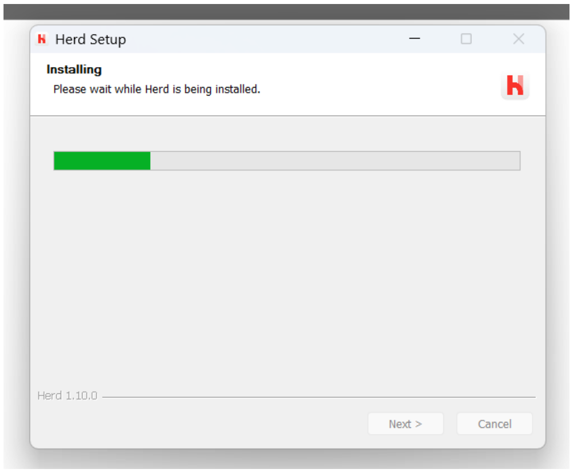
Prepared by: Janine Ishe M. Matamorosa **BSIT3C**

1. Install Laravel along with its dependencies.

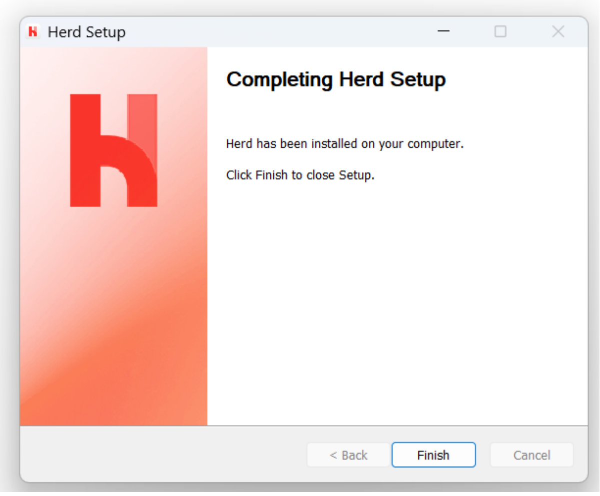


Herd Setup

Installing

Please wait while Herd is being installed.

Herd 1.10.0

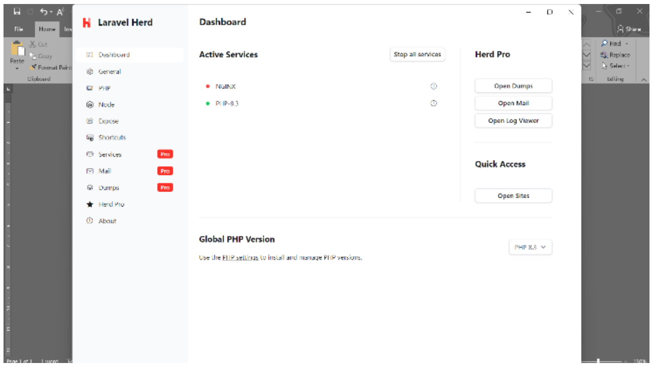


Herd Setup

Completing Herd Setup

Herd has been installed on your computer.

Click Finish to close Setup.



Dashboard

Active Services

- NGINX
- PHP-8.3

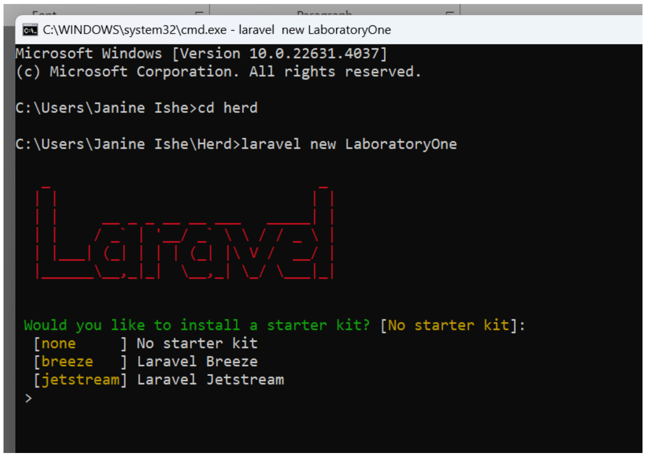
Herd Pro

Quick Access

Global PHP Version

Use the PHP settings to install and manage PHP versions.

