COMP3385 - Assignment 1 (20 Marks)

Due: February 12, 2024

Laravel

Laravel is a web application framework for PHP. In this lab, you will install Laravel, create some routes, controllers and a template.

Exercise 1 - Setup your Computer for Laravel Development

Before creating your first Laravel project, make sure that your local machine has PHP and Composer installed. You will need at least PHP 8.2 or 8.3

PHP should already be installed on MacOS.

To install PHP on Windows you can follow the steps in the following video:

https://www.youtube.com/watch?v=MPRLUd8Pmyo

Note: You will also need to edit your **`php.ini`** configuration file that is in your PHP installation directory. Within that file look for the lines:

```
;extension=fileinfo
;extension=zip
```

Remove the ";" in front of those lines to uncomment them (if they are not already uncommented) and save the file. Laravel needs those two extensions.

To install Composer on Windows, you can use the following download link: https://getcomposer.org/Composer-Setup.exe

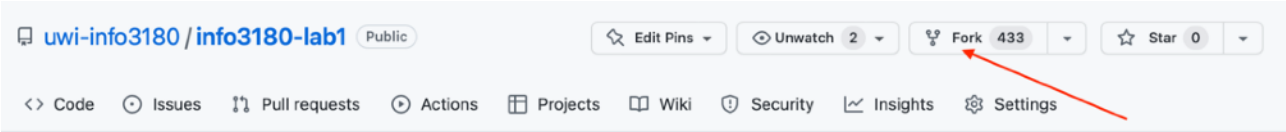
To install Composer on MacOS or Linux, you can follow the instructions at: https://getcomposer.org/doc/00-intro.md#installation-linux-unix-macos

Exercise 2 - Fork and Clone the Assignment 1 Repository

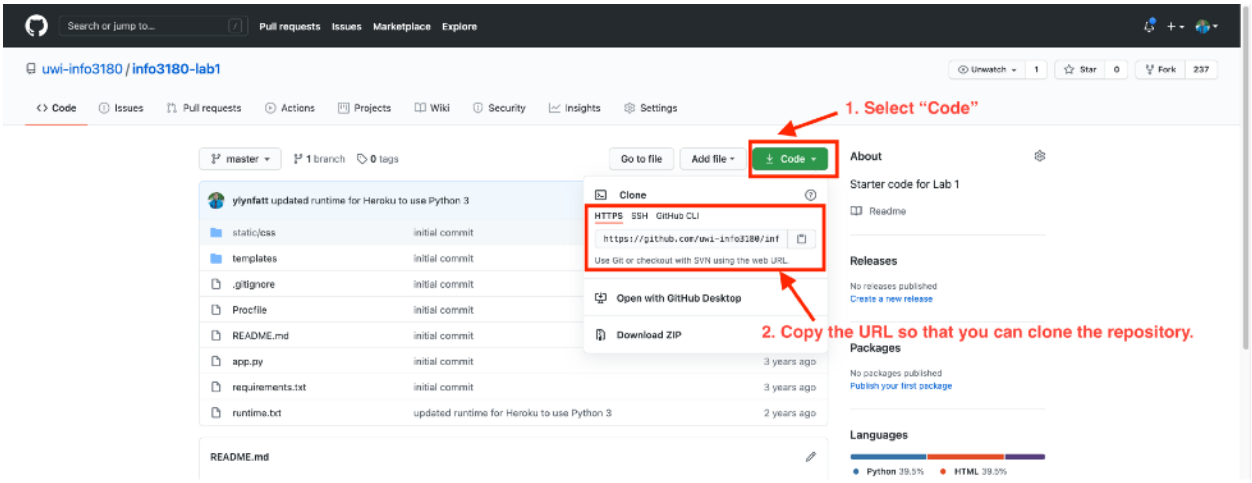
A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. Most commonly, forks are used to either propose changes to someone else's project or to use someone else's project as a starting point for your own idea. In our case for this lab we are using another project on Github as the starting point for this Lab.

Login to your Github account and fork the repository by visiting the link below and clicking on the "Fork" button:

https://github.com/uwi-comp3385/comp3385-assignment-1.git



Next, to clone it to your local machine you will first need to get the URL of the new repository that you just forked to your account:



Then do the following on your own local computer from the command line/prompt in a folder of your choice. Ensure you change **{yourusername}** to your actual Github username:

```
git clone https://github.com/{yourusername}/comp3385-assignment1
```

Exercise 3 - Install Laravel and it's dependencies and start the Development Server

1. Now that you've cloned the Assignment 1 repository, navigate to the folder and install the Laravel dependencies:

   ```
   cd comp3385-assignment1
   composer install
   ```

2. Next, start Laravel's local development server by using Laravel Artisan's **`serve`** command:

   ```
   php artisan serve
   ```

3. Once you have started the Artisan development server, your application will be accessible in your web browser at http://localhost:8000.
4. You should see the default Laravel welcome page.

