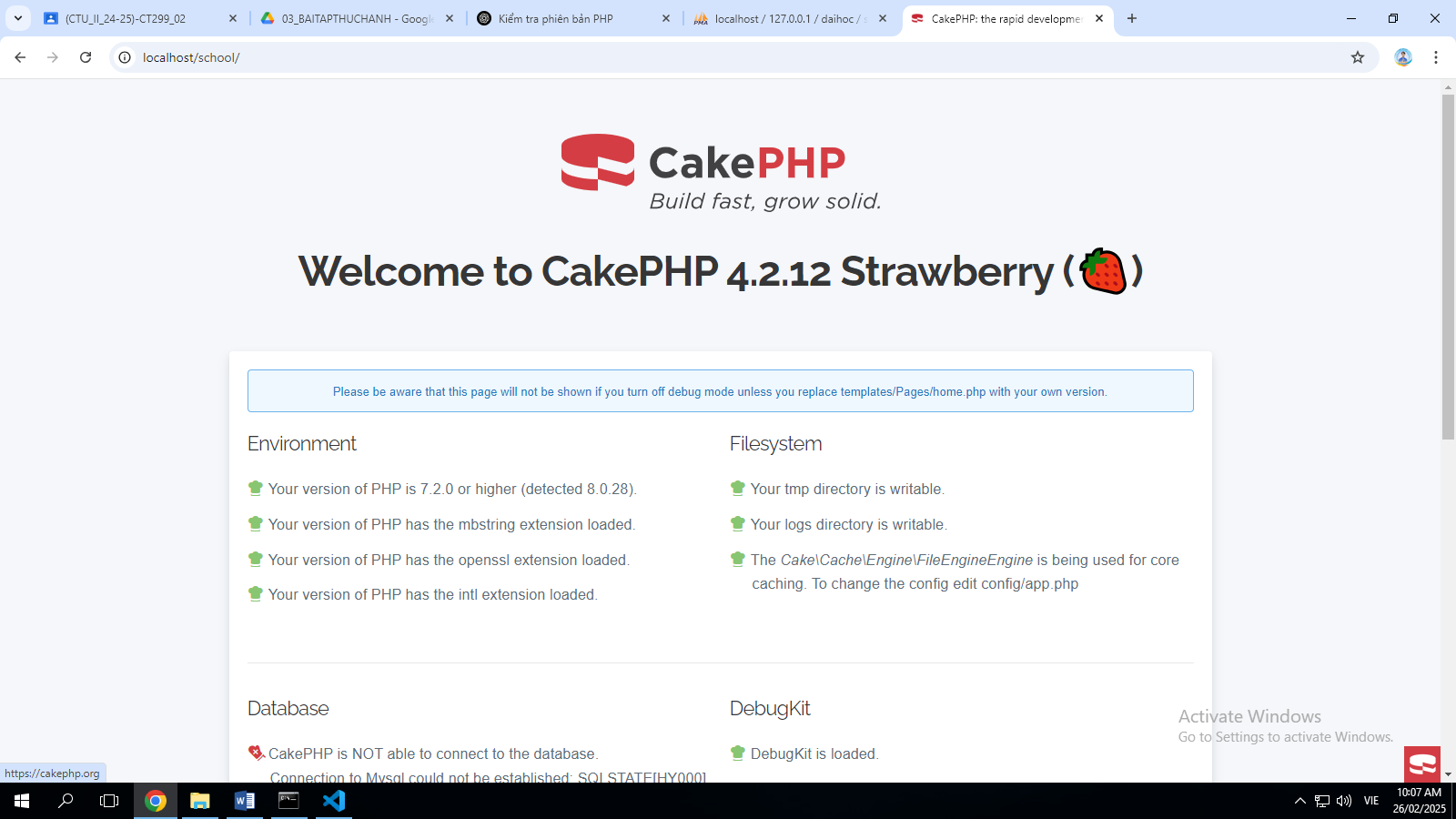
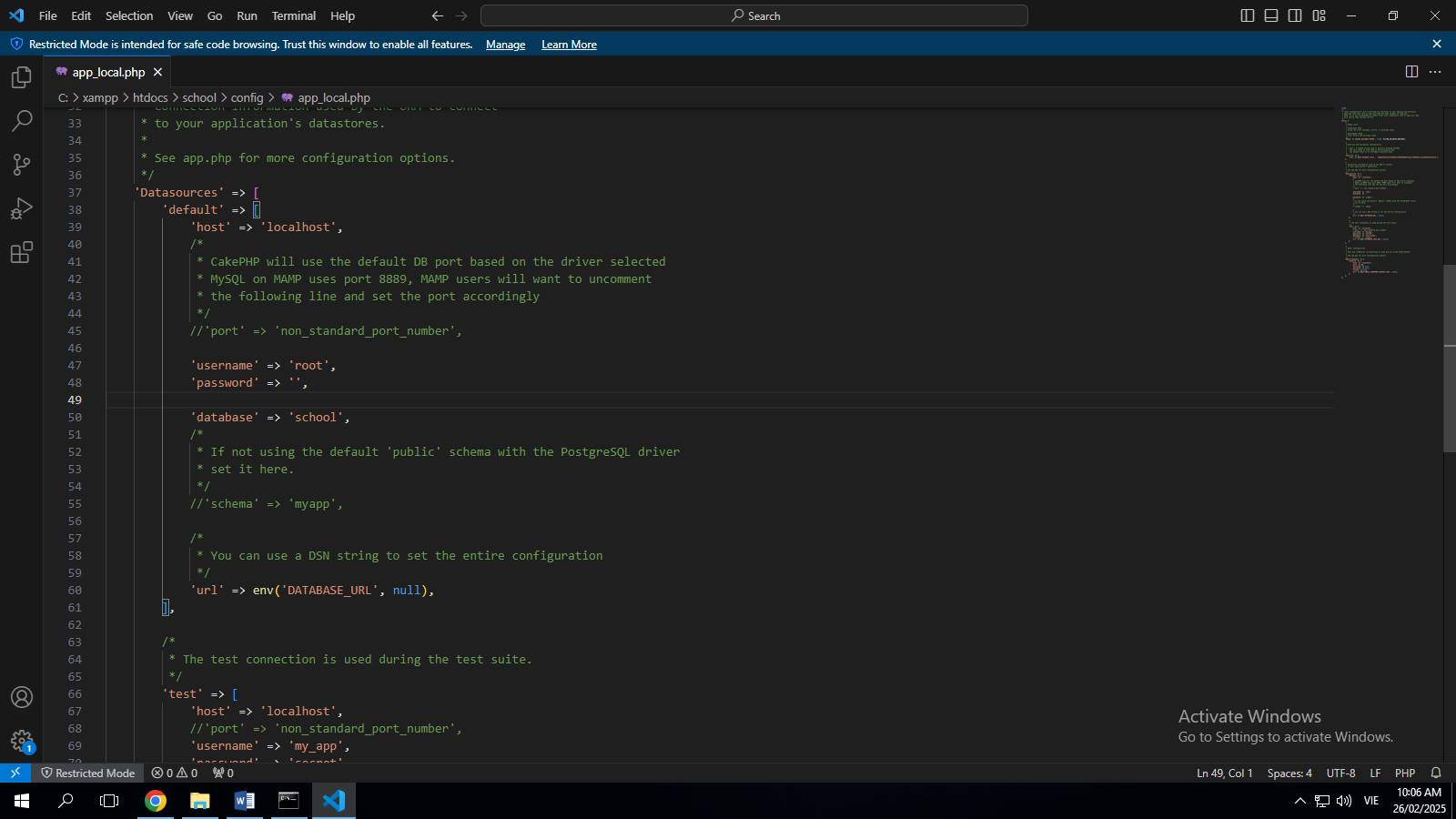