PHP 8.3



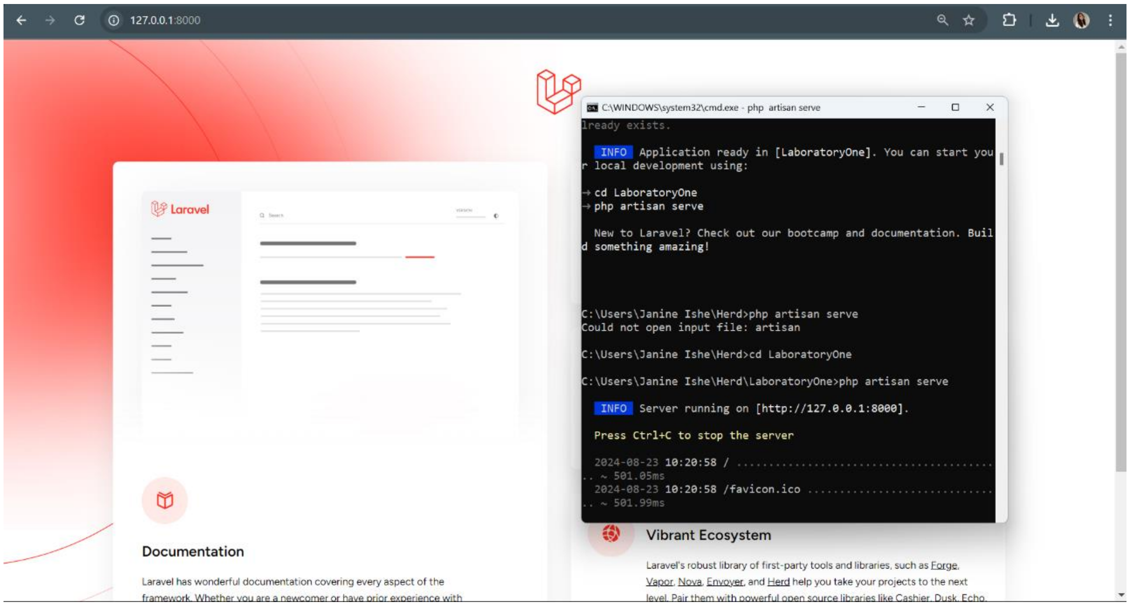
C:\WINDOWS\system32\cmd.exe - laravel new LaboratoryOne

```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Janine Ishe>cd herd

C:\Users\Janine Ishe\Herd>laravel new LaboratoryOne


 Would you like to install a starter kit? [No starter kit]:
  [none      ] No starter kit
  [breeze    ] Laravel Breeze
  [jetstream] Laravel Jetstream
 >
```



127.0.0.1:8000

C:\WINDOWS\system32\cmd.exe - php artisan serve

```
lready exists.

   INFO  Application ready in [LaboratoryOne]. You can start you
r local development using:

→ cd LaboratoryOne
→ php artisan serve

  New to Laravel? Check out our bootcamp and documentation. Buil
d something amazing!



C:\Users\Janine Ishe\Herd>php artisan serve
Could not open input file: artisan

C:\Users\Janine Ishe\Herd>cd LaboratoryOne

C:\Users\Janine Ishe\Herd\LaboratoryOne>php artisan serve

   INFO  Server running on [http://127.0.0.1:8000].

  Press Ctrl+C to stop the server

  2024-08-23 10:20:58 / ..........................................
.. ~ 501.05ms
  2024-08-23 10:20:58 /favicon.ico ...............................
.. ~ 501.99ms
```

Documentation

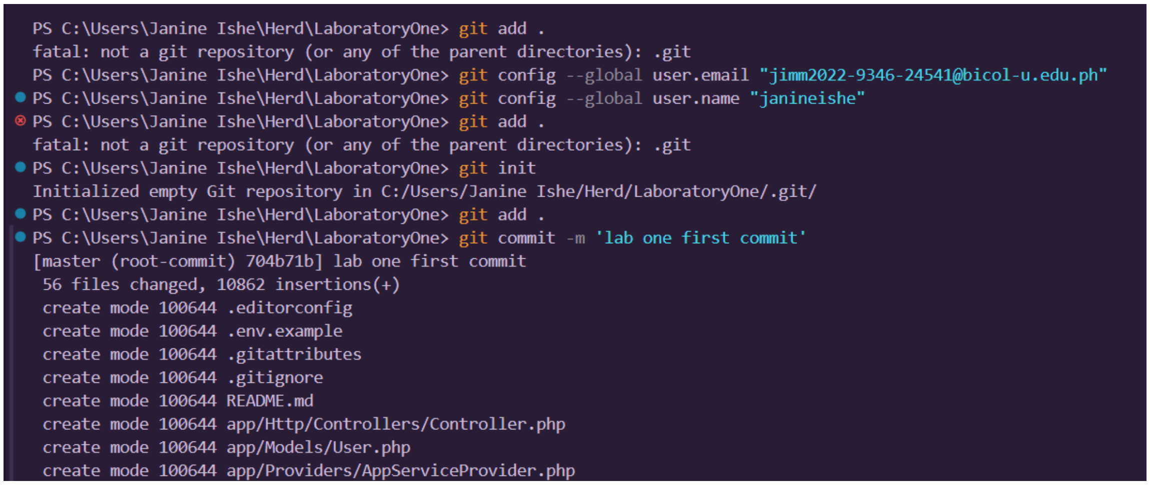
Laravel has wonderful documentation covering every aspect of the framework. Whether you are a newcomer or have prior experience with

Vibrant Ecosystem

Laravel's robust library of first-party tools and libraries, such as Forge, Vapor, Nova, Envoyer, and Herd help you take your projects to the next level. Pair them with powerful open source libraries like Cashier, Dusk, Echo,