Exercise 4 - Create a new Route

Now let us create our first route.

1. Open the **`routes/web.php`** file in your IDE.
2. You should see an existing route for "**`/`**" that represents the default Laravel welcome page.
3. Below that route, create your first route with the URL path "**`/about`**". For the 2nd parameter, use an anonymous PHP function (closure) and return the string "`My About Page`".

```
Route::get('/about', function() {
     return "My About Page";
});
```

4. Ensure your Laravel development server is running and open your web browser and visit "http://localhost:8000/about". You should see the page load with the string you returned from your route.
5. Now would be a good time to commit your changes to your Git repository and push the changes to Github.

Exercise 5 - Create a new Controller, Layout and View

Now that we know how to create a simple route, let us convert it to use a controller and also use a template for our view.

Instead of defining all of your request handling logic as closures in your route files, you may wish to organize this behavior using "controller" classes.

1. To create a new controller we could create it from scratch, however, we will use Laravel Artisan Command Line Interface (CLI) to help us to create one instead. Within your project root, run the following command:

```
php artisan make:controller AboutController
```

2. Now open your **app/Http/Controllers** folder. You should see your new controller file **AboutController.php**.
3. Within your new controller class, create a new method called **show()**. Inside of that method you will now return a template, using the Laravel **view()** function. The first parameter of that function should be the name of your template file. In this case let us call it "about".

```
public function show() {
     return view('about');
}
```

4. Let us create a layout template that we can use across multiple pages. Open your **resources/views** folder and create a new folder called **layouts**. Inside of that folder create a new file called **app.blade.php**. Note that all Laravel template files end with "**.blade.php**". Within that file, add the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>COMP3385 - Lab 1</title>
```

```
</head>
<body>
    <header>
        <h1>COMP3385 - Lab 1</h1>
    </header>
    <main>
        @yield('content')
    </main>
    <footer>
        <p>Copyright &copy; 2024, COMP3385 Inc.</p>
    </footer>
</body>
</html>
```

5. Now you will need to create a template file for your About page that will inherit that layout. Open your **`resources/views/`** folder and create a new file called "**`about.blade.php`**".
6. In that file, ensure you extend your app layout and open your 'content' section/ block. In your content section put your name within the level two heading (<h2></ h2>) and write a short paragraph within a paragraph tag ( ) about yourself.

```
@extends('layouts.app')

@section('content')
    <h2>Place Your Name Here</h2>
    <p>Write a short paragraph about yourself here.</p>
@endsection
```

7. Update the route for your *About* page so that instead of using the anonymous function/closure that it instead references your new controller and the show method.

   ```
   Route::get('/about', [AboutController::class, 'show']);
   ```

   also remember to import your AboutController class at the top of the file.

   ```
   use App\Http\Controller\AboutController;
   ```

8. Now refresh your web browser and you should see your new about page.
9. Now would be a good time to commit your changes to your Git repository and push the changes to Github.

Exercise 6 - Style your page and add an Image

Now that you know how to create a controller and a template, let us add a CSS file to our template, style the page and also add an image.

1. In your **`public`** folder, create a **`css`** folder and then add a file called **`app.css`** and add some CSS to change the `font-size`, `font-family` and `colour` of the text in your web page or any other styles that you would like to make your webpage look like your own.
2. Now, Open your **`app.blade.php`** layout template file and in the **`<head></head>`** tags add the following:

```
<link rel="stylesheet" href="{{ url('/css/app.css') }}" />
```

3. Next, in your **public** folder, create an **images** folder and put an image of your choosing in that **public/images** folder. Now in your **about.blade.php** template file load that image by using the **url()** function in an **<img />** tag.

```
<img src="{{ url('/images/your-image.jpg') }}" alt="Your
alternate description" />
```

4. Now check your webpage again and ensure that your image is showing and also that the styles you added to your CSS file are also working.
5. Now would be a good time to commit your changes to your Git repository and push the changes to Github.

Submission

Submit your code via the "Lab 1" link on the VLE. You should submit the following links:

- Github repository URL for your Laravel Exercise e.g. https://github.com/{yourusername}/comp3385-assignment1