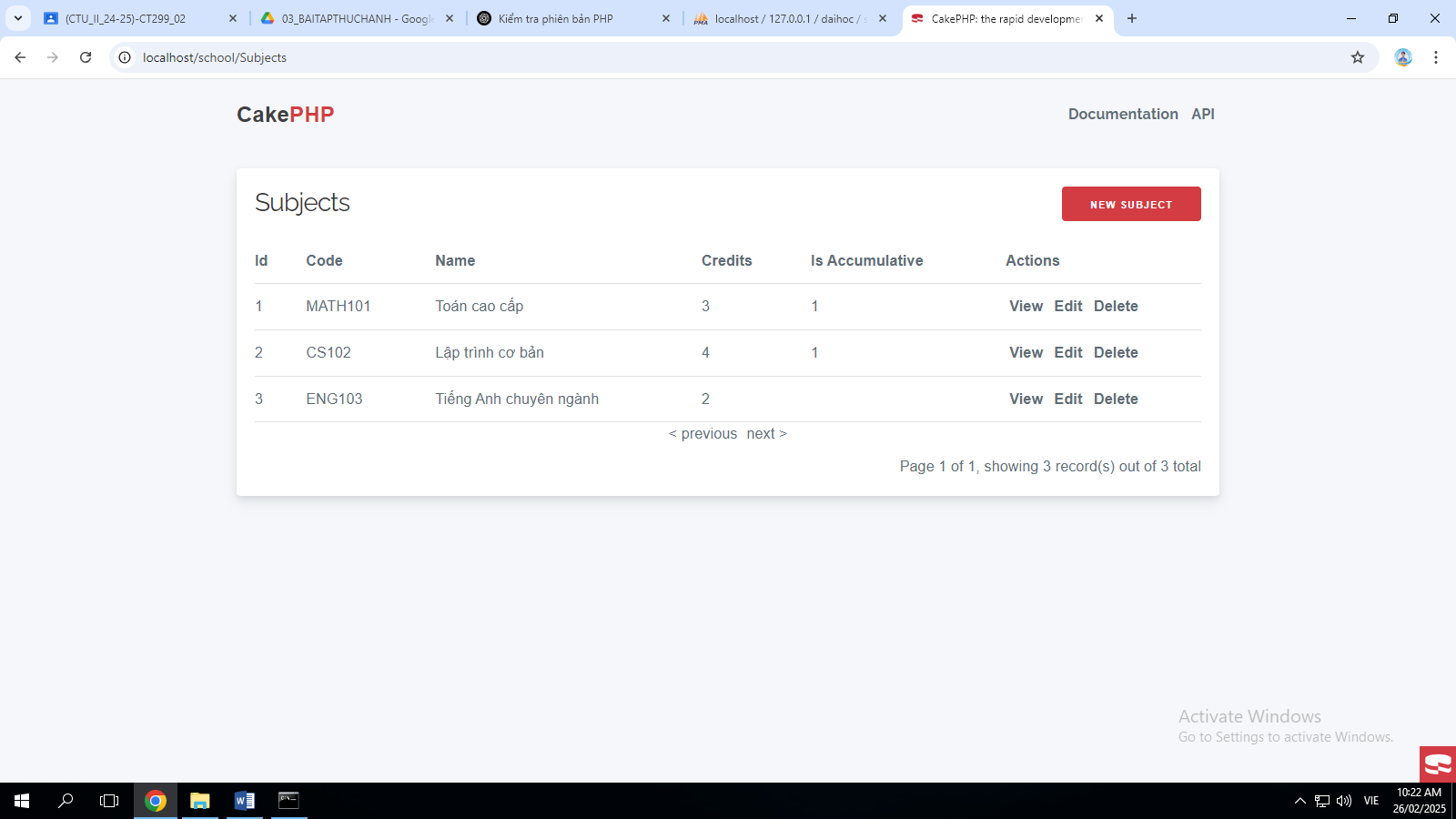