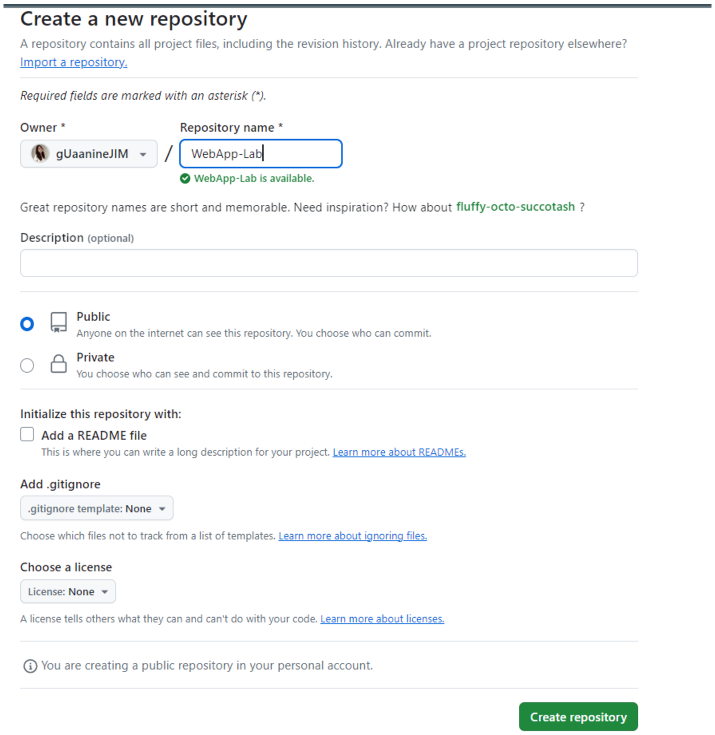
2. Create a Laravel project and put it in a Git repository

   Here are the steps I did when creating a git repository for my Laravel Projects.



```
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git add .
fatal: not a git repository (or any of the parent directories): .git
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git config --global user.email "jimm2022-9346-24541@bicol-u.edu.ph"
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git config --global user.name "janineishe"
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git add .
fatal: not a git repository (or any of the parent directories): .git
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git init
Initialized empty Git repository in C:/Users/Janine Ishe/Herd/LaboratoryOne/.git/
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git add .
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git commit -m 'lab one first commit'
[master (root-commit) 704b71b] lab one first commit
 56 files changed, 10862 insertions(+)
 create mode 100644 .editorconfig
 create mode 100644 .env.example
 create mode 100644 .gitattributes
 create mode 100644 .gitignore
 create mode 100644 README.md
 create mode 100644 app/Http/Controllers/Controller.php
 create mode 100644 app/Models/User.php
 create mode 100644 app/Providers/AppServiceProvider.php
```

Next is create a repository in GitHub.



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner *: gUaanineJIM / Repository name *: WebApp-Lab

✓ WebApp-Lab is available.

Great repository names are short and memorable. Need inspiration? How about fluffy-octo-succotash ?

Description (optional)

● Public — Anyone on the internet can see this repository. You choose who can commit.

○ Private — You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file — This is where you can write a long description for your project. Learn more about READMEs.

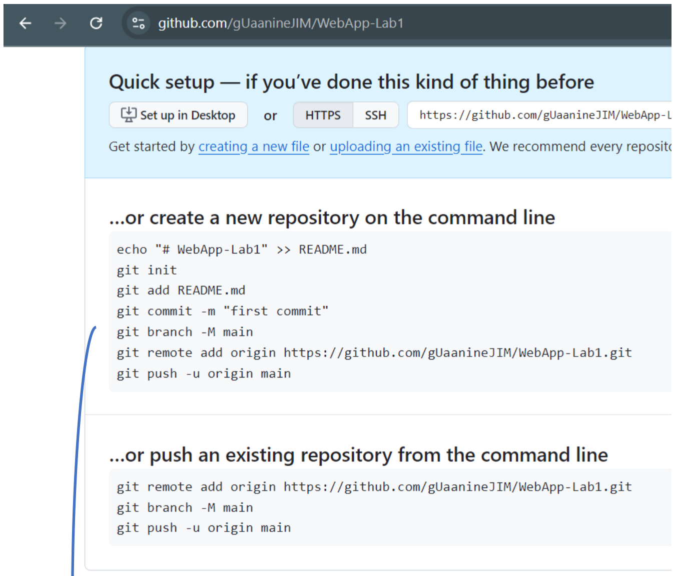
Add .gitignore: .gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license: License: None

A license tells others what they can and can't do with your code. Learn more about licenses.

ⓘ You are creating a public repository in your personal account.



github.com/gUaanineJIM/WebApp-Lab1

Quick setup — if you've done this kind of thing before

https://github.com/gUaanineJIM/WebApp-L

Get started by creating a new file or uploading an existing file. We recommend every reposit

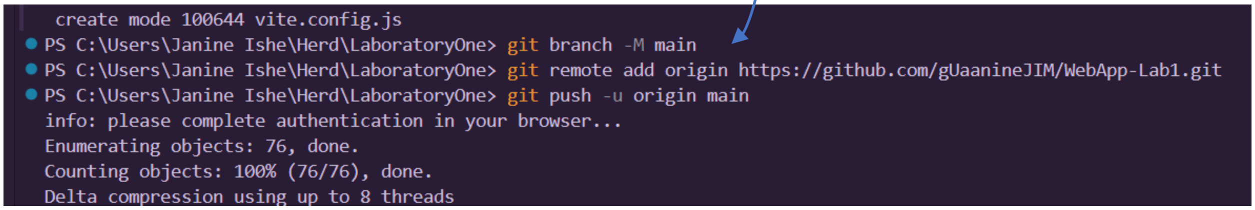
...or create a new repository on the command line

```
echo "# WebApp-Lab1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/gUaanineJIM/WebApp-Lab1.git
git push -u origin main
```

