1st Step: Setup environment languages.

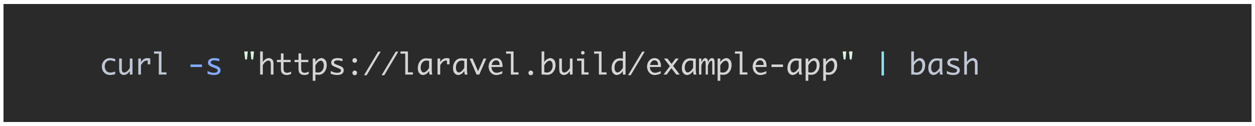
Install in your computer

- PHP 8.2
- nodeJS (v20.x)
- Composer
- Docker

2nd Step: create the environment with docker

https://laravel.com/docs/10.x#laravel-and-docker

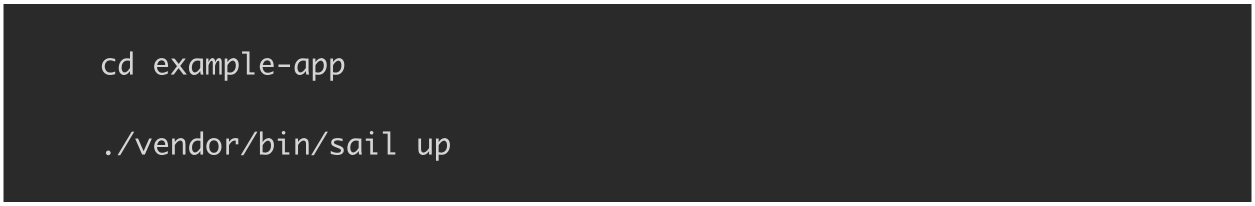
Run docker desktop or in command terminal. Then, run this command inside the folder where you want to store the project:



```
curl -s "https://laravel.build/example-app" | bash
```

This will install all the files, frameworks and development tools needed to work with the project.

Before start the docker environment, stop any httpd (Apache) or nginx process in your computer. Then go to the folder of the app:



```
cd example-app

./vendor/bin/sail up
```

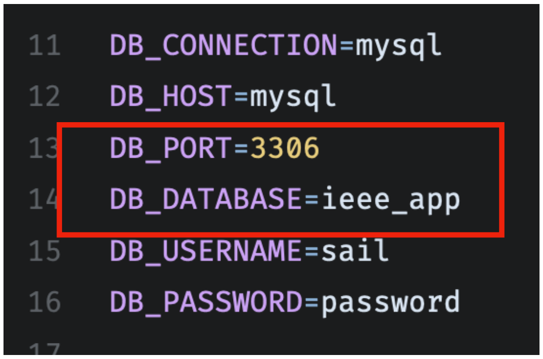
Your docker environment should start running after a couple of seconds… or more. You can look if the environment is running inside the dashboard of the docker desktop.

At this point, if you open http://localhost/ in your navigator, you will see base of the platform.

If you have an IDE for databases, like TablePlus, you can create a new connection and configure it with the credentials that are already created in the .env file inside the project folder:

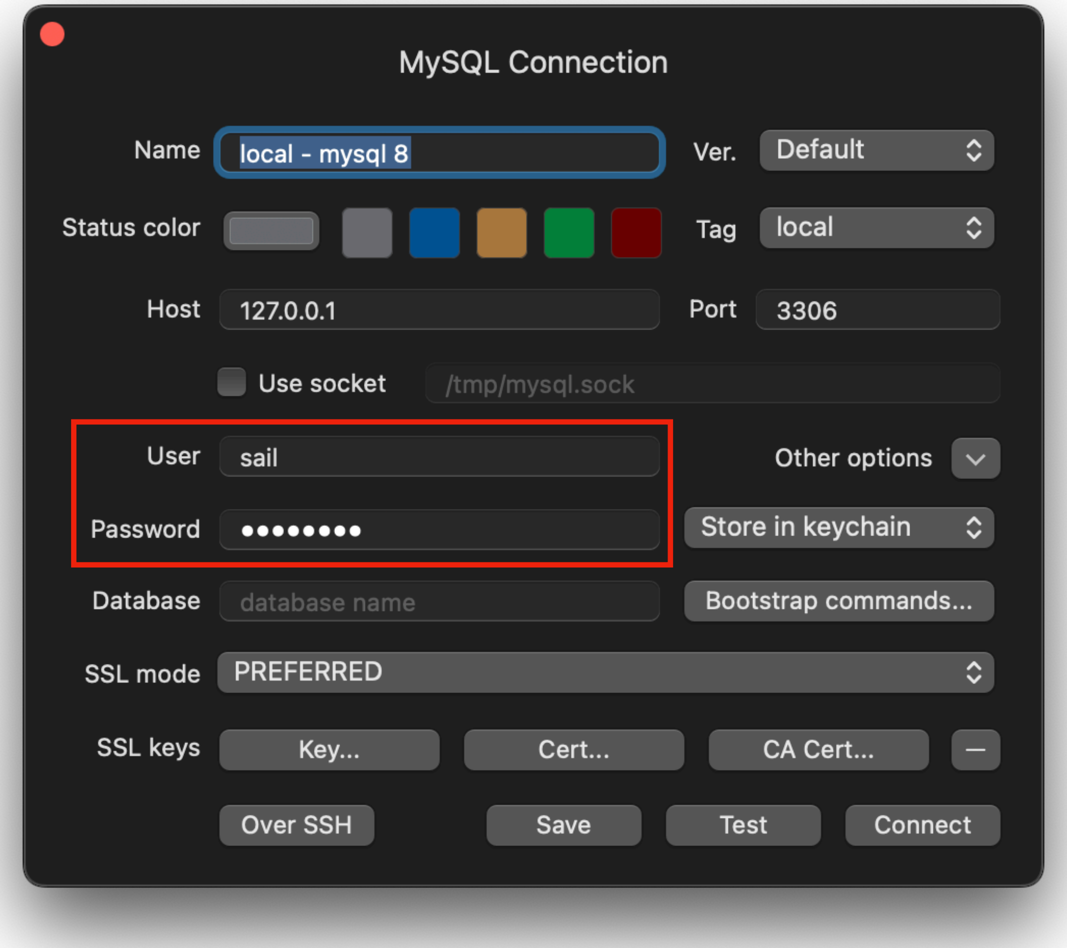
Username: sail
Password: password

From .env file ->



```
DB_CONNECTION=mysql
DB_HOST=mysql
DB_PORT=3306
DB_DATABASE=ieee_app
DB_USERNAME=sail
DB_PASSWORD=password
```

To the database connection.



MySQL Connection

Name: local - mysql 8

Ver.: Default

Status color

Tag: local

Host: 127.0.0.1

Port: 3306

☐ Use socket /tmp/mysql.sock

User: sail

Password: ●●●●●●●●●

Store in keychain

Database: database name

SSL mode: PREFERRED

After open the database that have the same name as your project, you will be able to see Your database tables. At this point, the database should be empty.

Step 3: Run first command to create some tables.

In a new terminal window, run:

```
./vendor/bin/sail artisan migrate
```

This will create some tables in the database.

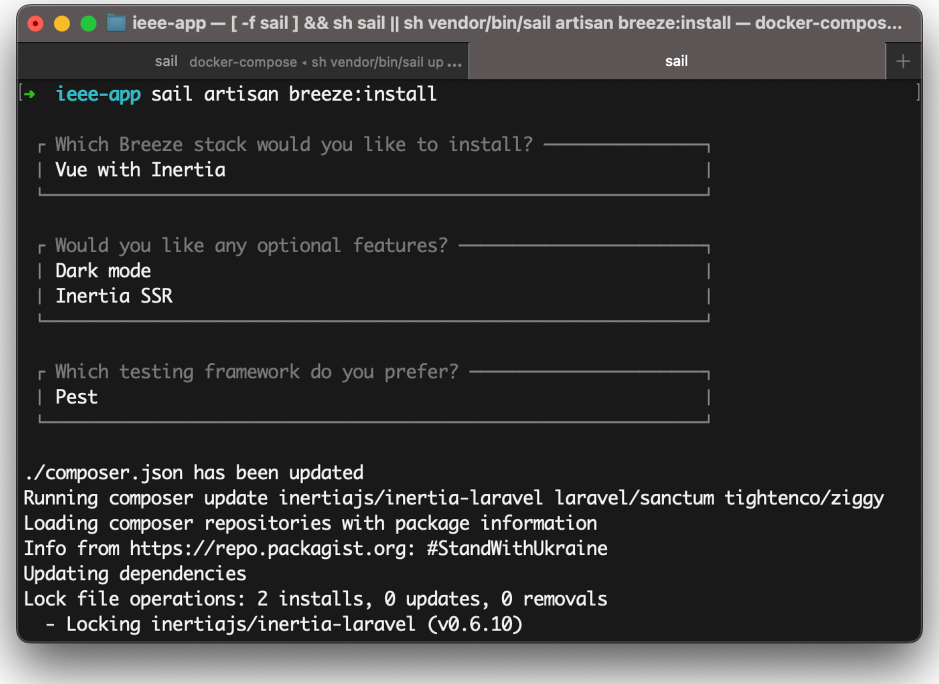
Step 4: Install the Front-End framework

https://laravel.com/docs/10.x/starter-kits#laravel-breeze-installation

To install the other framework to work with a VueJS or React (or others), you can run:

```
composer require laravel/breeze –dev

./vendor/bin/sail artisan breeze:install
```


ieee-app — [-f sail] && sh sail || sh vendor/bin/sail artisan breeze:install — docker-compos...
sail docker-compose ◂ sh vendor/bin/sail up ...
sail

```
→  ieee-app sail artisan breeze:install

 ┌ Which Breeze stack would you like to install? ─────────────┐
 │ Vue with Inertia                                           │
 └────────────────────────────────────────────────────────────┘

 ┌ Would you like any optional features? ─────────────────────┐
 │ Dark mode                                                  │
 │ Inertia SSR                                                │
 └────────────────────────────────────────────────────────────┘

 ┌ Which testing framework do you prefer? ────────────────────┐
 │ Pest                                                       │
 └────────────────────────────────────────────────────────────┘

./composer.json has been updated
Running composer update inertiajs/inertia-laravel laravel/sanctum tightenco/ziggy
Loading composer repositories with package information
Info from https://repo.packagist.org: #StandWithUkraine
Updating dependencies
Lock file operations: 2 installs, 0 updates, 0 removals
  - Locking inertiajs/inertia-laravel (v0.6.10)
```

```
./vendor/bin/sail artisan migrate

npm install    // Install tools in node_modules folder

npm run dev    // Run the front-end environment
```

At this point, if you refresh the web, you can see the *Login* and *Register* buttons. You're ready to create a platform.

Step 5 (Optional): Adding “Teams” to the profile

https://jetstream.laravel.com/installation.html

In a new terminal window, run:

```
composer require laravel/jetstream

artisan jetstream:install inertia --teams --dark --ssr
```

This will reinstall the front-end framework, but it will include the ability to put the users in teams. Then:

```
./vendor/bin/sail artisan migrate

npm install     // Install tools in node_modules folder

npm run dev     // Run the front-end environment
```

Good luck!