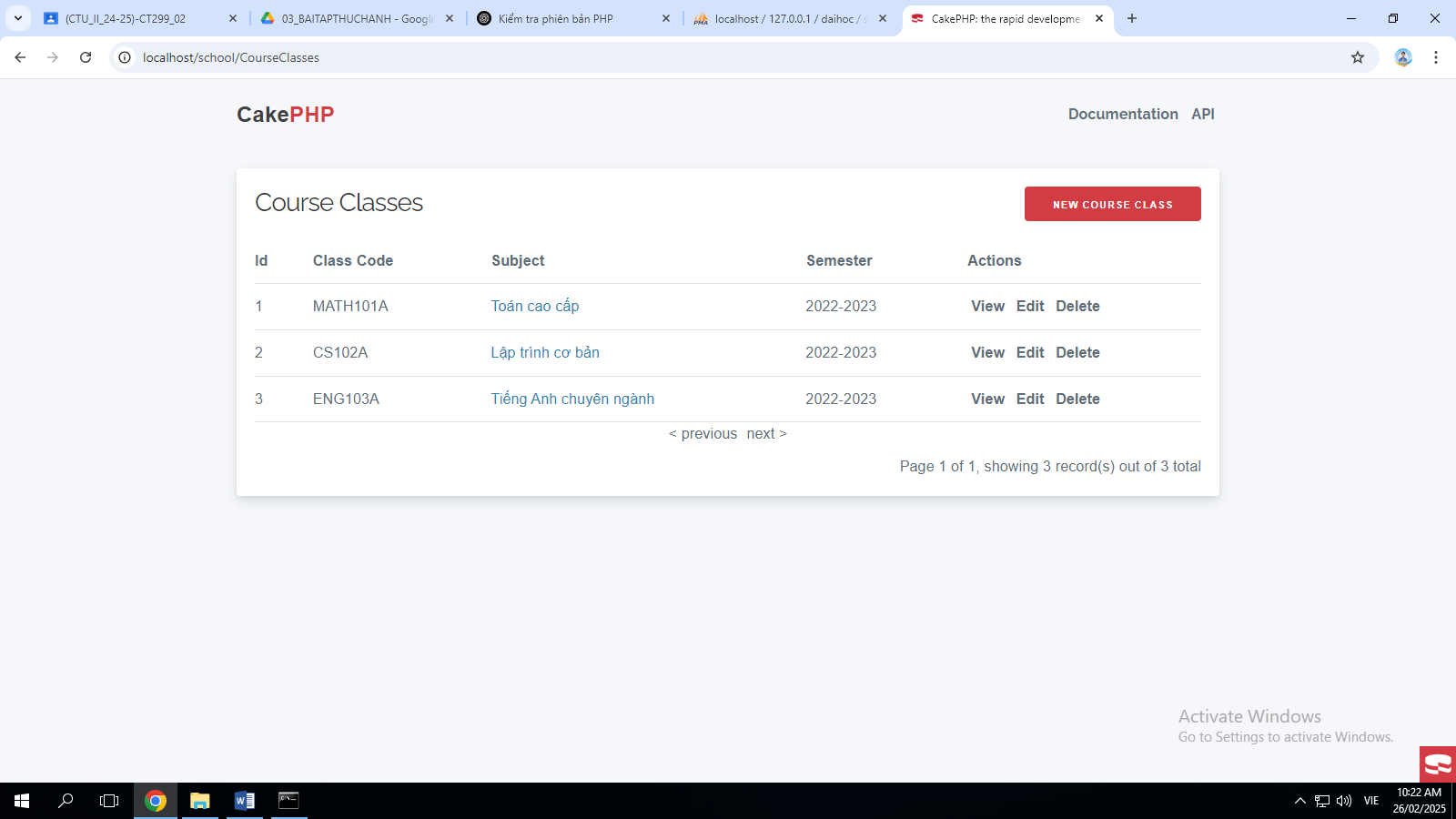
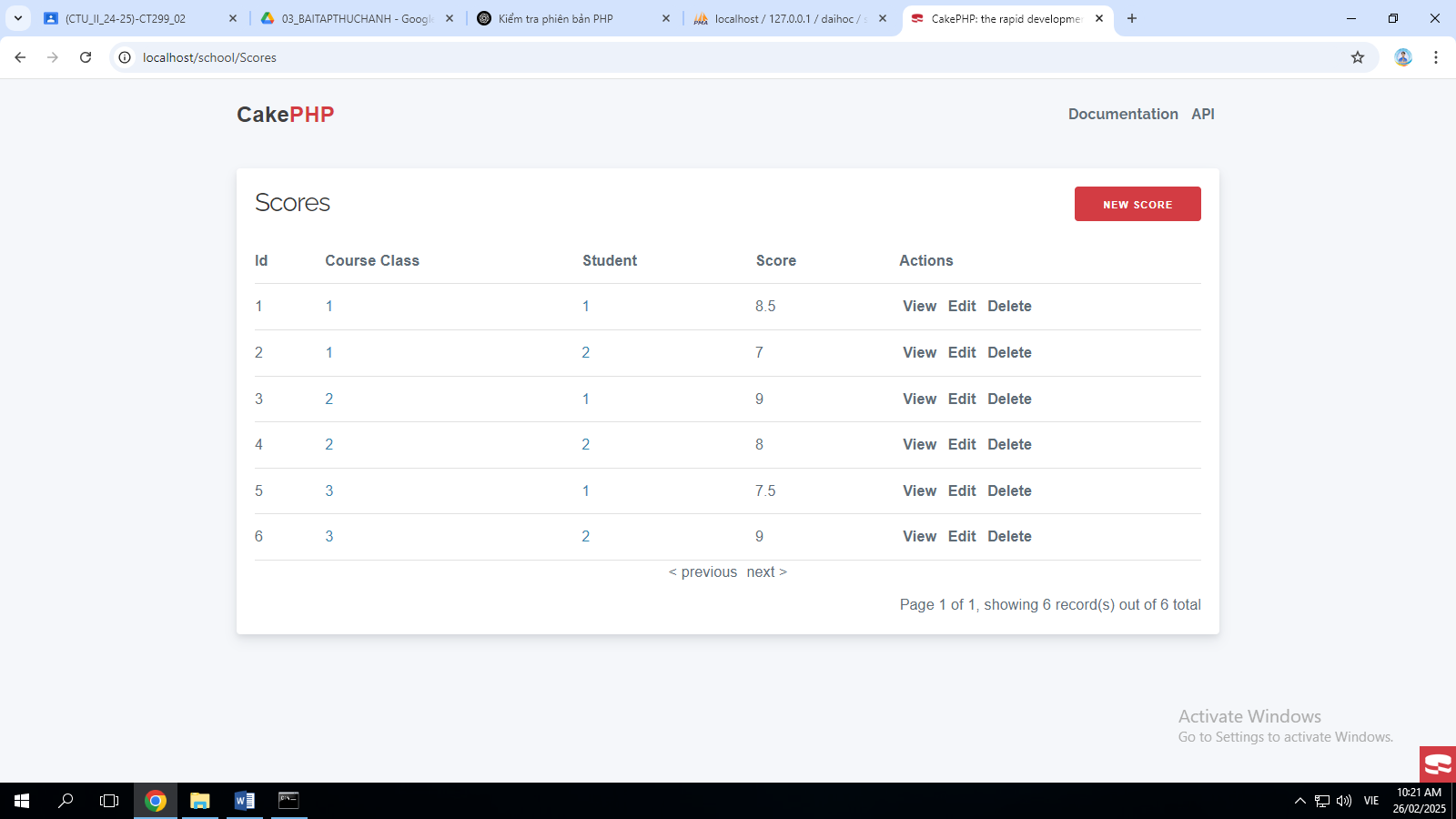