...or push an existing repository from the command line

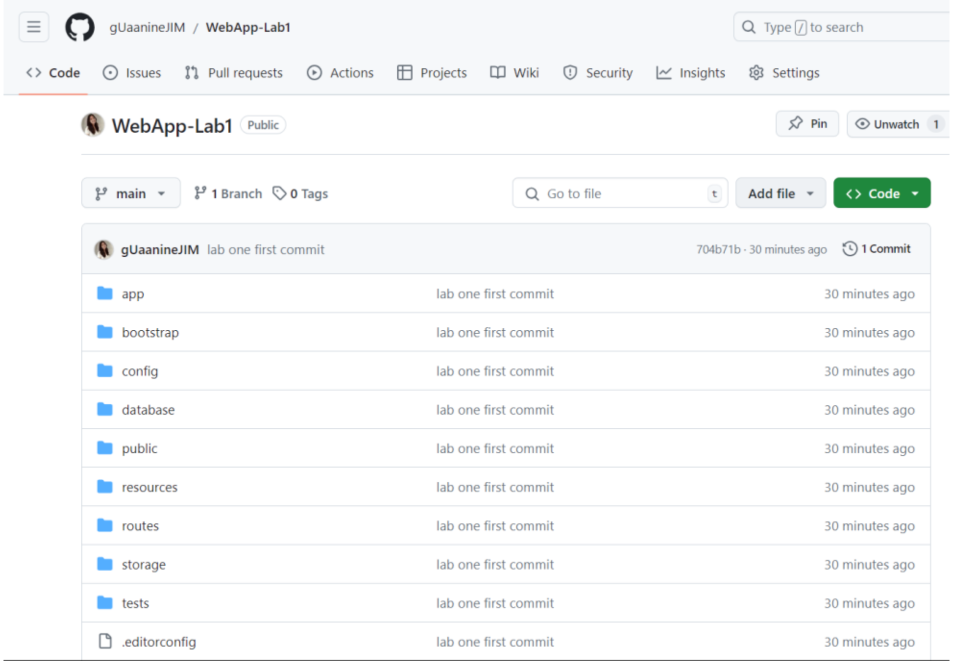
```
git remote add origin https://github.com/gUaanineJIM/WebApp-Lab1.git
git branch -M main
git push -u origin main
```

After clicking the 'Create repository' button. Copy the text 'git branch -M main to the terminal.



```
 create mode 100644 vite.config.js
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git branch -M main
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git remote add origin https://github.com/gUaanineJIM/WebApp-Lab1.git
PS C:\Users\Janine Ishe\Herd\LaboratoryOne> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 76, done.
Counting objects: 100% (76/76), done.
Delta compression using up to 8 threads
```

Go back to GitHub to check the repository.



gUaanineJIM / WebApp-Lab1

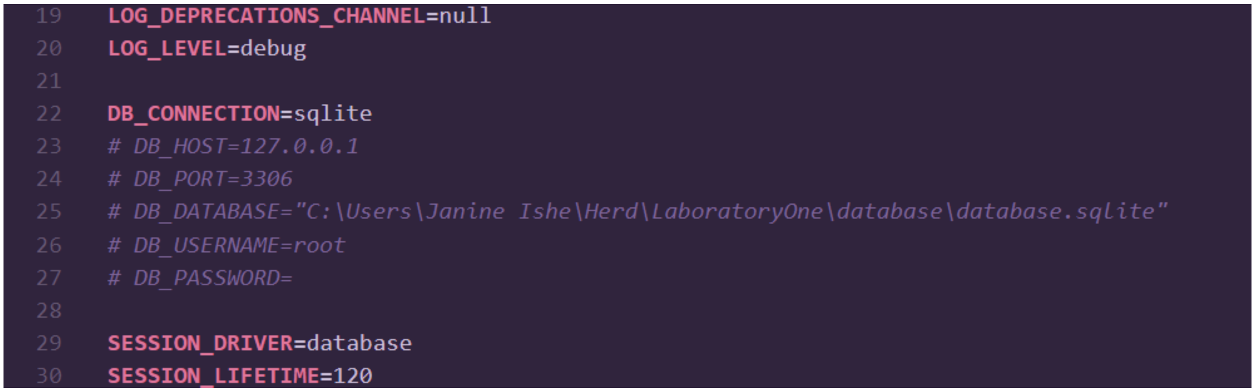
WebApp-Lab1 Public

1 Branch 0 Tags

gUaanineJIM lab one first commit 704b71b · 30 minutes ago 1 Commit

app	lab one first commit	30 minutes ago
bootstrap	lab one first commit	30 minutes ago
config	lab one first commit	30 minutes ago
database	lab one first commit	30 minutes ago
public	lab one first commit	30 minutes ago
resources	lab one first commit	30 minutes ago
routes	lab one first commit	30 minutes ago
storage	lab one first commit	30 minutes ago
tests	lab one first commit	30 minutes ago
.editorconfig	lab one first commit	30 minutes ago

3. Set the database on the env file.

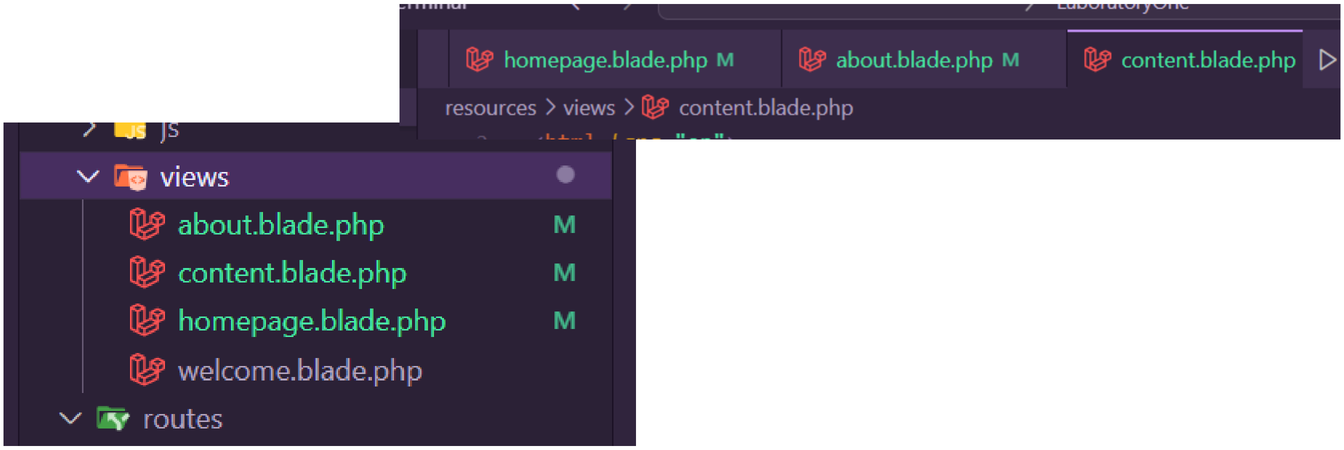


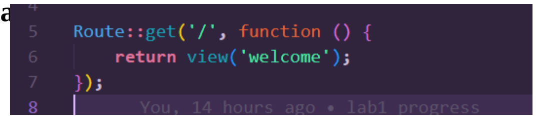
```
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug

DB_CONNECTION=sqlite
# DB_HOST=127.0.0.1
# DB_PORT=3306
# DB_DATABASE="C:\Users\Janine Ishe\Herd\LaboratoryOne\database\database.sqlite"
# DB_USERNAME=root
# DB_PASSWORD=

SESSION_DRIVER=database
SESSION_LIFETIME=120
```

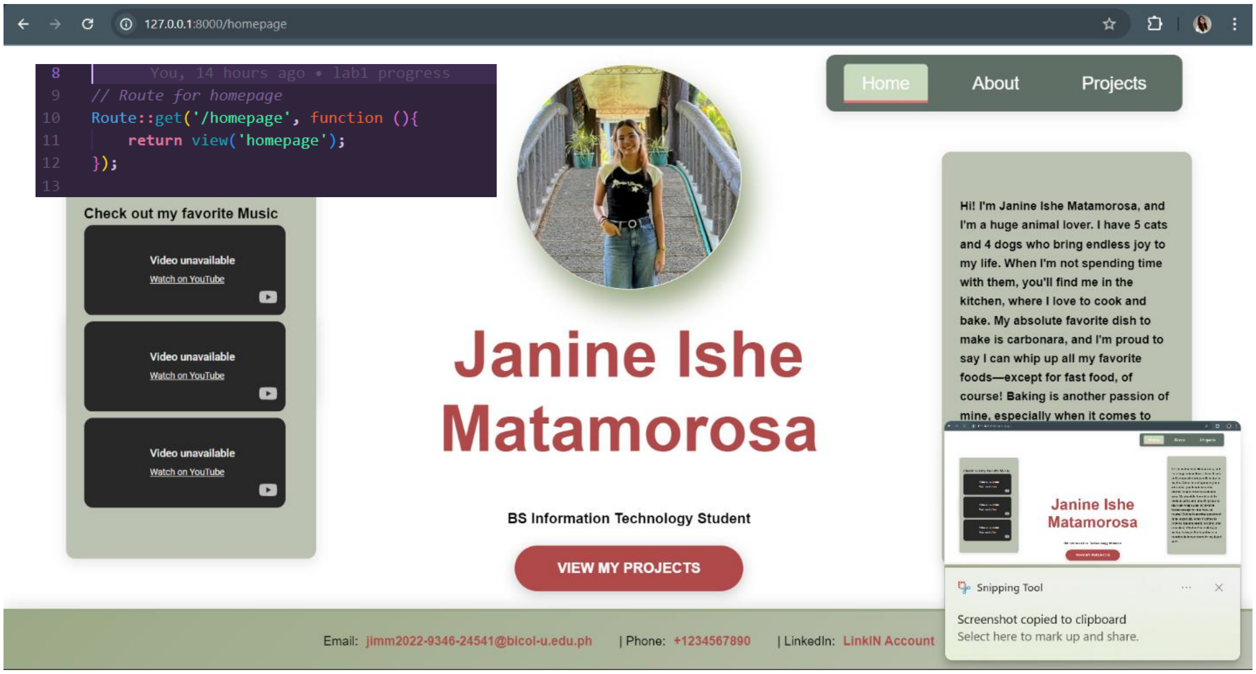
4. Create at least 3 views (homepage, about, content)





```
Route::get('/', function () {
    return view('welcome');
});
    You, 14 hours ago • lab1 progress
```