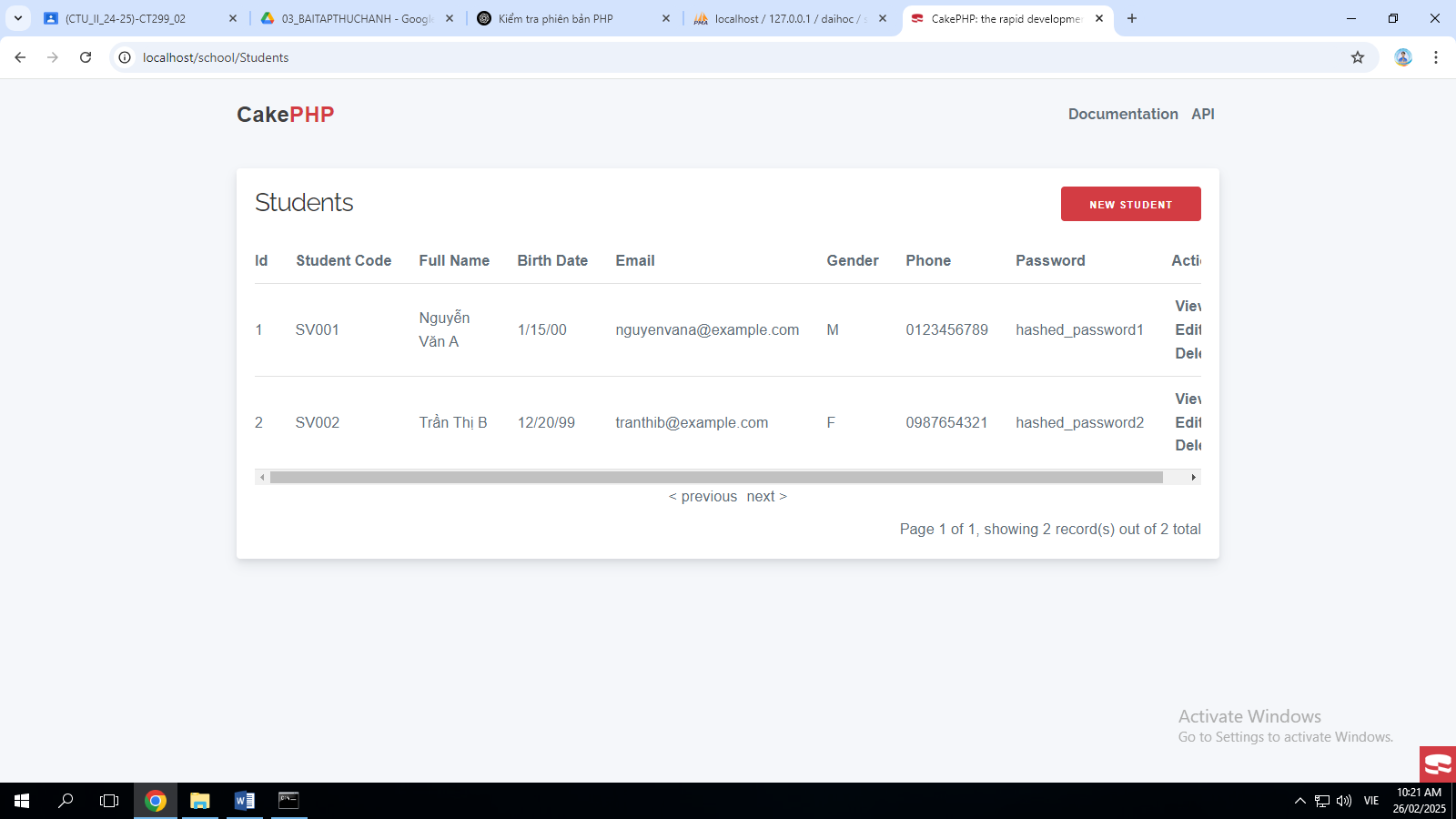