5. Create and run routes.



127.0.0.1:8000/homepage

```
    You, 14 hours ago • lab1 progress
// Route for homepage
Route::get('/homepage', function (){
    return view('homepage');
});
```

Check out my favorite Music

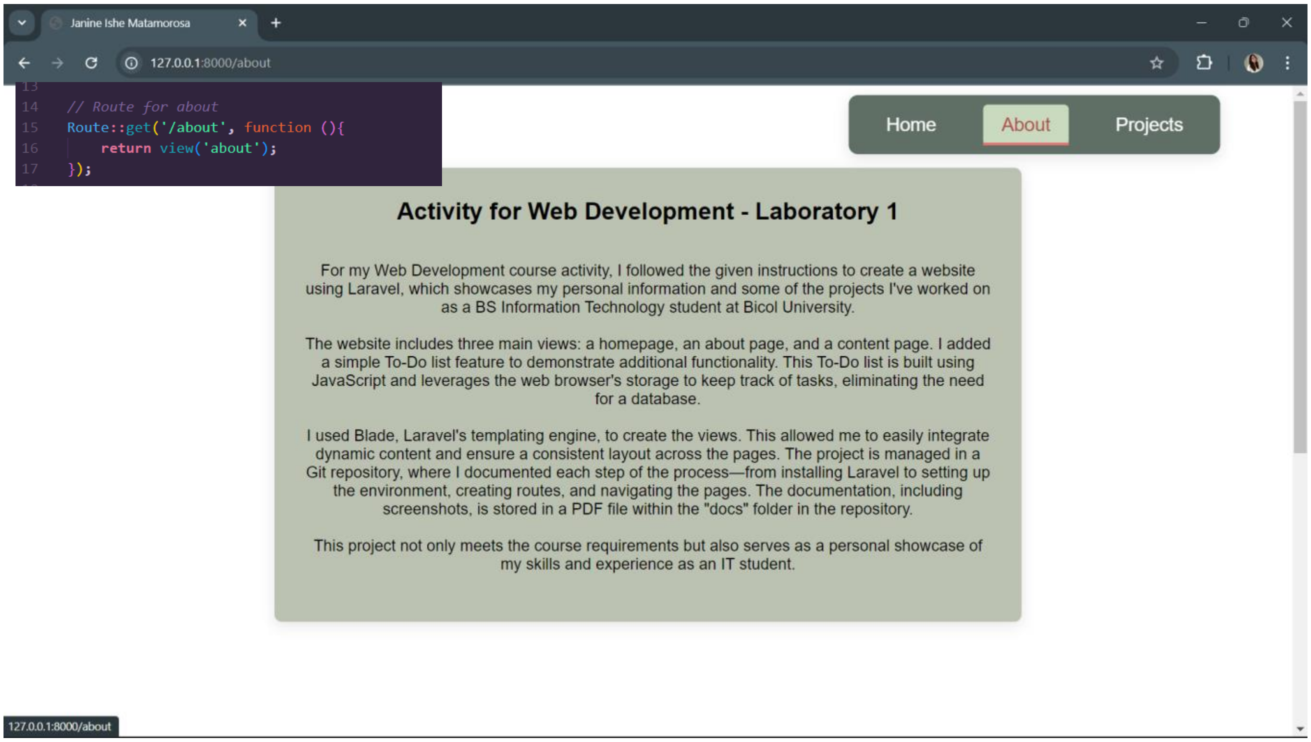
Janine Ishe Matamorosa

BS Information Technology Student

Hi! I'm Janine Ishe Matamorosa, and I'm a huge animal lover. I have 5 cats and 4 dogs who bring endless joy to my life. When I'm not spending time with them, you'll find me in the kitchen, where I love to cook and bake. My absolute favorite dish to make is carbonara, and I'm proud to say I can whip up all my favorite foods—except for fast food, of course! Baking is another passion of mine, especially when it comes to

Snipping Tool
Screenshot copied to clipboard
Select here to mark up and share.

Email: jimm2022-9346-24541@bicol-u.edu.ph | Phone: +1234567890 | LinkedIn: LinkIN Account



Janine Ishe Matamorosa

127.0.0.1:8000/about

```
// Route for about
Route::get('/about', function (){
    return view('about');
});
```

Activity for Web Development - Laboratory 1

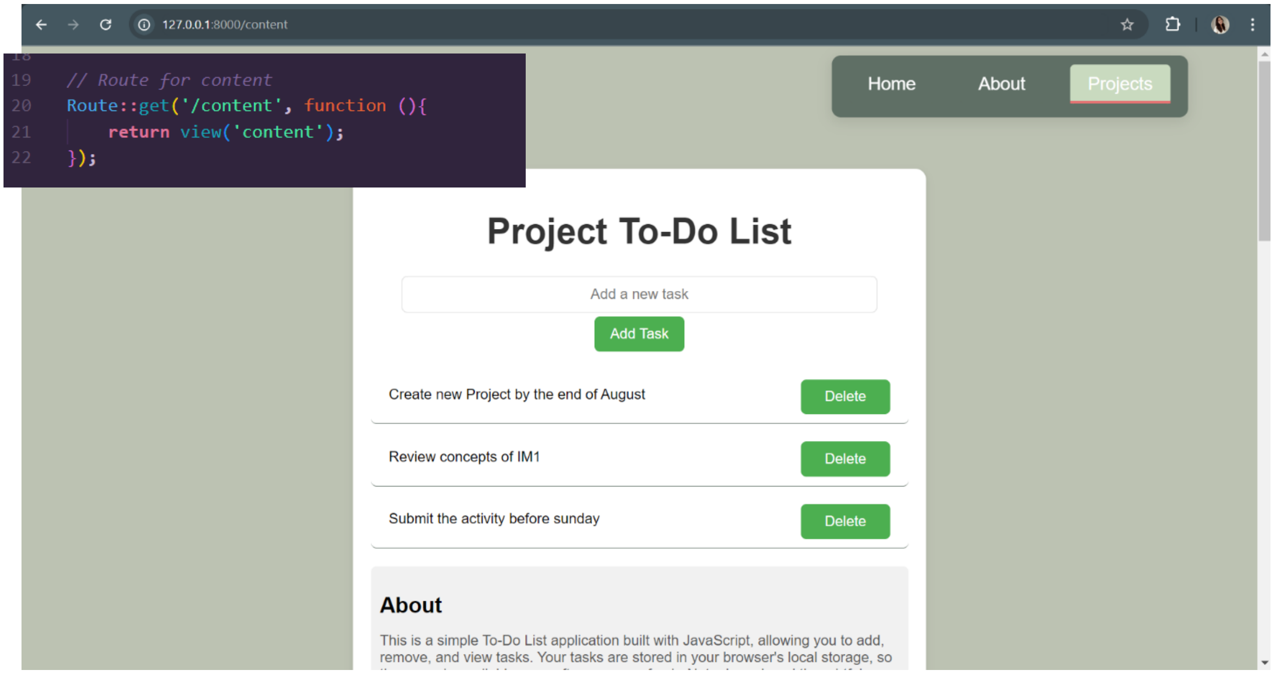
For my Web Development course activity, I followed the given instructions to create a website using Laravel, which showcases my personal information and some of the projects I've worked on as a BS Information Technology student at Bicol University.

The website includes three main views: a homepage, an about page, and a content page. I added a simple To-Do list feature to demonstrate additional functionality. This To-Do list is built using JavaScript and leverages the web browser's storage to keep track of tasks, eliminating the need for a database.

I used Blade, Laravel's templating engine, to create the views. This allowed me to easily integrate dynamic content and ensure a consistent layout across the pages. The project is managed in a Git repository, where I documented each step of the process—from installing Laravel to setting up the environment, creating routes, and navigating the pages. The documentation, including screenshots, is stored in a PDF file within the "docs" folder in the repository.

This project not only meets the course requirements but also serves as a personal showcase of my skills and experience as an IT student.

127.0.0.1:8000/about



```
// Route for content
Route::get('/content', function (){
    return view('content');
});
```