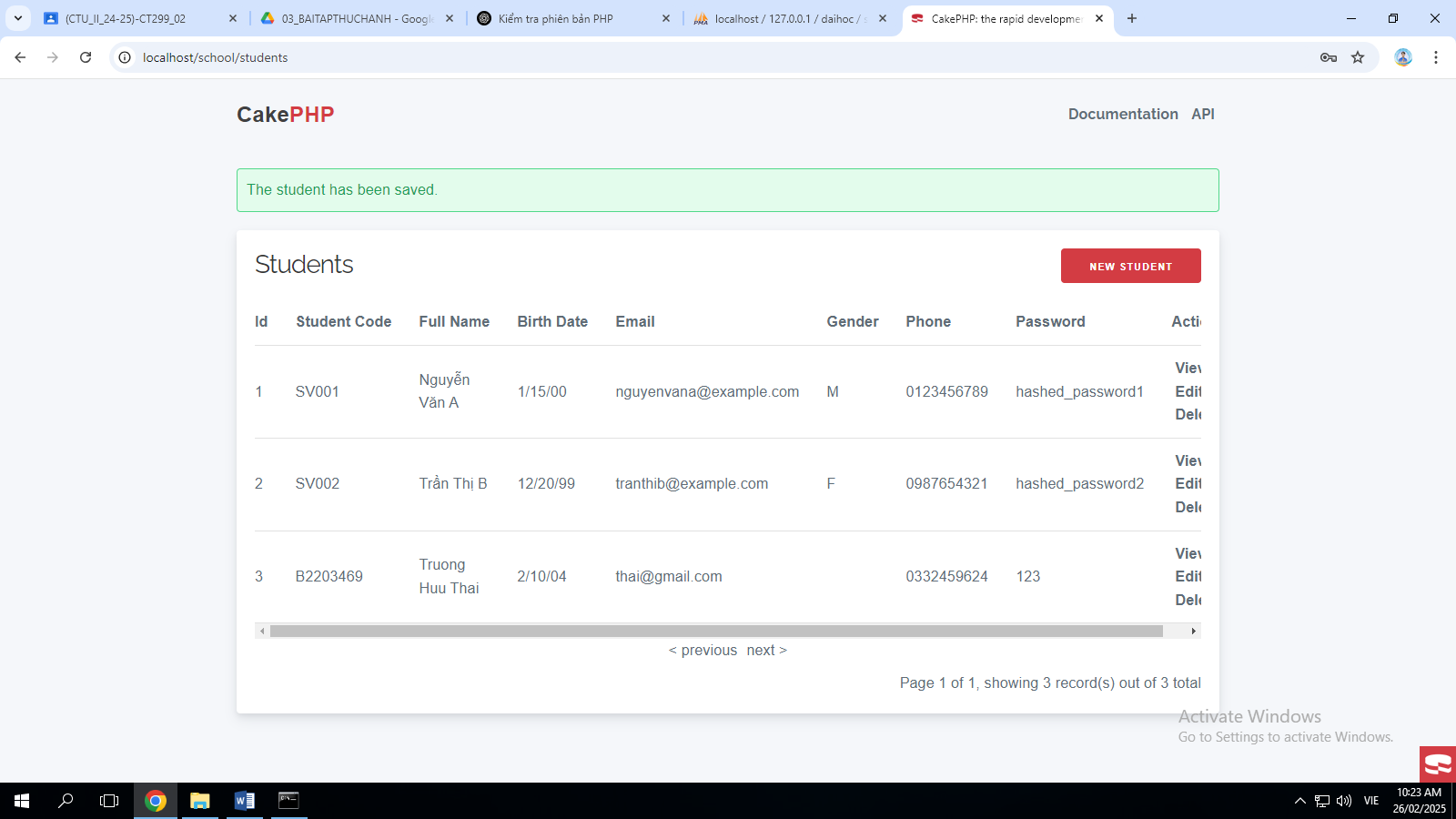
1. Bạn lựa chọn hoặc dùng Cakephp hoặc dùng Laravel để tạo project và tự động sinh CRUD cho CSDL đã tạo ở câu 6.



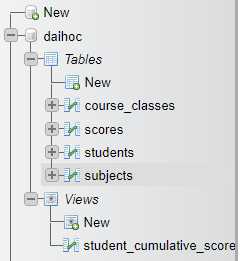
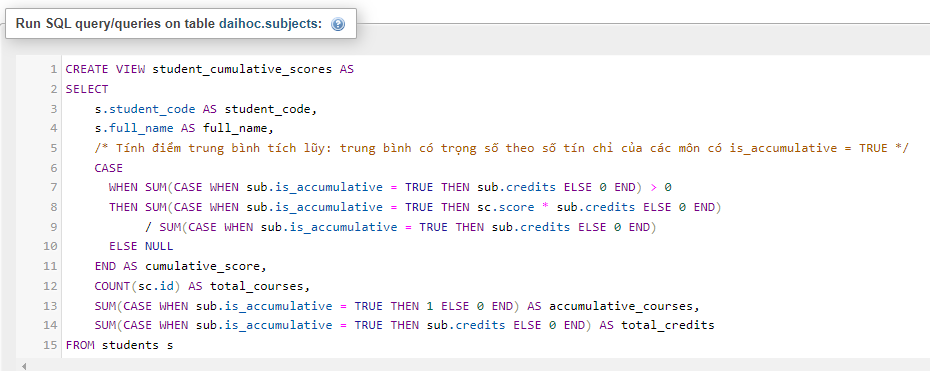




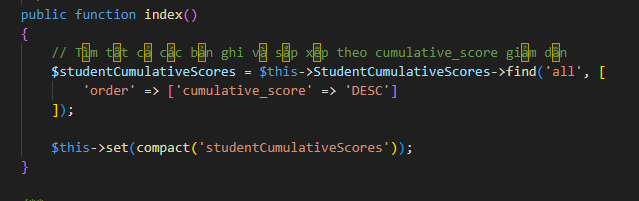
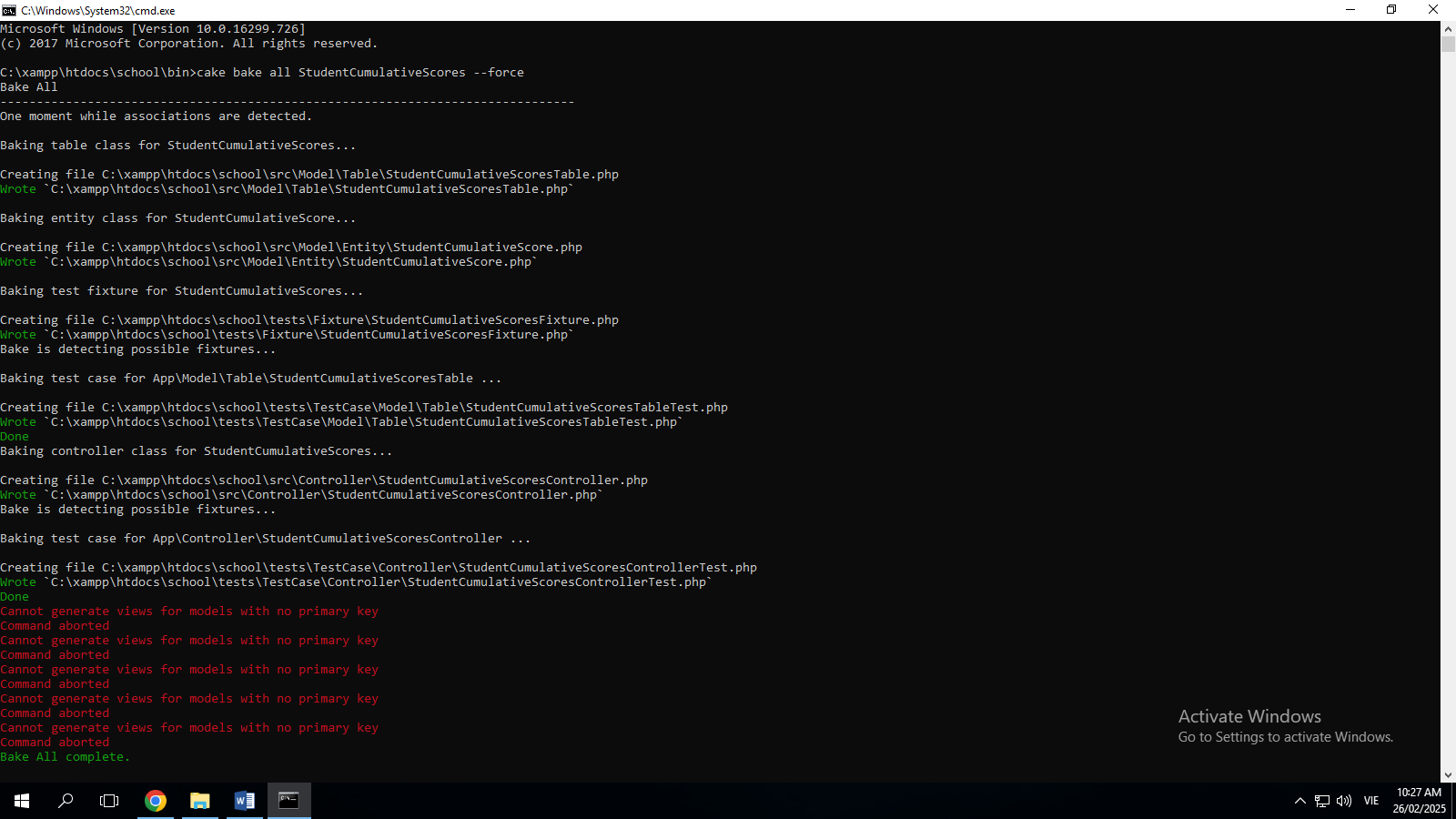
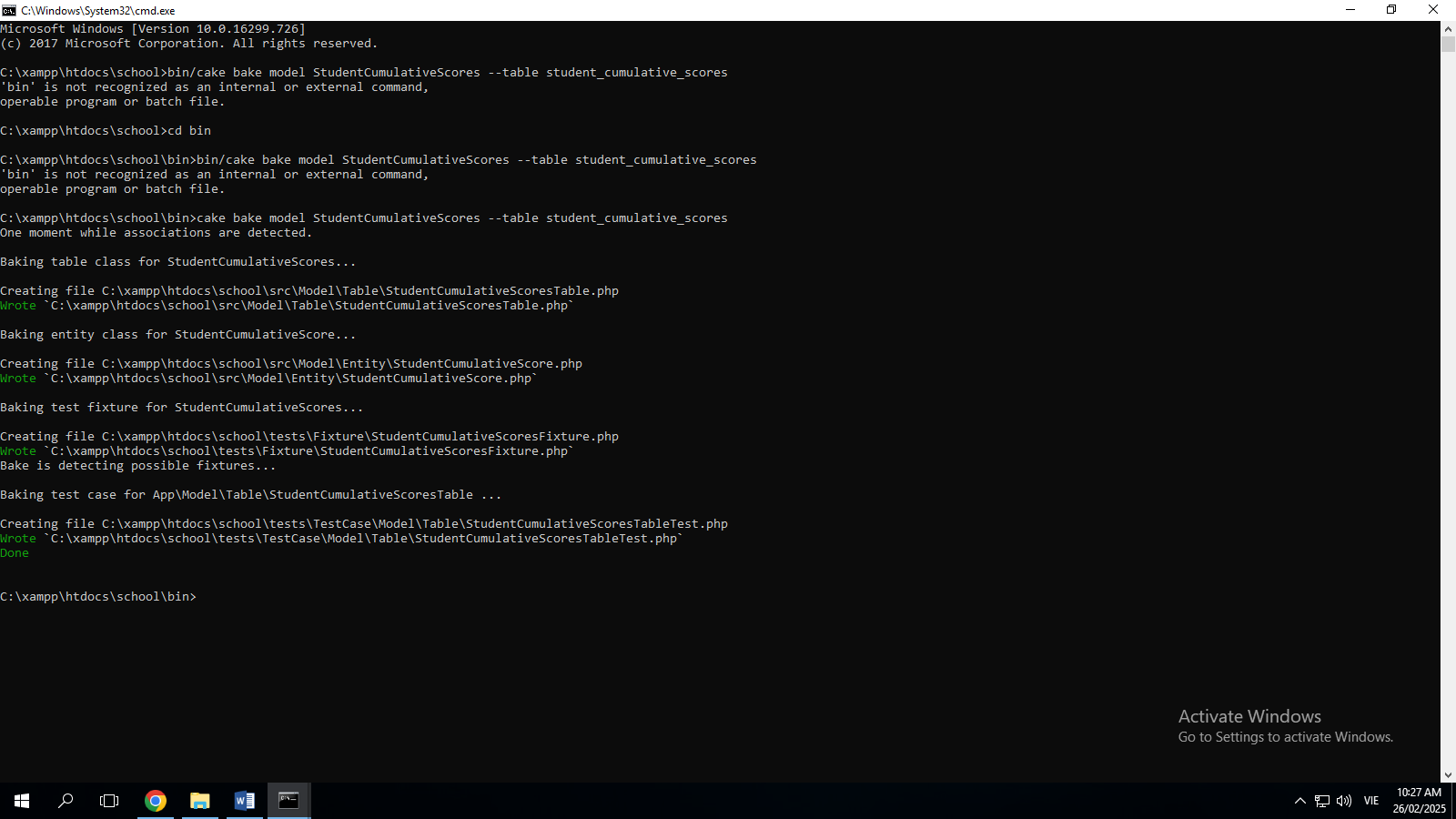
1. Từ các giao diện đã tạo bạn sẽ thêm các bạn sinh viên cùng nhóm làm bài tập của bạn, và nhập thêm 1 số dòng dữ liệu nữa để đảm bảo có ít nhất 10 sinh viên có điểm trong hệ thống.