Project To-Do List

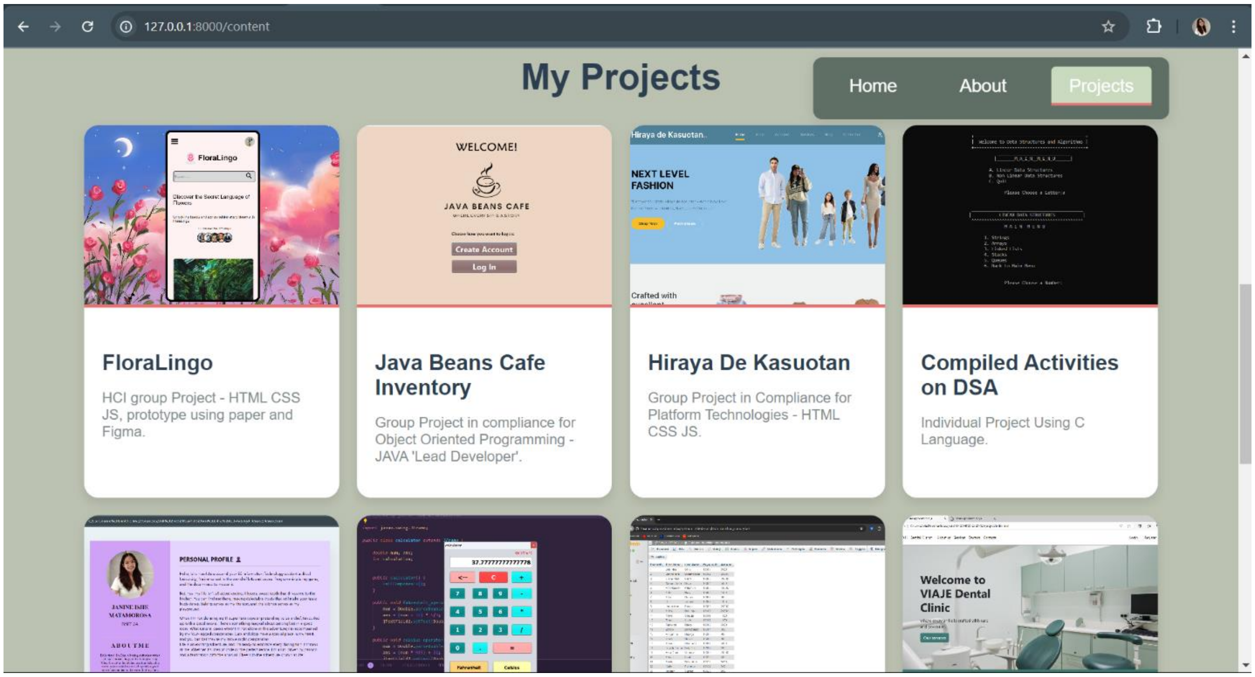
Create new Project by the end of August

Review concepts of IM1

Submit the activity before sunday

About

This is a simple To-Do List application built with JavaScript, allowing you to add, remove, and view tasks. Your tasks are stored in your browser's local storage, so



My Projects

FloraLingo

HCI group Project - HTML CSS JS, prototype using paper and Figma.

Java Beans Cafe Inventory

Group Project in compliance for Object Oriented Programming - JAVA 'Lead Developer'.

Hiraya De Kasuotan

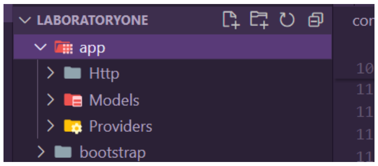
Group Project in Compliance for Platform Technologies - HTML CSS JS.

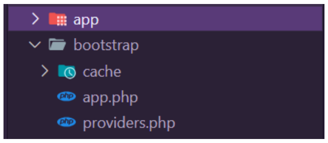
Compiled Activities on DSA

Individual Project Using C Language.

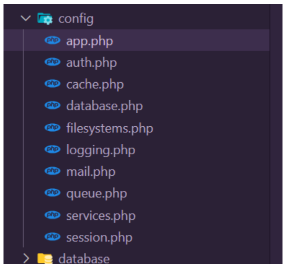
6. Identify and document the purpose of key directories and files in the Laravel project.

App Directory - This directory contains the application logic. It is divided into several subdirectories, each with a defined role in the application's structure. (HTTP Controllers, Models, Provider, Service). This serves as the brain of the application.

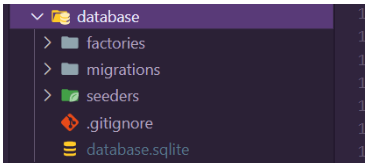




Bootstrap Directory - is responsible for initializing the application by setting up configurations and establishing the execution environment for your Laravel projects. Ex. Timezone.

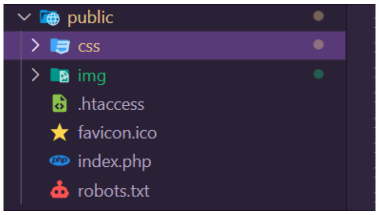


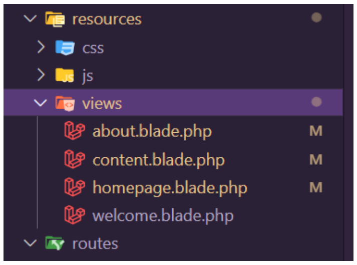
Config Directory - contains configuration files allowing us to modify the application's behavior to meet specific requirements.



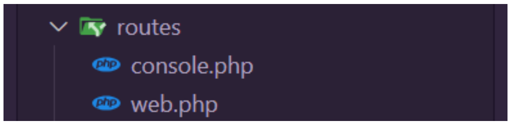
Database Directory - This holds migrations, facilitating streamlined database operations for the Laravel application. Basically, all the things related to database works here.

Public Directory - holds the index.php file as well as assets such as my newly added CSS files, and images. This contains the folder that are publicly accessible in the webserver. The images and CSS that I used on my website are stored here in the public Directory.

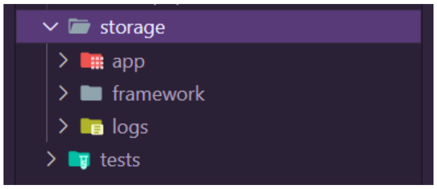




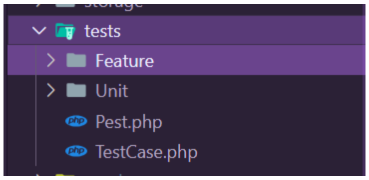
Resources directory - This manages views, language files, and other assets, ensuring efficient resource management for applications hosted on different platforms. The VIEW FOLDER is where we store our PHP file for our different webpages.



Routes Directory - controls the URL patterns, routing them to the correct controllers and methods.



Storage Directory- This manages files, including compiled Blade views and user-uploaded content.



Tests Directory - This organizes testing files, contributing to the stability and reliability of the Laravel application.

Vendor Directory - This centralizes all external dependencies and libraries, simplifying integration and management. All the dependen cies of the of the project and is managed by composer.

7. Document each step (from installation to navigation of the pages) by taking screenshots

8. For numbers 6 and 7, create a folder called "docs" and put your answers in a pdf file with the filename format: surname_fname_lab1 within the folder