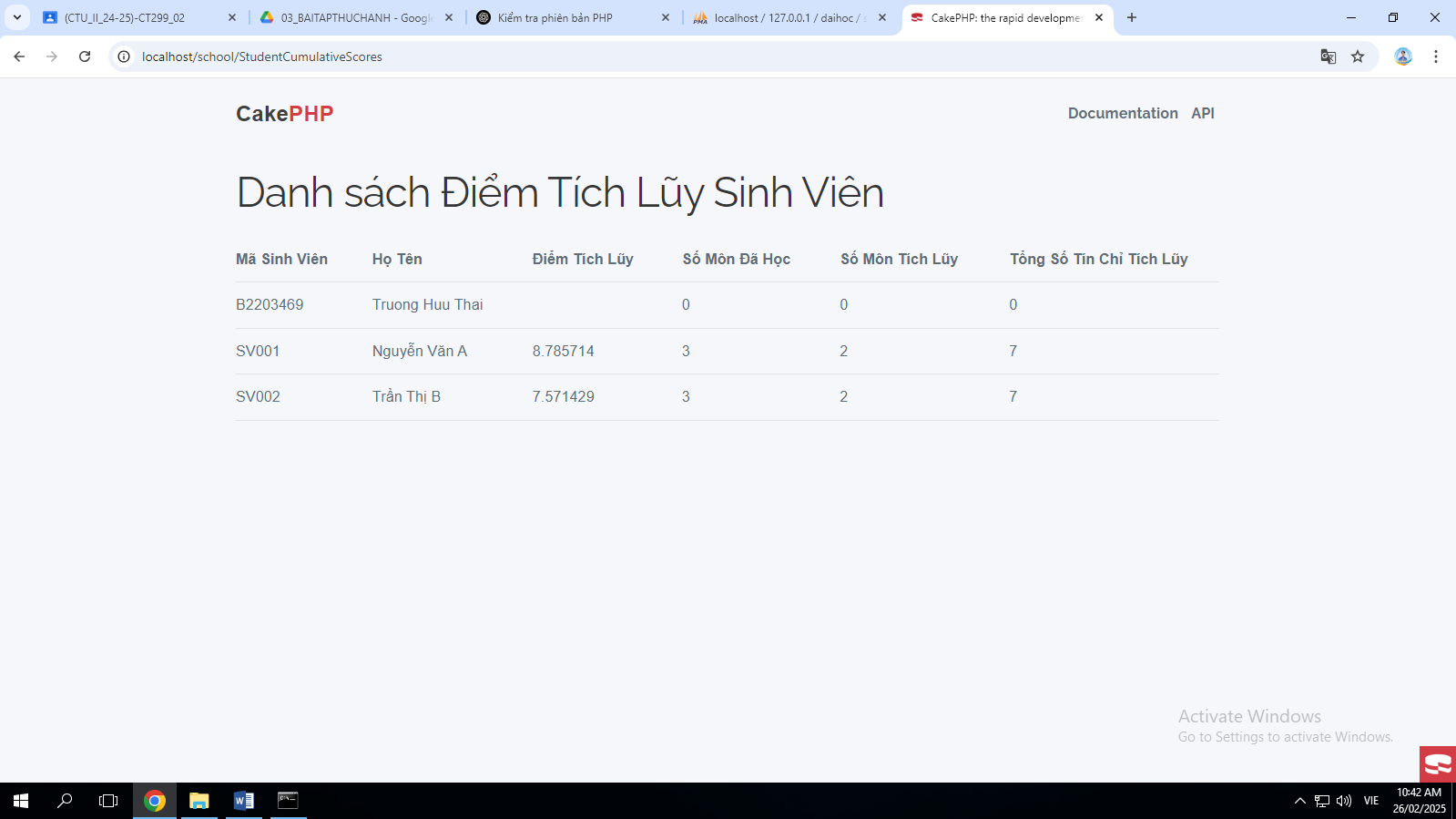


1. Tạo 1 khung nhìn (view CSDL) bằng câu lệnh SQL để tính điểm trung bình tích lũy của mỗi sinh viên, thông tin bao gồm: mã số sinh viên, họ tên, điểm tích lũy, số môn đã học, số môn đã tích lũy, tổng số tín chỉ đã tích lũy. Tên view CSDL nên đặt tên sao cho thỏa điều kiện để có thể tạo CRUD tự động.



1. Hãy tạo CRUD cho view CSDL đã tạo ở câu 9, quan sát và chụp hình kết quả. Vào controller chỉnh sửa sao cho trang index của CRUD đã được tạo hiển thị điểm tích lũy của các sinh viên sắp xếp giảm dần.





**Chú ý:**

* Các bạn nộp file word: Quy tắc đặt tên file: **<mssv>-<hoten>-<bai><stt\_bai thực hành>.docx** nộp lên Classroom (VD: **B123456-NguyenVanA-bai1.docx**), kèm với các file khác được yêu cầu như phần câu hỏi đã nêu. **Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file zip**. File zip đặt tên như file word.
* Mỗi câu các bạn trả lời bằng hình hoặc dạng text tùy vào yêu cầu của câu hỏi và **TRẢ LỜI THEO ĐÚNG THỨ TỰ CÂU HỎI**. Nếu câu nào không trả lời được các bạn cứ để số thứ tự câu hỏi và bỏ trống phần trả lời.
* Các câu trả lời có tham khảo trên Internet phải trích dẫn link/nguồn.
* *Vi phạm 1 trong các điều sau đây bài thực hành sẽ bị 0 điểm:*
  + Đặt tên KHÔNG ĐÚNG quy tắc được yêu cầu.
  + Bài không đủ các thành phần (word, code+data (nếu có),...) đã được yêu cầu.
  + Bài không thực hiện đúng yêu cầu “**Ngoại trừ file word trả lời câu hỏi, các file còn lại các bạn nén vào 1 file .zip”**
  + Bị phát hiện copy, sao chép từ các bạn khác
  + Phần trả lời không ghi rõ trả lời cho câu nào
  + Thứ tự câu trả lời không đúng thứ tự câu hỏi

1. https://book.cakephp.org/4/en/intro.html [↑](#footnote-ref-0)
2. [↑](#footnote-ref-1)
3. https://book.cakephp.org/4/en/intro/conventions.html [↑](#footnote-ref-2)